Energy efficiency with quality of service constraints in heterogenous networks

by

Abdullah M. Alqasir

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Co-majors: Computer Engineering; Electrical Engineering

(Computing and Networking Systems)

Program of Study Committee:
Ahmed E. Kamal, Major Professor
Daji Qiao
Sang W. Kim
Ashfaq Khokhar
Lu Ruan

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2019

DEDICATION

To the best girls in my life: Nora, Basmah and Warrd.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ACKNOWLEDGMENT

I would like to take this opportunity to express my thanks to those who helped me conducting research and the writing of this thesis. First, I would like to express my sincere gratitude to my advisor Prof. Ahmed Kamal for his continuous support of my Ph.D study and research, for his patience, motivation, and knowledge. Without his continuous help and support, I would not have been where I am today and what I am today.

I would also like to thank my committee members for their valuable feedback and contributions to this work: Dr. Daji Qiao, Dr.Sang Kim, Dr. Ashfaq Khokhar and Dr. Lu Ruan. Finally, I would like to thank my fellow labmates in Laboratory for Advanced Networks (LAN) Group for the stimulating discussions, and for all the help and fun I have had in the last years with them.

ABSTRACT

The Fifth Generation (5G) cellular network is a new technology that is driven by the demand of high data usage, large number of wireless devices and better Quality of Service (QoS). An important challenge for 5G is the energy consumption which is causing the wireless communication networks to be one of the main contributors of the global worming. Thus, energy efficiency becomes an important aspect in designing the wireless communication networks.

In this thesis, we study different approaches for Energy Efficient (EE) operation of Small Base Stations (SBSs) in Heterogenous wireless Networks (HetNets). First, we focus on enhancing energy efficiency in heterogenous networks, where Macro BSs (MBSs) and SBSs co-exist, by presenting a sleeping strategy. In the sleeping strategy SBSs serving few or no users are turned off and their users and resources are offloaded to neighboring SBSs. However, adapting the sleeping strategy will affect the lifetime of the electronics of the SBSs, due to the frequent change of power level between turning ON and OFF the SBS. Therefore, in order to maximize energy savings, we formulate an optimization problem that provides an optimal user association and SBSs sleeping strategy for the entire network, while minimizing the total numbers of the switching of SBSs ON and OFF.

Furthermore, an other approach consider is the deactivated SBSs are equipped with two power sources, a harvested energy (HE) source and a grid power source, where first the SBS will use its available HE to serve the associated users. Then, the SBS will request any shortage of its energy from other active or deactivated SBSs which have surplus of HE. Finally, if there is still shortage in energy, the SBS will use the power drawn from the grid. However, since the formulated problem is a Mixed Integer NonLinear Problem (MINLP), Generalized Bender Decomposition (GBD) is proposed to decompose the problem into two

subproblems. Moreover, a new heuristic approach is proposed to provide a computational efficient algorithm to solve and optimize the user association and energy harvesting of the system model.

A new UEs' prediction method is introduced to provide a future information for the model and to apply an accurate designing parameters. This method is based on a combined approach of Non-linear Autoregressive with External input(NARX) and probabilistic Latent semantic Analysis (pLSA) to provide accurate prediction for multiple steps.

Therefore, we consider integrating the powerful Machine Learning (ML) techniques to provide solution of the system model with less computation demands. Thus, we introduced an efficient less complex approach that is based on synthetically generating data from the optimization problem and employing it to train and configure an Artificial Neural Network. An extensive simulation results are presented to show the effectiveness of our approaches in comparison to the optimal results.

CHAPTER 1. INTRODUCTION

In this chapter, background information about the 5G technologies, small cells and green communication and energy efficiency in HetNets will be presented. Also this chapter discusses the emerging new methodology of machine learning and its application in wireless communication networks.

1.1 5G Networks

Over the past few years the cellular mobile communication technology has exponentially expanded from 2G with Small Messaging Service (SMS) to the video streaming capabilities of the 4G[1]. The main motivation behind this evolution is the rise of the data demands, where the data demands grew from around $7,000$ PB (Petabytes) per month [2] to $201,000$ PB per month and the numbers are predicted to grow more in the near future to reach around 400 thousands PB per month [3]. In addition, other types of requirements , e.g., ultra low latency, higher bandwidth per unit area, higher number of connected devices and more coverage, started to rise and require considerations. Moreover, new wireless applications keep on emerging and need to be included within the next mobile communication generation. Applications like Device-to-Device (D2D)communications, Machine-to-Machine (M2M), Internet of Things (IoT), Smart cities, Health care systems and automation are emerging with needs of more robust Quality of Service QoS and scalability. The legacy systems, unfortunately are not capable of matching these new and growing demands. Therefore, 5G includes new technologies that are able to accommodate these new demands. 5G presents abroad new technologies that are developed to overcome the drawback of the previous generations.

1. **Massive Multi-Input Multi-Output (Massive MIMO):** Massive MIMO is a key to the networking future that 5G is adapting in its structure. Massive MIMO is a multiuser MIMO where the number of antennas at the BS is much larger than the antennas at the device [4]. The concept of using large number of antennas at the BS is to simultaneously serve many devices at the same time. The main characteristics of Massive MIMO is the fully digital processing, where each antenna ha its own Radio Frequency RF and digital baseband. This gives Massive MIMO a superior performance compared with MIMO. The two main benefits of adapting Massive MIMO that are desirable in the 5G are the spectral efficiency which is achieved by spatial multiplexing, and energy efficiency, which the array gain permits a significant reduction in the transmission power [5]. However, a clear Channel State Information CSI is required to exploit the full advantage of Massive MIMO.

2. **millimeter Wave (mmWave)** mmWave is the waves with bandwidth that is ranging from 30GHz to 300GHz. mmWave has been introduced as a part of the 5G networks to provide high data throughput for short range application. The mmWave range (30GHz to 300GHz) can offer an abundance of spectrum that can be utilized in providing the required high demands of the 5G. Other advantage of mmWave is its short wavelength which helps install large number of antenna elements into a small area. This helps expoilting massive MIMO at both the BS and the devices [6].The high frequency of mmWave suffers from high propagation loss due to the rain attenuation compared with microwaves which limits the range of the mmWave[7]. However, with the emerging of small cells and indoor usage the rain attenuation will have less effects on the performance of the mmWave. Therefore, mmWave communications are perfectly suited for indoor environments and small cells of size on the order of 200m[8].

3. **Cognitive Radio Networks (CRN):** CRN is another promising technology in the 5G networks that is dynamically exploit the available spectrum. The spectrum sub 6 GHz is a crowded bandwidth with all the licensed application that use it to com-

municate. However, even though the sub 6 spectrum is totally licensed, CRN allow users without license to exploit the under utilized bandwidth opportunistically [9]. CRN technology assumes the existence of two types of networks, the primary networks (PNs) that serve primary UEs (PUEs) and secondary networks (SN) that serve secondary UEs (SUEs). In this topology the PNs have the priority in exploiting the licensed spectrum while the SNs are exploiting the licensed spectrum as long as they does not cause disturbance in the PUEs connection [10].

4. **Small Cells:** Spectrum reuse has been used as an approach to accommodate more demands. This can be applied using another new technology that is able to reuse the spectrum to its full exploitation. Small BSs are new approach that are used to accommodate the high demands and are discussed in the following section.

1.2 Small Base Stations in Heterogenous Networks

As state above, the demand for higher throughput in mobile broadband communications increases dramatically every year, wireless carriers must be prepared to match this growing demand. According to Cisco's Visual Networking Index [11], the cellular networks might have to deliver to users 1000 times the current throughput. There is a requirement of rearchitecting the wireless networks from traditional macro base stations (MBSs) covering wide areas, e.g., 10s of kilometres, to the introduction of much smaller coverage small base stations (SBSs). These typically include femtocells, with each covering 10s of meters and are deployed in an ultra densed manner. Therefore, one of the promising technologies of the next generation cellular systems (i.e., 5G) is the SBSs, where the smaller coverage of the SBS results in higher spectrum reuse rate which increases the network capacity and provides the required data rates. Furthermore, Base Stations (BS) with smaller area coverage ensures the reduction in the number of users competing for resources at every BS [12].

The SBSs are small, low energy, cost effective and self-configurable BSs. The SBSs are introduced to solve the problems that arise by the increasing demand for higher data

rates. Therefore, they are seen as an alternative to MBSs for better coverage, quality of service (QoS) and energy efficiency. The SBSs are proposed to operate within the coverage of the MBSs as an additional layer in HetNets where users with higher QoS demands can be offloaded from the MBS to the nearby SBSs. Moreover, the SBSs are typically installed in densed areas like malls, offices and crowded parks where they provide cellular services as closed, i.e., serving only the owners of the BS, or open, i.e., serving all subscribers, as configured by the service providers [13]. Thus, the benefits from introducing SBSs in the cellular networks are: First, higher signal strength due to shorter distance between the user and the SBS; second, lower transmitted power and interference mitigation and, third, fewer associated users lead to better distribution of the resources between the users [14].

However, the deployment of a large number of SBSs requires more attention since it can introduce more complications to the wireless networks in general. One problem that is caused by deploying the large number of SBSs is the interference that each SBS causes to others. The interference is a problem in HetNets since the SBSs are deployed without preplanning and in a decentralized manner. Thus, a power management is necessary to overcome the system degradation that may occur as a result of high interference in the wireless network [15].

1.3 Green Communication and Energy Efficiency in HetNets

The global temperature has been steadily rising through the last half century which is mainly due to the concentration of carbon dioxide (CO2) in the atmosphere. The rise of the global temperature encouraged scientists to work on new means to reduce the greenhouse gas emissions (GHG) that cause the concentration of CO2. The information and communication technology (ICT) is one of the contributors of the GHG emission that is not widely discussed on the media. Furthermore, the ICT sector is currently growing and developing fast that the GHG emission from this sector is going to double within a decade [16]. A study conducted by [17] showed that the ICT GHG emissions contribution could grow from

around 1.6% in 2007 to more than 14% of the 2016-level worldwide GHG emission by 2040, which accounts for more than half of the current contribution of the transportation sector. Thus, the deployment of the SBSs in large numbers contributes to increasing energy consumption in the whole network. The SBSs are introduced as an energy efficient BS compared relatively to larger BSs. However, cellular networks with densed SBSs deployment can be a challenge to energy efficient networks due to the increased number of SBSs, which adds to energy consumption [18]. Therefore, introducing new technologies to minimize the energy consumption is an important and active research topic.

The power consumption for every BS (including SBSs) includes three components [18]:

- **Transmission Power**: corresponds to the amount of energy used to transmit the signal from the BS to the users, which includes the power amplifier (PA) and signal modulation from the baseband to the wireless radio signal.

- **Computational Power**: which represents the energy consumed by the basic circuitry of the BS, that includes the digital signal processing function, the control functions, the management functions and the backhauling communication.

- **Other Types of Power**: corresponds to all other energy consumption that are not included into the Transmission and Computational types. More specifically, the energy consumed for cooling and energy lost to a non-ideal power transfer between the power grid and the BS power supply. Also, on the receiver end, the power consumed at the receiver circuit.

There are many new approaches to reduce the energy consumption of the BS in the literature, and two of the well known approaches are the sleeping strategy, where a SBS without any associated users is then turned off, and energy harvesting, where the SBS supports itself by harvesting energy from a renewable energy source (e.g., solar, wind... etc).

1.3.1 Sleeping Strategies in SBS

The BS sleep control has been considered as an effective approach for minimizing the BS's energy consumption. The data traffic fluctuates considerably during different times of the day and at different locations. For example, high traffic demands originate from the downtown areas or business districts during weekday mornings in comparison to weekend mornings, where the traffic is very limited. This fluctuation in data demand provides the opportunity to turn off under-utilized BSs to save energy. The sleeping strategy is an active research area due to its great potential, thus, many efforts have been dedicated to designing different sleeping strategies for different network scenarios [19]-[20]. However, as for next generation networks with their heterogeneous architectures, designing a sleeping strategy faces many challenges, which can be summarized as follow:

- **Applicability to New Technologies**: The next generation cellular network (i.e., 5G) is considering several new technologies, e.g., device-to-device (D2D), opportunistic and unlicensed user communication, unlicensed bands and closed/open SBSs. All these new technologies lead to difficulties in designing effective sleeping strategies. For example, the sleeping strategy of a closed SBS which serves a limited number of registered users

- **SBSs Dense Deployment**: The SBSs deployment is considered to be dense where a large number of SBSs are deployed in relatively small areas. The large number of SBSs increases the complexity of solving any proposed design. For example, solving the user association optimally requires deciding on large binary decision variables, which takes exponential time to solve and this makes it impractical for use in a time efficient manner.

- **Frequent Fluctuation in Data Demand**: Due to the small coverage of the SBSs, the user association is required to be performed over shorter periods (in seconds) even for slow moving users. This results in high fluctuations in the data demands from the

users of every SBS, where in every time slot the number of users associated with it changes greatly. The changes over short times require designing fast algorithms that can accommodate the fast changes.

- **High ON-OFF Frequency**: The sleeping strategy is very effective in saving energy in cellular networks. However, the frequent switching of devices ON and OFF has a negative impact on the SBS hardware lifetime [21]. In [22], the authors discuss the two parameters that impact the lifetime of the SBS. The first parameter is positive, and is the duration of the sleeping time (the longer the SBS is deactivated the better), while the second one has a negative impact which is the frequency of the ON/OFF on the lifetime (higher switching frequency causes a reduction in the lifetime). Therefore, considering the sleeping strategy only to build an energy efficient cellular network may not be the best approach to reliably operate networks, as the frequent power change may decrease the lifetime of the SBSs and cause more failures to the network.

An important feature of the SBSs is the unplanned deployment, such as the users plug-and-play deployment, in contrast of the MBSs where the service provider deploys them according to careful planning and dimensioning strategies.

1.3.2 Energy Harvesting in SBS

Renewable Energy (RE) is one of the clean energy sources in next generation networks to ease the effect of climate change. Since most of the energy consumption in the cellular networks caused by the BSs (from 75% to 80% according to [23]), reducing the energy consumption at this level is essential for energy efficient networks. Energy Harvesting is a promising solution for minimizing the energy consumption of a BS. The BSs can harvest energy from ambient environment, such as solar power, winds and Radio Frequency (RF). The Harvested Energy (HE) could partially sustain the needs of the BS and for the case of SBSs could be fully sustained by HE [24]. Suggesting HE to power MBS in cellular networks was proposed in 2001 in[25]. Nevertheless, according to [26] the renewable energy

has not been fully exploited in cellular networks due to economical reasons. However, with the dense deployment of the SBSs, installing HE parts could provide the network with a reliable source of energy.

Using HE in cellular networks could provide many benefits for customers, service providers and the environment. As stated before, the cellular networks contribute a considerable percentage of the GHG emissions, thus, HE could provide a great source of environmental friendly energy. Also, HE could be installed in BSs on rural areas where there is no power grid and are powered by diesel generators. As for service providers, HE is one method of minimizing the cellular network's daily OPEX. Also, HE could provide a backup power source in case of grid failures [27]. Furthermore, the On the other hand, the advanced technology of smart grids (SG) made energy cooperation between wireless networks different components feasible. The concept of SG can be regarded as an electric system that uses information and two-way power flow in an integrated fashion to achieve an efficient and sustainable system [28]. Exploiting such technology could provide enormous opportunities for wireless networks. One approach is by utilizing the SG to transfer harvested energy from one BS to another, with high transfer efficiency.

1.4 Machine Learning in 5G Networks

ML is a promising enabler for solving complex, dynamic and time-limiting problems in an efficient time. The ML is evolving from a mere pattern recognition tool to govern complex systems through learning processes [29]. The way ML works is that an algorithm running on the machine learns to execute a particular task based on historical knowledge of the behaviour of this task, then the machine aims to improve its performance while executing the task. ML can be widely employed in modeling new emerging problems of 5G systems, such as massive MIMO, D2D networks, HetNets with small cells[30].

In this thesis we focus on applying ML to two main problems, mobility prediction and ANN in optimization.

1.4.1 Mobility Prediction

One of the great advantages of wireless networks compared with wired networks is its ability to serve moving UEs without any loss of the quality of service. Therefore, knowledge of human mobility is essential in assuring the quality of service required in communication networks. In [31], the authors inspected the movement of $50,000$ individuals to study their trajectories in a limited time frame. By measuring the entropy of each individual's movement pattern, they found a %95 potential predictability in user mobility across the whole set of individuals that are under study. This means that users tend to follow predicted patterns and lack the variability in predictability. This predictable pattern of movement gives a great advantage for wireless networks operators to design the wireless networks to accommodate this predictable pattern.

Mobility prediction has two main ways of prediction: Handover (HO) prediction and location prediction. The HO prediction focuses on the aspects of the UE switching from one BS to another. HO prediction is one of the most widely used prediction methods in wireless communication networks, since it allows a proactive approach in resource allocation. Thus accurate HO prediction is a significant method in reducing the latency and call drop[32]. However, in SBS dense networks the UE's location is more important than the mere HO between two BSs. The HO in a dense network is more challenging in prediction aspect and does not provide the essential information that is needed in designing the other networks requirements. The power management is a key aspect in modern HetNets, where BSs can change their transmission power according to the level of interference, UE's location and UE association. Therefore, the location prediction provides more valuable information that helps improve the network's performance compared with HO prediction.

1.4.2 Artificial Neural Networks (ANN)

ANNs are promising computational tools that are developed to solve many complex problems. The advantage of the ANN comes from its ability of universally approximating

any input-output relation of any given system [33]. ANNs are defined as a structure that is comprised of interconnected simple nodes (called neurons) that are capable of performing simple tasks. The combination of these simple neurons gives a massive computational capability in processing the data [34]. ANNs have a wide range of applications, from computer science to medicine, engineering and statistics. Figure 1.1 shows the topology of typical ANN. The most important types of problems that ANN shows its power are the following:

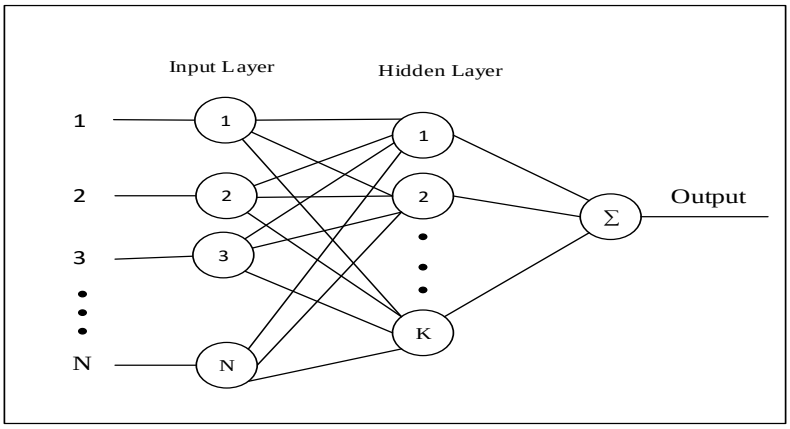

Figure 1.1 Artificial Neural Network Topology.

1. **Classification:** The task of ANN in classification is to assign any data represented by its features as an input to one of a set of predefined classes. Speech recognition and image recognition are well know applications of this type.

2. **Clustering:** Clustering is a kind of classification with unlabeled input. It is known as unsupervised learning since the training data has no labels. In clustering, ANN explores the similarities between the input data and group them in one cluster.

3. **Function Approximation:** In many engineering real-world problems, the mapping between certain inputs and outputs is unknown. The task of ANN in this application is to find an approximation function that can estimate the relation between the input and the output.

4. **Time Series Prediction:** This task is used in forecasting future data according to a time sequenced data input. ANN is trained to predict the pattern that the time series follows in order to generate a forecasting of where the series is heading in the future. Stock markets, weather forecasting and human mobility are some of the applications that use this type.

ANN generally consists of layers, neurons and weighted connections. The layers are of three types, input layer, hidden layer (or layers), and output layer. The input layer is the layer that receives the external input and usually has a number of neurons that is equal to the number of the input data. The layer that produces the final results is the output layer which consists of one or more neurons according to the specific application of this ANN. The layer or layers in between are called the hidden layers, with numbers of the neurons that differ from layer to layer according the requirement of every design. Each neuron in the hidden layers is equipped with an activation function. The hidden layers take inputs from the input layer and multiplies them by weights and produces an output through the activation function.

The most important unit in ANN structure is the activation function. Activation functions are used to decide if a neuron should be active or not according to the value of the input multiplied by the assigned weight [35]. The most important activation functions are listed below:

1. **Logistic Sigmoid Function:** This is a conventional activation function that has a mathematical representation of $f(x) = \frac{1}{1+\exp^{-x}}$. The function is differentiable and monotonic and the output of the function is always between 0 and 1. However, the function's values towards the ends are almost horizontal, i.e., it responds less to changes to the input values, which means the gradient in these regions is small or has vanished. Another problem with the sigmoid function is that its mean value is non-zero which induces singular values in the Hessian, which causes a slower learning time [36].

2. **Hyperbolic Tan:** Similar to logistic Sigmoid function, Tanh is an S-shaped function and differentiable its mathematical representation is $f(x) = \frac{\exp^{2x}-1}{\exp^{2x}+1}$. Tanh has an advantage over logistic sigmoid that its output ranges between -1 and -1 instead between 1 and 0 in logistic sigmoid case. This results in a zero mean that can speed up the learning time.

3. **Rectifier Linear Unit (ReLU):** is the most popular non-linear functions [37]. It is simply the half wave rectifier $f(x) = max(0, x)$. ReLU's popularity comes from its advantage of faster learning in networks with multilayers which creates sparse neurons[38]. However, due to horizontal line for negative x, the gradient in this region will be 0, which means the weights for these neurons will not be adjusted. This problem can be solved using Leaky ReLU, which is a modified version of ReLU that the negative part is not horizontal line but in fact is a linear line with a very small slope. Leaky ReLU has the following mathematical representation:

$$f(x) = \begin{cases} x & x \geq 0 \\ .01x & \text{otherwise} \end{cases}$$

The learning process in neural networks is carried out using a backpropagation. Backpropagation is the algorithm that is deployed to carry the learning process back to the neurons in neural networks. The learning problem is formulated as a minimization of the loss function, which measures the performance of the neural network on a data set. The loss function is composed of two parts, the error function and the regularization term. The error function evaluates the neural networks performance. The regularization term is to prevent the ANN from overfitting. There are many algorithms that are used in performing the backpropagation process, and the most popular algorithm is Levenberg-Marquardt which is designed to work with loss functions.

1.5 Thesis Contribution and Scope

The main theme of this thesis is green wireless communication. In the green wireless communication paradigm we applied new approaches to help reduce the energy consumption in wireless networks. Since most of the consumed energy in such networks comes from the BSs, we propose new ways in reducing the BSs energy consumption without affecting the UE's quality of service. In 5G, the networks will consist of large number SBSs that will increase the energy consumption and increase the complexity of the network. Thus, our new approaches will help reduce energy consumption in complex wireless HetNets. First, we propose a new sleeping strategy where the system will minimize the number of active SBSs and reduce the frequency of owering SBSs ON and OFF. In order to maximize energy savings, we formulate an optimization problem that provides an optimal user association and SBSs sleeping strategy for the entire network, while minimizing the total numbers of the switching of SBSs ON and OFF.

A second approach for reducing the energy consumption in SBSs is developed in which, we introduce a second source of energy, specifically, renewable energy. In this approach every SBS draws its power from two energy sources, a grid power source, and a harvestable energy source. Combining the harvested energy with the sleeping strategy results in reducing the energy drawn from the grid. We investigate the grid energy minimization problem by optimizing both the harvested energy transfer from one SBS to another and activation/deactivation (Dynamic Sleeping) of the SBSs. Furthermore, new UEs' movement prediction method is introduced to provide a future information for the model and to apply an accurate designing parameters. This method is based on a combined approach of Non-linear Autoregressive with External input(NARX) and probabilistic Latent semantic Analysis (pLSA) to provide accurate prediction for multiple steps.

Finally, due to the complexity of the formulated problem, we investigate other ways of solving the optimization problem. First we introduced a new method by using network centrality to develop a new metric, Base Station Centrality (BSC), of the SBS centrality

in the network. BSC is presented to mark the SBSs that have the most potential to be deactivated without affecting the QoS of users. Second, we consider integrating powerful Machine Learning (ML) techniques to provide solution to the optimization problem, with less computation demands. Thus, we introduce an efficient and less complex approach that is based on synthetically generating data from the optimization problem and employing it to train and configure an Artificial Neural Network.

1.6 Thesis Organization

The rest of the thesis is organized as follows:

Chapter 2 discusses the most recent related works. Chapter 3 introduces the a new method of lifetime prolonging and femto base station's sleeping strategy in HetNets by reducing the frequent change of power level between turning ON and OFF the SBS. Chapter 4 presents a cooperative small cell HetNet that employs both sleeping and energy harvesting, where renewable energy source is introduced into the network in order to minimize energy consumption. Chapter 5 proposes an enhanced cooperative small cell HetNet by integrating an ANN to provide a solution of the optimization problem with less computations demands. Chapter 6 concludes the thesis and outlines its main contributions. Some potential open problems and possible future works are also presented in this chapter.

CHAPTER 2. LITERATURE REVIEW

2.1 Small Base Stations in Heterogenous Networks

There is some recent work which focuses on energy efficient SBS networks. In [39] the authors consider minimizing the energy consumption by turning off the MBS and offload the users to SBSs or neighboring MBSs. Since the MBSs consume much more energy than SBSs, it is reasonable to turn the MBSs off. Nevertheless, SBSs are being deployed in large numbers, therefore, deactivating some of the deployed SBSs can reduce the networks energy consumption greatly. In [40] the authors proposed an energy efficient scheme that minimizes the total power consumption in SBS networks. However, the authors considered minimizing the power, not the ratio of power and rate, which results mostly in providing the users with the minimum required rate. The authors in [41] consider a two tier network, which consists of an MBS underlaid by SBSs. They also considered minimizing the ratio of the energy to the rate in order to maintain quality of service. However, the authors were concerned with optimizing the transmission power only, i.e., the authors did not consider minimizing the energy by sleeping strategy for either the SBSs or the MBS. Similarly, in [42] and [43] the authors considered two-tier networks, where MBS/SBS co-exist, they also consider maximizing the ratio of the system throughput and the energy consumption. However, the authors did not take into consideration the SBSs basic circuit energy consumption. While in [21] the authors considered the energy efficiency and lifetime of the BS, they did not consider the quality of service requirements of users.

On the other hand, several researches have dealt with powering cellular BSs with renewable energy sources. In [26] and [44], the authors highlighted the importance of combining

renewable energy systems and the smart grid for developing an energy efficient cellular network. In [45], the authors formulated a constrained optimization problem in order to minimize the total cost incurred by the Cellular Networks operators by harvesting and transferring the energy through the SG. Additionally, the authors of [46] and [47], used the dynamic sleeping to activate and deactivate BSs in order to minimize the energy drawn from the grid. In [48] they formulate an optimization problem for the system, and due to the problem's NP-hardness, they proposed a greedy decomposition to tackle the problem. On the other hand, the authors of [49] considered a model where the small BSs are powered solely by harvested energy, and minimized the grid energy by optimizing the Macro BS active probability and Small BSs transmission power. In [47], the authors considered a system of Cognitive Radio (CR) and formulated a constrained optimization problem to maximize the throughput by optimizing the power allocation from the renewable energy and smart grid.

Moreover, the authors in [50] considered the stochastic process of energy harvesting of the remote radio heads (RRHs) to develop an online resource allocation algorithm, which maximized user utility while ensuring the sustainability of each RRH in Cloud-RANs.

2.2 Mobility Prediction in HetNets

Mobility prediction as part of the wireless networks design has attracted attention from academia and industry. The main concept of the mobility prediction is that giving the current and previous locations for a unique UE, what will be the next location or locations for that UE. Such predictions help optimize the wireless networks resource allocation and increase its performance. There are many research done in predicting the UE mobility. Works [51]-[52], are based on Markov Chain, which is easy to implement but can suffer from overfitting for low data. The authors of [53]-[54], are proposing mobility prediction based on Hidden Markov Model (HMM), which is more accurate than Markov chain and is able to find more complex relation between different patterns. However, HMM requires

more complex structure and higher computational capacity. The works of [55]-[56], apply the ANN to predict users' mobility by learning their inherent characteristics. However, ANN has a huge computational expense during the training process and without proper validation it can suffer from overfitting.

Authors of [51] investigated Markov chain to predict the users movement in SBSs, specifically in Femto BSs, where their results shows the prediction accuracy is affected by the regularity of the user's movement. The authors used the generated historical database for the users to discover mobility pattern and improve the prediction performance.The authors of[57] is using an enhanced Markov chain algorithm to predict user mobility by introducing an algorithm that is composed of two components: Global Prediction Algorithm (GPA) and Local Prediction Algorithm (LPA). IF GPA fails when the cell does not exist in the training database, then LPA is used. In [52], the authors employed Markov chain user prediction in proactive caching in anywhere in the network rather than just at the edge. The authors discussed a system where vehicles are connected to Roadside Units (RSU), can be BSs or Access points (APs), to allow them to connect to the internet backbone. In order to proactively cache data into the RSUs, the authors presented a mobility prediction based on Markov chain to predict the vehicle's next RSU. In [58] the authors proposed a mobility prediction based on Markov Chains to predict the users trajectory to minimize the interruption time when the handover is triggered.

In [53] , the authors adapted HMM as mobility prediction model to forecast the next AP the user connects with. The authors proposed a location awareness AP selection algorithm to improve the number of connection to AP with better signal quality compared to random handover. The results shows the superiority of the system performance that is using the HMM in comparison with the random movement. In [59], the authors combined the HMM with pLSA to improve the probabilistic prediction performance. The authors implemented a history-based Expectation Maximization algorithm and GPS trajectory prediction to improve the performance. Their results show successful prediction with as low as 2.2%

inaccuracy. In [54], the authors constructed a hybrid Markov that takes the non-Gaussian spatio-temporal characteristics of human data into accounts. The authors improved the prediction accuracy of the Markov models by overcoming their drawbacks i.e., time independency , high memory cost and failure to predict new locations. The authors also presented an algorithm that discovers the mobility patterns to help increase the prediction accuracy. The results show a prediction accuracy of more than 56%. The authors in [60] combined the space-partitioning with their model of the frequent regions to predict trajectory patterns using Hidden Markov Models. In [61], the authors used a mixture model that uses the individual data to generate broader patterns (e.g., to work, to home... etc.).

The authors of [55], proposed a new algorithm based on Recurrent Neural Networks that dynamically selects the BS to form virtual cell topologies. The authors built a simple model to effectively select the next BS the user will probability connect to. The simulation results show an accuracy of 98% to predict the optimal virtual cell topology. The model does not predict the exact user's location but instead it consider the Received Signal Strength (RSS) to make the prediction. In [62], the authors proposed a handover prediction method based on NARX model. The authors formulated a mathematical model to solve the handover decision problem assuming the mobility scenario is known. The optimal solution for the mathematical model is then used to train the NARX to be able to automatically make the handover decision based on the RSS. The results show that the new model outperforms the threshold-based handover in terms of delay, which results in abetter Quality of Service. The authors of [63], discussed the different training algorithms that are used in training NARX network to predict the user's GPS location. The authors employed NARX to predict the GPS location for users in case of the GPS signal is lost. The results show that Levenberg Marquardt algorithm outperformed Conjugate Gradient with Flectcher Reeves updates in case of the user's height. In [56], the authors proposed a routing scheme, in which the controller uses the trained ANN to predict the vehicle arrival rate without continuously monitoring the vehicle locations. Based on the prediction of the arrival rate the BS can

estimate the traffic by building the traffic model and then make the routing decision. The simulation results show that the proposed scheme outperforms others in terms of delay.

2.3 Deep Learning (DL) in Wireless Networks

Deep learning has a broad range of applications in wireless networks that can contributes in designing and optimizing the performance. According to [64], deep learning can contributes to the wireless networks in many applications, where some of the advanced areas are: Firs, DL in network level, e.g., traffic classification, network prediction and resource allocation. Second, DL in user mobility analysis where DL is used in predicting user's movement and user patterns. Third, DL in network control such as network optimization, routing and scheduling.

In [65], the authors investigate the effects of the network's metrics such as user throughput , number of active users in a cell, data per user and channel status in user's Quality of Experience QoE. The authors employ ANN to predict the users' QoE as the network's metrics changes, where the results show high prediction accuracy. The authors of [66], developed a network traffic prediction method based on a deep belief network and a Gaussian model. The model is designed to predict the precise estimate of the network distributions. In [67],[68] the authors presented a city wide scale prediction based on modified ANNs. The authors of [67], propose a prediction on a city scale that deploy a deep Spatio-Temporal neural Network (STN) that exploits important relations between user traffic patterns at different locations and times, to achieve a precise prediction for the mobile traffic. In their work, they combined the Convolutional Long Short-Term Memory (ConLSTM) and 3D-Convolutional Networks to model long-term trends and short-term variations of the mobile traffic volume, respectively. Such scheme can take advantage of both models by using two layers that fuse the features extracted by each model from previous traffic measurements. In [68], the authors proposed a mobile traffic super resolution technique that reduces the complexity of measurements and analysis. A modified Generative Adversarial neural

Network(GAN) architecture is presented to infer fine-grained mobile traffic patterns, from aggregate measurements collected by network probes.

In traffic classification, the authors of [69] propose an end-to-end method of encrypted traffic classification with one-dimensional Convolution Neural Networks (CNN). One-dimensional CNN is trained by feeding it with the traffic features to learn to predict the encrypted traffic labels. The results in this end-to-end approach outperform the state of the art scheme that is employing divide and conquer. Moreover, in [70], the authors applied Self-taught Learning (STL) which is based on deep learning that helps detect network breaches. The results show that the Network Intrusion Detection System (NIDS) performed very well compared to previously implemented NIDSs for the normal/anomaly detection when evaluated on the test data. Similarly, the authors of [71], presents a novel deep learning technique for intrusion detection, which is based on nonsymmetric deep autoencoder (NDAE). Moreover, they proposed a deep learning classification model constructed using NDAEs and Random Forest learning. Combining the two approaches increased the performance of the classification since the analytical overhead is reduced.

In [72], the authors proposed a new idea of employing deep learning in wireless communication. The authors argue that the lack of considering deep learning in wireless communication is due to the strong theoretical model for their system design and optimization. However, the mathematical model in wireless communication became increasingly complex and consumes an ever increasing time in solving it. Thus, they propose combining the strong mathematical model in wireless communication and DL to produce a system that is capable of the near optimal solution for never observed inputs. This scheme use the accurate mathematical method in generating output for a given input, then this pair of input-output is then used to train ANN to be used for unobserved inputs. The results show good performance with relatively small computational burden.

CHAPTER 3. LIFETIME PROLONGING AND FEMTO BASE STATION'S SLEEPING STRATEGY IN HETNETS

3.1 Introduction

In this chapter, we introduce a holistic sleeping strategy for heterogenous networks by:

- Reducing the active time (ON time) of Femto Base Stationa (FBSs) to reduce the total energy consumption of the network.
- Relocating the released spectrum to other ON FBSs to reduce transmission power by transmitting the data using wider spectrum, hence further reducing energy consumption.
- Minimizing the switching frequency (the frequency of turning the FBS ON and OFF) to reduce the FBS's hardware failure probability on the long run.

In the proposed system, we mathematically formulate an optimization problem to optimize both the energy efficiency and the total switching frequency of all FBSs of the network. Additionally, due to the problems NP-hardness, we propose a heuristic clustering approach, where the network is divided into smaller disjoint clusters, which are solved independently. We use different clustering approaches and compare between their performance in terms of optimality and complexity.

3.2 System Model

We consider a heterogenous network, which consists of an MBS overlaid with many FBSs. The MBS is covering a large area including the area that is covered by the FBSs.

We assume that the FBSs are controlled by a central controller that the FBSs are backhauled through broadband connections. We consider a cellular network where the FBSs and users are distributed randomly within the coverage area of an MBS. We focus on the downlink communication to the users since it occupies most of the spectrum most of the time, we also assume that there is no interference between the MBS and FBSs, since they use different frequency bands in the downlink.

In the system model we consider the FBSs to be either open, open but owned by individuals, or hybrid. In open, the FBSs are owned and distributed in large scale in public areas (e.g., airports, malls, etc..) by the operator, where in open but owned by individuals FBSs are open but owned and operated by individuals. On the other hand, a hybrid FBS is an FBS that is individually owned and the owner allows part of the FBS's resources to be used openly.

Let $f = 1, ..., F$ denote the set of the FBSs in the macro cell coverage area, while $u = 1, ..., U$ denotes the set of randomly distributed users covered by the FBSs. Moreover, let $c = 1, ..., C$ denotes the available resources in the network. The system is slotted into $n = 1, ..., N$ slots, with τ as the duration of every time slot. We also assume that users associate with the FBS with the highest SINR. Furthermore, every user is associated with only one FBS but can use more than one channel to communicate with that FBS, depending on the bandwidth allocated to the user.

Figure 3.1 shows a two-tier network architecture, where we have an MBS covering a specific area. Inside the MBS coverage area we show a small area of FBSs that are underlaying the MBS. We assume that the MBS and FBSs are using OFDMA, therefore, the MBS will cause no interference to the FBS users, therefore. Hence, the cause of the interference to a certain FBS user will be other FBSs that are using the same channel.

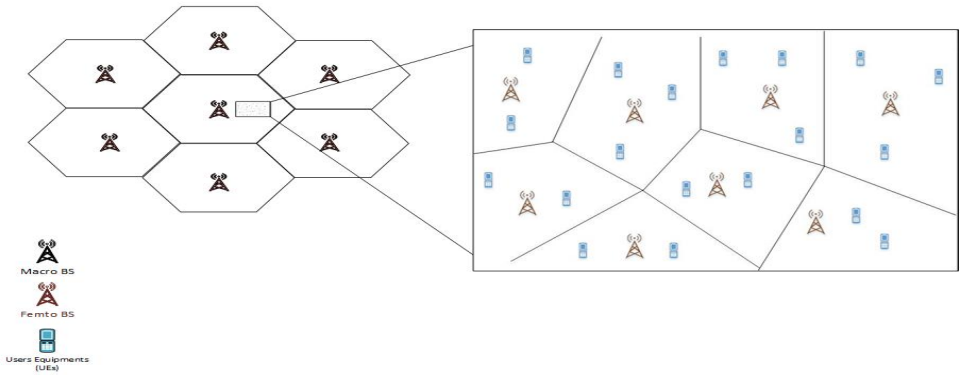

Figure 3.1 The Network Architecture.

3.2.1 Achievable Rate for All Users

Let $x^c_{uf}[n]$, be a binary indicator that is equal to 1 if channel c is used by user u and FBS f during time slot n, or 0 otherwise. Also, let $z_{uf}[n]$ be a binary indicator that is equal to 1 if user u is associated with base station f during time slot n, or 0 otherwise. Moreover, let $y_f[n]$ be a binary variable that represents the FBS on/off status, where $y_f[n] = 1$ if the FBS is on during time slot n, and $y_f[n] = 0$ if the FBS is off. Also, $p^c_{uf}[n]$ and $h^c_{uf}[n]$ denote the transmission power and channel gain from FBS f to user u using channel c during time slot n.

The interference at a user u which is associated with FBS f from all other FBSs are using the same channel c during time slot n will be:

$$I^c_{uf}[n] = \sum_{j \neq u}^{U} \sum_{i \neq f}^{F} p^c_{ji}[n] h^c_{ui}[n], \tag{3.1}$$

where $p^c_{ji}[n]$ and $h^c_{ui}[n]$ represent the power and channel gain for all other users which are connected to other FBSs with the same channel. Then, the signal to interference and noise ratio (SINR) at user u is:

$$\gamma^c_{uf}[n] = \frac{p^c_{uf}[n] h^c_{uf}[n]}{I^c_{uf}[n] + \omega\sigma}, \tag{3.2}$$

where ω is the available bandwidth for every channel, and σ is the channel noise spectral density which, is assumed to be Additive White Gaussian Noise AWGN. Thus, the data rate for every user using a single channel will be as follow:

$$R_{uf}^{c}[n] = \omega \log(1 + \gamma_{uf}^{c}[n]), \tag{3.3}$$

Hence, the total rate for every FBS is the summation of the aggregated rate for all users associated with the FBS:

$$R_f = \sum_{u=1}^{U} \sum_{c=1}^{C} \sum_{n=1}^{N} \omega \log(1 + \gamma_{uf}^{c}[n]), \tag{3.4}$$

Additionally, let the variable ϵ_f be defined as an indicator for switching between ON and OFF, and vice versa, for every FBS and is expressed as $\epsilon_f[n] = y_f[n] - y_f[n-1]$.

3.2.2 Energy Consumption Model For FBSs

The total energy consumption for every FBS is as follow:

$$E_f = \sum_{u=1}^{U} \sum_{c=1}^{C} \sum_{n=1}^{N} p_{uf}^{c}[n]\tau + \sum_{n=1}^{N} E_b y_f[n], \tag{3.5}$$

where E_b is the constant energy consumption for operating the FBS equipments. In (3.5), the first part represents the total transmission power, while the second part represents the basic energy consumption of the FBS.

Thus, the system energy efficiency, which is defined as the ratio between the total energy consumption to the total rate is given by:

$$\Gamma = \frac{\sum_{f=1}^{F} E_f}{\sum_{f=1}^{F} R_f}, \tag{3.6}$$

3.3 Problem Formulation and Solution

In this section, we model the proposed system as an optimization problem with QoS constraint. Then we proposed a clustering heuristic solution to the problem.

3.3.1 The Optimization Formulation of the Problem

Our goal here is to minimize the ratio of the total energy consumption divided by the total achieved rate for all FBSs. The minimization is subject to certain QoS constraints

which are defined by the minimum rates for each of the users. The formulation, which is a mixed integer nonlinear program (MINLP) is shown below:

Problem Γ :

$$\underset{p_{uf}^{c}[n],y_f[n],z_{uf}[n],x_{uf}^{c}[n]}{\text{Minimize}} \frac{\sum_{f=1}^{F} E_f}{\sum_{f=1}^{F} R_f} + \sum_{f=1}^{F}\sum_{n=1}^{N} |\epsilon_f[n]| \tag{3.7}$$

subject to

$$\sum_{f=1}^{F}\sum_{c=1}^{C} R_{fu}^{c}[n] \geq R_u^{min}, \qquad \forall n \forall\ u \tag{3.8}$$

$$\sum_{u=1}^{U} x_{uf}^{c}[n] \leq 1, \qquad \forall\ f, \forall\ c, \forall n \tag{3.9}$$

$$\sum_{f=1}^{F} z_{uf}[n] = 1, \qquad \forall\ u, \forall n, \tag{3.10}$$

$$\frac{\sum_{c=1}^{C} x_{uf}^{c}[n]}{Q} \leq z_{uf}[n] \leq \sum_{c=1}^{C} x_{uf}^{c}[n], \qquad \forall n, f, u, \tag{3.11}$$

$$\frac{\sum_{u=1}^{U} z_{uf}[n]}{Q} \leq y_f[n] \leq \sum_{u=1}^{U} z_{uf}[n], \qquad \forall\ f, \forall n \tag{3.12}$$

$$\sum_{u=1}^{U}\sum_{c=1}^{C} p_{uf}^{c}[n] \leq P^{max} \sum_{u=1}^{U}\sum_{c=1}^{C} x_{uf}^{c}[n], \qquad \forall f, \forall n, \tag{3.13}$$

where R^{min} is the minimum required rate for every user. P_f^{max} is the maximum allowed transmission power for every FBS. Also, $z_{uf}[n]$ is a binary variable representing the association between the FBS and user in every time slot. Q represents the total number of the available resource blocks.

Constraint (3.8) represents the quality of service (QoS) with a minimum service rate for every user. While constraint (3.9) guarantees that every channel is used at most once in every FBS, constraint (3.10) represents the users' association with the FBSs, where every user is associated with a single FBS and no user is allowed to associate with more than that.

Constraint (3.11) evaluates $z_{uf}[n]$, where if there is at least one user u that is associated with the FBS f, then $z_{uf}[n]$ is one, or zero otherwise. Similarly, constraint (3.12) captures the sleeping strategy of the variable $y_f[n]$, where if there is at least one user associated with the FBS f then this FBS is kept on, otherwise it will be turned off. Constraint (3.13) is the maximum power constraint.

Since the above problem is a Mixed Integer Nonlinear Problem MINLP, which is known to be NP-hard and there is no efficient algorithm with a reasonable time that can solve the problem, especially for large numbers of variables. Additionally, the absolute value in the objective function is causing the problem to be differentiable, while constraint (3.8) is causing the problem to be a non-convex, which will add to the complexity of the problem. Reformulating the problem is required by rewiring the absolute operator in a differentiable form. Finally, we suggest a clustering heuristic solution where we can divide the network into smaller clusters. Moreover, solving the clusters individually can reduce the calculation time greatly. The optimization problem will be as follows:

$$\text{Problem } \Gamma :$$

$$\underset{p^c_{uf}[n],y_f[n],z_{uf}[n],x^c_{uf}[n]}{\text{Minimize}} \frac{\sum_{f=1}^{F} E_f}{\sum_{f=1}^{F} R_f} + \sum_{f=1}^{F}\sum_{n=1}^{N} t_f[n] \tag{3.14}$$

subject to :

$$(3.8) - (3.13)$$

$$\epsilon_f[n] - t_f[n] \leq 0 \qquad \forall n \forall f \tag{3.15}$$

$$\epsilon_f[n] + t_f[n] \leq 0 \qquad \forall n \forall f$$

$$t_f[n] \leq 0 \qquad \forall n \forall f$$

3.3.2 The Clustering Approach

In this approach, we will divide the network into smaller disjoint clusters, where every cluster includes part of the available resources and part of the FBSs. Within a cluster, the neighboring FBSs will form a smaller network, where it will be dealt with independently

from other clusters. Therefore, the whole network is divided into many disjoint clusters that are independent from each other. Furthermore, the optimization problem (3.7)-(3.13), will be solved in every cluster independently.

The number of clusters is proportional to the network's size, i.e., if the network size increases, the number of the clusters increases. Dividing the network into clusters will help mitigate the optimization complexity. Since the problem's complexity increases exponentially with the network's size, then dividing the network into small clusters will keep the complexity under control.

Moreover, it is necessary to define a measure of similarity which will establish a rule for assigning FBS to a particular cluster. One such measure of similarity may be the Euclidean distance D between two points x and y defined by $D = ||x - y||$. The smaller the distance between x and y, the greater the similarity between the two and vice versa. There are several clustering techniques in the literature[73], K-means and Genetics Algorithm (GA) are two of the popular clustering approaches.

The K-means clustering approach is one of the simplest ways of clustering, which is used to cluster unlabeled data into K apriori known clusters. The algorithm works iteratively to assign every group to one of the K clusters based on the metric that was provided (e.g., Euclidean Distance). Every point is assigned to a particular cluster k according to $min||c_i - x||$, where c_i is the centroid of cluster i and x is data position. Hence, K-means will group neighboring FBSs into smaller clusters. We can then apply equations (3.7)-(3.15) to every cluster independently. However, since we will use a heuristic implementation of K-means and also since we do not treat all FBSs globally, our results maybe suboptimal.

Similarly, Genetics Algorithms can be used to cluster unlabeled data into K different clusters [74]. First, Genetic Algorithm involves a population of candidate solutions. Each individual of the population represents a clustering of the data (i.e., set of centroids). Then, data points is mapped to their nearest centroid. In each iteration the best solutions are selected for the next generation, and the rest is replaced by a new solution generated in

the crossover phase. Finally a mutation is performed to obtain a new candidate solution [75]. In general, Genetic Algorithms perform better than K-means, where it searches the entire space and is not affected by the initial points as in K-means. However, K-means converges faster than genetic algorithms. Therefore, in the results section we will compare both algorithms with the optimal approach and with each others.

3.4 Simulation Results

We evaluated the performance of the system by using computer simulation. In the simulation we used General Algebraic Modeling System (GAMS) to run the model. In GAMS, we used the Couenne solver, which is an open source solver for MINLP[76]. In the simulation experiments we have randomly distributed the FBSs and the users using a uniform distribution with maximum transmission power P_{max}=0.3 Watts for every FBS, and the constant energy consumption for operating the FBS equipments E_b=3 Watts for every τ = 1sec. The channel's bandwidth $\omega = 10MHz$. Moreover, for quality of service purposes we assumed $R_{min} = 1Mbps$.

Figure 3.2 represents a comparison between the optimal solution and the clustering approaches (K-means and GA), where we have the number of FBSs ranging from 3 to 12 with 24 users with $N = 20$ time slots. The maximum cluster size is 7 FBSs. As we can see from the figure, the performance of the clustering approaches is very close to the optimal solution. Moreover, with a small number of FBSs the optimal and the clustering approaches result in identical performance since the network size is equal to the cluster size. However, as we increase the network size, we start to see a small difference between the the optimal solution and the clustering approaches. This difference is due to the fact that in the optimal solution, we have a global view of the network, while in the clustering approaches we have a partial view of the network. However, the clustering approaches computation time is reasonable (up to two minutes compared to multiple hours for the optimal approach).

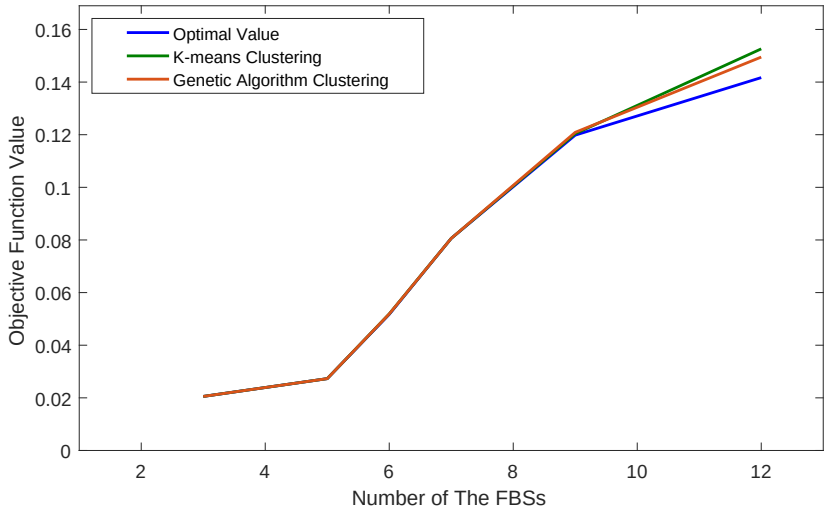

Figure 3.2 Comparing the Optimal with the Clustering.

In Table 3.1 we compare the two algorithms (K-means and GA), for the objective function value and the computational time for the algorithm to converge. Hence, as the table shows, the K-means is much faster than the GA with a very small difference in the optimal value. Therefore, using K-means algorithm is more reasonable, where the network adapts much faster than the GA without any significant loss in the energy efficiency.

Figure 3.3 shows two scenarios of the system behavior, the first scenario is without including ϵ in the objective function, which allows the FBSs to switch On and OFF freely, while the second scenario is including ϵ in the objective to limit the total number of switching ON and OFF. As the figure depicts, using ϵ on the objective function will reduce the frequent switching significantly, while the increase in energy consumption is negligible, especially for fewer FBSs. On the total number of ϵ, we are summing the total number of switching in the total network i,e,. $\sum_{f=1}^{F} \epsilon_f$. Thus, scenario 2 is outperforming scenario 1 with the total number frequent switching.

In Figure 3.4 we evaluate the achievable total rate using the K-means clustering approach for different numbers of users. As the figure depicts, the rate increases as the number of available FBSs increases. This is understandable, since the P_{max} is limited, then as the coverage of the FBS decreases the achievable rate increases. Moreover, as the number of

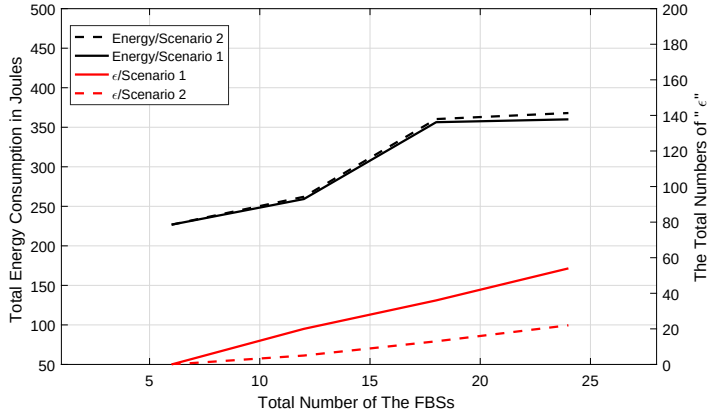

Figure 3.3 Comparing the Two scenarios, with and without ϵ.

Table 3.1Comparison Between K-means and GA for Different Numbers of FBSs

No. of FBSs	Obj. Fun. Value		Computational Time	
	K-means	GA	K-means	GA
24	0.238	0.234	1.5 sec	1.5 min
32	0.341	0.336	1.5 sec	1.5 min
40	0.366	0.361	2 sec	1.5 min
48	0.38	0.374	2.5 sec	2 mins

the users in the network increases the total aggregate rate in the network increases.However, as the number of users increases, the incremental increase in the total rate decreases, and the users receive rate closer to R_{min}.

Similar simulation parameters as in Figure 3.2, in Figure 3.5 we evaluate the energy consumption of the available FBSs of the K-means clustering approach. In this figure we can see that the energy consumption increases as the number of the FBSs increases. This increase in energy consumption is due to the increase of the number of FBSs, which will provide higher rate, and consequently consume more energy on their hardware. On the other hand, for fewer users scenario, the energy consumption will not increase as the FBSs number increases, which this is due to the fact that more FBSs are turned off.

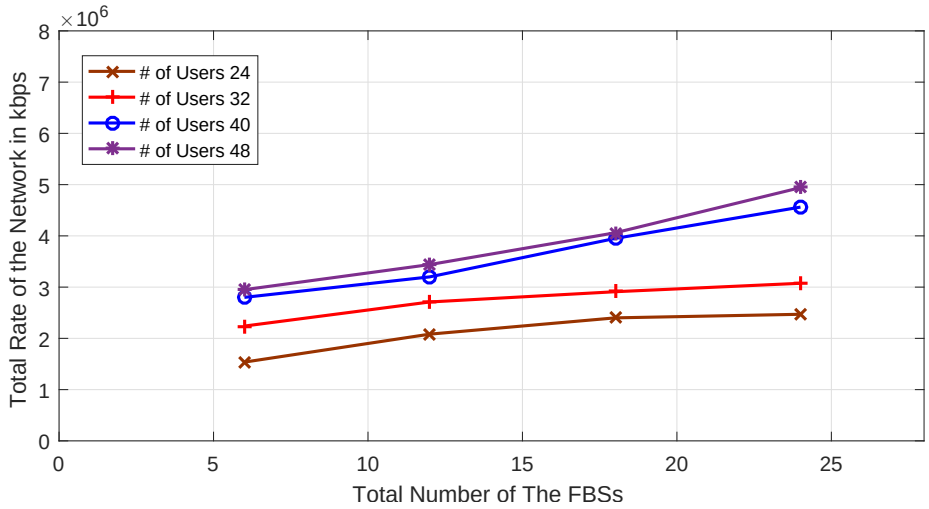

Figure 3.4 The Total Rate of the Network with Different Numbers of Users Using the K-means Approach.

In Figure 3.6 we use the same simulation parameters in Figure 3.2, to show how the system is turning OFF FBSs as the number of FBSs increases. As the figure shows, in the four different users scenarios, all FBSs are ON for a network with only 6 FBSs. This is due to the high number of users which causes the network to activate all the available FBSs to meet the demand. As we increase the number of available FBSs, the scenario with fewest users to serve will keep the number of active FBSs unchanged and increase it as the available number of FBSs increases, while the other scenarios will increase the number of active FBSs in order to serve the users with the required rate. Thus, as we increase the number of available FBSs the network will activate more FBSs, since this activation will increase the aggregated users rate and consume energy reasonably.

However, in Figure 3.4, Figure 3.5 and Figure 3.6 we can see that for larger numbers of FBSs, and scenarios with more users (40 and 48 users) the metrics will keep increasing while the other two scenarios with fewer users (24 and 32 users) we sustain a small increase. The explanation of this behaviour is that the network with fewer users is almost saturated, where adding more FBSs will not affect the overall performance of the network. On the other hand, for the scenarios with higher numbers of users, the network activates more FBSs to provide the users with higher rates, hence, increasing the energy consumption. However,

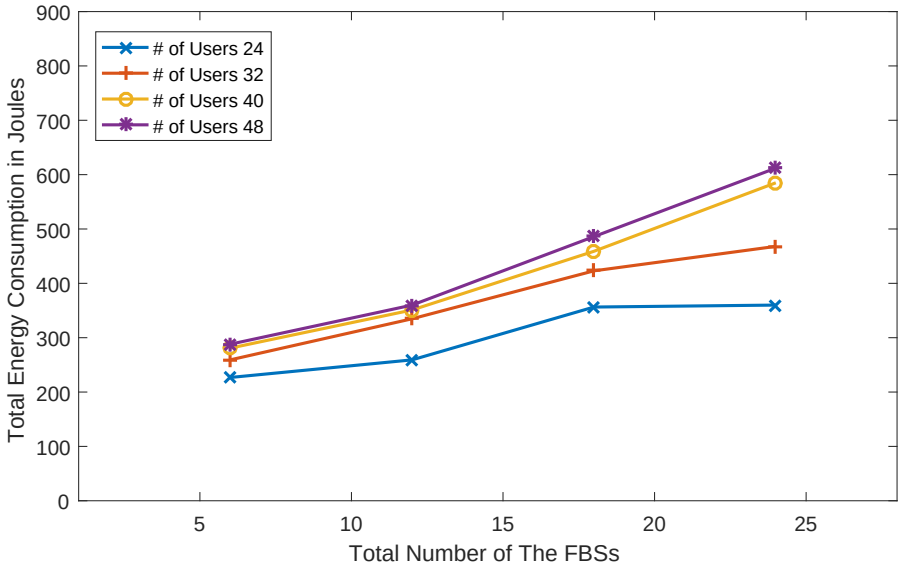

Figure 3.5 The Total Consumed Energy in the Network with Different Number of Users.

this increase in energy is compensated by the increase of the total rate, hence adjusting the objective function.

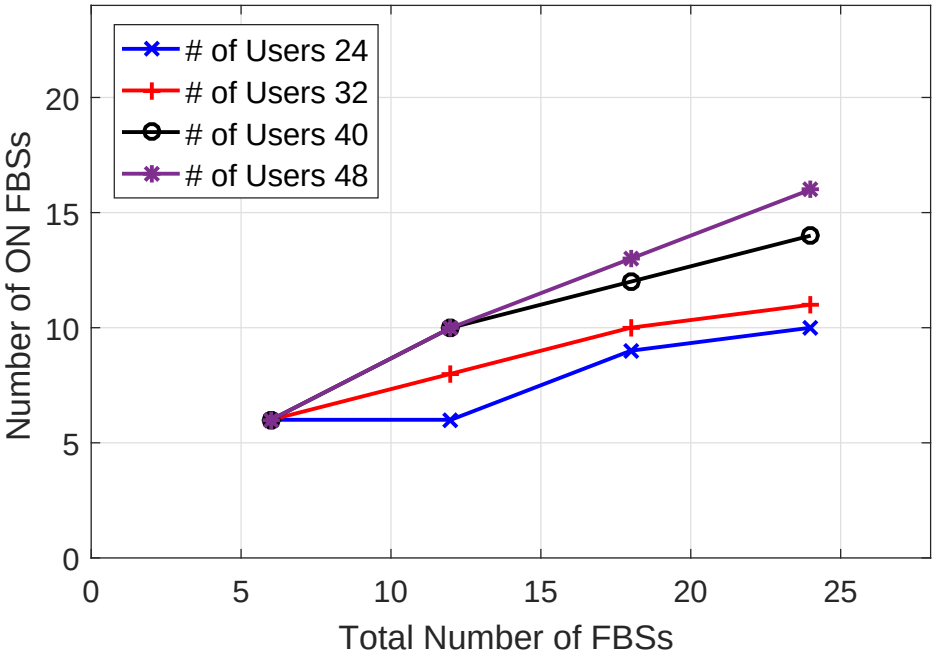

Figure 3.6 The Total Operating FBSs to Different Number of Users.

CHAPTER 4. COOPERATIVE SMALL CELL HETNETS WITH SLEEPING AND ENERGY HARVESTING

4.1 Introduction

The goal of the work of this chapter is to minimize the power drawn from the grid by exploiting harvested energy as much as possible. Therefore, in order to benefit from the HE and minimize the power driven from the grid, our model will push the network to deactivate as many Small BSs (SBSs) as possible and utilize the sleeping SBSs in harvesting energy. However, to ensure quality of service, we set a minimum required rate for every user that the network should not violate. The main contributions are summarized as follows:

1. We formulate an optimization problem to minimize the energy drawn from the network grid by utilizing the harvested energy and dynamic sleeping of the SBSs. The problem is formulated such that it motivates the network to deactivate as many SBSs as possible to minimize their energy consumption and benefits from their harvesting capability.

2. The formulated problem is a Mixed Integer NonLinear Problem (MINLP) which is NP-hard. Therefore, we decompose the problem into two subproblems: a convex nonlinear problem for solving the continuous variables of the harvesting energy and a Mixed Integer Linear Problem (MILP) for solving the user association and sleeping strategy.Then, we used the Generalized Bender Decomposition (GBD) algorithm to solve the two subproblems iteratively for an optimal solution.

3. We propose a new computationally efficient algorithm: Base Station Centrality (BSC) which is based on network centrality to solve and optimize the dynamic sleeping, user association and energy harvesting of the system model.

4. Finally, we evaluate the performance of our proposed algorithm through an extensive simulation study to verify its superiority over the GBD in terms of operational strategy.

4.2 System Model

This chapter considers a heterogenous network where several SBSs co-exist in a designated area. The deployment of the SBSs is a promising solution to provide higher quality of service (QoS) for users. However, between the SBSs interference is considered, since the SBSs are deployed in a dense environment and reusing the available resources provides higher throughput.

Figure 4.1 shows the architecture of the network where SBSs are equipped with energy harvesting methods (solar panels for example) and are serving users under their coverage. Moreover, SBS is connected to the Smart Grid with a two way energy flow.

4.2.1 Energy Harvesting Model For SBSs

Let $f = 1, ..., F$ denote the set of the SBSs that are randomly distributed in the macro cell coverage area of $\mathbb{A}$, while $u = 1, ..., U$ and $c = 1, ..., C$ denote the set of a randomly distributed users covered by the SBSs and the set of available resource blocks in the network, respectively. Moreover, we consider a time slotted system with fixed duration τ, and $n = 1, ..., N$ denote the index of the slot number. Furthermore, each user is assumed to be associated with only one SBS.

The SBSs harvest energy from a renewable source (e.g. wind, solar... etc), where the amount of harvested energy for every SBS f and time slot n is denoted by $hr_f[n]$ and it follows the truncated Gaussian distribution [77]. Moreover, each SBS is equipped with a

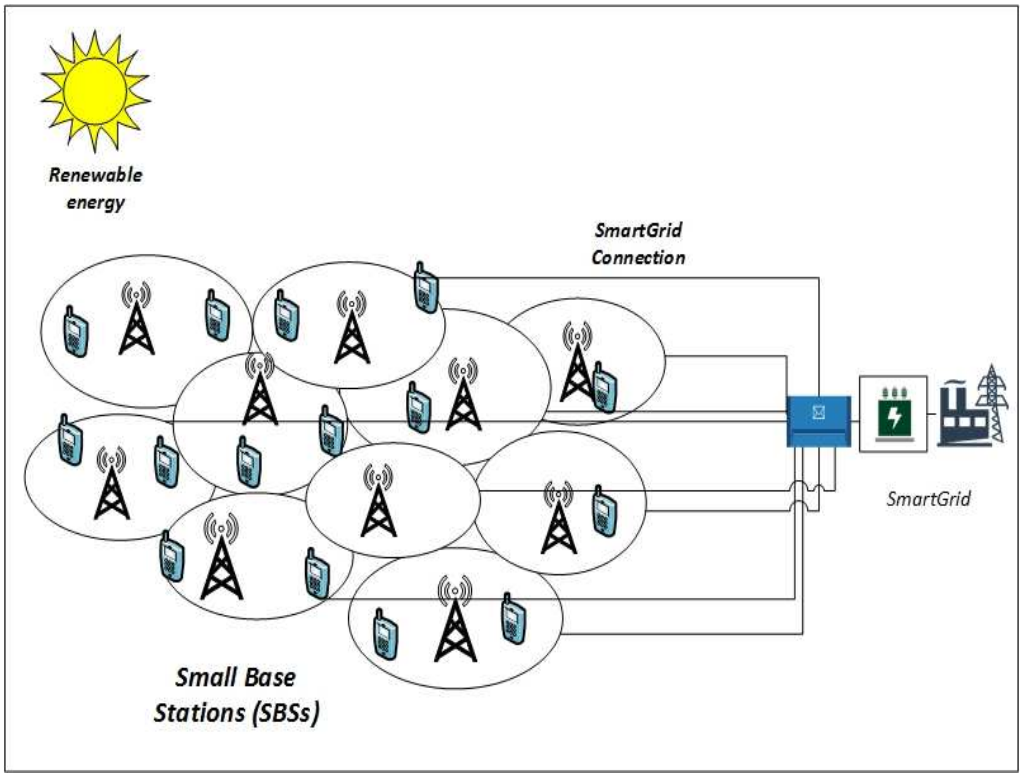

Figure 4.1 A network with SBSs powered by renewable energy and connected to a smart grid.

battery to store its harvested energy with a maximum capacity of B_{max} with battery level at time slot n is $B[n]$. However, due to the stochastic nature of energy harvesting, each BS is connected to a non-renewable energy source to compensate for the renewable energy shortage. In other words, each SBS is set to use the energy from renewable source first, and then request power from the grid. However, the promising technology of Smart grid which allows a two-way flow of power [28], can be used here to transfer the harvested energy between SBSs, i.e., the SBS with surplus harvested energy will transfer it to other SBSs that suffer from renewable energy deficit. Therefore, at the end of every time slot, the SBS will either transfer the surplus of its harvested energy or request energy from other SBSs to fulfill its deficit.

If the energy surplus of the other SBSs cannot match the energy demand of the SBS with the shortage, then the SBS will request a non-renewable energy from the smart grid directly. Hence, every SBS is equipped with two power sources: the non-renewable power from the

Table 4.1List of Notations used throughout chapter 4

Notation	Description
f	SBS index.
u	User index.
c	Resource block index.
n	Time slot index.
τ	Duration time of every slot.
B_{max}	Maximum battery capacity.
$p^c_{fu}[n]$	Transmission power from f to u by c.
$B_f[n]$	Battery level of SBS f during slot n.
$\lambda_f[n]$	Injected energy into the SG by f during n.
$hr_f[n]$	HE for every SBS during n.
$\mu_f[n]$	Drawn energy from the SG by f during n.
$h^c_{fu}[n]$	Channel Gain between f and u by c during n.
η	The transfer efficiency of the SG.
ν_f	Indicates if all users associated with SBS f are offloadable.
ϕ_f	The total number of SBSs that f can offload its associated users to them.
x^c_{fu}	The binary association between f and u through c.
y_f	The SBS f ON/OFF status.
ω	Available bandwidth for every channel.
E_b	The energy consumption of the SBS basic circuit.

grid and the power from the renewable sources. Therefore, the transmission power between user u and BS f using resource block c, during the time slot n is: $p^c_{fu}[n] = p^c_{fu,g}[n] + p^c_{fu,r}[n]$, where $p^c_{fu,g}[n]$ is the power drawn from the grid and $p^c_{fu,r}[n]$ is the power drawn from the renewable source (including the energy transferred from other SBSs).

Let $\lambda_f[n]$ and $\mu_f[n]$ denote the amount of the harvested energy the BS f is injecting into or receiving from the smart grid at the end of slot n, respectively. Then the amount of the harvested energy that is transferred into the smart grid equals the harvested energy that is drawn from the smart grid, where η is the transfer efficiency.

$$\sum_{f=1}^{F}\sum_{i=1}^{n}\mu_f[i] = \sum_{f=1}^{F}\sum_{i=1}^{n}\eta\lambda_f[i] \tag{4.1}$$

Therefore, at time slot $i = 1$ the battery will be zero, and at the end of every slot $i = 1, 2, ...N$ the battery storage will be the sum of the harvested energy subtracting the transmission power and the transferred energy $0 \leq B_f[i] \leq B_{max}$, where $B_f[i]$ is defined as:

$$B_f[i] = \sum_{n=2}^{i} hr_f[n] - \sum_{n=1}^{i} \sum_{u=1}^{U} p^c_{uf,r}[n]\tau - \sum_{n=1}^{i} \lambda_f[n] \tag{4.2}$$

4.2.2 User Association and Achievable Rate

Let x^c_{uf} be a binary indicator that is equal to 1 if user u and SBS f are associated using resource block c, or 0 otherwise. Also, let z_{fu} be a binary indicator that is equal to 1 if user u is associated with SBS f, or 0 otherwise. y_f indicates the SBS on/off status, where $y_f = 0$ if the SBS is OFF (where there are no users associated with it), and $y_f = 1$ if the SBS is ON. However, a deactivated SBS will keep harvesting energy and injecting it into the smart grid to serve other active SBSs.

The distance between the fth SBS and the uth user can be expressed as follow:

$$d_{uf} = ||l_u - l_f|| \qquad \forall u \in U, \forall f \in F \tag{4.3}$$

where the l_u and l_f are the $x - y$ coordinates for the location of the UE, and the fixed location of the SBS, respectively. It follows from (4.3) that the channel power gain can be modeled as:

$$|h^c_{uf}|^2 = \frac{\beta_0}{d^\alpha_{uf}} = \frac{\beta_0}{||l_u - l_f||^\alpha} \tag{4.4}$$

where β_0 denotes the channel gain at the reference distance of $d_0 = 1$m, and α is the path loss exponent.

Moreover, the interference at a user u which is associated with SBS f from all other SBSs at a time slot n will be:

$$I^c_{uf}[n] = \sum_{j \neq u}^{U} \sum_{i \neq f}^{F} p^c_{ji}[n]|h^c_{ui}[n]|^2, \tag{4.5}$$

Then, the signal to interference and noise ratio SINR for every user is:

$$\gamma_{uf}^{c}[n] = \frac{p_{uf}^{c}[n]|h_{uf}^{c}[n]|^2}{I_{uf}^{c}[n] + \omega N_0}, \tag{4.6}$$

where $h_{uf}^{c}[n]$ denotes the channel gain from SBS f to user u using resource block c at time slot n, ω is the available bandwidth for every channel, and N_0 is the channel noise spectral density which is assumed to be Additive White Gaussian Noise AWGN, and ωN_0 is the noise variance σ^2. Thus, the data rate for every user using a single channel will be as follow:

$$R_{uf}^{c}[n] = \omega \log(1 + \gamma_{uf}^{c}[n]) \tag{4.7}$$

4.3 Problem Formulation

In this section, an optimization problem, which minimizes the non-renewable energy consumption of the transmission power for a cooperative heterogenous network is formulated. First, we formulate a problem where users association, sleeping strategy and energy minimization are performed within a single optimization problem. There are two problems, the first is optimizing over N slots and this requires predicting the harvestable energy and channel conditions, where the second is optimizing for every slot separately, and this is not globally optimal, but more realistic, hence we focus on the second problem in this chapter. The problem can be stated as follows: given the number of users and SBSs, the problem will solve the user association, sleeping strategy and power consumption, then at every time slot the optimization problem will recalculate the users association and the transmission power, while not changing the status of the SBSs, this will help simplify the problem since the time slot is relatively very short. However, due to the non-convexity of the problem we present a more tractable and a convex approximation where we decouple the users association and sleeping strategy from the energy minimization. We can then mathematically state the main optimization problem as below:

Problem $\mathcal{F}$:

$$\underset{p^c_{uf}[n],\lambda_f[n],\mu_f[n],y_f,z_{uf},x^c_{uf}}{\text{Minimize}} \sum_{f,u,n,c=1}^{F,U,N,C} p^c_{fu,g}[n]\tau + \sum_{f=1}^{F} E_b y_f \tag{4.8}$$

subject to

$$\sum_{f}^{F}\sum_{c=1}^{C} x^c_{fu} R^{min}_u \leq \sum_{f=1}^{F}\sum_{c=1}^{C} x^c_{fu} R^c_{fu}[n], \qquad \forall u, \forall n \tag{4.9}$$

$$\sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu,r}[n]\tau \leq B_f[n-1] + \mu_f[n], \qquad \forall f, \forall n, \tag{4.10}$$

$$B_f[n] \leq B_{max} \qquad \forall f, \forall n, \tag{4.11}$$

$$\sum_{f=1}^{F}\sum_{i=1}^{n} \mu_f[i] = \sum_{f=1}^{F}\sum_{i=1}^{n} \eta\lambda_f[i] \qquad \forall n, \tag{4.12}$$

$$\sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu}[n] \leq P^{max}_f \qquad \forall f, \forall n, \tag{4.13}$$

$$\sum_{u=1}^{U} x^c_{fu} \leq 1 \qquad \forall f, \forall c, \tag{4.14}$$

$$\sum_{f=1}^{F} z_{uf} = 1, \qquad \forall u, \tag{4.15}$$

$$\frac{\sum_{c=1}^{C} x^c_{uf}}{\#ofUs} \leq z_{uf} \leq \sum_{c=1}^{C} x^c_{uf}, \qquad \forall\ f, \forall u, \tag{4.16}$$

$$\frac{\sum_{u=1}^{U} z_{uf}}{\#ofSBSs} \leq y_f \leq \sum_{u=1}^{U} z_{uf}, \qquad \forall f, \tag{4.17}$$

Constraint (4.9) represents the QoS for every user. The constraints from (4.10) to (4.13) are dealing with energy transfer and cooperation between SBSs, while constraints from (4.14) to (4.17) are dealing with the users association and SBSs sleeping strategy. Constraint (4.10) represents the energy consumption causality where the BS cannot use energy more than what is available. Constraint (4.11) limits the battery capacity. Constraint (4.12) is for energy conservation, where the total injected energy into the smart grid equals the total received energy by all BSs. Constraint (4.13) limits the maximum allowed transmission power for every BS.

4.3.1 Generalized Bender Decomposition

However, due to the coupling of the user associations and sleeping strategy with energy harvesting, the above problem is clearly intractable and difficult to solve. Since we have three binary variables (y_f, z_{fu}, and x^c_{fu}) with four different indices (f, u, and c), the time needed to find the optimal solution will increase exponentially as the network size increases linearly. This is due to the fact that the problem is MINLP, for which there is no efficient algorithm for solving it. Therefore, we propose a decomposition approach where problem ($\mathcal{F}$) is decomposed into two subproblems, the first subproblem include the integer variables, while the second problem contains the continues variables.

Generalized Benders Decomposition (GBD) is a well known method for solving mathematical programming problems with mixed integers nonlinear problems [78]. GBD is a generalization of the Benders' decomposition[79] to include broader class of problems which can be nonlinear. The duality theory for nonlinear convex problems is exploited to derive the cuts corresponding to those in the original Benders' Decomposition. However, GBD approach requires the optimization problem to be a convex nonlinear problem, which is not the case for Problem ($\mathcal{F}$). Hence, ($\mathcal{F}$) is nonconvex problem due to constraint (4.9), where the SINR inside the log function will cause the constraint to be nonconvex.

4.3.2 Linearizing QoS Constraint:

In this section we derived a concave lower bound of constraint (4.9) using the first order Taylor series to transfer problem ($\mathcal{F}$) to the standard form in GBD. Therefore, constraint (4.9) can be lower bounded by the following:

$$
\begin{aligned}
R^c_{uf}[n] &= \omega \log(1 + \frac{p^c_{uf}[n]|h^c_{uf}[n]|^2}{\sum\limits_{j\neq u}^{U}\sum\limits_{i\neq f}^{F} p^c_{ji}[n]|h^c_{ui}[n]|^2 + \omega N_0}) \\
&= \omega \log(\sum_{u}^{U}\sum_{f}^{F} p^c_{uf}[n]|h^c_{uf}[n]|^2 + \omega N_0) \\
&- \omega log(\sum_{j\neq u}^{U}\sum_{i\neq f}^{F} p^c_{ji}[n]|h^c_{ui}[n]|^2 + \omega N_0) \\
&\geq \omega \log(\sum_{u}^{U}\sum_{f}^{F} p^c_{uf}[n]|h^c_{uf}[n]|^2 + \omega N_0) - \hat{R}_{Ty}
\end{aligned}
\tag{4.18}
$$

where $\hat{R}_{Ty}$ is the first-order Taylor approximation around point $(p^c_{0ji}[n])$, and is as follows:

$$
\begin{aligned}
\hat{R}_{Ty} &\triangleq \omega log(\sum_{j\neq u}^{U}\sum_{i\neq f}^{F} p^c_{0ji}[n]|h^c_{ui}[n]|^2 + \omega N_0) \\
&+ \sum_{j\neq u}^{U}\sum_{i\neq f}^{F} \frac{|h^c_{ui}[n]|^2}{ln(2)p^c_{0ji}[n]|h^c_{ui}[n]|^2 + \omega N_0}(p^c_{ji}[n] - p^c_{0ji}[n])
\end{aligned}
\tag{4.19}
$$

Hence, the approximated concave lower bound of equation (4.9) will be as follow:

$$
\hat{R}^c_{uf}[n] \triangleq \omega \log(\sum_{u}^{U}\sum_{f}^{F} p^c_{uf}[n]|h^c_{uf}[n]|^2 + \omega N_0) - \hat{R}_{Ty} \tag{4.20}
$$

Thus, replacing (4.9) by its approximated concave lower bound $\hat{R}^c_{uf}[n]$ will transfer problem $(\mathcal{F})$ into a convex MINLP, which can be rewritten as follows:

Problem $\mathcal{P}$:

$$
\underset{p^c_{uf}[n],\lambda_f[n],\mu_f[n],y_f,z_{uf},x^c_{uf}}{\text{Minimize}} \sum_{f,u,n,c=1}^{F,U,N,C} p^c_{fu,g}[n]\tau + \sum_{f=1}^{F} E_b y_f \tag{4.21}
$$

subject to

$$
\sum_{f}^{F}\sum_{c=1}^{C} x^c_{fu} R^{min}_u \leq \sum_{f=1}^{F}\sum_{c=1}^{C} x^c_{fu}\hat{R}^c_{uf}[n], \qquad \forall u, \forall n \tag{4.22}
$$

(4.10) – (4.17)

4.3.3 Derivation of the Primal Problem:

The problem $(\mathcal{P})$ now falls into the standard forms of GBD problems, where the problem has two sets for variables, the first is the binary variables (y_f, z_{fu}, and x^c_{fu}) that can

$$\Gamma(\alpha_u[n],\rho_f[n],\zeta_f[n],\xi[n],\beta_f[n]) = \tag{4.23}$$
$$\sum_{f,n,u,c=1}^{F,N,U,C} p^c_{fu,g}[n]\tau + \sum_{f=1}^{F} E_b y_f + \sum_{u=1}^{U}\sum_{n=1}^{N} \alpha_u[n]\Big[- \sum_{f=1}^{F}\sum_{c=1}^{C} \hat{R}^c_{fu}[n] + \sum_{f}^{F} x^c_{fu} R^{min}\Big] + \sum_{f=1}^{F}\sum_{n=1}^{N} \rho_f[n]\Big[B_f[n] - B_{max}\Big]$$
$$+ \sum_{f,n=1}^{F,N} \zeta_f[n]\Big[\sum_{u,c=1}^{U,C} p^c_{fu,r}[n]\tau - B_f[n-1] - \mu_f[n]\Big] + \sum_{n=1}^{N} \xi[n]\Big[\sum_{f=1}^{F}\sum_{i=1}^{n} \mu_f[n] - \sum_{f=1}^{F}\sum_{i=1}^{n} \eta\lambda_f[n]\Big] + \sum_{f=1}^{F}\sum_{n=1}^{N} \beta_f[n]\Big[\sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu}[n] - P^{max}_f\Big]$$

$$\hat{\Gamma}(\hat{\alpha}_u[n],\hat{\rho}_f[n],\hat{\zeta}_f[n],\hat{\xi}[n],\hat{\beta}_f[n]) = \tag{4.24}$$
$$\sum_{u=1}^{U}\sum_{n=1}^{N} \hat{\alpha}_u[n]\Big[- \sum_{f=1}^{F}\sum_{c=1}^{C} \hat{R}^c_{fu}[n] + \sum_{f}^{F} x^c_{fu} R^{min}\Big] + \sum_{f=1}^{F}\sum_{n=1}^{N} \hat{\rho}_f[n]\Big[B_f[n] - B_{max}\Big]$$
$$+ \sum_{f,n=1}^{F,N} \hat{\zeta}_f[n]\Big[\sum_{u,c=1}^{U,C} p^c_{fu,r}[n]\tau - B_f[n-1] - \mu_f[n]\Big] + \sum_{n=1}^{N} \hat{\xi}[n]\Big[\sum_{f=1}^{F}\sum_{i=1}^{n} \mu_f[n] - \sum_{f=1}^{F}\sum_{i=1}^{n} \eta\lambda_f[n]\Big] + \sum_{f=1}^{F}\sum_{n=1}^{N} \hat{\beta}_f[n]\Big[\sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu}[n] - P^{max}_f\Big]$$

be considered as the complicating variables and the second set is the continuous variables $(p^c_{uf}[n], \lambda_f[n], \mu_f[n])$. Therefore, it is much easier to solve problem $(\mathcal{P})$ when the binary variables are fixed. In fact, fixing the binary variables, problem $(\mathcal{P})$ become a Convex optimization problem which can efficiently be solved using the Lagrangian method. Consequently, the problem is decomposed into two subproblems, the primal problem and the master problem. The primal problem is a convex nonlinear optimization problem and is formulated as follow:

Problem $\mathcal{L}$:

$$\underset{p^c_{uf}[n],\lambda_f[n],\mu_f[n],}{\text{Minimize}} \ \mathbb{S} = \sum_{f,u,n,c=1}^{F,U,N,C} p^c_{fu,g}[n]\tau + \sum_{f=1}^{F} E_b \bar{y}_f$$

subject to

(4.20), (4.10) − (4.13)

The goal of solving the primal problem is to find an upper bound for the solution given by algorithm 1. The problem $(\mathcal{L})$ is convex, since the objective function is linear and all the constraints are convex [80] (Note: the second part of the objective function is constant and has no effect on the final solution.). Therefore, problem $(\mathcal{L})$ can be solved using the lagrangian to obtain the optimal solution. Hence, the Lagrangian of $(\mathcal{L})$ is given in (4.23), where $\big[\alpha_u[n], \rho_f[n], \zeta_f[n], \xi_f[n], \beta_f[n]\big]$ are the lagrangian multipliers.

4.3.4 Derivation of the Master Problem:

The master problem is derived by fixing all continuous variables and solve the problem with only the binary variables $\mathbb{Z} = \left[y_f,\, z_{fu},\, x^c_{fu}\right]$. Thus, the master problem will be pure binary optimization problem. The master problem goal is to find the lower bound solution of the algorithm1. Hence, the master problem can be formulated as follows:

$$\text{Problem } \mathcal{M}: \tag{4.25}$$

$$\underset{y_f, z_{uf}, x^c_{uf}}{\text{Minimize}}\ \nu$$

$$\Gamma(\alpha_u[n], \rho_f[n], \zeta_f[n], \xi[n], \beta_f[n]) \leq \nu \tag{4.26}$$

$$\forall \alpha_u[n], \rho_f[n], \zeta_f[n], \xi[n], \beta_f[n] \geq 0, \forall u \in U, \forall f \in F, \forall n \in N$$

$$\hat{\Gamma}(\hat{\alpha}_u[n], \hat{\rho}_f[n], \hat{\zeta}_f[n], \hat{\xi}[n], \hat{\beta}_f[n]) \leq 0 \tag{4.27}$$

$$\forall \hat{\alpha}_u[n], \hat{\rho}_f[n], \hat{\zeta}_f[n], \hat{\xi}[n], \hat{\beta}_f[n] \geq 0, \forall u \in U, \forall f \in F, \forall n \in N$$

$$(4.14) - (4.17)$$

The master problem uses Lagrange equation (4.23) that is associated with the the primal problem to get a lower bound for the solution of algorithm 1. However, when the solution of the primal problem is not feasible, algorithm 1 uses Lagrange multipliers for the feasibility problem instead. Feasibility problem is stated as follows:

$$\text{Problem } \mathcal{U}:$$

$$\underset{p^c_{uf}[n], \lambda_f[n], \mu_f[n]}{\text{Minimize}}\ \kappa \tag{4.28}$$

$$\text{subject to}$$

$$(4.20), (4.10) - (4.13)$$

Thus, algorithm 1 is presented to solve the optimization problem iteratively between the master $\mathcal{M}$ and primal $\mathcal{L}$ problems. First, we initialize feasible points for the binary

Algorithm 1: Generalized Bender Decomposition.

Input: $f; u; n; c; h^c_{fu}[n]; hr_f[n]; R_{min}; P_{max}; \omega N_0; \epsilon$
Initialize: $y^0_f; x^{0c}_{fu}; z^0_{fu}; i = 0; UB = \infty; LB = -\infty; \Delta = \infty$
while $\Delta \geq 1$ **do**
  Solve the primal NLP problem $[\mathcal{L}]$ and let the solution be: $\bar{\mathbb{T}}^{[k]} = \left[\bar{p}^c_{uf}[n]^{[k]}, \bar{\lambda}_f[n]^{[k]}, \bar{\mu}_f[n]^{[k]}\right]$
  and the multipliers: $\bar{\mathbb{Q}}^{[k]} = \left[\bar{\alpha}_u[n]^{[k]}, \bar{\rho}_f[n]^{[k]}, \bar{\zeta}_f[n]^{[k]}, \bar{\xi}[n]^{[k]}, \bar{\beta}_f[n]^{[k]}\right]$
  Set $UB = \bar{\mathbb{S}}^{[k]}$
  if $\bar{\mathbb{T}}^{[k]}$ is feasible **then**
    if $UB - LB \leq \epsilon$ **then**
      $\mathbb{T}^* := \bar{\mathbb{T}}^{[k]}$
      $\mathbb{Q}^* := \bar{\mathbb{Q}}^{[k]}$
    else
      $k = k + 1$
    end if
  else
    Solve the feasibility problem and find the lagrangian multipliers $\hat{\mathbb{Q}}^{[l]}$
    $l = l + 1$
  end if
  Solve the Master MIP problem $[\mathcal{M}]$ and let the solution be: $\bar{\mathbb{Z}}^{[k]} = \left[\bar{y}^{[k]}_f, \bar{z}^{[k]}_{fu}, \bar{x}^{c[k]}_{fu}\right]$
  Set $LB = \bar{\mu}^{[k]}$
  if $UB - LB \leq \epsilon$ **then**
    $\mathbb{Z}^* := \bar{\mathbb{Z}}^{[k]}$
    $\Delta = 0$
  else
    Go to step 4
  end if
end while
Output: $\mathbb{T}^*, \mathbb{Z}^*$

variables: $[y^0_f, z^0_{fu}$, and $x^{0c}_{fu}]$ along with other parameters. Second, the convex subproblem $(\mathcal{L})$ is solved generating an optimal solution $\bar{\mathbb{T}}^{[1]} = \left[\bar{p}^c_{uf}[n], \bar{\lambda}_f[n], \bar{\mu}_f[n]\right]$ and the multipliers $\bar{\mathbb{Q}}^{[1]} = \left[\bar{\alpha}_u[n], \bar{\rho}_f[n], \bar{\zeta}_f[n], \xi[\bar{n}], \bar{\beta}_f[n]\right]$, then, this solution is set as the upper bound of the algorithm.

However, if the primal problem $(\mathcal{L})$ is infeasible, the algorithm then solves the feasible problem $(\mathcal{U})$ and sets its solution as the upper bound. The second step is to solve the master problem $(\mathcal{M})$ with any efficient Integer Linear algorithm applying the solution of

the primal problem ($\mathcal{L}$): $[\bar{\mathbb{T}}, \bar{\mathbb{Q}}]$ from the first step. Then we set the binary solution of the master problem's output to be the lower bound.

On every iteration the algorithm evaluate the difference between the UB and LB and if the difference is greater than ϵ, then the algorithm will uses the solution of the binary variables from the master problem ($\mathcal{M}$) to solve the revised primal ($\mathcal{L}$) problem and repeat the previous steps. The algorithm iterates until the termination condition is met. It is shown in [78] that GBD algorithm terminates in a finite number of steps.

4.4 User Association and Sleeping Dynamic using Centrality Analysis

In this section a new approach to choose which SBS to deactivate and offload its associated users to neighboring SBSs is introduced. This approach exploits the centrality analysis that is used in graph theory to indicate the most important vertices within a graph [81]. In this section, the Base Station Centrality (BSC) analysis is presented to mark the SBSs that have the most potential to be deactivated without affecting the users QoS. The basic idea of BSC is to introduce a metric that shows the fitness of every individual SBS to deactivate. The fitness of every SBS is evaluated according to its ability to offload its users to more neighboring SBSs, i.e., the SBS that is able to offload its users to more neighboring SBSs will have more potential to deactivate without interrupting the service of the associated users. Thus, BSC will provide an intuition on which SBS to choose to deactivate instead of a random approach.

We introduce two parameters to calculate the Base Station centrality, ν_f and ϕ_f. ν_f indicates wether all users associated with the SBS f can be offloaded or not, while ϕ_f denotes the total number of SBSs that SBS f can offload its associated users to them. Moreover, the user can be offloaded to any SBS that can support its QoS requirements, i.e., R_{min}. Thus, we define the Base Station Centrality of an SBS in the network as follows:

$$BSC_f = \nu_f \phi_f \qquad \forall f \tag{4.29}$$

Eq.(4.29) captures the effect of the number of the SBSs whose associated users can be offloaded based on the BSC of SBS f. Hence, BSC works as an indicator of how every SBS is centered within the network, which means as the BSC increases the SBS f is placed close to many other SBSs. Therefore, the higher the BSC the more likely the SBS to be deactivated and its associated users are offloaded to other SBSs.

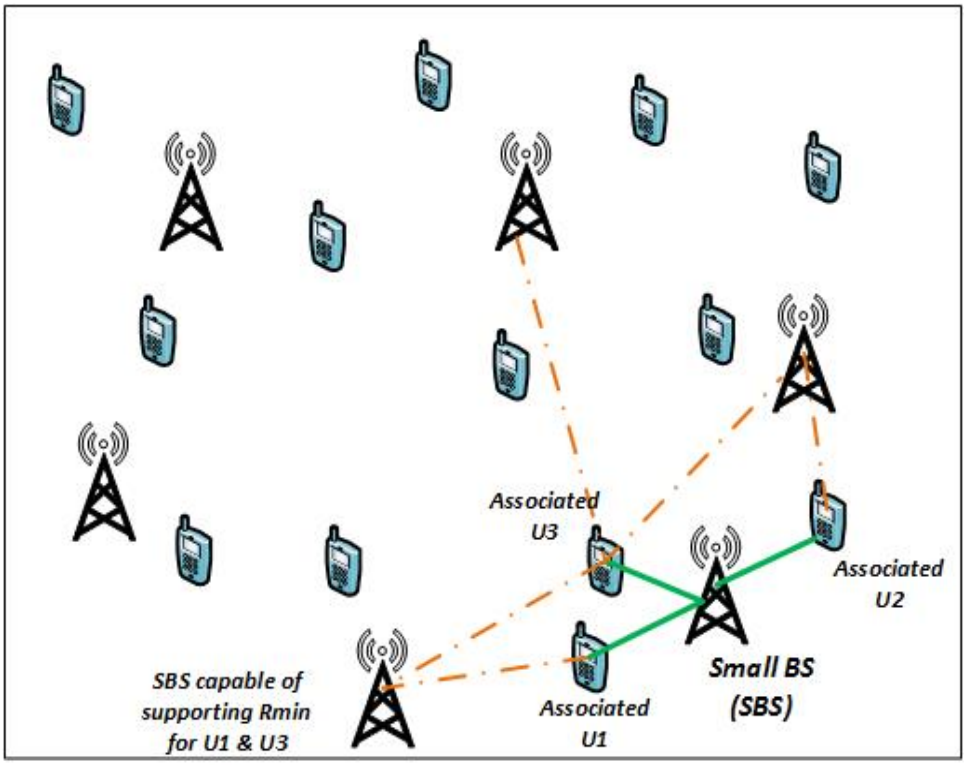

Figure 4.2 The evaluation of BS Centrality (BSC) through the calculation of ν_f and ϕ_f.

Figure 4.2 shows an illustrative example of how to evaluate the BSC of SBS f on a topology with dense SBSs network. In the figure users $U1, U2$ and $U3$ are associated with SBS $f = 1$ according to the best SINR. However, those users who are associated with this SBS can also be associated with other SBSs that can support their QoS (i.e., R_{min}), e.g., U3 can be associated with three other SBSs, while the other two users each of them can be associated with one SBSs. In this example ϕ_1 will be equal to five since the associated users can be offloaded to five SBSs in total, and ν_1 will be equal to one since all associated users can be offloaded to other SBSs. Thus, the BSC of SBS $f = 1$ will equal to five.

Algorithm 2 is presented to calculate the value of BSC for all the active SBSs. First, In steps (1,2) we provided the algorithm with the parameters and set some initial values. Then, for every iteration (steps 4-26) SBS (f): First, in step (4) the algorithm will associate the users with the SBS that provides the best SINR. Second, in steps (5-7) all SBSs that have no associated users are deactivated. Third, in steps (8-11) the BSC is evaluated for every SBS by calculating ν_f and ϕ_f. Then, in steps (12,13), the algorithm will deactivate the SBS with the highest BSC and offload its associates to the neighboring SBSs that can support their QoS requirements.

After the association, the algorithm will solve the optimization problem ($\mathcal{L}$) and produce a candidate solution for the continuous variables $\bar{\mathbb{T}}^{[k]} = \left[\bar{p}^{c}_{uf}[n]^{(k)}, \bar{\lambda}_f[n]^{(k)}, \bar{\mu}_f[n]^{(k)}\right]$. Fourth, in steps (14-20), if the optimization problem is found to be infeasible, then we set ν_f to be zero to indicate that SBS f cannot be deactivated, and else if, objective function value for problem ($\mathcal{L}$) at iteration (k) is less than its value at iteration ($k-1$), then the candidate solution is set as the best solution. Finally, in steps (22-26), if $\nu_f = 0$ for all SBSs then the algorithm break with $\mathbb{T}^*, y^*_f, x^{*c}_{fu}$ as the final solution. Otherwise, go back to step (4) and recalculate the BSC for the remaining active SBSs.

4.5 Simulation Results

In this section, simulation results are provided to demonstrate the performance of the our proposed work on system model in Figure 4.1, to minimize the energy consumption of heterogenous networks. The parameters in all simulations, unless stated otherwise, are presented in Table 4.2. Moreover, the harvested energy levels are estimated by a truncated normal distribution with a mean equal to 0.2 and standard deviation of 0.07, the truncated normal distribution is set to have values higher than 0.001 [82]. Moreover, in all simulations we consider an area of 100X100 m^2 where the SBSs and associated users are uniformly distributed.

In Figure 4.3 we compare the performance of the optimal solution of $\mathcal{F}$ to BSC algorithm and the uncooperative approach, where the SBSs do not exchange harvested energy. However, $\mathcal{F}$ is very difficult to solve especially for large networks. Therefore, we solve the network for small number of SBSs $F = 5$, and 10 users with increasing the minimum required rate. From the figure we can see that using sleeping SBSs to harvest energy and transfer it to other SBSs provide better performance than the noncooperative approach. On the other hand, BSC algorithm performs close to the optimal solution of $\mathcal{F}$ within a reasonable time.

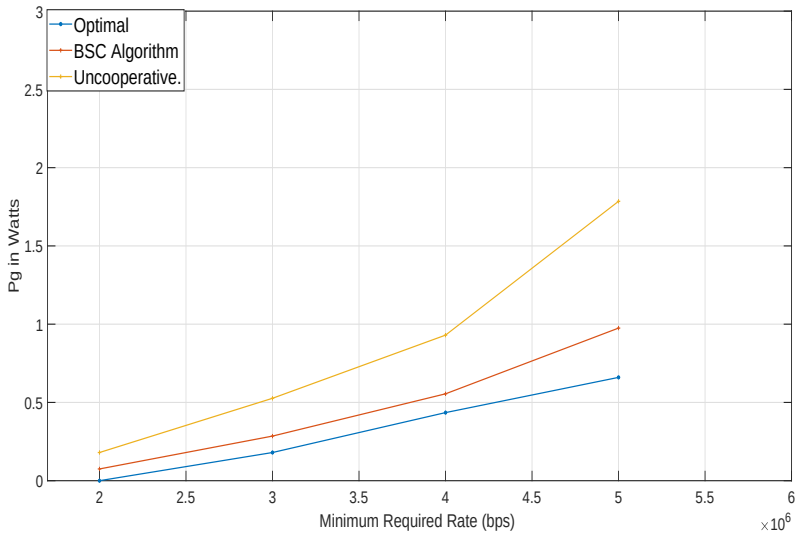

Figure 4.3 The optimal results compared to BSC algorithm and the uncooperative as the R_{min} increases.

In Figure 4.4, we compare the performance of the optimal solution of the GBD problem of algorithm 1 to our proposed approach of BSC in algorithm 2 in case of the p_g consumption and the computational time. In this simulation we have SBSs of [5, 10, 15, 20] and the numbers of their associated users are [10, 20, 30, 40], respectively, where $N = 4$. As the figure depicts the BSC approach performs close to the optimal solution of the GBD, in fact for small networks the two approaches are very close, while for larger network the difference is around %10 . On the other hand, the optimal GBD required longer times

for computations than BSC algorithm, and as the network increases in size, the required computational time increases exponentially (around 2000s for 20 SBSs and 40 users), while BSC achieved a reasonable solution for a much shorter computational time (around 160s).

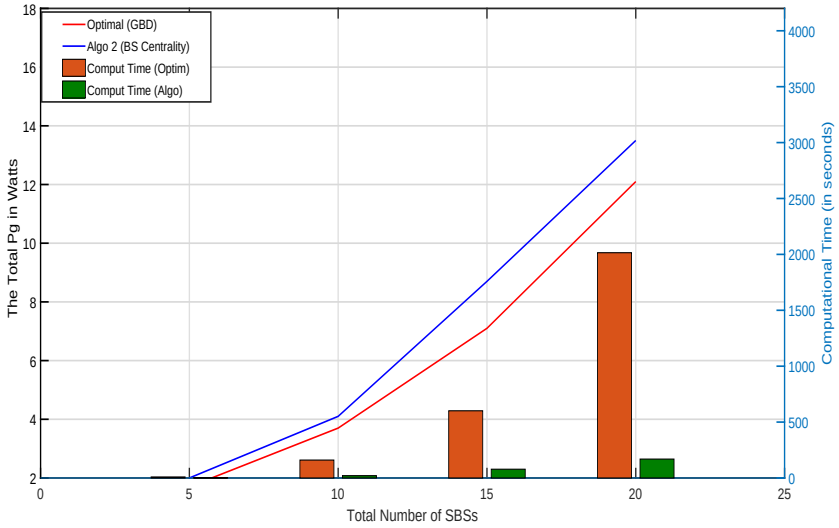

Figure 4.4 The optimal results compared to BSC algorithm with respect to the computational time and total p_g consumption.

Figure 4.5 shows the behavior of the Algorithm in every step. In this scenario we have a topology of $100X100m^2$ with 20 SBSs and 40 users i.e., $F = 20$ and $U = 40$, which all of the SBSs and associated users are uniformly distributed with minimum received rate is 1.5 Mbps. As in algorithm 2, in every step BSC-algorithm chooses the SBS with the highest BSC to deactivate and offload its associated users to other neighboring SBSs. The offloaded users are associated with the SBS that provide them with the second highest SINR. In the enlarged area we see how the BSC algorithm deactivate one SBS and associate its users to close by SBSs.

In Figure 4.6 we investigate the effect of the efficiency parameter (η) and the increase of the QoS requirement i.e., R_{min}, on the amount of power used from both sources (P_g an P_g). In this simulation we have In this simulation we used 20 SBSs and 40 users while N = 10. As the results show, for $\eta = 0.9$, the network will rely solely on P_r until R_min reaches

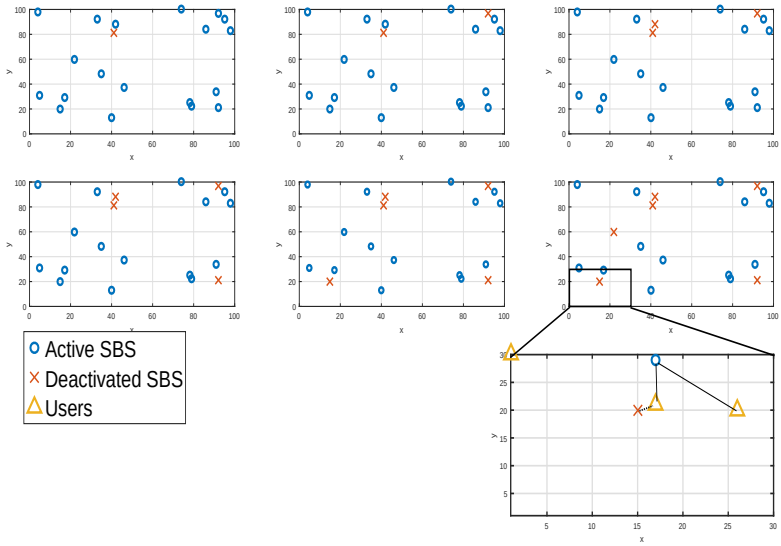

Figure 4.5 The behaviour of BSC algorithm on every iteration.

3.00 Mbps then we see that the network will start demanding power from the grid as P_g starts increasing to provide the necessary power to match the increase of users demands. A similar pattern happens in both $\eta = 0.75$ and $\eta = 0.60$, with lower R_{min} required to trigger the demand of the power from the grid (P_g). This is understandable since the lower the efficiency means the lower the energy transferred between SBSs in the network. On the other hand, for lower minimum rates ($R_{min} \leq$ 2Mbps), P_r values for the three scenarios are very close to each other, despite the differences in efficiencies, this is due to the fact that for lower rates almost all SBSs are using their harvested energy and not receiving or transferring it through the network where the η factor will come into effect.

Figure 4.7 shows the effect of increasing the number of BSs and the number of users on the amount of injected energy λ. As shownin the figure, we have two trends. First, the increasing number of SBSs will increase the amount of injected energy λ into the network. Second, as the number of users in the network increases λ decreases until it becomes almost zero. This can be explained as the number of users increases, the active SBSs will have no energy left to inject into the network, and only the deactivated SBSs that will be injecting energy into the network. However, for larger numbers of users, the algorithm cannot de-

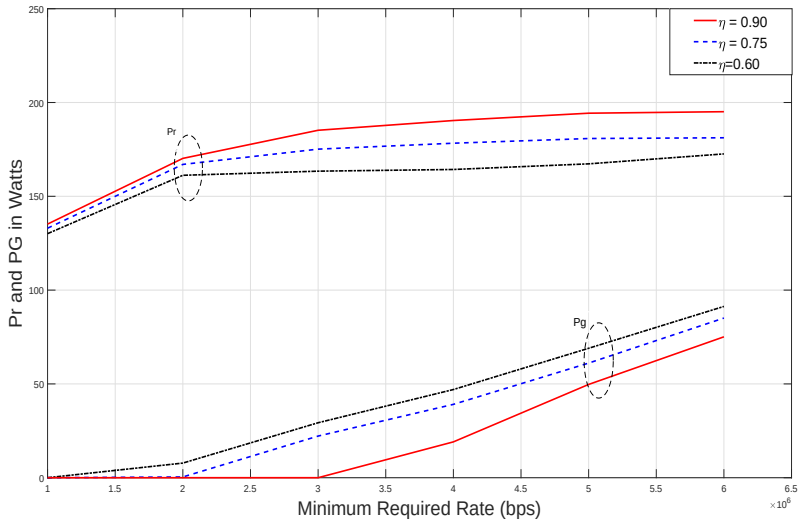

Figure 4.6 The Minimum Rate Compared to the Efficiency.

activate any SBS since the demand is high. On the other hand, increasing the number of SBSs in the network will result in more HE to be injected into the SG causing the network to be less reliable on the grid power.

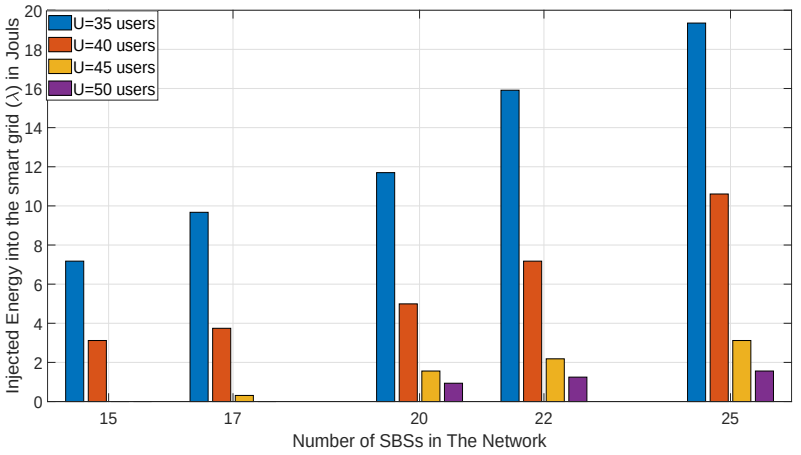

Figure 4.7 The relation between the increase of the SBSs and the Injected Energy.

Algorithm 2: Sleeping approach using Centrality.

Input:$h^c_{fu}[n]; hr_f[n]; R_{min}; P_{max}; \omega N_0$
Initialize: $p^c_{fu}[n]^{[0]}; y_f^{[0]} = 1; k = 0$
while True **do**
  Calculate SINR $\forall u, \&\forall f$, and associate users with BSs according to the highest SINR.
  if $z_{uf} = 0 \quad \forall u \in U$ **then**
    $y_f := 0 \quad \forall f \in F$
  end if
  for `<f=1:|Active SBSs|>` **do**
    Calculate ν_f, ϕ_f
    $BSC_f = \nu_f \phi_f$
  end for
  $y_{f'}^{[k]} \leftarrow 0$, BS f' is the BS with the highest BSC, and associates its users with the neighboring BS $\bar{x}^{c[k]}_{fu}$.
  Solve problem $[\mathcal{L}]$ and let the solution be: $\bar{\mathbb{T}}^{[k]} = \left[\bar{p}^c_{uf}[n]^{(k)}, \bar{\lambda}_f[n]^{(k)}, \bar{\mu}_f[n]^{(k)}\right]$
  if $\bar{\mathbb{S}}^{[k]} \leq \bar{\mathbb{S}}^{[k-1]}$ **then**
    $\mathbb{T}^* := \bar{\mathbb{T}}^{[k]}$
    $x^{*c}_{fu} := \bar{x}^{c[k]}_{fu}$
    $y^*_f := \bar{y}^{[k]}_f$
  else if problem $[\mathcal{L}]$ is infeasible **then**
    $\nu_f := 0$
  end if
  k:=k+1
  if $\nu_f = 0 \quad \forall f \in F$ **then**
    Break
  else
    Go to step 4
  end if
end while
Output: $\mathbb{T}^*, y^*_f, x^{*c}_{fu}$

Table 4.2Simulation Parameters

Parameter	Value	Parameter	Value
P_{max}	2 W	R_{min}	4 Mbps
N_0	-174 dbm/Hz	ω	5 MHz
B_{max}	6 Joules	τ	100 ms
η	0.9	E_b	4 W

CHAPTER 5. ENHANCING COOPERATIVE SMALL CELL HETNET USING DEEP LEARNING (DL)

5.1 Introduction

In this chapter, problem 4.8 is reformulated to be more practical. Problem 4.8 considers the UEs association to SBSs to be unchanged during all time slots, which helps easing the problem's difficulty since updating the users association in every time slot will increase the complexity of the problem and enforce it to be updated every time slot. Solving this is very challenging knowing the problem is already an MINLP. However, this assumption cannot be practical all time, specially in scenarios where the UEs locations are changing rapidly. Therefore, we reformulated the problem to include the UEs association in every time slot n.

The issue of the reformulated optimization problem is not the accuracy rather its complexity, where tackling the problem directly requires an exponential amount of computation. This is critical since mobile communication networks are highly active and their parameters required often updates. Thus, a joint approach of deep learning and optimization theory can significantly reduce the computation time and provide a near optimal solution. The main contributions of this chapter are summarized as follows:

1. We formulate an optimization problem to minimize the energy drawn from the network grid by utilizing the harvested energy and dynamic sleeping of the SBSs. The problem updates the UEs association with SBSs in every time slot to accommodate the UEs new locations.

2. A new UEs prediction method is introduced which employs a combined approach of Nonlinear Autoregressive with External input (NARX) and probabilistic Latent Semantic Analysis (pLSA) to provide an accurate multistep Neural Network prediction. A new algorithm, Nonlinear AutoRegressive with external input and probabilistic latent Semantic Analysis (NARSA), is presented to show the details of the computation steps for this new joint method.

3. The formulated problem is a Mixed Integer NonLinear Problem (MINLP) which is NP-hard. Therefore, we decomposed the problem into two parts. First, the UEs association and sleeping strategy (The binary variables). The second part is the continuous variables of the harvesting energy and transmission power where we use Deep Learning (DL) to solve it.

4. We employ a deep learning approach specifically, Artificial Neural Network (ANN), to configure the relationship between the input-output in our communication system. Moreover, ANN is trained and configured to work as a communication system to provide a computationally efficient solution every time the system parameters change.

5. An updated form of algorithm 2 in chapter 4 is developed to solve the two parts of the optimization problem iteratively. In the first part, the new algorithm is using the same centrality approach to solve the UEs' association according to their predicted location. In the second part, the algorithm uses ANN to approximate the optimization problem.

6. Lastly, we evaluate the performance of our proposed algorithm through an extensive simulation study to verify its superior performance compared with the optimal solution.

5.2 System Model and Problem Formulation

In this part, the system model is similar to the system model in Chapter 4, where UEs are randomly distributed in a designated area and are covered with SBSs that also are randomly distributed. However, in this part the UEs association is updated in every time step therefore, where the UEs' locations are collected using the pilot signal between the UE and the SBS. The following section is for the updated notations.

5.2.1 User Association and Achievable Rate

Let $x^c_{uf}[n]$ be a binary indicator that is equal to 1 if user u and SBS f are associated using resource block c in n, or 0 otherwise. Also, let $z_{fu}[n]$ be a binary indicator that is equal to 1 if user u is associated with SBS f in n, or 0 otherwise. $y_f[n]$ indicates the SBS on/off status, where $y_f[n] = 0$ if the SBS is OFF during the time slot n and $y_f[n] = 1$ if the SBS is ON. However, a deactivated SBS will keep harvesting energy and injecting it into the smart grid to serve other active SBSs.

The time-varying distance between the fth SBS and the uth user can be expressed as follow:

$$d_{uf}[n] = ||l_u - l_f|| \qquad \forall u \in U, \forall f \in F \tag{5.1}$$

where the $l_u[n]$ is the $x - y$ coordinates for the location of the user at time slot n, and the fixed location of the SBS, respectively. It follows from (5.1) that the channel power gain can be modeled as:

$$|h^c_{uf}[n]|^2 = \frac{\beta_0}{d^\alpha_{uf}[n]} = \frac{\beta_0}{||l_u - l_f||^\alpha} \tag{5.2}$$

where β_0 denotes the channel gain at the reference distance of $d_0 = 1$m, and α is the path loss exponent.

Moreover, the interference at a user u which is associated with SBS f from all other SBSs at a time slot n will be:

$$I^c_{uf}[n] = \sum_{j \neq u}^{U} \sum_{i \neq f}^{F} p^c_{ji}[n]|h^c_{ui}[n]|^2, \tag{5.3}$$

Then, the signal to interference and noise ratio SINR for every user is:

$$\gamma_{uf}^{c}[n] = \frac{p_{uf}^{c}[n]|h_{uf}^{c}[n]|^2}{I_{uf}^{c}[n] + \omega N_0}, \tag{5.4}$$

where $h_{uf}^{c}[n]$ denotes the channel gain from SBS f to user u using resource block c at time slot n, ω is the available bandwidth for every channel, and N_0 is the channel noise spectral density which is assumed to be Additive White Gaussian Noise AWGN, and ωN_0 is the noise variance σ^2. Thus, the data rate for every user using a single channel during a time slot is as follow:

$$R_{uf}^{c}[n] = \omega \log(1 + \gamma_{uf}^{c}[n]) \tag{5.5}$$

However, Eq.5.5 is non-convex we will use Taylor Expansion to linearize it, similar to the process in 4.20 to get the following:

$$\hat{R}_{uf}^{c}[n] = \omega \log(\sum_{u}^{U}\sum_{f}^{F} p_{uf}^{c}[n]|h_{uf}^{c}[n]|^2 + \omega N_0) - \hat{R}_{Ty} \tag{5.6}$$

5.2.2 Problem Formulation

In this section, the optimization problem 4.21 is updated that it will minimize the non-renewable energy consumption of the transmission power for a cooperative heterogenous network while changing the UEs association every time slot. As before, we formulate a problem where users association, sleeping strategy and energy minimization are performed within a single optimization problem. The problem can be stated as follows: given the number of users and SBSs, the problem will solve the user association, sleeping strategy and power consumption, then at every time slot the optimization problem will recalculate the users association and the transmission power and the status of the SBSs. We can then mathematically state the optimization problem as below:

Problem $\tilde{\mathcal{P}}$:

$$\underset{p^c_{uf}[n],\lambda_f[n],\mu_f[n],y_f[n],z_{uf}[n],x^c_{uf}[n]}{\text{Minimize}} \sum_{f,u,n,c=1}^{F,U,N,C} p^c_{fu,g}[n]\tau + \sum_{f=1}^{F} E_b y_f[n]$$

subject to

$$\text{C1}: R_u^{min} \leq \sum_{f=1}^{F}\sum_{c=1}^{C} x^c_{fu}[n]\hat{R}^c_{fu}[n] \quad \forall u, \forall n$$

$$\text{C2}: \sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu,r}[n]\tau \leq B_f[n-1] + \mu_f[n] \quad \forall f, \forall n,$$

$$\text{C3}: B_f[n] \leq B_{max} \quad \forall f, \forall n,$$

$$\text{C4}: \sum_{f=1}^{F}\sum_{i=1}^{n} \mu_f[i] = \sum_{f=1}^{F}\sum_{i=1}^{n} \eta\lambda_f[i] \quad \forall n$$

$$\text{C5}: \sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu}[n] \leq P_f^{max} \quad \forall f, \forall n$$

$$\text{C6}: \sum_{u=1}^{U} x^c_{fu}[n] \leq 1 \quad \forall f, \forall c, \forall n$$

$$\text{C7}: \sum_{f=1}^{F} z_{uf}[n] = 1 \quad \forall u \forall n$$

$$\text{C8}: \frac{\sum_{c=1}^{C} x^c_{uf}[n]}{\#ofUs} \leq z_{uf}[n] \leq \sum_{c=1}^{C} x^c_{uf}[n], \quad \forall\ f, \forall u, \forall n$$

$$\text{C9}: \frac{\sum_{u=1}^{U} z_{uf}[n]}{\#ofSBSs} \leq y_f[n] \leq \sum_{u=1}^{U} z_{uf}[n], \quad \forall f, \forall n$$

Constraint (C1) represents the QoS for every user. The constraints from (C2) to (C5) are dealing with energy transfer and cooperation between SBSs. Constraint (C2) represents the energy consumption causality where the BS cannot use energy more than what is available. Constraint (C3) limits the battery capacity. Constraint (C4) is for energy conservation, where the total injected energy into the smart grid equals the total received energy by all BSs. Constraint (C5) limits the maximum allowed transmission power for every BS.

Constraints from (C6) to (C9) are dealing with the UEs association and SBSs sleeping strategy.

Problem $\tilde{\mathcal{P}}$ is too complex due to the coupling of the binary and continuous variables. Moreover, due to the dynamic nature of UEs movement, where each UE is expected to move in every time slot which causes the system to update its association and sleeping strategy accordingly, problem $\tilde{\mathcal{P}}$ is required to be solved every time slot to match the changes.

One solution for this is to predict the future location of the UEs and solve the problem according to the predicted location of each UE. Therefore, we present a user mobility prediction approach, where the UEs' movements are predicted for the next N slots. This will help simplify the problem where it will be solved for the next N predicted time slots instead of every slot. Table 5.1 shows the notations that are used in this chapter.

5.3 Prediction Model for User Mobility

In this section we investigate two approaches to predict the UEs' future mobility pattern and location. First, we solely apply Nonlinear Autoregressive (NAR) method to predict each UE individually. Second, a joint approach to predict the UEs' future mobility pattern and location by exploiting ANN Nonlinear Autoregressive with External input (NARX) and probabilistic Latent Semantic Analysis (pLSA) to provide an accurate multistep Neural Network prediction. This joint approach leverages ANN NARX outstanding results in time series prediction tasks and the (pLSA) ability in detecting hidden patterns between different UEs.

5.3.1 Nonlinear Autoregressive (NAR) Time Series Prediction

The Nonlinear Autoregressive (NAR) are types of Neural Networks that are used to forecast samples framed in a one-dimensional time series. In many cases, time series applications are characterized by highly variant data, e,g,. human movement. Highly variant movement, as in the human case, can only be modeled with a nonlinear model, where the

Table 5.1List of Notations used throughout chapter 5

Notation	Description
$x^c_{fu}[n]$	The binary association between f and u through c during time slot n.
$z_{fu}[n]$	The binary association between f and u during time slot n.
$y_f[n]$	The SBS f ON/OFF status during time slot n.
$y(n)$	The true UE movement data that is used in training NARX networks.
$\hat{y}(n)$	The output prediction of the UE movement in NARX.
$k(n)$	The external set of data that are used to increase the accuracy of the prediction in NARX.
$e(n)$	The error caused by the imperfect prediction in NARX networks.
$h(.)$	The unknown function that map the relation between the input and the output.
m_k	The unobserved class variable k.
$l[n]$	The UE's location during time slot n.
$\mathcal{L}[n]$	The direction of the movement during time slot n.
$P(u, \mathcal{L}[n])$	The joint probability of the observed data.
$m(u, \mathcal{L}[n])$	The data point of UE u that made a movement $\mathcal{L}[n]$.
$P(m_k/u)$	The probability that UE u is following mobility class m_k at any given time.
$P(\mathcal{L}[n]/m_k)$	The probability of the direction of movement given the mobility class m_k.
α^u_k	The UE u that belongs to class k.
α^*_k	The UE that has the highest probability in class k.
Ω_k	The group that contains all UEs that belong to class k.

input data is processed by a nonlinear autoregressive model. NAR networks is used to predict the value of certain time series data by using the past data. The following equation shows how NAR networks work:

$$y(n) = f(y(n-1), y(n-2), ..., y(n-d)) \tag{5.7}$$

where the function f is unknown and the neural network is used to approximate. This equation describes how NAR works in predicting the future values of y. In every step, NAR uses the past d values of y to predict the future data point in series y. In case of UEs movement, NAR will use the past d locations of each UE to predict the next location of this UE.

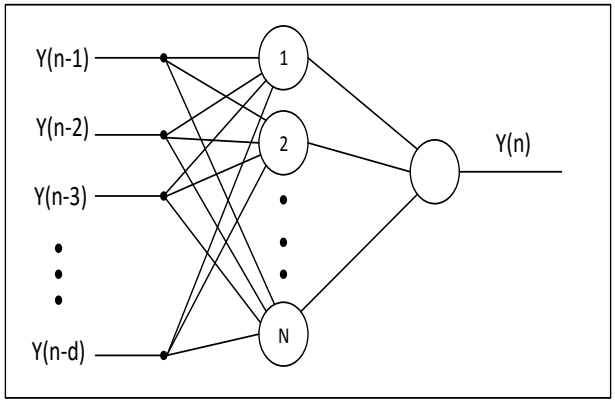

Figure 5.1 Standard Topology for NAR Networks.

Figure 5.1 shows the topology of NAR Network with three layers, input layer, hidden layer with N neurons and an output layer. The figure depicts the training stage where it is in open loop. In the typical case the output $y(n)$ is fed back to the feedforward neural network as part of the standard NAR. Since in the training stage the true output is available it is better to be used instead of feeding back the estimated output. This has the advantage that the input to the network is more accurate. After training the network, the output will be fed back as a delayed input to predict future points. The advantage of NAR is its simple structure and requires less computation time. However, for multistep prediction where the predicted output $\hat{y}(n)$ is used to predict future results, the error will adds up in every step which is accumulated to produce inaccurate results. Therefore, NARX is better in predicting future works since it relies not only on the data but on other external set of correlated data.

5.3.2 Nonlinear Autoregressive with External input (NARX) Time Series Prediction

In many applications, there are an important correlation between the predicted time series and some other external data. Some stock exchange prices are correlated with certain

durations of the years. For example, technology companies stock prices increase during the holiday seasons or after offering new products while other times of the years the prices either decrease or stay unchanged. Thus, utilizing the time of the year as an external data could benefit the time series model in providing accurate prediction [83]. Moreover, for data that hold long dependency NARX are proven that can retain information for two to three times as long as conventional recurrent neural networks [84]. The following equation shows how NARX networks work:

$$y(n) = h(y(n-1), y(n-2), ..., y(n-d), k(n-1), k(n-2), ..., k(n-d)) \tag{5.8}$$

Similar to NAR networks, NARX predicts the future of y according to the past d values, where for every one future prediction NARX will employ the past data to make its prediction. However, NARX includes the external data set $k(n)$ to approximate the function h. Moreover, NARX network has two configurations: Parallel Architecture and Series-parallel Architecture.

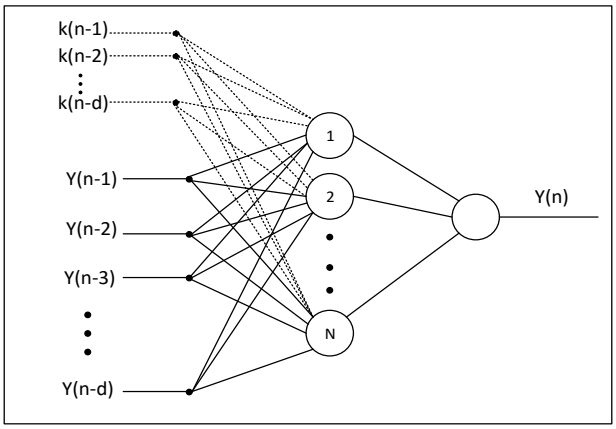

Figure 5.2 Network Topology for NARX Networks in The Training Stage.

1. Series-Parallel Architecture: NARX in this configuration the future value of the time series $y(n+1)$ is predicted from the past true values of $y(n)$ and the past values of external data $k(n)$. Figure 5.2 shows the Series-Parallel architecture where the input

is the combining the true past values of the data and the external past data that are shifted by d data points. This configuration is usually used in the training stage, because using the available true past values as input will results in more accurate prediction and the resulting network architecture is feedforward where the Multi-Layer Perceptron (MLP) networks can be used [85]. The following equation shows how NARX works in this configuration.

$$\hat{y}(n+1) = h(y(n), y(n-1), y(n-2), ..., y(n-d), k(n), k(n-1), k(n-2), ..., k(n-d)) + e(n) \tag{5.9}$$

where $\hat{y}[n]$ is the predicted value and $e[n]$ is the prediction error.

2. Parallel Architecture: This is the standard configuration of NARX where the output is fed back to the network as an input. As in Figure 5.3 the prediction is performed using the past values of the external data and from the past predicted values of the time series. This configuration is used to perform the actual prediction since the true values are not available. Thus, the NARX equation will be as follows:

$$\hat{y}(n+1) = h(\hat{y}(n), \hat{y}(n-1), \hat{y}(n-2), ..., \hat{y}(n-d), k(n), k(n-1), k(n-2), ..., k(n-d)) + e(n) \tag{5.10}$$

NAR is simpler to perform and requires less computation power. However, NARX can have better performance, specially when there is strong correlation between the predicted data and the the external data. Therefore, in the following section a statistical method is employed to detect the correlation between different data sets to use them in performing the prediction.

5.3.3 probabilistic Latent Semantic Analysis (pLSA)

pLSA was originally introduced to derive a representation of the observed variables in terms of certain hidden variables [86]. pLSA is a statistical technique used for the analysis

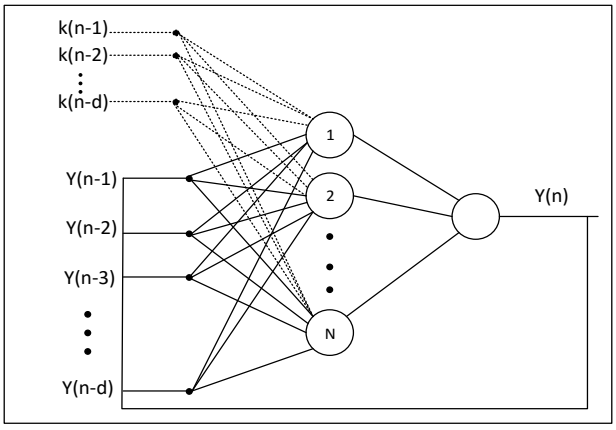

Figure 5.3 Standard Topology for NARX Networks in The Prediction Stage.

of co-occurrence data which is based on a mixture decomposition derived from a latent class model. The goal of pLSA is to map high-dimensional data to a lower dimensional representation called latent semantic space. Thus, pLSA is used to reveal hidden relationship between the data of interest. Due to its generality, pLSA has been used as an analysis tool for a wide range of applications, e.g., information retrieval, data caching, UE's movement prediction and information filtering.

Mobile UEs usually follow daily movement patterns, e.g., some UEs use public transportation daily, others drive to business districts. For example, every morning a group of users move from their homes (usually in the suburbs) to their work (usually in downtown), while during afternoons they move back to their homes. Therefore, finding the pattern that each UE is following every day is very helpful in predicting their movement. Thus, pLSA is used in this work to reveal the hidden patterns that each UE is using and employ this pattern as an external data, i.e., $k(n)$ in the previous section, to provide more accurate predictions.

Moreover, the pLSA model is a latent variable model for co-occurrence data which associates an unobserved class variable $m_k \in m_1, m_2, ..., m_K$ with each observation, where the observation is the user's movement from one location to another.

Let us denote the user's location at time slot n to be equal to $l[n] \in \mathbb{A}$, where, $\mathbb{A}$ is a predefined area. The user's movement from location a to location b is denoted by $l^a[n] \rightarrow l^b[n]$. Here, we are considering the direction of the user's movement without considering the traveled distance. Thus, we denote $\mathcal{L}[n]$ as the direction of the movement during time n, where the movement direction is either north, south, east or west. Hence, the probability of the movement toward direction $\mathcal{L}[n]$ is:

$$P(\mathcal{L}[n]) = \frac{\text{Number of movements toward direction } \mathcal{L}[n]}{\text{Total number of movements}} \tag{5.11}$$

Moreover, every observed data item is a pair of data $(u, \mathcal{L}[n])$, which means that user u moved toward direction $\mathcal{L}[n]$. Therefore, the joint probability of the observed data will be:

$$P(u, \mathcal{L}[n]) = P(u)P(\mathcal{L}[n]/u) \tag{5.12}$$

However, equation (5.12) provides no information about the class that the user and movement belong to. Thus, the joint probability can be rewritten as the following:

$$P(u, \mathcal{L}[n]) = \sum_{k=1}^{K} P(u)P(m_k|u)P(\mathcal{L}[n]|m_k) \tag{5.13}$$

where parameter $m_k \in [m_1, m_2, ..., m_K]$ denotes the mobility class, and the probability that a user u is following mobility class m_k at any given time, is defined as $P(m_k|u)$. Moreover, the probability of the direction of movement given the mobility class m_k is $P(\mathcal{L}[n]|m_k)$, while $P(u) = \frac{m(u,\mathcal{L}[n])}{\sum_{u=1}^{U} m(u,\mathcal{L}[n])}$ denotes the probability that u made a movement, where $m(u, \mathcal{L}[n])$ is the number of times UE u made a movement.

However, the two probabilities $P(\mathcal{L}[n]|m_k)$ and $P(m_k|u)$ cannot be calculated analytically, since the classes m_k are unknown. Therefore, their values must be estimated and updated iteratively. Moreover, there is no known formula that calculates the optimal number of classes K in the network, where K values can range from 1 class to number of classes equal to the total number of UEs in the network. Thus, employing different K to find the optimal trade off between the performance and computation requirement is the only way.

The Expectation Maximization (EM) algorithm is a well-known algorithm that is used to compute Maximum Likelihood Estimates (MLE) [87]. In many estimation models the direct access to the data necessary for estimating the parameters of the distribution is impossible. EM algorithm is ideally suited for this sort of problems [88]. The EM algorithm consist of two consecutive steps: an expectation step, followed by a maximization step. The first step: the expectation (E) step where posterior probabilities are calculated for the latent variables, based on the current estimates of the parameters. The second step the Maximization (M) step where parameters are updated to maximize the expected complete data log-likelihood, which depends on the posterior probabilities computed in the E step.

In our model the classes $m_k \in M = \{m_1, m_2, ..., m_K\}$ are unobserved and unknown, which can only be estimated using the EM algorithm. Applying the EM algorithm to the pLSA model as in [86] introduces the two steps as follows: The expectation (E) step the Bayes' estimator is calculated for the latent variables, based on the current values of $P(m_k|u)$ and $P(\mathcal{L}[n]|m_k)$. The Maximization (M) step is used to update the parameters to maximize the expected complete data log-likelihood, which depends on the $P(m_k|u)$ and $P(\mathcal{L}[n]|m_k)$ equations that are computed in the E step.

In the Expectation step, we use Bayes' estimator to calculate the posterior probabilities based on the current estimates of the parameters. The Bayes' estimator is as follows:

$$P(m_k|u, \mathcal{L}[n]) = \frac{P(\mathcal{L}[n]|m_k, u)P(m_k|u)}{\sum_{m \in \mathbb{M}} P(\mathcal{L}[n]|m_k, u)P(m_k|u)} \tag{5.14}$$

In the Maximization step, the expectation of the complete data log-likelihood $\mathbb{E}[\mathbb{L}]$ is maximized as follow:

$$\mathbb{E}[\mathbb{L}] = \sum_{u=1}^{U} \sum_{\mathcal{L}[n]} m(u, \mathcal{L}[n]) * \sum_{k=1}^{K} P(m_k|u, \mathcal{L}[n]) log\big[P(\mathcal{L}[n]|m_k, u)P(m_k|u)\big] \tag{5.15}$$

where $m(u, \mathcal{L}[n])$ indicates the number of times the user moved according to direction $\mathcal{L}[n]$ during time slot n. Then, the following two re-estimation equations are used in the M-step:

$$P(\mathcal{L}[n]|m_k) = \frac{\sum_u m(u, \mathcal{L}[n])P(m_k|u, \mathcal{L}[n])}{\sum_{n=1}^{N} \sum_u m(u, \mathcal{L}[n])P(m_k|u, \mathcal{L}[n])} \tag{5.16}$$

$$P(m_k|u) = \frac{\sum_u \sum_{n=1}^{N} m(u, \mathcal{L}[n]) P(m_k|u, \mathcal{L}[n])}{\sum_{n=1}^{N} \sum_u m(u, \mathcal{L}[n])} \tag{5.17}$$

After evaluating Eqs. (5.16) and (5.17) each UE will be assigned to the class with the highest probability, which it will share with other UEs where they share a common pattern. Let us denote α_k^u as the UE u that belongs to class k and Ω_k as the group that contains all UEs that belong to class k. Lets denote α_k^* as the UE that has the highest probability in class k.

In the prediction stage using NARX network, every group Ω_k will choose the UE that has the highest probability in $P(m_k|u, \mathcal{L}[n])$ as the external data $k(n)$ for predicting the movement of every UE in that Group. This ensures the external data that is used in NARX has a strong correlation with all of its associates.

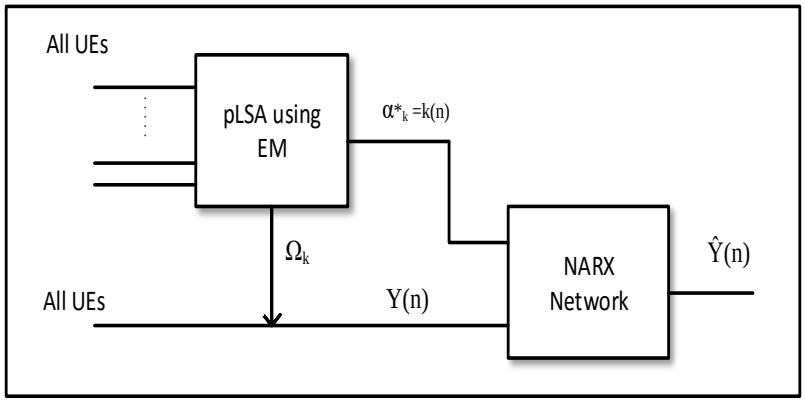

Figure 5.4 Diagram Showing the Stages of NARSA Algorithm.

Figure 5.4 depicts the flowchart of the combined approach between NARX and pLSA. First, all UEs are fed to the EM algorithm to decide the UE with highest probability α_k^* in each class k. Then α_k^* is fed to NARX network as the external data $k(n)$. Then, each UE within the group Ω_k is fed to NARX as the input $y(n)$. Finally, NARX network is used to estimate the function h in Eq.(5.9) during the training stage. In the prediction stage,

the trained network is used to predict the future location of each UE and this output is fed back to get the next movement. This process is repeated for every UE in every class.

Algorithm 3: UEs' Movement Prediction using NARX and pLSA (NARSA).

Input: $u; \mathcal{L}[n]; K; \epsilon$
Initialize $P^0(m_k|u)$;$P^0(\mathcal{L}[n])|m_k)$; $\Delta \leftarrow \infty$;$i \leftarrow 1$
while $\Delta \geq \epsilon$ **do**
    Compute $P^{[i]}(m_k|u, \mathcal{L}[n])$ in (5.14) using $P^{[i-1]}(m_k|u)$ and $P^{[i-1]}(\mathcal{L}[n])|m_k)$;
    Update $P^{[i]}(m_k|u)$ and $P^{[i]}(\mathcal{L}[n])|m_k)$ using (5.16) and (5.17);
    Compute the expectation of the complete data log-likelihood $\mathbb{E}[\mathbb{L}]^{[i]}$ in (5.15);
    $\Delta = |\mathbb{E}[\mathbb{L}]^{[i]} - \mathbb{E}[\mathbb{L}]^{[i-1]}|$;
    $i \leftarrow i + 1$;
end while
Output: $\mathbf{P}(m_k|u)$, $\mathbf{P}(\mathcal{L}[n])|m_k)$,$\alpha_k$,$\alpha_k^*$,$\Omega_k$.
for k=1:K **do**
    $k(n) \leftarrow \alpha_k^*$
    for $\forall$ UE $\in \Omega_k$ **do**
        $y(n) \leftarrow \alpha_k^u$
        Train NARX Network by estimating Eq.5.9
    end for
end for
Output: $\hat{y}(n)$

Algorithm 3: NARSA is constructed of two parts: The first is the EM algorithm, and the second is the NARX network training. First we initialize the probabilities of $P(\mathcal{L}[n])|m_k)$ and $P(m_k|u[n])$ with random initial values, then the algorithm will alternate between the Expectation and maximization parts i.e., between Eq.(5.14) and Eqs.(5.16) and (5.17). In every step $\mathbb{E}[\mathbb{L}]^{[i]}$ is calculated using eq.(5.15), and compared to $\mathbb{E}[\mathbb{L}]^{[i-1]}$. If the difference is less than Δ the algorithm will stop and $\mathbf{P}(m_k|u)$, $\mathbf{P}(\mathcal{L}[n])|m_k)$ will be the local optimal values and α_k^* is computed. Otherwise the algorithm will repeat the previous two steps again. At the second part, NARSA will employ the results from EM to feed NARX network the external data and train it to predict $\hat{y}(n)$. The authors of [87], proved that $\mathbb{E}[\mathbb{L}]$ is monotonically increasing in every iteration and that EM will converge within finite

iterations. However, since the EM is nonconvex, the algorithm is repeated with different initials multiple times to find the maximum value for $\mathbb{E}[\mathbb{L}]$.

The amount of consumed energy for computing algorithm 3 depends on the number of UEs in the network, where the consumed energy increases as the number of UEs increases. The energy required for computing NARSA algorithm is worth investigating. However, since NARSA is computed in Cloud Radio Access Networks (CRAN), it is out of scope in this thesis, which focuses on the consumed energy within the SBSs. The amount of consumed energy for computing algorithm 3 depends on the number of UEs in the network, where the consumed energy will increase is the UEs number increases. The energy required for computing NARSA algorithm is worth investigating. However, since NARSA is computed in Cloud Radio Access Networks (CRAN), it is out of scope in this thesis, which focuses on the consumed energy within SBSs.

5.4 Artificial Neural Networks (ANN) in Communication Systems

In this section we present a new approach in solving the optimization problem ($\mathcal{P}$) by implementing a deep learning method by employing ANN. Optimization problem ($\mathcal{P}$) is able to produce a very accurate solution to the proposed system model. However, the problem is very complex and solving it for the optimal solution is extremely difficult that it becomes unpractical to apply the problem in highly dynamic communication systems. Thus, deep learning with ANN is a promising new methodology in solving such problems within a reasonable computation time[72]. Theoretically, ANN is able to derive the relationship between any pair of input-output. This makes ANN a very powerful tool in solving complex models, where traditional optimization tools are unpractical.

The general approach is built from four consecutive stages as shown in Figure 5.5, system parameters, mathematical modeling, ANN stage and system output.

1. System Parameters: This stage is the first stage where all the required parameters are generated or decided, e.g., number of users, number of SBSs, channel gain, expected

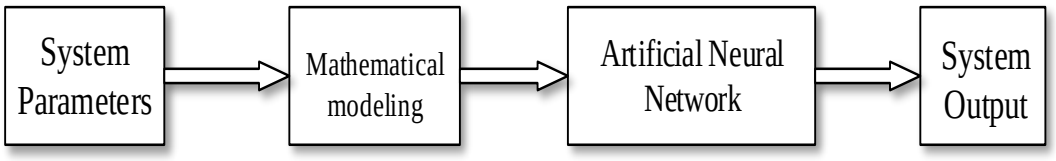

Figure 5.5 A Block Diagram Showing the Steps in Using ANN in Communication Systems.

harvested energy...etc. This stage is essential since it will decide the complexity of the problem and the accuracy of the ANN output. Moreover, these parameters are used in solving the optimization problem and also are used as the input for ANN.

2. Mathematical Modeling: This stage includes and solves the optimization problem that models the designed system. The optimization problem here is solved using the traditional optimization theory to find the optimal solution of all the variables. Thus, this stage is used to generate accurate solution of the problem and employ it to train the ANN in the next stage. Therefore, the problem is solved several times with different system parameters in every time to generate the training data set that will be used in the next stage.

3. Artificial Neural Network: ANN stage is implemented to approximate the input-output relation between the system parameters and the system design. The generated data set from the optimization problem is used to train the ANN to learn the input-output map. According to [89] ANN is considered as a universal approximator where it virtually can learn any input-output relation.

4. System Output: This is the final stage where the system output is used to design the communication network, i.e., the resource distribution. Moreover, the output is updated using the trained ANN every time one of the system parameters has changed without repeating the four stages.

However, the four stages need to be modified to fit our particular optimization problem. First, a single ANN cannot perform a classification and regression at the same time, since the two use different loss functions, where problem 4.8 is deciding the user association and sleeping strategy, which is a classification problem, and the power consumption and energy distribution, which is a regression problem. Therefore, in this section, we will decouple the user association and sleeping strategy from the energy transfer and power transmission in problem 4.8. The user association and sleeping strategy will be solved by the method that was developed in the previous chapter, while the remaining optimization problem is as follows:

$$
\begin{aligned}
&\text{Problem } \tilde{\mathcal{P}}2: \\
&\underset{p^c_{uf}[n],\lambda_f[n],\mu_f[n]}{\text{Minimize}} \\
&\mathbb{S} = \sum_{f,u,n,c=1}^{F,U,N,C} p^c_{fu,g}[n]\tau + \sum_{f=1}^{F}\sum_{n=1}^{N} E_b \bar{y}_f[n] \\
&\text{subject to} \\
&\text{C1}: R_u^{min} \leq \sum_{f=1}^{F}\sum_{c=1}^{C} \bar{x}^c_{fu}[n]\hat{R}^c_{fu}[n] \quad \forall u, \forall n \\
&\text{C2}: \sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu,r}[n]\tau \leq B_f[n-1] + \mu_f[n] \quad \forall f, \forall n, \\
&\text{C3}: B_f[n] \leq B_{max} \quad \forall f, \forall n, \\
&\text{C4}: \sum_{f=1}^{F}\sum_{i=1}^{n} \mu_f[i] = \sum_{f=1}^{F}\sum_{i=1}^{n} \eta\lambda_f[i] \quad \forall n \\
&\text{C5}: \sum_{u=1}^{U}\sum_{c=1}^{C} p^c_{fu}[n] \leq P_f^{max} \quad \forall f, \forall n
\end{aligned}
\tag{5.18}
$$

where $\bar{y}_f[n]$ and $\bar{x}^c_{fu}[n]$ are the sleeping strategy and user association that are determined outside the optimization problem. Therefore, the problem here is a convex Nonlinear Problem NLP. This formulation is solved several times to produce synthetic data which will be used in training the ANN.

Solving problem ($\tilde{\mathcal{P}}2$) will generate four continuous variables $p^c_{uf,r}[n], p^c_{uf,g}[n], \lambda_f[n], \mu_f[n]$, and it is worth noting that beside the harvested energy, all other parameters can be considered constant for relatively long time. The minimum required rate, maximum power

consumption, maximum battery level and energy transfer coefficient cannot be updated in every time slot, since updating these parameters requires new hardware to be installed inside each SBS. The channel gain, on the other hand, is related to the location of the UE which can be predicted in the same way the UEs' movement is predicted. Thus, keeping these parameters constant during the training stage of ANN is more practical than including them during this stage. Therefore, to produce one pair of data from problem ($\tilde{\mathcal{P}}2$) we only change the harvested energy and keep everything else constant. This is understandable since the harvested energy is a random process that is changing every time slot. Hence, the generated synthetic data for every run consist of pairs of $hr_f[n]$ and $\mathbb{T} = p^c_{uf,r}[n], p^c_{uf,g}[n], \lambda_f[n], \mu_f[n]$.

However, the data pair $hr_f[n], \mathbb{T}$ has a problem with the big difference in dimensionality between the input and the output. $hr_f[n]$ is a matrix with dimensions $f \times n$, while $p^c_{uf,r}[n]$ and $p^c_{uf,g}[n]$ both has dimensionality of $f \times n \times c \times u$. This difference in the dimensionality degrade the performance of ANN since few input parameters are required to predict large number of output variables. To overcome this problem we propose two solutions, first, we employ the total consumed power in every channel in each SBS during time slot, i.e., $p^c_{f,r}[n]$ and $p^c_{f,g}[n]$ instead of $p^c_{fu,r}[n]$ and $p^c_{fu,g}[n]$. This will decrease the dimensionality of the output since the UEs' numbers are high. The second is training three ANNs independently instead of a single ANN, the first and second ANNs will have $hr_f[n]$ as input and as an output $p^c_{f,r}[n]$ and $p^c_{f,g}[n]$, respectively, while the output of the third ANN is $\lambda_f[n]$ and $\mu_f[n]$. The training of the three ANN will be conducted in parallel. Thus, the modified diagram of Figure 5.5 will be updated as shown in Figure 5.6:

Figure 5.6 shows the modified block diagram of the proposed system. The proposed system consists of two parts, the UE association and sleeping strategy and the optimization problem with ANN. The UE association and sleeping strategy implemented by using the developed Algorithm 2, where we use BSC to assign certain weight to each SBS. This assignment is to decide the deactivation of the SBSs according to each weights, the higher the weight the higher the probability of this SBS to be deactivated. Moreover, this part

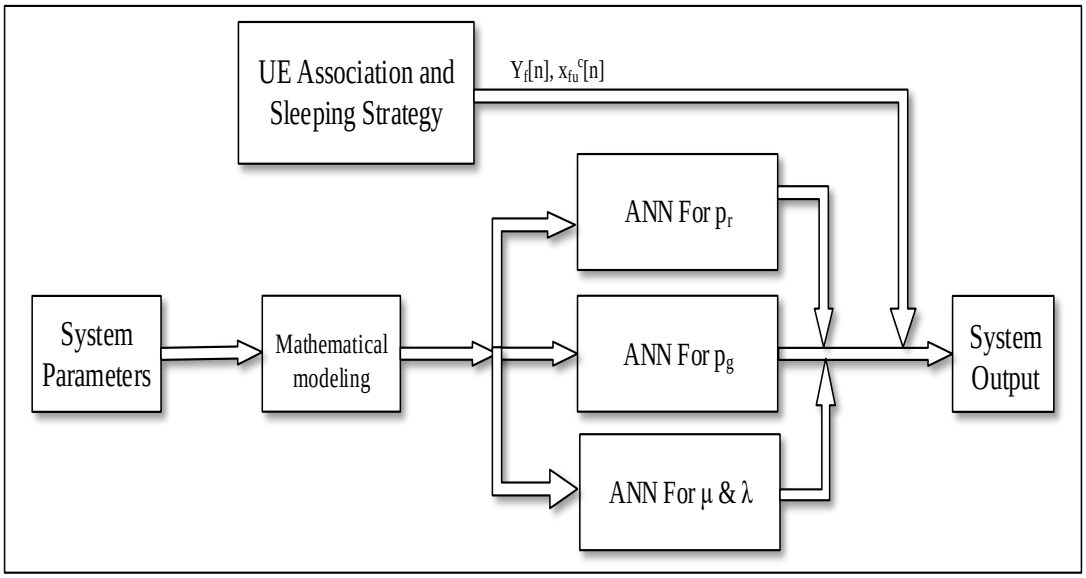

Figure 5.6 A Block Diagram Showing the Modified Steps in Using ANN in Communication Systems.

takes advantage of the developed Algorithm 3 that employ NARX and pLSA to predict the UEs mobility. The second part is applying the ANN by generating the data from the mathematical model that was presented in 5.18. As observed from the previous figure the ANN part is split into three parallel ANNs to train the three outputs $p^c_{f,r}[n], p^c_{f,g}[n], \lambda_f[n], \mu_f[n]$. At the final phase the two parts are combined to produce the full output design, i.e., $p^c_{f,r}[n], p^c_{f,g}[n], \lambda_f[n], \mu_f[n], y_f[n]$ and $x^c_{fu}[n]$. These two parts are performed offline, since solving the mathematical model and training the ANN usually take long time, hence not suitable for online applications. The stages that are implemented to reach the final design are summarised as follows:

1. Algorithm 3 is activated to predict the next step or steps for the UEs movement using NARX and pLSA. This will generate the UE predicted locations $\bar{y}[n]$ using the closed loop as in Figure 5.3 that is applied in the prediction stage.

2. Next, the predicted locations $\bar{y}[n]$ are used as input for Algorithm 4 to calculate the UE association and sleeping strategy according to the BSC. The output of this stage are the binary variables $y_f[n]$ and $x^c_{fu}[n]$.

3. Then, Problem 5.18 is used to generate the synthetic data that consists of the continuous variables of the transmission power and energy transfer. The data pairs that are generated from this stage are the changing parameter $hr_f[n]$ and the outputs $p^c_{f,r}[n], p^c_{f,g}[n], \lambda_f[n], \mu_f[n]$.

4. Then, the generated synthetic data is used to train the ANN to be able to approximate the input output relation of Problem 5.18. The ANN is split into three parallel ANNs that are trained to approximate the three different continuous variables.

5. Finally, the results from the two parts are used to design the communication system.

After training the ANNs, it is possible to update the power allocation and energy transfer without having to solve the optimization problem every time the system parameters change, i.e., the new system parameter $hr_f[n]$ needs only to be fed as an input to the already trained ANN to obtain the new power transmission and energy transfer. This process requires a negligible computation time since the ANN is already trained and requires performing few elementary functions to compute the output. The following algorithm is presented to show the procedure that the proposed scheme follows to update the communication system every time the system parameter changes during operational time.

Algorithm 4 shows the steps that are implemented to configure the systems design. First, the system parameters are fed to the algorithm, i.e.,$hr_f[n]; R_{min}; P_{max}; \omega N_0$, also algorithm 3 is used to predict the UEs' future locations to calculate the channel gain $h^c_{fu}[n]$. Second, we initialize the power transmission $p^c_{fu}[n]$ randomly and set all SBSs ON, i.e., $y_f[n] = 1$. Then steps from $5-13$ invoke the SBC algorithm to assign UEs to the SBS with the highest SINR and calculate the BSC for each SBS. Then, steps $14-16$ use the ANN to generate the candidate solution for $p^c_{r,f}[n], p^c_{g,f}[n], \lambda_f[n], \mu_f[n]$. Steps $17-23$ make sure that deactivating

Algorithm 4: The ANN Implementation Using BSC and NARSA.

Input:$hr_f[n]; R_{min}; P_{max}; \omega N_0$;
Get the predicted UEs' locations $\hat{y}(n)$ from Algorithm 3 and calculate $h^c_{fu}[n]$ accordingly
Initialize: $p^c_{fu}[n]^{[0]}; y_f^{[0]} = 1; k = 0$
while True **do**
  Calculate SINR $\forall u, \& \forall f$, and associate users with SBSs according to the highest SINR.
  if $z_{uf}[n] = 0 \quad \forall u \in U$ **then**
    $y_f[n] := 0 \quad \forall f \in F$
  end if
  for `<f=1:|Active SBSs|>` **do**
    Calculate ν_f, ϕ_f
    $BSC_f = \nu_f \phi_f$
  end for
  $y_{f'}^{[k]}[n] \leftarrow 0$, BS f' is the BS with the highest BSC, and associates its users with the neighboring BS $\bar{x}^{c[k]}_{fu}[n]$.
  Use ANN_{p_r}, to generate $[p^c_{r,f}[n]]$
  Use ANN_{p_g}, to generate $[p^c_{g,f}[n]]$
  Use $ANN_{\lambda,\mu}$, to generate $[\lambda_f[n], \mu_f[n]]$
  if Any $p_f[n] \geq p_{f,max}$ **then**
    $\nu_{f'} \leftarrow 0$
  else if $p_f[n]^{[k]} \leq p_f^{[k-1]}$ **then**
    $\mathbb{T}^* := [p^c_{r,f}[n] y_f[n], p^c_{g,f}[n] y_f[n], \lambda_f[n], \mu_f[n]]^{[k]}$
    $x^{*c}_{fu}[n] := \bar{x}^{c[k]}_{fu}[n]$
    $y^*_f[n] := \bar{y}^{[k]}_f[n]$
  end if
  $k \leftarrow k + 1$
  if $\nu_f = 0 \quad \forall f \in F$ **then**
    Break
  else
    Go to step 4
  end if
end while
Output: $\mathbb{T}^*, y^*_f[n], x^{*c}_{fu}[n]$

Table 5.2Simulation Parameters

Parameter	Value	Parameter	Value
P_{max}	4 watts	R_{min}	1 Mbps
N_0	-174 dbm/Hz	ω	5 MHz
B_{max}	600 Joules	τ	10s
η	0.9	E_b	20 Joules

the SBS will not force other SBSs to consume more power than its maximum allowance, and compare this solution to the previous one it assigns as the optimal solution. Finally, these steps are repeated for all SBSs, then the final answer is produced as an output.

5.5 Simulation Results

In this section we evaluate the performance of algorithms 3 and 4, on the Mobile Data Challenge (MDC) data set [90] [91]. DMC data contains GPS traces for both pedestrians and vehicular from Lake Geneva region in Switzerland. The data is gathered from participants using GPS where their locations are recorded every 10 seconds for over one year of time. The data is processed to improve their quality, i.e., some outliers are removed. Figures 5.7, 5.9 and 5.10 evaluate the performance of Algorithm 3 to predict UEs' movements and compare it with NAR method. The system parameters are listed in Table 5.2 unless stated otherwise.

Figure 5.7 shows the comparison between NAR and NARX. In this figure we use an ANN with three layers, the first is the input layer, the second is the hidden layer and the third is the output layer. The hidden layer consists of 10 neurons with Tanh function. In this part we applied a delay of 4 steps, i.e., $d = 4$. Moreover, 10 UEs are used from DMC data set with $m_k = 3$ classes. One UE data from every class is used as an external data in NARX, while NAR system does not require external data. Moreover, 2000 data pairs are used to train the network, with 70% for training, 15% for validation and 15% for testing purposes. The figure shows the accumulated error percentage of both NAR and

NARX systems. As shown in the figure NAR predicts the next two steps with very accurate prediction, however, after the fifth prediction NAR accumulated error, which is defined as how far the prediction is compared to the actual location, increases rapidly. On the other hand, NARX kept a relatively accurate prediction until the tenth step with error percentage of around 10%. This is understandable since the external data has a strong correlation with the input data that is being predicted.

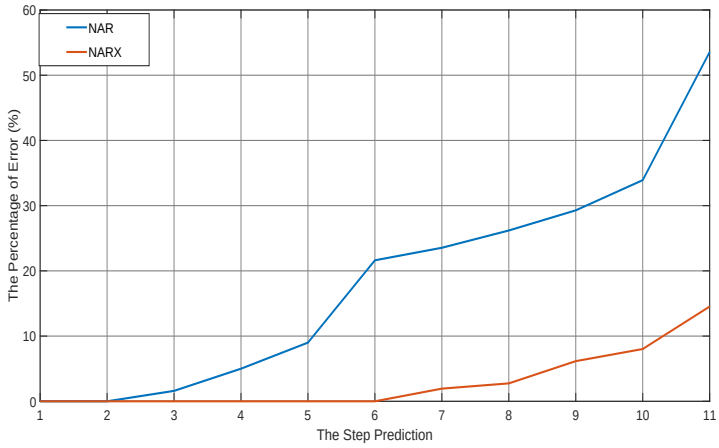

Figure 5.7 The Performance Comparison between NARX and NAR for 11 Steps.

With the same setting as in Figure 5.7, Figure 5.8 show the probability of success as the number of prediction increases. In this figure we also compare NAR with NARX, and as the figure depicts both approaches start with high probability of success but as the number of prediction steps increases the performance of NAR decreases rapidly, while the performance of NARX decreases slowly. This is understandable since the highly correlated external input that is used in NARX helped sustaining the prediction for more steps.

Figure 5.9 shows the effect of the number of classes on the performance of NARX compared to NAR. The number of classes that are used in pLSA is essential to group UEs with strong correlation and results in better prediction. Therefore, in Figure 5.9 we used 10 UEs as before with 2000 training pairs for each UE. As for the ANN, we used the same architecture as Figure 5.7 with 70% for training, 15% for validation and 15% for testing

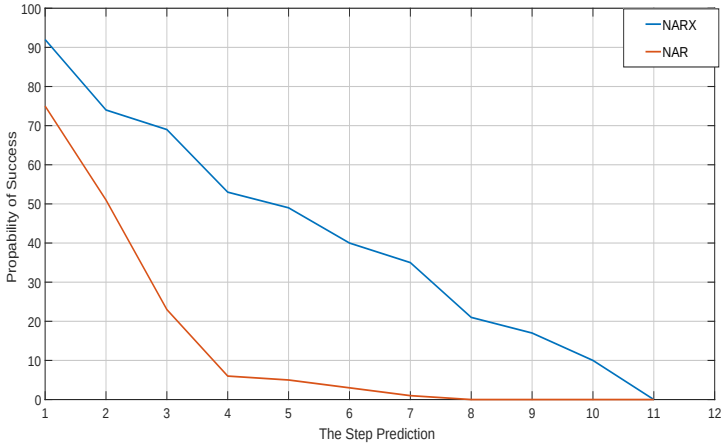

Figure 5.8 The probability of success as the number of prediction steps increases.

with three layers and 10 neurons. As the figure shows, NARX outperform NAR for 2 or more classes. This is understandable since using pLSA helps find correlation between different UEs which results in NARX to employ this correlation to improve its prediction. On the other hand, NAR does not benefit from any extra knowledge that comes from the correlation. Moreover, NARX accuracy increases as the number of classes increase. However, after $m_k = 4$ the percentage error did not improve as the number of classes increased. This is because the UEs are classified into 4 classes even if we increased the number of classes more than 4, i.e., the UEs can be grouped into at most 4 classes.

Figure 5.10 shows the change of power consumption due to the prediction inaccuracy. In this figure we solved problem $\mathcal{P}$ according to the actual locations and predicted locations of both NAR and NARX. We used 8 SBSs that are placed to cover the area of the future movement of 1 UE. First the problem was solved according to the actual locations in order to decide the user association in every time slot n. Then, using this association we changed the UEs locations according to the prediction to calculate the consumed power. The figure shows how the inaccurate prediction affected the energy consumption in both NAR and NARX. In NAR, after the fifth step the assigned SBS to the UE cannot maintain the communication since it used the maximum allowed power, which is represented by the

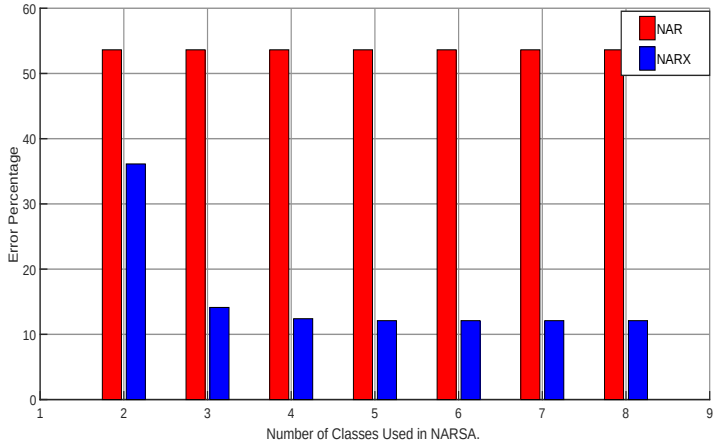

Figure 5.9 The performance of NARX and NAR in different number of classes.

straight red line. On the other hand, NARX has a more accurate prediction of the system that kept the UE associated with SBS until the tenth step.

The following figures show the performance of ANN in approximating the transmission powers $p^c_{f,g}[n]$ and $p^c_{f,r}[n]$ and the performance of ANNp_g and ANNp_r with $hr_f[n]$ as the input. In the following two figures we use 20 UEs and 10 SBSs and 10 time slots, where the UEs assigned to the SBS with the highest SINR. Problem 5.18 is solved several times with different inputs to generate the data that is used to train the ANN. Furthermore, a four layered ANN is applied, where the first layer is the input layer, the second and the third layers are the hidden layers with 10 neurons and 4 neurons, respectively, the fourth layer is the output layer. The activation function for the hidden layers is Rectified Linear units (ReLu), while the output layer uses a Linear function. We divided the generated data into three parts 70% for training, 15% for validation and 15% for testing. Figure 5.11 shows the Mean squared Error (MSE) of both $p^c_{f,g}[n]$ and $p^c_{f,r}[n]$ as the size of the training matrix increases. The training matrix consists of the number of times problem 5.18 is solved to generate the data. As the figure shows the MSE decreases as the training size increase which is expected because the greater the training size is the higher the accuracy will be.

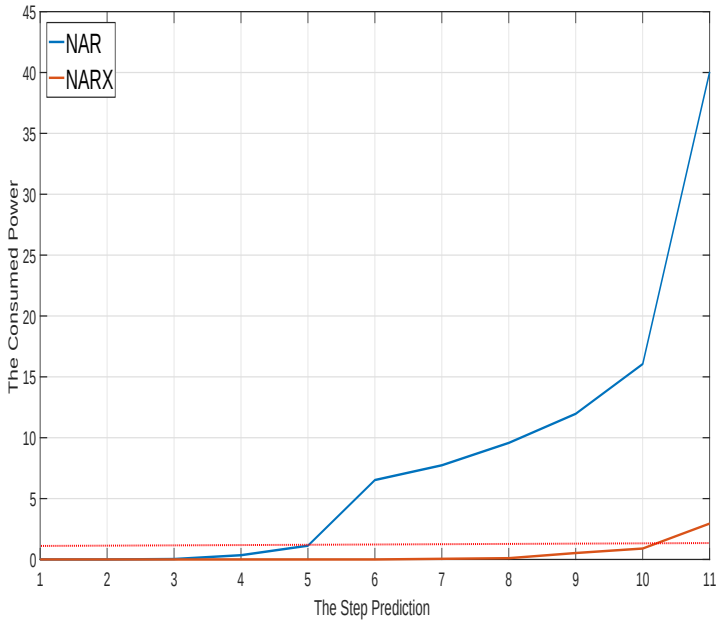

Figure 5.10 The Difference in Consumed Power Due to Prediction Error.

Figure 5.12 shows the computation time needed to train the ANNp_g and ANNp_r as the training matrix increases. From the figure, as the training size increases the computation time increases for both ANNs. The computation time for both ANNs is almost the same since both p_g and p_r have similar training data. However, even as the computation time is very high this will not affect the computational complexity of the system since the training stage is performed offline.

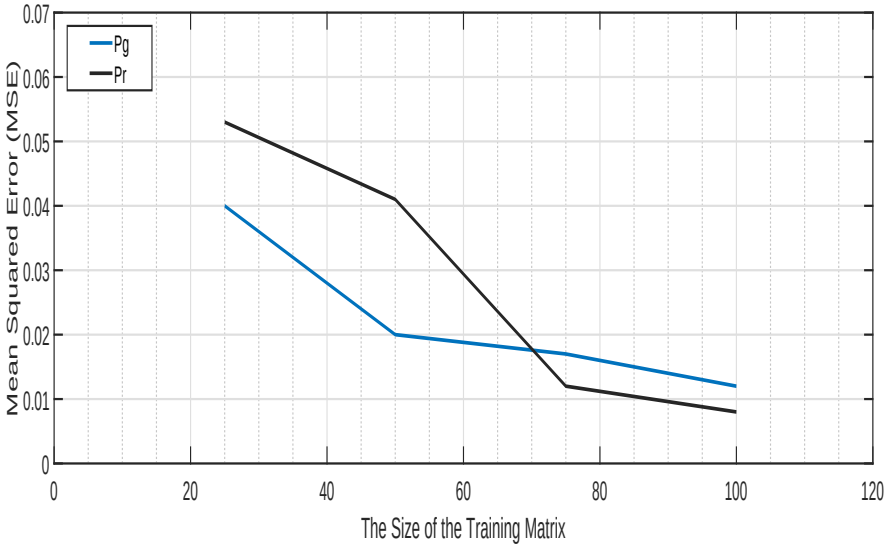

Figure 5.11 The Performance of ANNp_g and ANNp_r with different size of the Training Input .

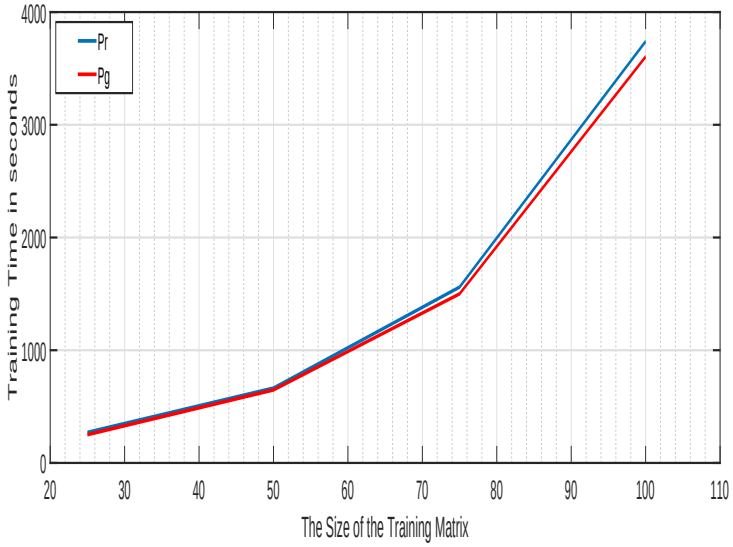

Figure 5.12 The computation time needed to train ANNp_g and ANNp_r as the training matrix increase.

CHAPTER 6. CONCLUSIONS

6.1 Conclusions and Chapters Summaries

In this thesis, we focused on proposing green energy communication systems based on three approaches: First, deactivating Base Stations (BSs) with either no UEs association, or with few associations that can be offloaded to neighbouring BSs. Second, employing the energy harvesting technique to equip SBSs with an auxiliary source of energy to minimize the demand from traditional energy sources. Third, utilizing the smart grid to transfer energy from one SBS with surplus energy to another SBS that requires more energy.

1. Chapter 3 proposed an FBS sleeping strategy for energy efficient networks, where lightly used FBSs are turned off and their associated users and available bandwidth resources are distributed to neighboring FBSs. Additionally, we included the effect of the adaptation of the sleeping strategy on the lifetime of the FBSs, due to the frequent power switching of every FBS. Based on this proposed idea, we formulated an optimization problem that computes energy efficient sleeping strategy while considering the power change on the FBS's lifetime. Moreover, we used two clustering algorithms (K-means and GA) to help reduce the problem's complexity and provide a solution within a reasonable time. In addition we compared the performance of both algorithms for different numbers of FBSs. By using simulation based on the GAMS optimization solver, we showed that the K-means clustering algorithm provided a near optimal solution with a much smaller computational time and memory requirements.

2. Chapter 4 investigated energy harvesting in cooperative SBSs heterogenous networks with a dynamic sleeping strategy, where the deactivated SBSs cooperate with the rest of the network by harvesting then injecting the energy into the network to be transferred to other SBSs. Each of the SBSs is equipped with a harvesting device and a finite battery to store the HE. An optimization problem was formulated which aims at minimizing the transmission power driven from the grid under user QoS constraints. Since the formulated problem is MINLP which is NP-hard, a decomposition of the problem into two subproblems was proposed: a users association problem, and convex optimization problem. They were solved iteratively using GBD. Moreover, a computational efficient algorithm was introduced based on network centrality to solve and optimize the user association and energy harvesting of the system model. Finally, performance evaluation was carried out to examine the performance of both the optimal GBD algorithm and the heuristic BSC algorithm on the energy consumption and computational time. As shown in the results, the BSC algorithm showed superiority in terms of computational time with near optimal results compared to the GBD algorithm which required longer time to reach the optimal solution. Additionally, the results showed the benefit of densifying the network with more SBSs, as the increase of the SBSs number will lead to more cooperation that adds more HE to the network.

3. Chapter 4 reformulated the optimization problem to minimize the energy drawn from the network grid, with updating the UEs association with SBSs in every time slot instead of keeping the association stationary. A new UEs' prediction method is introduced that is based on a combined approach of Non-linear Autoregressive with External input(NARX) and probabilistic Latent semantic Analysis (pLSA) to provide accurate prediction for longer times. Moreover, a new algorithm NARSA is presented to show the general steps that are employed to predict the UEs' next locations. Due to the complexity of the developed optimization problem, a decomposition of the problem is introduced where the UE association and sleeping strategy is solved

using the Base Station Centrality algorithm and the continuous variables are solved using a deep learning approach. The deep learning approach is based on ANN where it was introduced to approximate the input-output relation of the optimization problem. The ANN is trained and configured for generating synthetic data from the decomposed optimization problem where the input is the harvested energy and the output is the transmitted power and the transferred energy. The results show that the employment of UEs prediction and ANN in solving the optimization problem gave efficient computational performance with a near optimal solution.

6.2 Future Research Direction

The future research will be directed into exploring new technologies that are expected to appear in 5G and beyond networks, specifically we will concentrate on two approaches: The effects of mmWaves in small cells environments, and exploring other Machine learning approaches to mitigate the complexity of the wireless communication networks.

6.2.1 mmWave in Small Cell HetNets

The millimeter wave is a spectrum ranging from 30GHz to 300GHz that was under-utilized and idle in the past. The reason behind the under-utilization is its high attenuation rate compared to the sub-6GHz spectrum, which makes it unsuitable for cellular communication networks [92]. Other reason for not utilizing mmWave in the past is the unfriendly channel conditions like high path loss, susceptibility to signal blocking, and rain absorption. However, the unlicensed band around 60GHz is suitable for very short range communication [93]. This will make mmWave suitable for the emerging technology of small cells. One of the main challenges in Small cells deployment is the interference mitigation in dense areas, but with the highly attenuated mmWave the interference can be resolved effectively in small cells. There is a sizeable amount of research that has been done regarding mmWave in HetNets.

In [94], the authors introduced a new dual connectivity with the users associated with two BSs, Macro BS and SBS. The proposed is formulated to maximize the throughput by associating users with the best BSs and improving the access fairness for all users at the same time. Similarly, the authors of [95], propose a Rate-based Cell Range Expansion (CRE) into two tier HetNets where the macro BS is operating in sub-6GHz and SBS is operating in the mmWave. They propose a user association scheme that updates the macro BS region to offload users to SBSs to maximize their QoS. The authors of [96], is proposing a prediction scheme to avoid interference and blockage between the moving user and the stationary BS. In their approach, the mmWave-enabled HetNet cells record the fingerprints of the incidents and the cells' transmission parameters. When a user is moving, and the serving cell's transmission parameters are approaching a fingerprint, the HetNet takes necessary actions to mitigate interference or avoid disconnection before it happens.

In our future work we would like to investigate the effect of integrating the mmWave in our work. This investigation will work into two approaches:

1. Interference Mitigation: The short propagation distance in mmWave imposes a natural limit on interference between SBSs that can help reuse resources in high efficiency. However, high signal attenuation can cause the QoS to decrease rapidly after short distance. This can be solved by deploying SBSs in a densed manner, which in this case affects the SINR in the network. Thus, in our future research we will consider updating the channel gain to include an accurate SINR in our proposed wireless network and come up with new approaches to mitigate the interference between SBSs.

2. Backhauling: In SBSs that employed mmWave, wireless backhauling in most cases cannot support the QoS promised by mmWave. Integrated Access and Backhaul (IAB) is a new approach that is provided by 5G new radio and is presented to overcome the wireless backhauling rate shortage. IAB is presented to minimize the capital and operational expenditure by connecting a fraction of the network SBSs to traditional fiber wired backhaul and the rest relay the backhaul traffic through multiple

hops, which can use mmWave frequencies [97]. In our future work we would like to include wireless backhauling in designing energy efficient wireless networks, where we implement sleeping strategies by considering the energy minimization and backhauling connection. Furthermore, we would like to investigate the possibility of deactivating SBSs with few associated UEs and reassociate them to neighboring SBSs, then employ the deactivated SBS in energy harvesting, backhauling or both.

6.2.2 Applying different ML approaches for Solving Optimization Problems

Applying ANNs in solving optimization problems has great advantages, especially the ANN ability in approximating highly complex problems. However, ANN suffers from high computational requirements which can easily diminish the advantage over solving the optimal problem directly for large wireless networks. Moreover, the coupling of the binary continuous variables in MINLP makes solving the problem using a single ANN intractable. In this thesis, we decoupled the MINLP into two subproblems, the continuous variables are solved using a configured ANN and binary variables are solved heuristically. Thus, in the future work we would investigate the possibility on using other ML approaches that are able to perform the combined classification and regression problems.

One candidate method is Cluster-Wise Regression (CR), which is a method that perform clustering and regression simultaneously [98]. The objective of CR is to find a partition of UEs association that minimize the sum of squared error across all clusters [99]. This is done by using an algorithm that transfers objects across clusters until no further improvement in the sum of squares criterion is achievable [100].

Applying CR in problems such as MINLPs can give the benefits of solving both kinds of variables, i.e., binary and continuous variables simultaneously. In future work we would like to apply CR in our own problem, where each UE is grouped into one of the clusters which represents the SBSs. The clustering will predict the UE association with SBSs, while the regression will predict the power transmission and energy transfer.

BIBLIOGRAPHY

[1] S. Sarraf, "5g emerging technology and affected industries: Quick survey," *American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)*, vol. 55, no. 1, pp. 75–82, 2019.

[2] C. V. N. I. Cisco, "Global mobile data traffic forecast update, 2008–2013," *white paper*, 2009.

[3] C. V. N. I. Cisco, "Global mobile data traffic forecast update, 2017–2022," *white paper*, 2018.

[4] Z. Gao, L. Dai, D. Mi, Z. Wang, M. A. Imran, and M. Z. Shakir, "Mmwave massive-mimo-based wireless backhaul for the 5g ultra-dense network," *IEEE Wireless Communications*, vol. 22, no. 5, pp. 13–21, 2015.

[5] E. G. Larsson and L. Van der Perre, "Massive mimo for 5g," 2017.

[6] T. E. Bogale and L. B. Le, "Massive mimo and mmwave for 5g wireless hetnet: Potential benefits and challenges," *IEEE Vehicular Technology Magazine*, vol. 11, pp. 64–75, March 2016.

[7] Z. Qingling and J. Li, "Rain attenuation in millimeter wave ranges," in *2006 7th International Symposium on Antennas, Propagation EM Theory*, pp. 1–4, Oct 2006.

[8] Y. Niu, Y. Li, D. Jin, L. Su, and A. Vasilakos, "A survey of millimeter wave (mmwave) communications for 5g: Opportunities and challenges," *Wireless Networks*, vol. 21, 02 2015.

[9] J. Mitola, "Cognitive radio. an integrated agent architecture for software defined radio.," 2002.

[10] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, pp. 201–220, Feb 2005.

[11] C. V. N. I. Cisco, "Global mobile data traffic forecast update, 2013–2018," *white paper*, 2014.

[12] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5g be?," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.

[13] D. Muirhead, M. A. Imran, and K. Arshad, "A survey of the challenges, opportunities and use of multiple antennas in current and future 5g small cell base stations," *IEEE Access*, vol. 4, pp. 2952–2964, 2016.

[14] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Communications Magazine*, vol. 46, no. 9, pp. 59–67, 2008.

[15] D. Lopez-Perez, I. Guvenc, G. De la Roche, M. Kountouris, T. Q. Quek, and J. Zhang, "Enhanced inter-cell interference coordination challenges in heterogeneous networks," *arXiv preprint arXiv:1112.1597*, 2011.

[16] J. Malmodin, A. Moberg, D. Lunden, G. Finnveden, and N. Lovehagen, "Greenhouse gas emissions and operational electricity use in the ict and entertainment and media sectors," *Journal of Industrial Ecology*, vol. 14, no. 5, pp. 770–790.

[17] L. Belkhir and A. Elmeligi, "Assessing ict global emissions footprint: Trends to 2040 and recommendations," *Journal of Cleaner Production*, vol. 177, pp. 448–463, 2018.

[18] X. Ge, J. Yang, H. Gharavi, and Y. Sun, "Energy efficiency challenges of 5g small cell networks," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 184–191, 2017.

[19] M. Feng, S. Mao, and T. Jiang, "Base station on-off switching in 5g wireless networks: Approaches and challenges," *IEEE Wireless Communications*, vol. 24, no. 4, pp. 46–54, 2017.

[20] C. Liu, B. Natarajan, and H. Xia, "Small cell base station sleep strategies for energy efficiency.," *IEEE Trans. Vehicular Technology*, vol. 65, no. 3, pp. 1652–1661, 2016.

[21] L. Chiaraviglio, F. Cuomo, M. Listanti, E. Manzia, and M. Santucci, "Sleep to stay healthy: Managing the lifetime of energy-efficient cellular networks," in *2015 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7, 2015.

[22] L. Chiaraviglio, P. Wiatr, P. Monti, J. Chen, J. Lorincz, F. Idzikowski, M. Listanti, and L. Wosinska, "Is green networking beneficial in terms of device lifetime?," *IEEE Communications Magazine*, vol. 53, no. 5, pp. 232–240, 2015.

[23] M. Etoh, T. Ohya, and Y. Nakayama, "Energy consumption issues on mobile network systems," in *2008 International Symposium on Applications and the Internet*, pp. 365–368, 2008.

[24] G. Piro, M. Miozzo, G. Forte, N. Baldo, L. A. Grieco, G. Boggia, and P. Dini, "Hetnets powered by renewable energy sources: Sustainable next-generation cellular networks," *IEEE Internet Computing*, vol. 17, no. 1, pp. 32–39, 2012.

[25] B. Lindemark and G. Oberg, "Solar power for radio base station (rbs) sites applications including system dimensioning, cell planning and operation," in *2001 Twenty-Third International Telecommunications Energy Conference INTELEC 2001*, pp. 587–590, 2001.

[26] G. Piro, M. Miozzo, G. Forte, N. Baldo, L. A. Grieco, G. Boggia, and P. Dini, "Hetnets powered by renewable energy sources: Sustainable next-generation cellular networks," *IEEE Internet Computing*, vol. 17, no. 1, pp. 32–39, 2013.

[27] H. A. H. Hassan, L. Nuaymi, and A. Pelov, "Renewable energy in cellular networks: A survey," in *2013 IEEE Online Conference on Green Communications (OnlineGreenComm)*, 2013.

[28] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart grid — the new and improved power grid: A survey," *IEEE Communications Surveys Tutorials*, vol. 14, no. 4, pp. 944–980, 2012.

[29] A. Azari, M. Ozger, and C. Cavdar, "Risk-aware resource allocation for urllc: Challenges and strategies with machine learning," *IEEE Communications Magazine*, vol. 57, pp. 42–48, March 2019.

[30] C. Jiang, H. Zhang, Y. Ren, Z. Han, K.-C. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Communications*, vol. 24, no. 2, pp. 98–105, 2016.

[31] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, vol. 327, no. 5968, pp. 1018–1021, 2010.

[32] M. Ozturk, M. Gogate, O. Onireti, A. Adeel, A. Hussain, and M. A. Imran, "A novel deep learning driven, low-cost mobility prediction approach for 5g cellular networks: The case of the control/data separation architecture (cdsa)," *Neurocomputing*, vol. 358, pp. 479 – 489, 2019.

[33] I. Basheer and M. Hajmeer, "Artificial neural networks: fundamentals, computing, design, and application," *Journal of Microbiological Methods*, vol. 43, no. 1, pp. 3 – 31, 2000. Neural Computting in Micrbiology.

[34] A. K. Jain, Jianchang Mao, and K. M. Mohiuddin, "Artificial neural networks: a tutorial," *Computer*, vol. 29, pp. 31–44, March 1996.

[35] B. Karlik and A. V. Olgac, "Performance analysis of various activation functions in generalized mlp architectures of neural networks," *International Journal of Artificial Intelligence and Expert Systems*, vol. 1, no. 4, pp. 111–122, 2011.

[36] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 249–256, 2010.

[37] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–44, 05 2015.

[38] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics* (G. Gordon, D. Dunson, and M. Dudík, eds.), vol. 15 of *Proceedings of Machine Learning Research*, (Fort Lauderdale, FL, USA), pp. 315–323, PMLR, 11–13 Apr 2011.

[39] W. Li, H. Zhang, W. Zheng, Y. Xie, and X. Wen, "Dynamic cooperative power management algorithm for virtual cell-based femto networks," in *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC)*, pp. 425–430, 2012.

[40] P. Ren and M. Tao, "A decentralized sleep mechanism in heterogeneous cellular networks with qos constraints," *IEEE Wireless Communications Letters*, vol. 3, no. 5, pp. 509–512, 2014.

[41] J. Kim, W. S. Jeon, and D. G. Jeong, "Base-station sleep management in open-access femtocell networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 5, pp. 3786–3791, 2016.

[42] K. S. Manosha, N. Rajatheva, and M. Latva-aho, "Energy efficient power and time allocation in a macrocell/femtocell network," in *Vehicular Technology Conference (VTC Spring), 2013 IEEE 77th*, pp. 1–5, IEEE, 2013.

[43] B. Han, W. Wang, and M. Peng, "A power allocation scheme for achieving high energy efficiency in two-tier femtocell networks," in *Communication Technology (ICCT), 2011 IEEE 13th International Conference on*, pp. 352–356, IEEE, 2011.

[44] H. A. H. Hassan, A. Pelov, and L. Nuaymi, "Integrating cellular networks, smart grid, and renewable energy: Analysis, architecture, and challenges," *IEEE access*, vol. 3, pp. 2755–2770, 2015.

[45] J. Leithon, T. J. Lim, and S. Sun, "Energy exchange among base stations in a cellular network through the smart grid," in *Communications (ICC), 2014 IEEE International Conference on*, pp. 4036–4041, IEEE, 2014.

[46] C.-Y. Chang, K.-L. Ho, W. Liao, and D.-s. Shiu, "Capacity maximization of energy-harvesting small cells with dynamic sleep mode operation in heterogeneous networks," in *Communications (ICC), 2014 IEEE International Conference on*, pp. 2690–2694, IEEE, 2014.

[47] P. He, L. Zhao, and B. Venkatesh, "Power allocation for cognitive energy harvesting and smart power grid coexisting system," in *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th*, pp. 1–5, IEEE, 2016.

[48] Y.-H. Chiang and W. Liao, "Green multicell cooperation in heterogeneous networks with hybrid energy sources," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 7911–7925, 2016.

[49] Y. Song, H. Wei, M. Zhao, W. Zhou, P. Dong, and L. Zhao, "Optimal base station sleeping control in energy harvesting heterogeneous cellular networks," in *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th*, pp. 1–5, IEEE, 2016.

[50] D. Zhang, Z. Chen, L. X. Cai, H. Zhou, S. Duan, J. Ren, X. Shen, and Y. Zhang, "Resource allocation for green cloud radio access networks with hybrid energy supplies," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 2, pp. 1684–1697, 2018.

[51] N. A. Amirrudin, S. H. S. Ariffin, N. N. N. A. Malik, and N. E. Ghazali, "User's mobility history-based mobility prediction in lte femtocells network," in *2013 IEEE International RF and Microwave Conference (RFM)*, pp. 105–110, Dec 2013.

[52] N. Abani, T. Braun, and M. Gerla, "Proactive caching with mobility prediction under uncertainty in information-centric networks," in *Proceedings of the 4th ACM Conference on Information-Centric Networking*, pp. 88–97, ACM, 2017.

[53] K. Yap and Y. Chong, "Optimized access point selection with mobility prediction using hidden markov model for wireless network," in *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 38–42, July 2017.

[54] Y. Qiao, Z. Si, Y. Zhang, F. B. Abdesslem, X. Zhang, and J. Yang, "A hybrid markov-based model for human mobility prediction," *Neurocomputing*, vol. 278, pp. 99 – 109, 2018. Recent Advances in Machine Learning for Non-Gaussian Data Processing.

[55] D. S. Wickramasuriya, C. A. Perumalla, K. Davaslioglu, and R. D. Gitlin, "Base station prediction and proactive mobility management in virtual cells using recurrent neural networks," in *2017 IEEE 18th Wireless and Microwave Technology Conference (WAMICON)*, pp. 1–6, April 2017.

[56] Y. Tang, N. Cheng, W. Wu, M. Wang, Y. Dai, and X. Shen, "Delay-minimization routing for heterogeneous vanets with machine learning based mobility prediction," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 3967–3979, April 2019.

[57] A. Hadachi, O. Batrashev, A. Lind, G. Singer, and E. Vainikko, "Cell phone subscribers mobility prediction using enhanced markov chain algorithm," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pp. 1049–1054, June 2014.

[58] A. Mohamed, O. Onireti, S. A. Hoseinitabatabaei, M. Imran, A. Imran, and R. Tafazolli, "Mobility prediction for handover management in cellular networks with control/data separation," in *Communications (ICC), 2015 IEEE International Conference on*, pp. 3939–3944, IEEE, 2015.

[59] D. Stynes, K. N. Brown, and C. J. Sreenan, "A probabilistic approach to user mobility prediction for wireless services," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2016 International*, pp. 120–125, IEEE, 2016.

[60] H. Jeung, H. T. Shen, and X. Zhou, "Mining trajectory patterns using hidden markov models," in *International Conference on Data Warehousing and Knowledge Discovery*, pp. 470–480, Springer, 2007.

[61] M. Lichman and P. Smyth, "Modeling human location data with mixtures of kernel densities," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 35–44, ACM, 2014.

[62] Y. Luo, P. N. Tran, C. An, J. Eymann, L. Kreft, and A. Timm-Giel, "A novel handover prediction scheme in content centric networking using nonlinear autoregressive exogenous model," in *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*, pp. 1–5, June 2013.

[63] S. Fahmeen, Masroorah, S. Raziuddin, and Q. Sultana, "Prediction of gps user position using lm and cgf algorithms," in *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pp. 588–591, July 2016.

[64] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Communications Surveys & Tutorials*, 2019.

[65] L. Pierucci and D. Micheli, "A neural network for quality of experience estimation in mobile communications," *IEEE MultiMedia*, vol. 23, pp. 42–49, Oct 2016.

[66] L. Nie, D. Jiang, S. Yu, and H. Song, "Network traffic prediction based on deep belief network in wireless mesh backbone networks," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–5, March 2017.

[67] C. Zhang and P. Patras, "Long-term mobile traffic forecasting using deep spatio-temporal neural networks," in *MobiHoc*, 2017.

[68] C. Zhang, X. Ouyang, and P. Patras, "Zipnet-gan: Inferring fine-grained mobile traffic patterns via a generative adversarial neural network," in *Proceedings of the 13th International Conference on emerging Networking EXperiments and Technologies*, pp. 363–375, ACM, 2017.

[69] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-to-end encrypted traffic classification with one-dimensional convolution neural networks," in *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pp. 43–48, July 2017.

[70] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, pp. 21–26, ICST (Institute for Computer Sciences, Social-Informatics and ..., 2016.

[71] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, pp. 41–50, Feb 2018.

[72] A. Zappone, M. Di Renzo, M. Debbah, T. T. Lam, and X. Qian, "Model-aided wireless artificial intelligence: Embedding expert knowledge in deep neural networks towards wireless systems optimization," *arXiv preprint arXiv:1808.01672*, 2018.

[73] K. Fukunaga, *Introduction to statistical pattern recognition*. Elsevier, 2013.

[74] R. H. Sheikh, M. M. Raghuwanshi, and A. N. Jaiswal, "Genetic algorithm based clustering: a survey," in *First International Conference on Emerging Trends in Engineering and Technology*, pp. 314–319, IEEE, 2008.

[75] Z. Feng, "Data clustering using genetic algorithm," *Evolutionary Computation: Project Report*, 2012.

[76] P. Belotti, "Couenne: a user's manual," tech. rep., Technical report, Lehigh University, 2009.

[77] A. Alsharoa, H. Ghazzai, A. E. Kamal, and A. Kadri, "Optimization of a power splitting protocol for two-way multiple energy harvesting relay system," *IEEE Transactions on Green Communications and Networking*, vol. 1, no. 4, pp. 444–457, 2017.

[78] A. M. Geoffrion, "Generalized benders decomposition," *Journal of optimization theory and applications*, vol. 10, no. 4, pp. 237–260, 1972.

[79] J. F. Benders, "Partitioning procedures for solving mixed-variables programming problems," *Numerische mathematik*, vol. 4, no. 1, pp. 238–252, 1962.

[80] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[81] S. P. Borgatti, "Centrality and network flow," *Social networks*, vol. 27, no. 1, pp. 55–71, 2005.

[82] A. Masadeh, A. E. Kamal, and Z. Wang, "Cognitive radio networking with cooperative and energy harvesting," in *proceedings of the IEEE VTC Fall*, 2017.

[83] L. Ruiz, M. Cuéllar, M. Calvo-Flores, and M. Jiménez, "An application of non-linear autoregressive neural networks to predict energy consumption in public buildings," *Energies*, vol. 9, no. 9, p. 684, 2016.

[84] Tsungnan Lin, B. G. Horne, P. Tino, and C. L. Giles, "Learning long-term dependencies in narx recurrent neural networks," *IEEE Transactions on Neural Networks*, vol. 7, pp. 1329–1338, Nov 1996.

[85] "Design time series narx feedback neural networks."

[86] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, (New York, NY, USA), pp. 50–57, ACM, 1999.

[87] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[88] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, pp. 47–60, Nov 1996.

[89] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359 – 366, 1989.

[90] J. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T.-M.-T. Do, O. Dousse, J. Eberle, and M. Miettinen, "The mobile data challenge: Big data for mobile computing research. nokia research center," 01 2012.

[91] N. Kiukkonen, B. J., O. Dousse, D. Gatica-Perez, and J. K. Laurila, "Towards rich mobile phone datasets: Lausanne data collection campaign," in *Proc. ACM Int. Conf. on Pervasive Services (ICPS,',',') , Berlin.*, 7 2010.

[92] A. Gupta and R. K. Jha, "A survey of 5g network: Architecture and emerging technologies," *IEEE Access*, vol. 3, pp. 1206–1232, 2015.

[93] T. Baykas, C. Sum, Z. Lan, J. Wang, M. A. Rahman, H. Harada, and S. Kato, "Ieee 802.15.3c: the first ieee wireless standard for data rates over 1 gb/s," *IEEE Communications Magazine*, vol. 49, pp. 114–121, July 2011.

[94] Y. Shi, H. Qu, J. Zhao, and G. Ren, "Downlink dual connectivity approach in mmwave-aided hetnets with minimum rate requirements," *IEEE Communications Letters*, vol. 22, pp. 1470–1473, July 2018.

[95] S. Fang, X. Zhu, X. Xu, M. Sun, and T. Svensson, "Rate-based cell range expansion for mmwave massive mimo enabled two-tier hetnets," in *2018 24th Asia-Pacific Conference on Communications (APCC)*, pp. 275–279, IEEE, 2018.

[96] C. Liu and L. Xiao, "Interference and blockage prediction in mmwave-enabled hetnets," in *2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, pp. 201–208, Sep. 2018.

[97] M. Polese, M. Giordani, T. Zugno, A. Roy, S. Goyal, D. Castor, and M. Zorzi, "Integrated access and backhaul in 5g mmwave networks: Potentials and challenges," *arXiv preprint arXiv:1906.01099*, 2019.

[98] Y. W. Park, Y. Jiang, D. Klabjan, and L. Williams, "Algorithms for generalized clusterwise linear regression," *INFORMS Journal on Computing*, vol. 29, no. 2, pp. 301–317, 2017.

[99] H. Späth, "Algorithm 39 clusterwise linear regression," *Computing*, vol. 22, no. 4, pp. 367–373, 1979.

[100] M. J. Brusco, J. D. Cradit, D. Steinley, and G. L. Fox, "Cautionary remarks on the use of clusterwise regression," *Multivariate Behavioral Research*, vol. 43, no. 1, pp. 29–49, 2008.