QoE Based Predictive Handover Mechanism in Software-Defined Enterprise Wi-Fi Networks

by

Sadegh AGHABOZORGI NAFCHI

THESIS PRESENTED TO ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT OF A MASTER'S DEGREE
WITH THESIS IN AUTOMATED MANUFACTURING ENGINEERING
M.A.Sc.

MONTREAL, NOVEMBER 25, 2019

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Mohamed Cheriet, thesis supervisor
Department of Systems Engineering , École de technologie supérieure ÉTS

Mr. Charles Despins, co-supervisor
Department of Electrical Engineering, École de technologie supérieure ÉTS

Mr. Abdelouahed Gherbi, president of the board of examiners
Department of Software Engineering and IT, École de technologie supérieure ÉTS

Mr. Chamseddine Talhi, external independent examiner
Department of Software Engineering and IT, École de technologie supérieure ÉTS

THIS THESIS WAS PRESENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC

ON "19 NOVEMBER 2019"

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincerest thanks to my supervisor Prof, Mohamed Cheriet and Prof, Kim Khoa Nguyen and Prof, Charles Despins for their permanent guidance, the continues support and encouragement during my master study.

I'm also very grateful to Prof, Abdelouahed Gherbi and Prof, Mohamed Chamseddine Talhi my board of examiners, for giving me the honor of accepting to judge my thesis.

My thanks also go to all my friends and lab mates at Synchromedia laboratory, who kindly supported me and whom I often disturbed.

Without question, none of this was possible without my wonderful family. I owe my profound gratitude to my parents for their unconditional love and support thought my life.

Mécanisme de transfert prédictif basé sur la QoE dans un réseau Wi-Fi d'entreprise défini par logiciel

Sadegh AGHABOZORGI NAFCHI

RÉSUMÉ

Au cours des dernières décennies, les fournisseurs de services et les entreprises ont tenté de répondre aux besoins des utilisateurs en matière de connexions Wi-Fi dans les bâtiments résidentiels, les campus et les palais publics avec de nouvelles infrastructures Wi-Fi centralisées. Bien qu'ils aient en grande partie résolu le problème de l'accès et de la convergence des réseaux Wi-Fi en les optimisant et en les adaptant à l'aide d'un réseau SDN (Software-Defined Networking), il est encore possible de rendre cette optimisation aussi intelligente que possible. Cela est dû à la croissance rapide de la demande et de l'application des utilisateurs sur les smartphones. De plus, SDN nous permet d'améliorer les performances des systèmes centralisés. Une connexion garantie est une fonctionnalité essentielle pour les utilisateurs de réseaux sans fil, leur permettant de continuer à utiliser leurs applications même s'ils se déplacent d'un côté du réseau à un autre. Le processus de transfert rend cela possible en dirigeant l'utilisateur d'un point d'accès à un autre ou d'un réseau à un autre. La décision de déplacer l'utilisateur de l'interface 1 à l'interface 2 peut affecter la qualité de service des utilisateurs.

Dans un réseau Wi-Fi d'entreprise, les utilisateurs mobiles peuvent être couverts par plusieurs points d'accès. Pour optimiser l'allocation des ressources, un transfert progressif est requis dans lequel le périphérique de l'utilisateur est transféré de manière transparente d'un point à un autre cette décision est prise de manière centralisée par un contrôleur de réseau Wi-Fi. Malheureusement, les mécanismes de transfert progressifs les plus avancés sont souvent conçus pour optimiser les ressources du point de vue du fournisseur de réseau et ne tiennent pas compte des comportements en temps réel des utilisateurs, ce qui peut affecter la qualité de l'expérience (QoE) de l'utilisateur. Dans ce mémoire, une nouvelle méthode basée sur l'apprentissage automatique (ML) a été présentée pour trouver un mécanisme de transfert optimal. Cette méthode permet de prédire si le transfert qui va se produire conservera la qualité d'expérience lorsque les utilisateurs se déplacent à l'intérieur d'un bâtiment. Notre premier objectif est de présenter un cadre pour la prédiction du transfert intercellulaire en introduisant une échelle de score continue basée sur la QoE de l'utilisateur. Nous étudions le comportement du locataire et l'effet de ce comportement sur le mécanisme de passation à l'aide d'un ensemble de données obtenu à partir d'une étude de cas réelle sur un campus universitaire. Ensuite, nous définissons un ensemble de règles basées sur nos résultats de prévision et d'observation à l'intérieur du réseau. Notre cadre de prédiction du transfert intercellulaire est complété par l'alimentation des caractéristiques définies par l'expert dans une régression vectorielle de support (SVR). La méthode proposée s'appliquait à plus d'un an de données collectées à partir de points d'accès du campus mentionné. L'évaluation des résultats prouve l'efficacité, la puissance de généralisation et la robustesse du cadre présenté pour la prévision d'un mécanisme de transfert indépendant du temps. Notre méthode proposée permet une amélioration de 34% du débit utilisateur par rapport aux algorithmes de pointe .

Dans ce travail, notre base de référence est le réseau d'auto-organisation XcellAir, qui est le fournisseur de services du campus. Nous menons et évaluons notre expérience en fonction des résultats de leur système d'optimisation. Le cadre proposé repose sur l'hypothèse que le transfert intercellulaire a lieu lorsqu'un utilisateur se déplace entre deux stations et que le controle intervient après que l'utilisateur a subi une dégradation des performances du service reçu. Ce fait suggère l'idée d'une méthode proactive plutôt que d'utiliser une méthode basée sur des seuils pour développer notre cadre. Nous avons proposé une approche en prévoyant un score que nous avons défini sur la base de la QoE des utilisateurs (du point de vue de l'utilisateur) et de notre modèle prédectif, etégalement à partir des commentaires des utilisateurs. En raison des biais que nous pouvons avoir dans les prédicteurs dépendant du temps et du fait que la migration à l'intérieur d'un réseau peut être très rapide, nous nous sommes concentrés sur l'utilisation de fonctionnalités plus importantes et l'apprentissage du comportement des utilisateurs en fonction de paramètres plutôt que du temps. En testant le cadre introduit sur notre ensemble de données [??]et les résultats confirment l'efficacité de la méthode proposée par rapport au modèle de base.

Mots-clés: Handover-Mechanism, Réseaux Wi-Fi, Apprentissage automatique, ML prédictif, Réseau défini par logiciel

QoE Based Predictive Handover Mechanism in Software-Defined Enterprise Wi-Fi Networks

Sadegh AGHABOZORGI NAFCHI

ABSTRACT

In recent decades, service providers and enterprises have tried to fulfill the need of users to Wi-Fi connections inside residential buildings, campus, and public palaces with new centralized Wi-Fi frameworks. Although they have mostly solved the issue of Wi-Fi networks access and convergence by optimizing and softwarization of them using Software-Defined Networking (SDN), there is still room to make this optimization as intelligent as possible. specially with rapid growth of user's demand and application on smartphones. Moreover, SDN will allow us to improve the performance of centralized systems. A guaranteed connection is a key feature for wireless network users so that they can continue to use their application even if they are moving from one side of a network to another side. Handover process makes this happen by steering user from one access point to another access point or from one network to another network. Deciding when to move the user from one interface to another interface can affect the QoS for users.

In an enterprise Wi-Fi network, mobile users may be covered by multiple access points (APs). To optimize resource allocation, a soft handover is required in which the user's device is seamlessly transferred from one AP to another, and this decision made centrally by a Wi-Fi network controller. Unfortunately, state-of-the-art soft handover mechanisms are often designed to optimize resources from the network provider's point of view and do not take into account user's real-time behaviors, which may affect user's Quality of Experience (QoE). In this thesis, a new machine learning (ML)-based method presented to define an optimal handover mechanism. This method allows predicting whether the handover that is going to happen will maintain QoE when users are moving inside a building. Our first goal is to present a framework for handover prediction by introducing a continues score scaling based on user's QoE. We study the behavior of tenant and effect of this behavior on the handover mechanism using a data-set obtained from a real case study on a university campus. Then we define a set of rules based on our prediction results and observation inside the network. Our framework for handover prediction is completed by feeding the handcrafted features to a Support vector regression (SVR). The proposed method applied to more than one year of collected data from access points of the mentioned campus. The evaluation of results proves the efficiency, generalization power, and robustness of our presented framework for predicting a time-independent handover mechanism. Our proposed method improves 34% of user throughput compared to state-of-the-art algorithms.

In this work, our baseline is the XcellAir self-organization network, which is the service provider of the campus. We run and evaluate our experiment based on the results of their optimization system. The proposed framework is based on the hypothesis that handover happened when a user is moving between two stations, and steering will happen after the user faced performance degradation in the received service. This fact suggests the idea of a proactive method rather than using a threshold-based method for developing our framework. We proposed an approach by predicting a score that we defined based on QoE of users (From user perspective of view) and our handcrafted feature also from user feedback. Due to the bias that we can have in time-dependent predictors and the fact that moving inside a network can occur very quickly, we focused on using more important features and learning the behavior of users based on parameters rather than time. We test the introduced framework on our data-set, and the results confirm the efficiency of the proposed method in comparison to the baseline model.

Keywords: Handover-Mechanism, Wi-Fi networks, Machine learning, Predictive ML, Software defined network

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABREVIATIONS

ASC	Agence Spatiale Canadienne
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
FV	Feature Vector
SVR	Support Vector Regression
QoS	Quality of Service
QoE	Quality of Experience
VHO	Vertcal Handover
AP	Access Point
APs	Access Points
CG	Cooperative Game
MADM	Multi Atribute Decision Making
AHP	Analytic Hierarchy Process
SDN	Software Defiend Network
ANN	Artificial Neural Network
WAN	Wireless Sensor Network
ML	Machine Learning
DL	Download Throughput
UL	Upload Throughput

Tx	Transmit Bit Rate
Rx	Receive Bit Rate
App	Application
SON	Self Organization Network
RRM	Radio Resource Management

CHAPTER 1

INTRODUCTION

Even though the research in the field of handover in enterprise Wi-Fi network received a vast amount of attention over last decades, it is still appealing but challenging; especially when we are dealing with a large number of users and access points. The user is connected to an AP which provides the highest QoS (usually the AP with the shortest distance). When the user moves inside the building, the connection is switched from an AP to another AP to ensure the QoS. This process is called the *handover* mechanism. In some cases, it is referred to as steering. There are two forms of handover mechanism:(i) soft handover in which the source and target APs are in the same network, and (ii) hard handover in which the source AP and destination AP belong to different networks. Handover is a recommendation to the client device. The client device can ignore the request or change to another AP based on its internal algorithm. Many studies have been conducted in the wireless community to improve the soft-handover mechanism. Those studies focused on reducing the transfer time at physical and software layers and decreasing the effects of handover on the QoS Yang, Wu, Chu & Song (2016), selecting the optimal AP Elhadj, Elias, Chaari & Kamoun (2016), etc. Our goal in this work is to maximize the total throughput (QoS) of users while reducing the number of unnecessary handovers and balancing load across APs. Indoor Wi-Fi networks are established in enterprise buildings and residence area based on the IEEE 802.11 standard.

1.1 Motivation

Wi-Fi networks provide connectivity and convenience to internet users and in general moving users inside buildings or in a public place. Users depend on Wi-Fi every data at home and work, in a way that more often they don't realize this dependency. Research has shown the Wi-Fi industry did not slow down over the years and in 2019 world will face cumulative device shipment surpassing 20 billion units. Thus Wi-Fi continues to impact everything from home networking and retail applications to critical business operation around the world. Many companies such as Netflix, Amazon, Facebook, Instagram, and most of the major airlines are all dependant on Wi-Fi to perform their daily operations. Therefor it's vital for Enterprise to provide these users with an intelligent Wi-Fi platform which can guarantee the performance and quality of the network for users and receivers of the Wi-Fi service. Figure 1.1 depict an example of an intelligent platform. Up to some point, this service is very dependant to user's devices and access points which serving the users. Moreover, management systems for optimizing resources and radio frequency are more important.

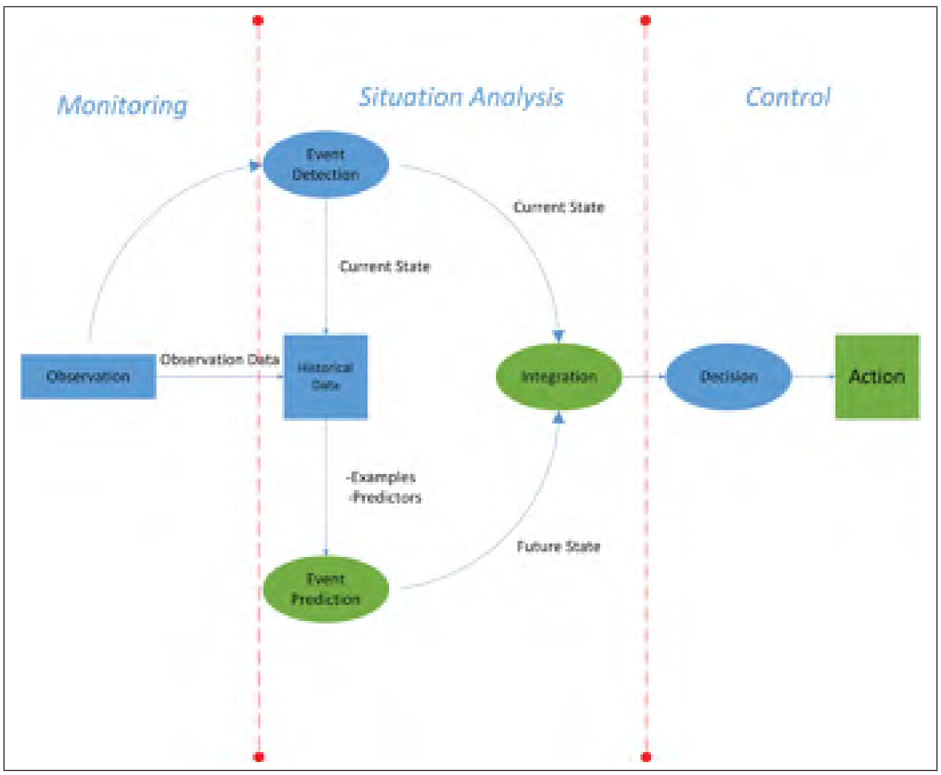

Figure 1.1   General Dataflow of our system.

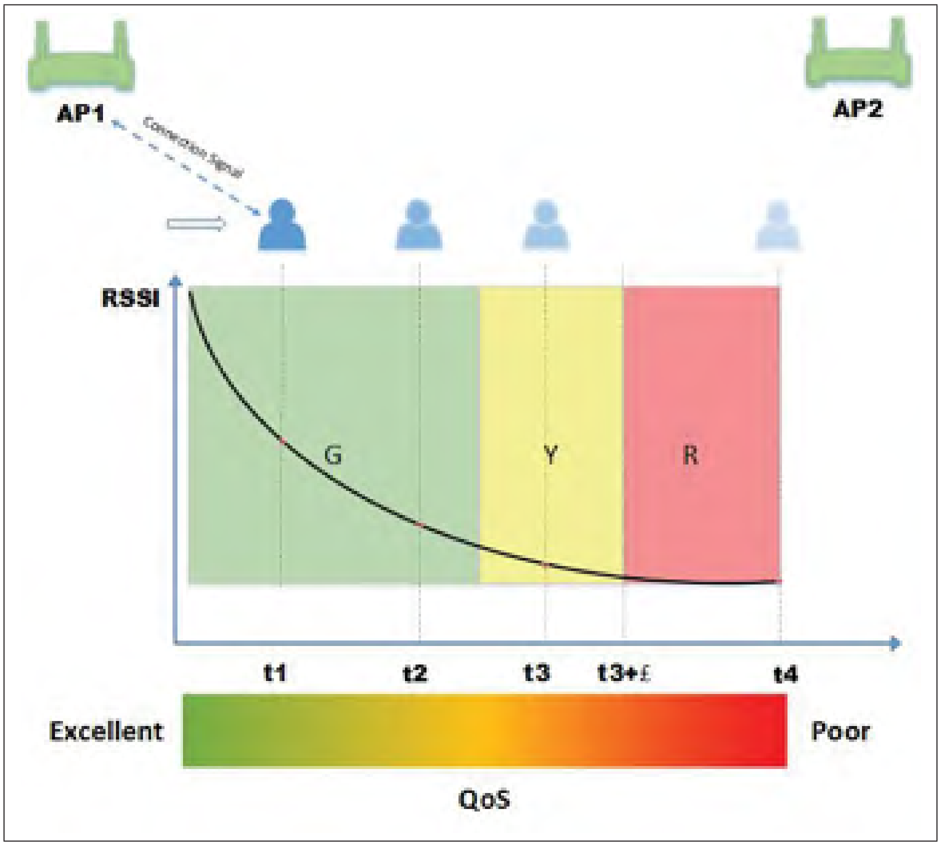

Figure 1.2 The QoS of connection changes when user's distance is increasing from AP. Green zone is the zone that user is receiving it service without any trouble and degradation while service is maintaining in yellow zone and is facing degradation in red zone.

1.2 Problem Statement

A key issue in the handover mechanism is to find the appropriate moment for triggering the process (i.e., switch the AP). For example, consider the scenario in Fig. 1.2 A user is connected to AP1 and moves toward AP2. As the users move along the path, the connection QoS drops because of the distance to AP1 increases. At some point, the connection must be transferred to AP2. The network suggests handover, then the device performs handover by itself. The connection information (e.g., signal strength, throughput) is periodically monitored and logged by the network controller. The network controller frequently checks the connection and triggers the handover process when the status of connection meets the handover conditions, which is defined based on a threshold. In baseline system, the thresholds are constant and manually configured by the network operator. This approach is reactive and does not ensure real-time quality of service. The handover mechanism should be predict prior to it's happening or before user connection encounters QoS degradation. In other word in Figure 1.2, handover should take place in yellow zone. Moreover, as the space inside the building changes (e.g., the decoration, the locations of objects, obstacles, and walls) the threshold value does not change based on the modification inside buildings, this will cause interference in propagation of signals inside buildings; We believe a dynamic threshold can help the system to adapt itself with these modification inside buildings. Another issue is the ping pong effect, which happens when the user connection is frequently transferred between APs in a short-period of time.

To this end, our research is aimed to research the solutions for the following research questions:

- Q1 Is it possible to develop an intelligent and efficient algorithm to learn the behavior of the users within a Wi-Fi enterprise network?
- Q2 As we do not have the precise time of handover event, how we can detect the handover event from user data?
- Q3 To what extent we can improve the overall performance of the Wi-Fi network by predicting the optimal handover parameters?
- Q4 Which methodology will help us to achieve the best prediction accuracy?
- Q5 Is it possible to build a framework from users historical data as well as preserving the privacy of users' data content?
- Q6 What are the drawback of the system and the recommended solutions to overcome the drawback?

Our aim in this work is to propose a new method for the handover mechanism to improve or at least maintain the quality of experience of the users, when they are connected to a centralized Wi-Fi network. Unlike the existing approaches which are based on the fixed threshold values, our approach relies on a machine-learning algorithm to find the optimal moment to trigger the handover. The connection QoS is measured before and after the handover process. A continuous handover score is defined, which scales from 0 to 1 to evaluate the handover event according to the connection QoS (before and after the transfer). In the proposed approach, the handover score is predicted (before the handover occurs) using a machine learning algorithm. The user data (e.g., the signal strength, connection throughput, and the number of previous handover events) used as the features in the prediction algorithm. The predicted score determines whether the handover can preserve the connection QoS.

In summary our contribution in this work is to detect the handover in the user data collected from centralized Wi-Fi network. Then we propose an automatic predictive framework for predicting the handover event prior to it's happening and at the end we will evaluate the robustness of our prediction by validating the prediction algorithm on a real-case study.

1.3 Objective of the thesis

In this thesis, we aim to design an advanced platform for re-optimizing enterprise WiFi network based on historical data mining. This platform should be capable of learning the behavior of users from their collected data and predicting the optimal handover parameters to maintain the user's quality of service. In this thesis, our focus is on the analyzing the user's utilization data to find the drawback of the baseline system and predicting the optimal parameters of handover then optimizing the handover process across the Wi-Fi access points and also between the bands and the channels within an access point. More specifically, we aim to:

- O1: Develop new QoE metric to score the handovers
- O2: Develop an algorithm to predict the moment of steering to guarantee the QoS
- O3: To design new algorithm that trigger Handover decision, based on our prediction

1.4 Thesis Outline

The thesis is organized into five chapters. The introduction chapter gives a big picture of the whole thesis. It contains the problem statement and the background of the handover mechanism in Wi-Fi networks. It also gives an overview of the objectives of this work.In Chapter 2 we reviewed the literature. Chapter 3 is dedicated to the methodology and presents the definition of the different handover in different condition. It also describes how we analyzed the feedback data to find the drawback of the baseline system — moreover our new approach for extracting features and predicting the optimal parameters using different ML approach. Chapter 4 focuses on introducing the baseline model and our new proposed algorithm to make a decision based on the predicted path and explain the experimental results used to validate our proposed platform. Finally, in the last chapter, conclusion and research direction for future works are discussed.

CHAPTER 2

BACKGROUND

2.1 Machine Learning

We know that humans learn from pas experience and machine follow human instructions. Now, what is humans can train machines to learn from past data and do what humans can do and much faster. From enhancing Wi-Fi network to detecting skin cancer or sorting fruits to detecting escalator needing repair, machine learning has granted computer systems entirely new abilities. We can have an answer from our data using machine learning. Machine learning is an algorithm than can find out how to decide for important tasks by learning from past examples. When the programming cost is high, this can be a feasible option. New research from McKinsey shows that machine learning will be the next big wave in technology Manyika, Chui, Institute, Brown, Bughin, Dobbs, Roxburgh & Byers (2011). Many different machine learning approach exists. From supervised learning to unsupervised, semi-supervised learning, classification, and clustering. In this work, we will focus on predictive modeling using support vector regression, which is a variety of support vector machine. In the following section, we describe how we choose machine learning to solve the problem or how learning can help us to solve the problem statement.

2.1.1 Learning = Representation + Evaluation + Optimization

Consider we have a problem that we think we can solve it through machine learning. The first thing that we have to think about is the variety of available algorithms. Which one we have to use? Many algorithms are available and much more publishing every year. The key to do not lost in this variety is to understand the following components (Table 2.1).

2.1.2 Representation

A predictor has to represent in a set of formal language that machine can read and handle it — this formal language so-called hypothesis space of the learner. If the predictor is not in the hypothesis space, it cannot be learned. Then we will look at the question of how to represent the input, which parameter use as features.

2.1.3 Evaluation

There is an evaluation function, or so-called objective function is needed to differ between a good and a bad predictor. To aim this goal, this function has been used internally by the algorithm, and it is different from the external algorithm that we want the predictor to optimize.

2.1.4 Optimization

Optimization is key to find the efficiency of the learner and helps to determine that the predictor in our case is following our evaluation function.

2.1.5 Feature Engineering

Some machine learning projects are successful, and some will fail, the question is, what makes the difference? The most important factor is the feature that been used. If we have many independent features that are correlated with each other, then the learning process is so easy. In contrary If the values are a very complex function of feature, learning is so hard. Usually, the raw data that we have is not in the form that can be learned easily, but we can construct features from our data. This is often the time-consuming part of each machine learning project. Gathering the data, integrate it, cleaning and pre-processing usually seem easy for first-timer, then they will realize how much trail and error is going to feature design and extraction. while learners can be general-purpose, being domain-specific make feature engineering difficult.

Table 2.1 The three components of learning algorithms.

Representation	**Evaluation**	**Optimization**
K-nearest neighbor	Accuracy/Error rate	Greedy search
Support vector machine	Persision and recall	Beam search
Naive Bayes	Squared error	Gradient descent
Logistic regression	Likelihood	Conjugate gradient
Neural networks	Posterior probability	Linear programming
Bayesian networks	K-L divergence	Quadratic programming

2.1.6 Prediction Modeling

Prediction is the core of our purposed method and makes our system pro-active. For prediction model several feature used (table 2.2) to train our model. These important features are average throughput in 1 minutes as well as RSSI and number of handovers. We aim to learn the behavior of the handover mechanism, train our model, and predict the handover score label associated with each handover.

Table 2.2 Feature metrics

Metrics	Notation
Signal Strength Value	$RSSI$
Throughput of user	dl, ul
Number of previous handover	$N_{handover}$

2.2 Software Defined Network (SDN)

A software-defined network attempts to make a network by separating it into two systems (figure 2.1); the primary system is the management plane that provides performance and fault via internet flow IP6 SNMP and conventional alternative protocols, it generally handles configuration management of the SDN criticism devices and perceive the configuration loaded with these details. The controller will request supported desired needs like QoS levels. The controller conjointly performs link management between devices. The second system is the information plane that is chargeable for forwarding traffic to the chosen destination; switches will either be dependent on the controller to create forwarding choices or make a choice on their own.

The control plane configures affiliation methods or flows into the information plane through the employment of an impact protocol. The controller employs the management protocol in a very software package outlined network to perform necessary operate like affiliation setup once attempts to speak with another host over an SDN the primary packets from the shopper involved the new flow are used. Forwarding call will be created domestically by the switch or if the switch must raise the controller what try to, if the switch determines that it must ask the controller, therefore via secure channel mistreatment the control protocol. The controller decides supported policies if the flow ought to be granted; If allowed, details regarding the flow might be entered into the controllers' affiliation table. The controller may then send directions to program the switches within the best path on the information plane; then the flow would be directed through the network. The switches can also tell the controller once a flow is not any longer active, this removes it from the table.

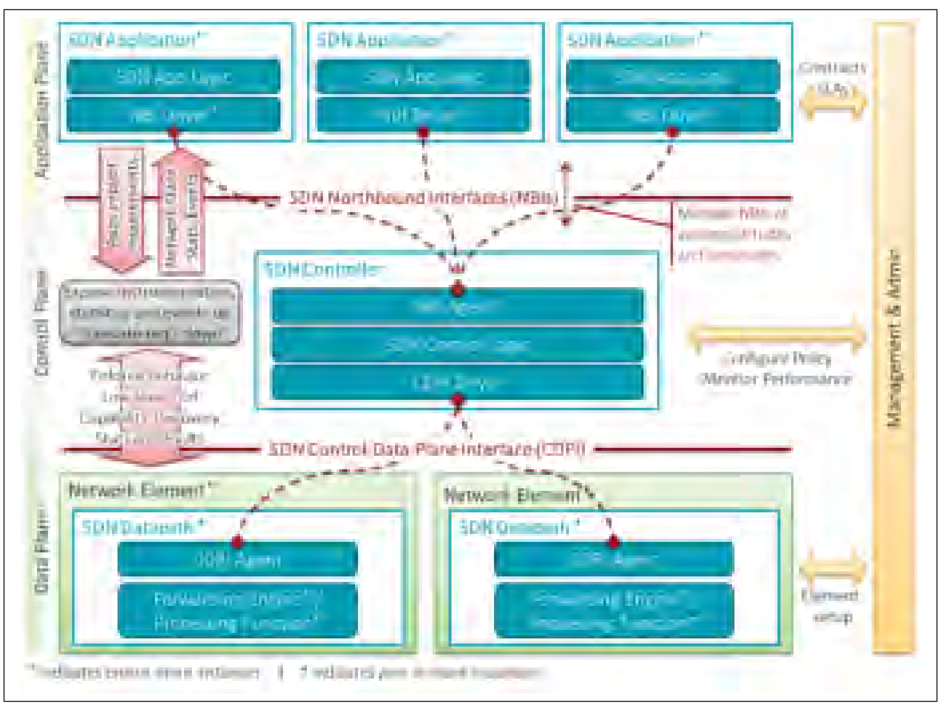

Figure 2.1 Overall scheme of SDN (Software Defined Network)

2.2.1 Controller Benefits

Centralizing a number of all of the connection has many advantages. Thanks to the configuration policies within the controller, some connection requests might be born such GOS attacks and increased discovery traffic. The policies on the controller that are leveraged to create selections on flows will be supported ranges of information science addresses, time of day, and alternative characteristics. SDN conjointly claimed to beat measurability problems; it's unlikely that one controller would be processing all of the connection requests for all of the access points on the network. The problem will be managed during several alternative ways which exist. The primary plan is to interrupt up the network into multiple management and information planes. Policies will then be synchronous across multiple controllers. Every controller still sets up connections end-to-end even once another information plane is concerned. A second thanks to unloading number of the process on every controller is to allow the switch receiving the initial connection requests to create some forwarding selections, permitting the switches to create most or some of the forwarding selections give support for surroundings that are not able to commit 100% to a protocol. Traffic analysis of a software-defined network comes in 2 major formats. Usually, the switches and routers during a software-defined network are SNMP compatible and that they will generally export some kind of NetFlow or information science mounted information even the controller might be found out to export flows from its connection table, guaranteeing that every detail is obtainable for network traffic analysis. SDN has gained tremendous momentum as a result of a minimum of six of the biggest networks (Google, Facebook...) within the world are supporting it. The advantages of SDN may lead to the flexibility to buy cheap switches that have little or no resident computer code and process wants. Centralization of the forwarding database or FIB permits optimum routs to be calculated deterministically for every flow, end-to-end across the topology. SDNs dynamically responds to application needs. SDN optimize the use of the network, while not sacrificing service quality. SDNs will filter packets as they enter the network and therefore these switches can act as straightforward firewalls at the sting of the network. SDN switches will send sure suspicious traffic flows to higher-layer security controls such as IPS systems, application firewalls, and information loss interference devices. SDN switches that support the modification of the packet headers will be ready to operate as an easy cost-efficient load equalization device. SDN controllers will be clustered for fault tolerance and high handiness. Interest can increase once applications utilize the centralized management obtainable in most SDN architectures.

2.3 Privacy Preserving

By growth of Wi-Fi network inside residential buildings, enterprise attempts for enhancement of Quality of service of the users and their Quality of experience. By considering that Quality of experience usually come from user side, preserving the privacy of the collected data is important. How to use the collected user experience, which in this case is the user footprint, without exposing their private data is still challenging. In this work, we will choose the raw data without looking at the running application, which result in a level of security and privacy in our handover detection pipeline.

2.4 System Description

This work has been done based on ETS Wi-Fi residence project. The ETS students are heavy users of several different internet services, and most of them spend their spare time in the residence. The dorm contains over 75 Wi-Fi access points, all of which can be optimized in relation to each other and to the users accessing them. The controller system has been designed and implemented by XcellAir. The central controller includes (Figure 2.2): SON is automated management of the wifi network, providing AP self-provisioning, self-healing, radio environment mapping, and monitoring. RRM is a powerful radio resource optimization tools to dynamically provision and tune radio resources.

There are two essential modules in the controller called Self Optimization Network (SON), and Radio Resource Management (RRM). SON will create a granular radio map, which can give a dynamic view of RF environment (Where the APs are, what channels they are operating on, the received signal strength from APs and location down to a 2m*2m 2-dimensional “Pixel” level).

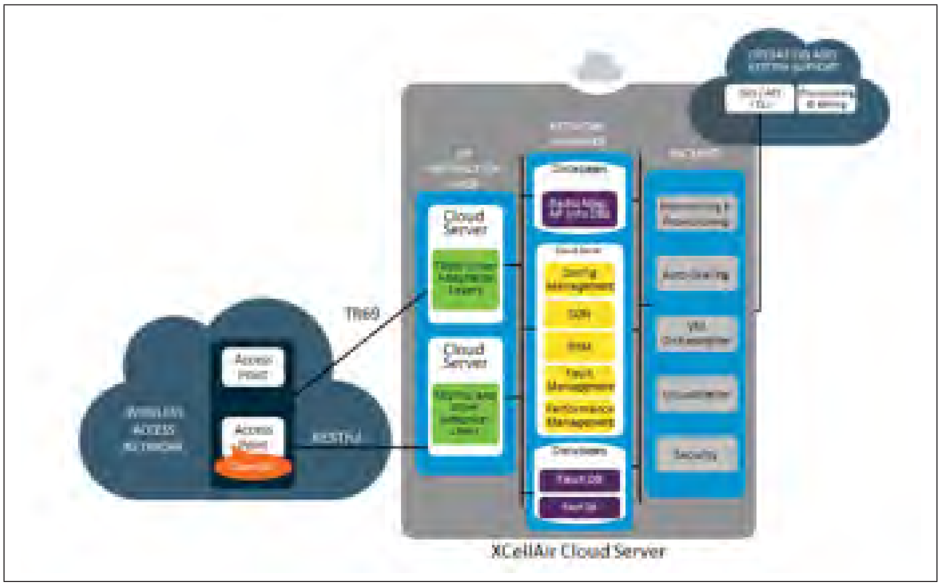

Figure 2.2 Overall big picture Xcellair (Our baseline system).

RRM is an optimization algorithm which is running on the cloud server and interact with the AP to set and change the channel, power levels, and other parameters.

2.4.1 Baseline Setup

For the baseline system, we assumed a Wi-Fi network with a centralized controller on top of the system. The network is comprising N APs where several mobile users using the network. The system is considering time as an interval, and each interval t, the system logs the user's metrics. Like other traditional systems, When a user is moving around, the handover mechanism gives an option to mobile user to change the serving AP, based on the signal strength. The controller on top of the centralized system continuously monitors the $RSSI\Gamma(t)$ of the users and the serving AP at the time A(t). When signal strength (RSSI) of users falls below a predefined static threshold Γ for a given time, the handover mechanism starts.

$$\Gamma(t) > \Gamma_{th} \tag{2.1}$$

2.4.1.1 Self Organization Network (SON)

SON is a self-optimization network top of the controller system which dynamically provisions the managed APs and localizes UN-managed APs and measures the interference within provisioned network. In figure 2.3 you can see the Access points before SON scanning and figure 2.4 after scanning of SON. SON draw a grid within the area that APs are located to obtain the coordinate of them, then try to identify the UN-managed APs. SON then will provide RRM with map of all the UN-managed and managed APs to furthermore optimization of bands and Channels.

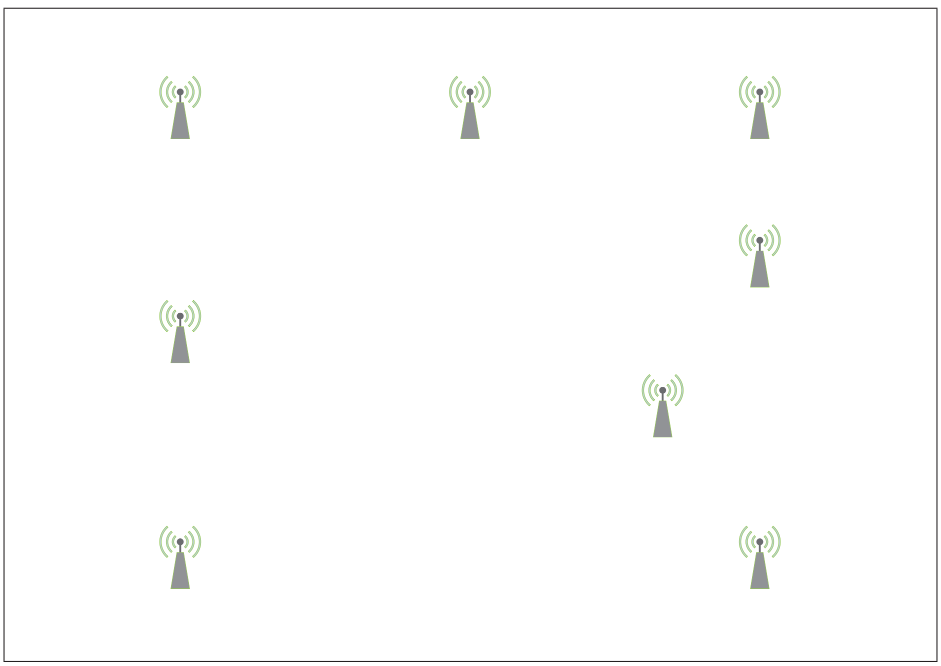

Figure 2.3 Overwiew of APs before SON. Where Green APs denotes Managed APs.

2.4.1.2 Radio Resource Management (RRM)

When Deploying a large-scale Wi-Fi network, it is important to have the ability to discover and auto-provision managed access points. A managed AP is an access point deployed by the operator, and manual configuration of APs in a large network becomes a non-scalable proposition. SON discovers and registers Service Provider Manages APs in its databases. A core feature of

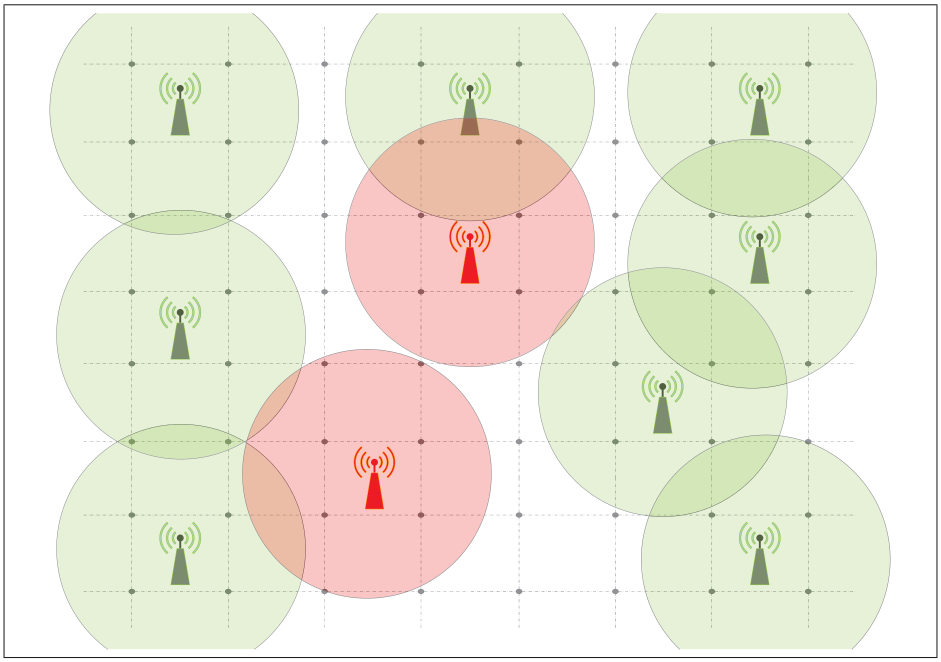

Figure 2.4 SON Pixel level visualization of managed and Un-Managed APs. Where Red APs denotes Un-Managed APs and Green one denotes Managed APs. SON will identify the coverage of each managed AP as well as identification of the Un-Managed AP and their coverage.

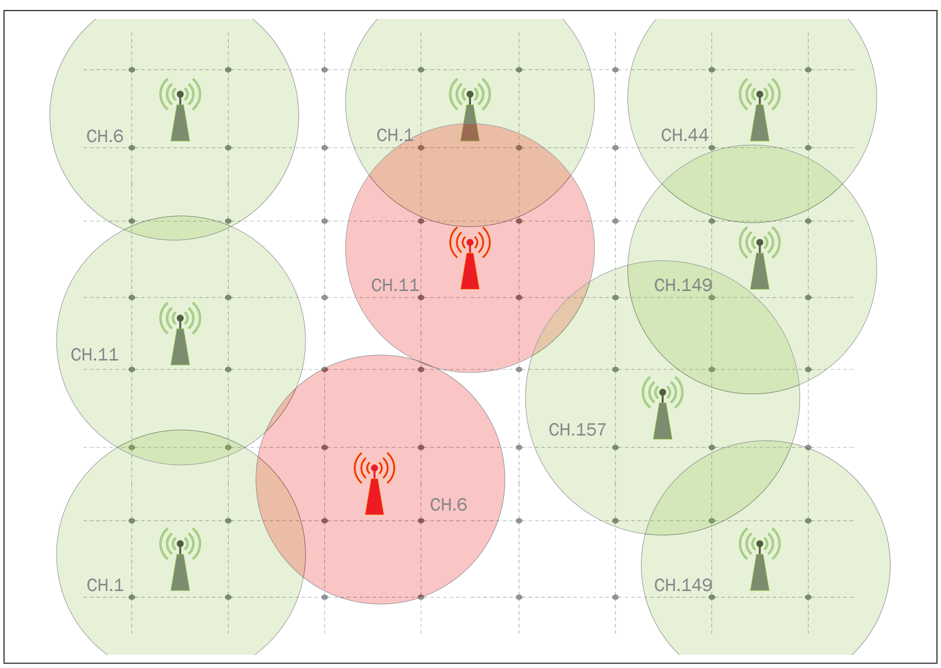

Figure 2.5 RRM radio resource management of channels and Band connectivity

Wi-Fi SON is a real-time, geo-mapped radio environment map that provides a view of Managed and Un-managed APs in the system and depicts channel usage and signal strength level on a angular pixel basis. in Figure 2.4 A pixel is a 2m multiple 2m area within the network.

RRM receives network details from the SON module. Details such as the current operating channel of each managed and UN-managed APs to optimize the radio resource management (Figure 2.5) such as channel allocation, Band steering, and Client steering.

CHAPTER 3

LITERATURE REVIEW

Handover mechanism helps Wi-Fi users to maintain their connectivity inside a centralized network or between two of several different networks. Networks can be either different Wi-Fi network or 4G or 5G network. A user handover inside one network so-called soft handover (Figure 3.1). When the same user handover between two Wi-Fi networks or Wi-Fi and 5G, for example, this action called Hard Handover. When users are connected to one Wi-Fi access point, subject to their distance from an access point, their connection varies from 2.4GHz for long-range and 5GHz for short-range. Therefor when they are moving inside a building within these two range, they could seamlessly hand between these two bands. To provide users with more performance, some access points would force users device to move from 2.4GHz to 5GHz. This action is called steering or in other words, 5G preferences, which often is not optimal for a user who is receiving service.

Figure 3.1 Overall scheme of a soft-handover inside Wi-Fi network. Soft-handover is when a user will move from one AP to another AP Inside one network.

Recently, Wi-Fi handover mechanism has been investigated in many studies Sarma, Chakraborty & Nandi (2016), Liang *et al.* (2017), Chen, Wang, Li & Wang (2018). Existing approaches for improving handover mechanism are based on optimization techniques Zhang, Qiu, Chu, Long & Leung (2017a), Liang *et al.* (2017) and machine learning algorithms Ali *et al.* (2018), López-Raventós *et al.* (2018), Aibinu *et al.* (2017).

Authors in Sarma *et al.* (2016) they tried to address the problem when Wi-Fi has been failed to provide desired quality of service (QoS) requirements for users thus their work will migrate the users from Wi-Fi network to Wi-Max. However, the users prefer to stay connected to Wi-Fi because of its low-cost availability and less power consumption. A key issue in this between is not the mobility or distance; it's more about the unbalanced traffic load distribution among the Access points. Although the traffic load in Wi-Fi access points is highly dynamic, and it varies from one access point to other access points depend on the geographical location and environment which the access points are located in. It this work proposed a bandwidth management control system to the more proper distribution of the total network traffic between access points. They used the Wi-Max network to distribute the traffic among access points. In the end, A handover policy has been designed, which defined when a user has done a handover between Wi-Fi and Wi-Max interfaces.

Work in Liang *et al.* (2017) focuses on VLC-Femto system in a family apartment. They propose an efficient Vertical Handover for hybrid VLC-Femto system. They employed cooperative game (CG) and analytic hierarchy process (AHP) to handle the multi-attribute decision making (MADM). In this work, they replaced the network selection with the decision of "Perform VHO" and "Not Perform VHO" when the access point or service provider (in this case VLC) is inaccessible or overloaded. They've chosen a two-person CG, which is very powerful in the calculation of the average marginal contributions of both cooperators. They considered decisions as cooperators and utilize the cooperative game to compare the criteria values of different decisions, then Analytic hierarchy process will give a score of several criteria based on different traffic type, and MADM do the decision job for them.

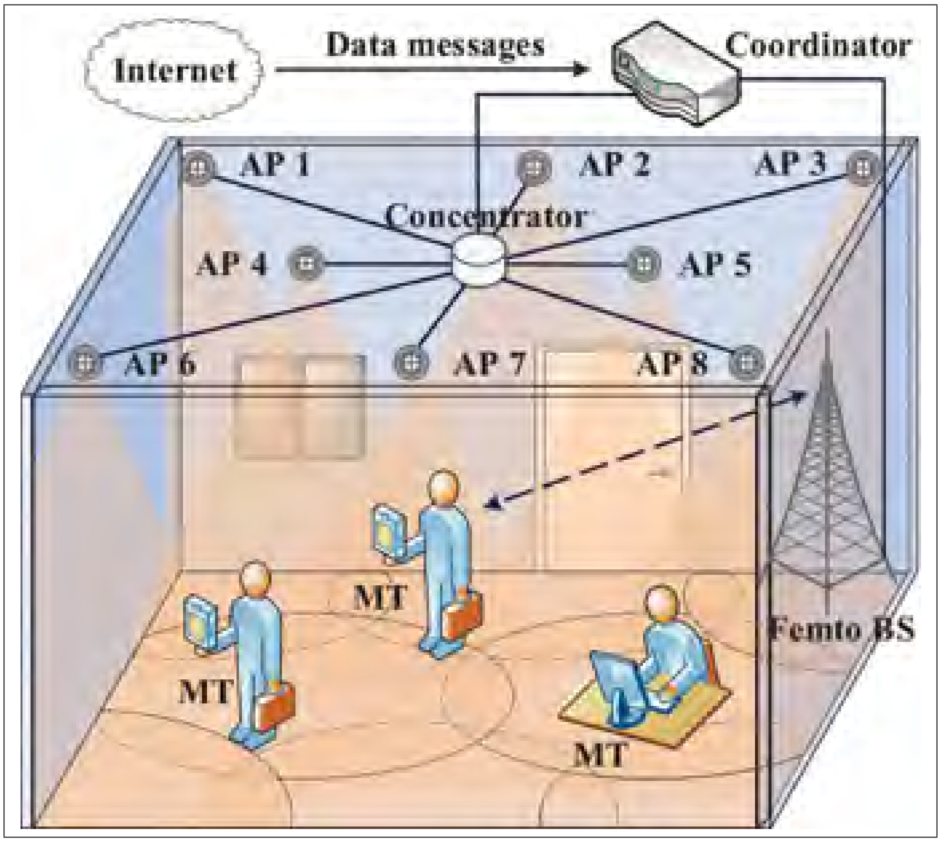

Figure 3.2 A family hybrid VLC-Femto system model Liang *et al.* (2017).

The flow diagram of their proposed algorithm and basic schematic of their system is shown in Figure 3.2 and 3.3.

In traditional vertical handover scheme for heterogeneous wireless access network, Quality of Service (QoS) has been commonly taken into account. QoE compares to QoS is more representer of the subjective feelings of users. In Chen *et al.* (2018), they introduced a Quality of Experience for vertical handoff, they employed a neural network based on QoE to find out the correlation between QoS and QoE. In the end, a Q-learning based handover algorithm has been designed to maximize the QoE utility of users within the network.

To achieve the demand of high data rate and good quality of service for supporting the video streaming or heavy streaming, handover mechanism employed to maintain the connectivity.

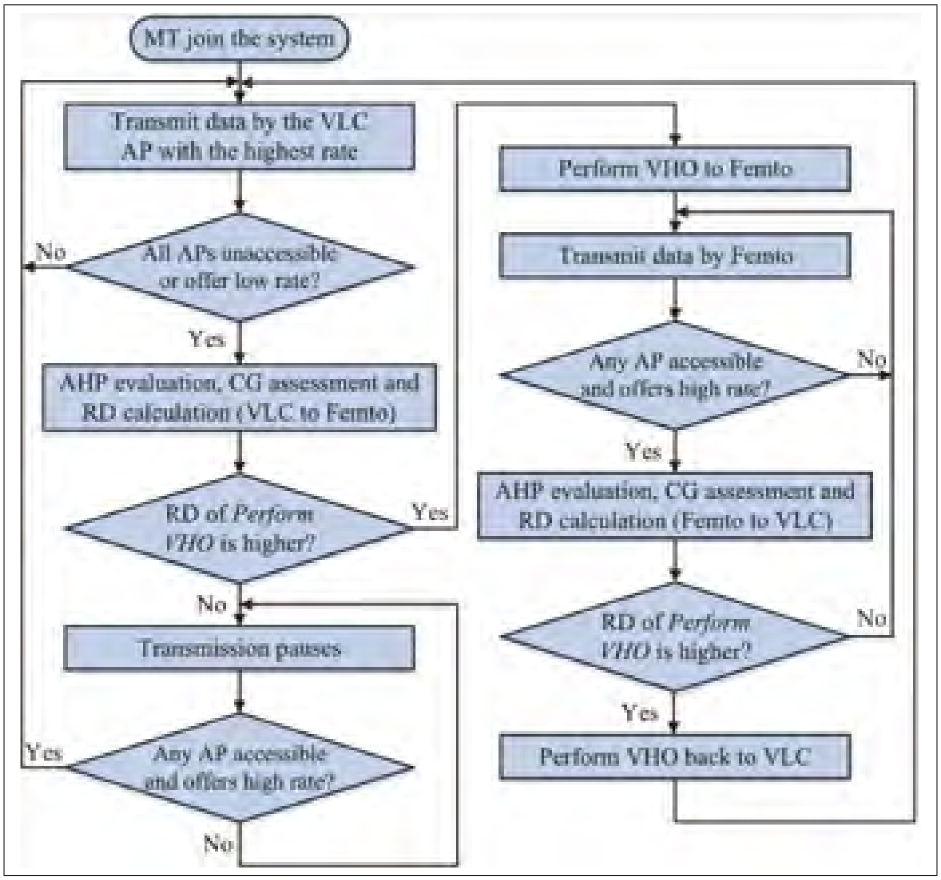

Figure 3.3 The flow diagram of the proposed AHP-CG VHO (Vertical Handover) algorithm Liang *et al.* (2017)

In Ali *et al.* (2018), they proposed a vertical handover mechanism which addresses resource allocation estimation, radio resource allocation decision, and radio resource allocation or allocation notification. To do so, they used a protocol stack to execute in all the devices for handover. The goal of this stack protocol is to trigger the essential communications between devices to optimize the handover decision process. Also, they introduced a new link-layer service access point (SAPs) for a common interface for link-layer function (Figure 3.4). Their main contribution is a novel mechanism to optimize the resource allocation and handover process between different technologies such as LTE-eNB, Wi-Fi, and Wi-Max.

Software define network (SDN) will provide the network with more capabilities to deal with users' demands while optimizing the radio resources. In this work López-Raventós *et al.* (2018),

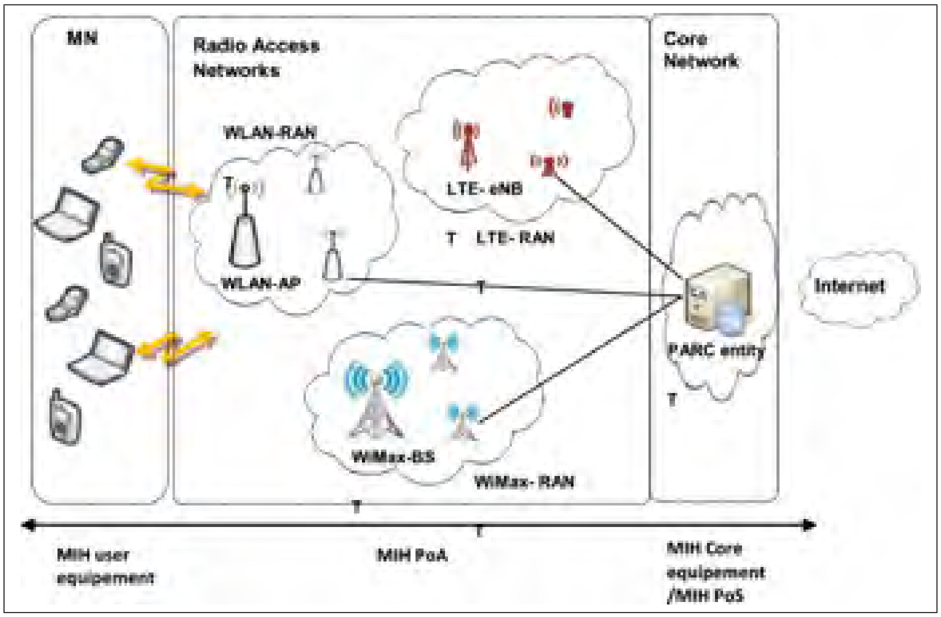

Figure 3.4 Proposed interworking achitecture in Ali *et al.* (2018)

they used machine learning to find possible optimize configuration through the learning process. Machine learning can improve the efficiency of the network by finding the optimal parameters for dynamic conditions. One potential issue in performance of the dense network is that (CSMA) have been designed to work in non-dense scenarios, then in campus or public environment, they'll face performance degradation. To overcome this problem, they introduced a wireless network that contains both (SDN) paradigm and Machine learning algorithms (Figure 3.5). To do so, a neural network has been used to predict the traffic and forecast the total amount of traffic along with a learning window to predict network behavior. This method is more about giving a general overview of the system in the next time interval without taking to account the characteristics of the handover process and mobility of users.

In the context of received signal strength, common usage is for positioning and location estimation. In Zhang *et al.* (2017b), authors proposed a novel positioning estimation strategy (Figure 3.6), which can avoid the AP selection problem in RSS-based Wi-Fi. Moreover, they offer a domain clustering technology for a more robust and reliable Wi-Fi management system using the classification method of machine learning. To this aim, they've used Naive Bayes

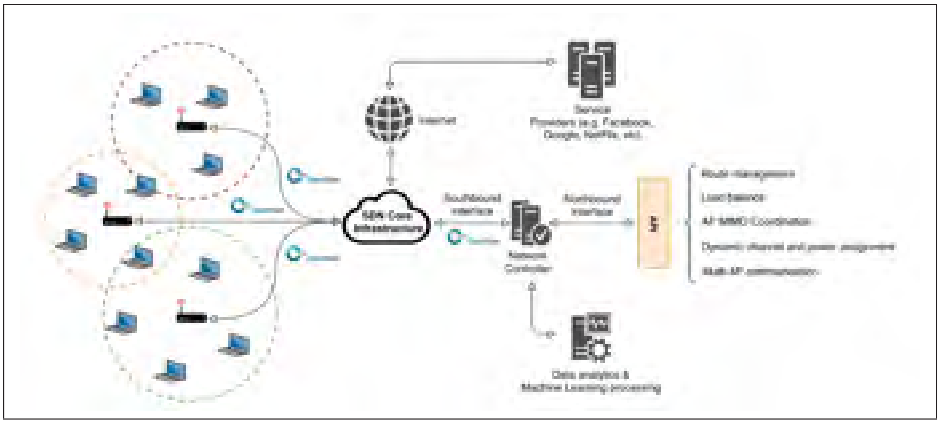

Figure 3.5 SDWN architecture with knowledge plane López-Raventós *et al.* (2018)

classifier and Weighted k nearest neighbor to classify the RSS and perform the clustering within a Wi-Fi network.

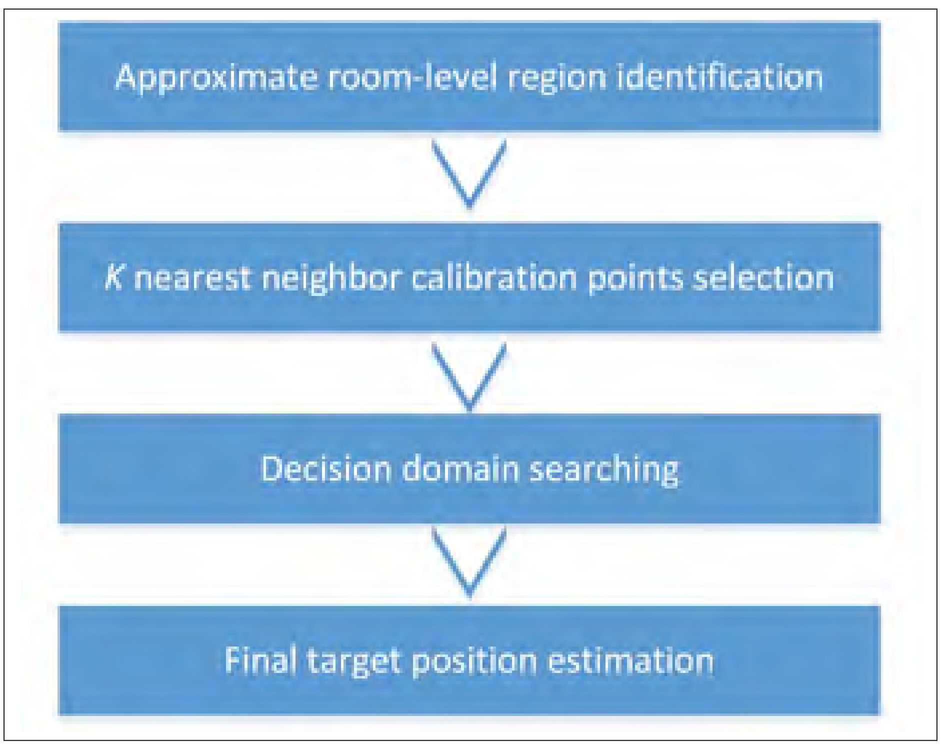

Figure 3.6 Flow chart of RDC (Desision Domain-based positioning) Zhang *et al.* (2017b)

As the seamless handover remains an important feature of wireless network, responsible factors for this feature have to be investigated. In this paper, Aibinu *et al.* (2017), a hybrid handover process, has been introduced, which is taking advantage of Machine learning techniques and fuzzy logic. In the machine learning phase, they build a prediction model (Figure 3.7) based on Artificial Neural network (ANN). The data that been used is time-series data of Received Signal Strength (RSS). Despite the power of this hybrid method, we have to mention that their model is time dependant wich makes it hard to generalize it over the different network service provider. Also, their prediction is limited to the estimation of only RSS wich makes this model a single parameters prediction and miss of other important traffic dependant parameters. However, they considered the other parameters at the end of their decision-making process.

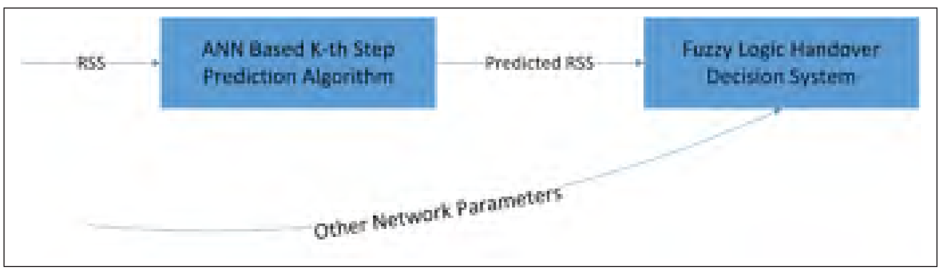

Figure 3.7 The block diagram of the proposed Hybrid AI based Handover Decision Algorithm Aibinu *et al.* (2017)

In this other work, they more focused on prediction of network parameters Shen *et al.* (2012). They established a prediction as well as a function (Figure 3.8) to compute the QoS. Then a hand of an algorithm proposed to make a decision based on the predicted parameters. Their experimental results showed that their results are more accurate than the other works based on the cost function.

In this study from 2013 Çalhan & Çeken (2013) they proposed an artificial neural network-based vertical handover decision algorithm for seamless handover between different wireless technologies. They also developed a Smart Mobile Terminal (SMT) to scan the wireless environment to optimize the radio resource allocation. Their decision-making algorithm

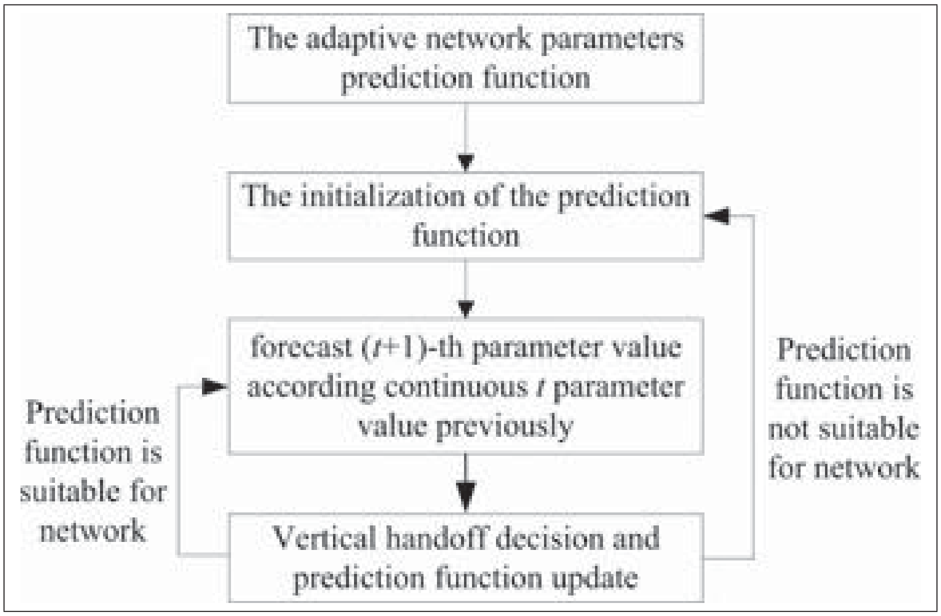

Figure 3.8 The proposed network parameter prediction algorithm
Shen *et al.* (2012)

considers handover between three different wireless technologies such as Wi-Fi, GSM, and GPRS.

Mobility prediction is also one of the key enablers for seamless handover in extremely dense cellular networks which employed to reserve the allocation and traffic prediction. To this aim author in this work, Farooq & Imran (2017) used a Semi-Markov model for Spatio-temporal mobility prediction. However, due to differences between Wlan and LTE network such as less mobility of users, we cannot directly apply this idea on Wlan network or vice versa. Still, their idea in the context of mobility prediction, which can help handover prediction inside a residential Wi-Fi network would be helpful for future works.

In this work, Hasbollah *et al.* (2017) authors proposed a prediction handover algorithm for vehicular application. Their goal is to predict the handover decision using optimal forwarding probability. The inputs of the model are vehicle location. For prediction, they used VLPHA implemented in NS-3 to find the best optimal forwarding probability value (Algorithm 3.9).

They've claimed that instead of focusing on a variety of parameters, our approach gives better performance.

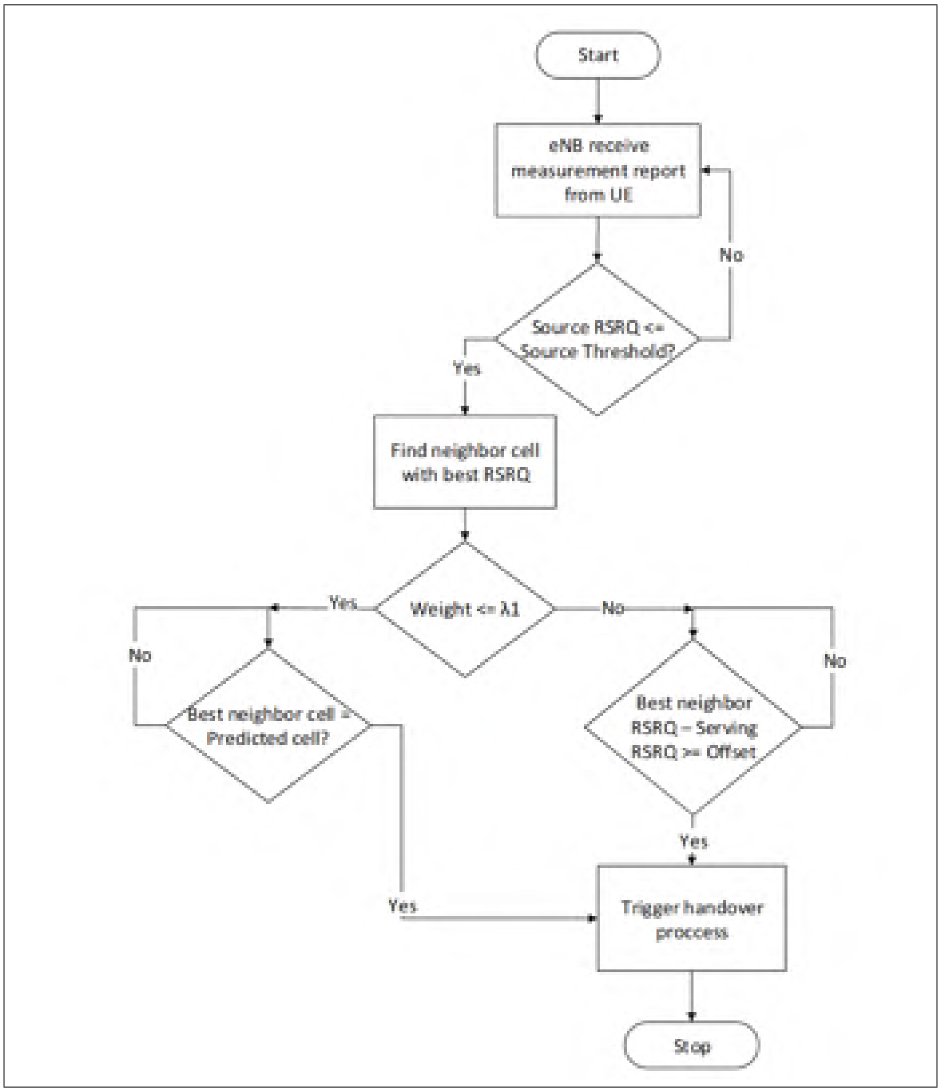

Figure 3.9 Vehicular Location Prediction Handover Algorithm (VLPHA) Hasbollah *et al.* (2017)

Another learning base approach for handover optimization in fog nodes proposed in Memon & Maheswaran (2019). Their focuses are on the Internet of Vehicles that would assist the optimal handover between fog nodes. To do so, they've taken advantage of machine learning power to learn the interaction of vehicle and fog nodes (Figure 3.10). A three-layer-feed-forward neural network has been used to predict the correct fog node at a given location and time. To learn the latency or cost associated with these services, they implemented a dual-stacked recurrent neural network (RNN) with long short term memory (LSTM).

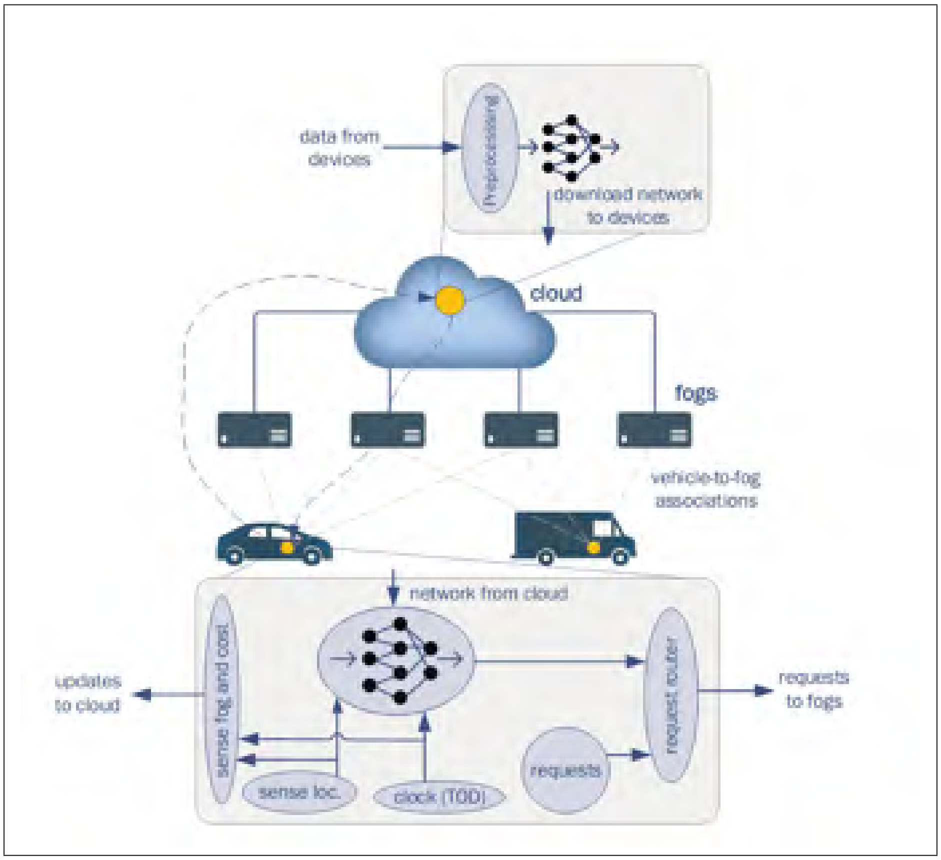

Figure 3.10 A system diagram showing the flow of information of our system through an edge computing architecture
Memon & Maheswaran (2019)

VLC (Visible light communication) is not only used for illumination but also offer connectivity. It is also taking advantage of huge bandwidth, high security, low cost, and health safety. In Bao, Adjardjah, Okine, Zhang & Dai (2018), authors propose a new (VHO) mechanism to guarantee continues transmission and maximize the QoE (Quality of Experience) for a user. They treated the problem with Markov Decision Process (MDP).

Various handover has been proposed based on the RSSI, where handover decision is made by comparing the RSSI value to a predefined threshold. In Zhang *et al.* (2017a), authors considered the handover as a dynamic resource allocation problem. They prioritize the allocation of APs to maintain the QoS of the urgent communications based on the standard IEEE 802.11 handover mechanism without taking into account the user behavior. The proposed VHO (Vertical Handover) algorithm in Liang *et al.* (2017) employed (AHP) analytic hierarchy process and (CG) cooperative game to make (MADM) multi-attribute decision-making thus supporting various traffic types, Such a way they distribute the workload or even before wireless station became overloaded or user starts to move through the network.

In Ali *et al.* (2018) authors present vertical decision algorithm for handover with considering the traffic class which mobile user are using. They used two modules in their algorithm, first to estimate the parameters for handover and to select the optimal network. To have an intelligent vertical handover decision, fuzzy logic and genetic algorithm have been used. Software-defined networking (SDN) model provide new capabilities to deal with demands while achieving better levels of efficiency and flexibility in those dynamic and complex scenarios. In López-Raventós *et al.* (2018) Authors used machine learning techniques (ML) to improve network resource usage and management by identifying feasible configuration through learning. In Aibinu *et al.* (2017), they build a time series prediction model made up of a hybrid of artificial neural network (ANN) and fuzzy logic to decrease the effect of ping-pong caused by handover mechanism. The data was fed to the newly proposed k-step ahead ANN-based RSS (Received Signal Strength) prediction system for estimation of prediction model coefficients.

Many studies have been done to move from decentralized to centralized Wi-Fi network. Also, various studies focused on maximizing the overall throughput during the handover mechanism and reducing the number of ping-pong transfers. While the widely used approach for handover is a comparison between RSS value and the defined RSS threshold, our method is not using any threshold. Moreover, many of studies are based on single metrics; several metrics used to measure a score and then using some of that metrics as our features in the prediction model.

Our proposed approach aimed to address the performance degradation issue in overlapping APs, to find out which handover is suitable for the system to trigger. It is shown that if we predict the best metrics, handover can happen in the optimal moment. This methodology works without any significant loss of QoS.

Table 3.1 depict the comparison between our proposed method and related works.

Table 3.1 Comparison of related work with proposed method

Related Works	**Handover Score**	**QoE**	**Resource Opt.**	**ML Pred.**
Sarma *et al.* (2016)	No	Yes	Yes	No
Liang *et al.* (2017)	No	Yes	Yes	No
Chen *et al.* (2018)	Yes	No	No	Yes
Ali *et al.* (2018)	No	Yes	Yes	No
López-Raventós *et al.* (2018)	No	No	No	Yes
Zhang *et al.* (2017b)	No	No	No	Yes
Aibinu *et al.* (2017)	No	Yes	Yes	Yes
Shen *et al.* (2012)	No	Yes	No	Yes
Çalhan & Çeken (2013)	No	No	Yes	Yes
Farooq & Imran (2017)	No	No	Yes	Yes
Hasbollah *et al.* (2017)	No	No	Yes	Yes
Memon & Maheswaran (2019)	No	Yes	Yes	Yes
Bao *et al.* (2018)	Yes	No	No	Yes
Zhang *et al.* (2017a)	No	Yes	Yes	No
Baseline System	No	No	Yes	No

CHAPTER 4

METHODOLOGY

Our methodology consists of different stages from calculating the handover score to designing a model to predict the parameters of an optimal handover mechanism.

At the end of this section we will dive in our proposed policy based on handover score calculated from feedback of user's data.

4.1 System Description

The handover mechanism guarantees the connectivity of users that are moving within a Wi-Fi network. It can affect the QoE of users. The moment that handover is triggered is very important to maintain the throughput and satisfaction of users. To this end goal ML techniques are used to predict the best parameters of handover which help the system to maintain the QoS for service provider and mitigate the QoE drop for users as well as ping pong avoidance in the Wi-Fi network. Handover prediction problem is treated as a supervised learning problem where handover can be associated with a set of the possible continuous tag. A preliminary step before the prediction process turns our data to a supervised learning process. To do so and to obtain this tag, we will measure a score for each handover.

All the implementation was done in Python for this project. Scikit-learn was used for the implementation of machine learning algorithms as well as the implementation of algorithms for testing and comparison purpose. The details of the implementation are as follows:

4.2 Handover Detection

In order to detect the handover event from the user data we have to set of rule for a detection pipeline. Our main source of data is coming from 3 major table in dataset. Combined performance analysis which gives us the throughput and bit rate value for each user in the interval of one minutes. Client Dashboard gives us radio resource information about user and AP Dashboard gives the same information for Access Points. By joining these tables together we can have a big table including infromation form Users, Access Points and their radio resource details such as MAC address of users, APs and band connectivity of specific user to one Access Points. Our aim is to spot each handover which happend inside network using user data to evaluate the performance of each handover and later on calculate an handover score based on Quality of Experience coming from user data. To achieve our goal we defined following assumptions in Figure 4.1.

START
INPUT: Data from 3 table (Combined Performance, Client Dashboard, AP Dashboard)
DETERMINE: IF User getting far from AP1:
  5GHz then 2.4GHz
  AND User getting close to AP2:
    Enter 2.4GHz zone AND AP1 MAC ≠ AP2 MAC AND 2.4GHz
APPEND(HANDOVER_LIST)

Figure 4.1 Assumption for detecting the handover event.

4.3 Proposed QoE metrics

AP to AP scenario considered for handover. We take different metrics to measure a continues (Non-Discrete) score. Metrics (Table 4.1) are from our data-set, which is a real-world data collected from access points installed in the smart residence of ETS campus located in Montreal. The difference of RSSI values of source and destination APs which user handover-ed to calculated as well as the difference of throughput and number of handover happened for that user within the last three-time slot. Then a score function was built to map a value to each handover as the label. This score is from 0-1, which 0 is a poor score for that handover and 1 is the best score that one handover can achieve. All the metrics have the same weight in our handover score.

$$Score = \sum_{i=1}^{N} RSSI_{b_i - a_i} + N_{Handover_i} + dl_{b_i - a_i} + ul_{b_i - a_i} \tag{4.1}$$

$$HandoverScore = MinMaxScaler(Score) \tag{4.2}$$

The score obtain using Equation 4.1 that we will scale in Equation 4.2 to get a handover score in range of 0 to 1.

Table 4.1 Score metrics

Metrics	Notation
RSSI diff (Before-After)	$RSSI_{(b-a)}$
Number of Previous handover	$N_{handover}$
Throughput diff (Before-After)	$dl, ul_{(b-a)}$

4.3.1 Data Pre-Processing

Pre-processing is transformation applied to our raw collected data before using it in machine learning algorithm. In order to achieve better result from model in machine learning project we have to provide the model with specific format.Some models doesn't support null values, therefore these values have to be managed from original collected data. Datasets are often containing some outlier as well that we have to remove. Download throughput and upload throughput in some case have very big value, in order to have a unified format, we will omit this cases from our data. The dataset comprised of attributes with varying scales, our model can benefit from re-scaling the attributes to all have the same scale. We used $MinMaxScale$ class from scikit-learn to re-scale our attributes. After scaling all of the values are in the range between 0 and 1.

4.3.2 Data Cleaning

Data cleaning is an important part of the pre-processing step. To this aim we have to remove the Outlier and handle the Null value inside the data. For Null value observed in our data we took the average of that attribute in that column and replace it with average of the column. This helps us to keep all the data from our population and do not miss any pattern in system behaviour.

4.3.3 Feature Extraction and Selection

Feature selection will help us to reduce the overfitting in the learning process. Less redundant daya means less opportunity to make decision based on noise. Feature selection will also lead to improve the accuracy. Less misleading data means modeling accuracy improves. It also reduce the training time. Less data means that algorithm train faster and converge to solution in optimal time. In this section of our work we used a method for feature extraction and feature selection. Principle component analysis used as our feature extraction. The main idea of the PCA is to convert a set of observation of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. PCA is basically used to reduce the number of variables and make sure that variables are independant of one another to avoid multicollinearity.

first step in PCA is to calculate a matrix that summarizes how our variables all related to one another the separating the matrix into two components, direction and magnitude. Answering this question of what would fitting a line of best fit to this data look like, help us to transform our original data to align with these important directions. At the final step by projecting the data into a smaller space, we reduce the dimensionality of our feature space.

Before everything we should have tabular data organized with n rows and likely p+1 columns, where one column corresponds to your dependant variable and p columns where each corresponds to an independent variable.

1 If a variable exist and is part of our data then separate our data into Y and X.

2 Take the matrix of independent variables X and, for each column, subtract the mean of that column from each entry. (This ensures that each column has a mean of zero.)

3 Decide whether or not to standardize. Given the columns of X, are features with higher variance more important than features with lower variance, or is the importance of features independent of the variance? (In this case, importance means how well that feature predicts Y.) If the importance of features is independent of the variance of the features, then divide each observation in a column by that column's standard deviation.

4 Take the matrix Z, transpose it, and multiply the transposed matrix by Z.

5 Calculate the eigenvectors and their corresponding eigenvalues of $Z^t Z$.

6 Take the eigenvalues , ... ,p and sort them from largest to smallest. In doing so, sort the eigenvectors in P accordingly.

7 Calculate $Z^* = ZP^*$. This new matrix, Z^*, is a centered/standardized version of X but now each observation is a combination of the original variables, where the weights are determined by the eigenvector.

Following the aforementioned steps, Figure 4.2 shows the extracted feature among other attributes of our dataset.

User Upload Thr	User Download Thr	Noise	Transmit power	Signal noise ratio	RSSI
Channel Usage	Channel Frequency	AP Download Thr	AP Upload Thr	Transmit bitrate	Receive Bitrate

Figure 4.2 Highlighted attributes are the selected features to feed to our prediction algorithm.

4.3.4 Prediction Model

Predictive modeling often use statistics to predict the outcome an event. This type of modelling can apply to almost any type of events without considering time of occurrence. In proposed method support vector regression will estimate the handover score or our proposed QoE metrics. Our task is referred as regression. Prediction of continuous values based on observation from the data.

4.3.4.1 Support Vector Regression

(SVR) Employed as our prediction model. This model is a version of the support vector machine (SVM) for a regression introduced by Vapnik in 1995 Vapnik (2013). The idea of this method is to map the training data into high dimensional feature space by using a non-linear mapping function and then obtaining a linear regression problem. Details of this method can be found in Vapnik (2013). Also, a brief overview of support vector regression-based modeling is given in this section Javed, Chan, Savkin, Middleton, Malouf, Steel, Mackie & Lovell (2009).

Training data considered as set of input vector $\{x_i\}_{i=1}^{N}$. Output vector considered as $\{y_i\}_{i=1}^{N}$ where N is the number of input data. This algorithm aim to find a function $f(x)$ that has at most ϵ deviations calculated from the targets for all the input vector. The function is as follow:

$$f(x) = \langle \omega, \phi(x) \rangle + b \tag{4.3}$$

in function (4.1) $\langle\rangle$ is the dot product. high-dimensional feature spaces presented as $\{\phi(x_i)\}_{i=1}^{N}$ which are non-linearly transformed from x. By minimizing the following regularized risk function The coefficients ω and b are estimated Vapnik (2013).

$$R(w) = \frac{1}{2} \|\omega\|^2 + C \frac{1}{N} \sum_{i=1}^{N} L_\epsilon(y_i, f(x_i)) \tag{4.4}$$

The first term of function (4.2) $\frac{1}{2} \|\omega\|^2$ is the regularized term, and it is used as flatness measurement of $f(x)$, To determine the tradeoff between the VC dimension of the model and training error a fixed constant called C used. The ϵ-insensitivity loss function is L_ϵ, which defined as:

$$L_\epsilon(y_i, f(x_i)) = \begin{cases} |y_i - f(x_i)| - \epsilon & |y_i - f(x_i)| > \epsilon \\ 0 & otherwise \end{cases} \tag{4.5}$$

An ϵtube defined using function (4.3). C and ϵ are user-defined. C is regularization constant and radius ϵ of the tube. The parameter ϵ used to controls the size of the ϵ-insensitive zone and to fit the training data Vapnik (2013).

By solving the optimization problem with the mentioned constrains, we have:

$$f(x) = \sum_{i=1}^{N} \beta_i \phi(x_i).\phi(x) + b \tag{4.6}$$

Where the coefficients β_i corresponds to each (x_i, y_i) and is nonzero only for a small subset of the training data named as support vectors. In SVR, by only using support vectors, the same solution can be obtained as using all the training data points.

We will use a kernel function to calculate the inner product in feature space to do all the computation directly in the input space. By putting the kernel function $K(x_i, x_j)$, equation (4.4) can be written as:

$$f(x) = \sum_{i=1}^{N} \beta_i k(x_i, x) + b \tag{4.7}$$

To obtain good generalization, there is several kernel function such as Sigmoid, polynomial, linear, and RBF. In this work, we used the RBF kernel to model our score prediction Javed *et al.* (2009). The RBF kernel function is as follow:

$$k(u, u') = exp(-\frac{\left\|u - u'\right\|^2}{2\sigma^2}) \tag{4.8}$$

Change in behavior or pattern drift is an issue in the context of supervised learning data-driven approach. Our Online support vector regression (SVR) approach introduced in this work is an efficient online learning method for SVR. This method is capable of handling the learning of behavior and changes in handover mechanism users' within the Wi-Fi network. This approach can effectively detect and add a new pattern or update the change of behavior to our prediction model.

4.3.4.2 Decremental and Incremental learning

Decremental and Incremental learning, as proposed in Cauwenberghs & Poggio (2001) provide us with an effective method for SVR to adaptively update the model with new data and in formations. Instead of the running model from the beginning, this method can add new data-points and remove an existing point in the model. In this work, we used decremental and incremental learning to update our data with new incoming data from the centralized system.

This method, including two connected components which are working together: one is offline training, which is training the model based on an available and previous data point that we have. The other component is online learning, which it's duty is to identify if new data are new patter, or change pattern or previous existing pattern then taking relative action regarding input.

4.3.4.3 Offline Training of Online-SVR

This section includes two steps. In the first step, we have to select the feature vector in the training dataset. The goal of this step is to find the feature space S transformed from a part of our training dataset. The second step is to train an SVR model with the founded features using a classical algorithm.

4.3.4.4 Online Learning of Online-SVR

This section is including the detection of new or changed pattern regarding the characteristic of the inputs and the bias of the prediction of the new data points and then making action about update task. The difference between predicted data and the real output of the model will take as bias and then to decide the change of the existing patterns. Suppose a new data point is (X_N, Y_N) and the prediction model for this instance is M trained on feature space S. The first step is to verify if (X_N, Y_N) is a new pattern by calculating its local fitness J_S, N with (4.3). To verify if the mapping γ_N of (X_N, Y_N) can be expressed by a linear combination of all feature space in S. If $1 - J_S, N$ is bigger than the predefined threshold π, the linear combination of feature vector in S cannot sufficiently approximate γ_N of (X_N, Y_N) is then taken a new pattern and added directly to the model using incremental learning in Cauwenberghs & Poggio (2001); the model M and the feature space S are updated at the same time and await for the next new data point without going to the second step of checking the bias of the predicted values compared to the true output. Otherwise $1 - J_S, N\pi$ it is not a new pattern, and we proceed to the second step to verify if there is any change in the existing patterns.

The second step of online learning feeds new data point to the model and calculate the difference between the predicted values using M and the real output y_N of the new data point. bias = $(|y_N(pred) - y_N|$ with $y_N(pred)$ value of the new data point. If the bias is smaller than the predefined threshold σ (the second tolerance parameter), there is no change in the existing patterns, and the model M is kept unchanged and awaits for the next data point. The procedure is as follows:

1. A vector $m = (m_1, m_2, ..., m_i)$ is used to reduce the contribution of each feature vector to the SVR models. Each value in m corresponds to a feature vector in the model.

2. m is set to be a zero vector before offline Training.

3. When the model M is trained during Offline Training with the selected feature vectors from the training dataset, m_i is increased by 1 if the corresponding feature vector is a support vector.

4. Each time the model is added with one new data point, a mew $m_l + 1$ is added to m to record the contribution of the new feature vector in the model. After the model is updated with addition, the contribution m_i of each feature vector in the model is updated with the contribution update rules: if the data point is a Support Vector in the new updated model, its new contribution is calculated as $m_i^n ew \leftarrow \tau * m_i + 1$, with τ a positive constant smaller than 1. The contribution of a feature vector in the new model is more veighted than that in the old models; otherwise it is kept unchanged.

5. when a change is detected with respect to the old patterns, the first step is to calculate the values a_N for new data point according to (5). Then, among all the feature vectors in the model with non-zero values in a_N, the one with least contribution, say m_1, is deleted from the model using Decremental learning as in Cauwenberghs & Poggio (2001) and m_1 is reset to zero. If there are several Feature vector with the same contribution and the least contribution, the Feature Vector to be replaced is selected as the oldest one among them.

6. The new data point is added to the model using Incremental Learning in Cauwenberghs & Poggio (2001) and it inherits the contribution m_1, which is zero for now. The vector m and the feature space S are updated, and also the contribution of the feature vector is updated according to the rules in step 4 above.

Note that the new data point replaces the feature vector in the model with least contribution to the SVR models among all those with non-zero values in the linear combination (according to (4.4)). This strategy for updating a changed pattern must and can keep the feature vectors in the model linearly independent so that the kernel matrix K_s, s in (4.3) is invertible and the online learning can continue to be carried out. If a new pattern is added because of the noise, this strategy can decrease the influence of the new data points and keep the capability of the model, as only one existing feature vector with least contribution is replaced. Note also that if a new data point is a new pattern, is added instantly in the model, without consideration of the bias of its prediction, so that a maximal richness of the patterns are kept in the model. This is different from the online learning methods which consider only the prediction accuracy. The changing patterns are made of the points which can be expressed as a linear combination of existing patterns, but with a bias of prediction larger than the present threshold γ. This allows replacing a changed pattern instead of adding it in the model, to keep the feature vectors in the model linearly independent and up-to-date. Note that proper selection of the (positive) values for the tolerance parameters, ϕ, and γ, cam efficiently decrease the influence of noise and avoid over-fitting by selecting only informative parts of the dataset.

Figure 4.2 show how our SVR model can update itself with new unseen data collected by controller.

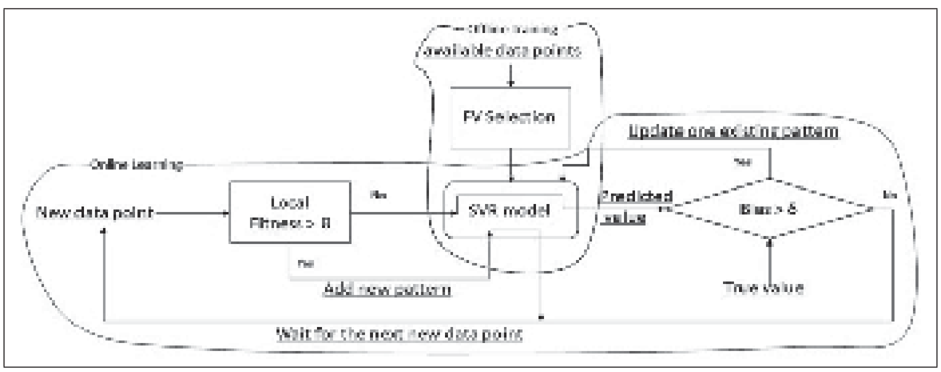

Figure 4.3 Paradiagram of SVR Liu & Zio (2016)

4.4 Policy

After deployment of our trained model in the centralized system controller, assume the scenario that a user is walking the path between two access points and an unknown application is running in his/her mobile device. The system sends a request along with historic data of the user and asks whether, if the system triggers the handover would be any degradation in QoS or improvement or no difference after all. For this aim, we defined a policy based on the score $[0 - 1]$ that system have to follow. This policy will take to account the RSSI value of user and possible

Access points around the user as well as the throughput of the user at that moment. Such a way the users should have an expected value of RSSI. To this aim, we defined specific spans. Handovers, which receives poor [0 – 0.4] scores and expected RSSI value, central system have to postpone them and check the user status later again because they cause degradation in QoS. Handovers with a score between [0.4 – 0.6] maintain the QoS, which means this handover don't cause degradation and improvement, it only maintain the QoS. Scores [0.6 – 10] have a positive impact on QoS, such a way that they can improve the throughput of users.

Table 4.2 Score Policy

Score Category	Range
Degradation	< 0.3
Maintaining	$0.3 < Score > 0.6$
Improvement	> 0.6

4.4.1 Model Deployment

Our software package is containing five main components. Cloud server comprise of Access point abstraction level, network manager and back-end. Wireless access network is the other component with access points located on it. Our data set is collected from this component using RESTful API. Last component is operation and system support which is responsible for provisioning and billing. Our model will deploy on network manager along with fault management and performance management modules. Figure 4.4 shows there three components on details.

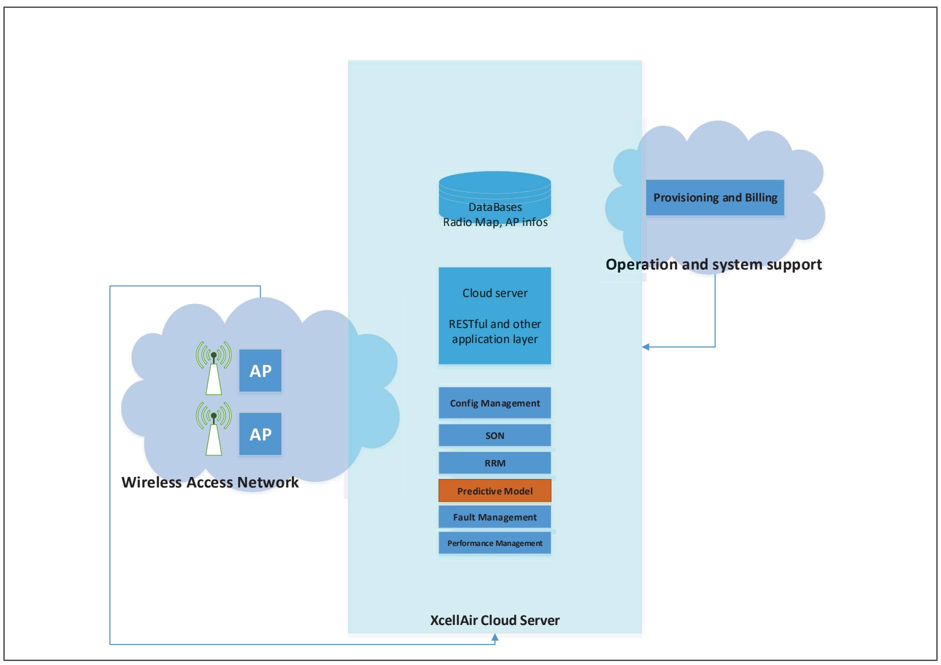

Figure 4.4 Three main component of management software and deployment of our prediction model on network manager component

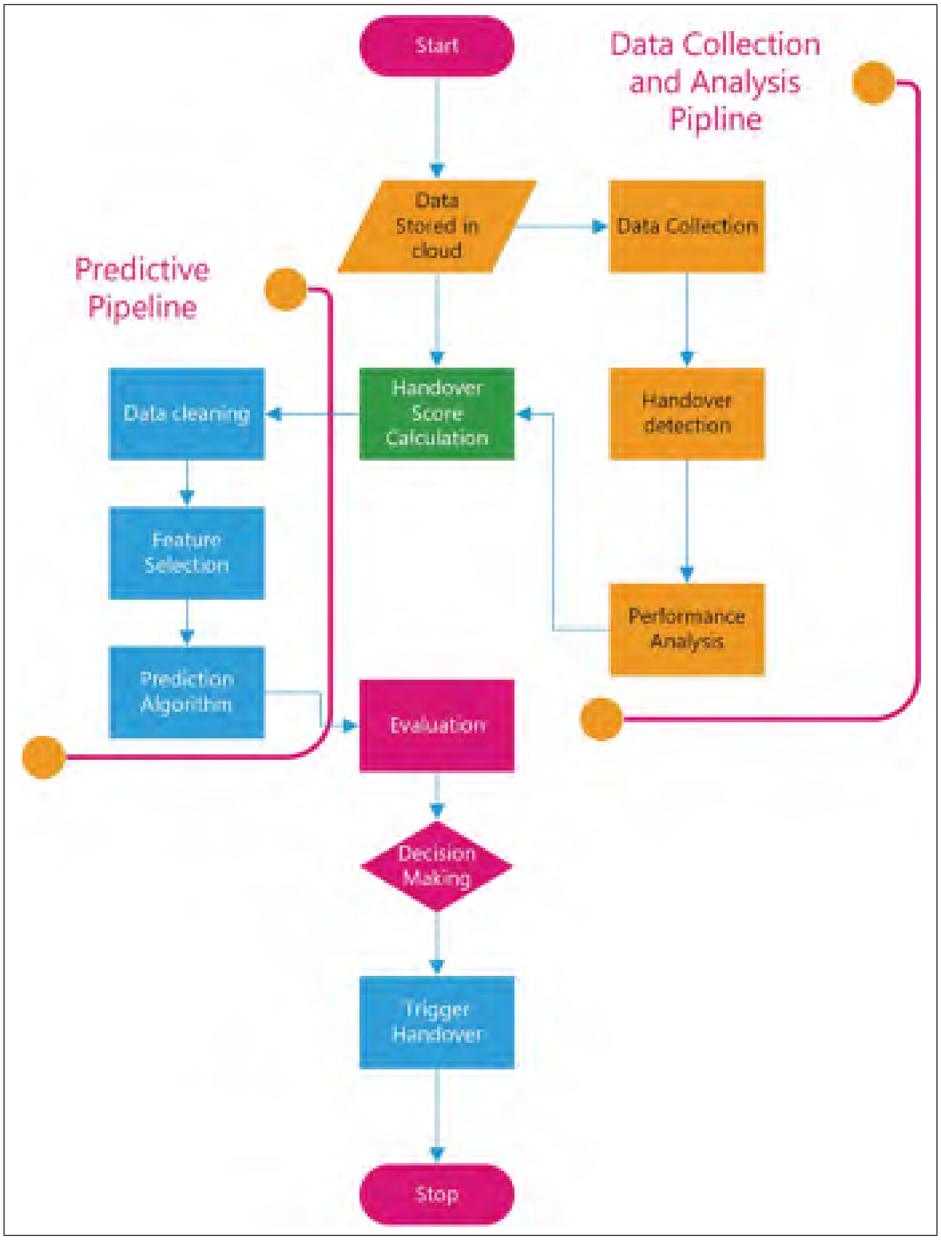

Figure 4.5 Overall flowchart of our proposed framework. The system will log the historical data in a cloud based storage. Our platform will use the historical data to train the algorithm and update itself using new incoming data, then making action based on current state and and predicted parameter for future state.

CHAPTER 5

EXPERIMENTAL RESULTS AND ANALYSIS

In this section we will demonstrate the validation of our method in test bed. This section consist of some analysis for quality of experience degradation and results of proposed method on test bed. We found the moment that handover happened in the Wi-Fi network. We analysed different metrics such as throughput before and after each handover to observe the effect of handover on quality of experience. Then we will dive in to the prediction results of effect of prediction on improvement of handover mechanism as well as performance of our prediction model.

5.1 TestBed

The test bed is located in student dorm building of ETS (École de Technologie Supérieure) situated in the heart of downtown Montreal, a city of roughly 2M people. This provides an ideal mix of circumstances for our experiment and a highly challenging Wi-Fi environment, as well as typical and replaceable MDU (Multi-Dwelling Unit) location. ETS students are heavy users of several different internet services, they tend to move around the building, visiting each others' rooms or study in the areas. The dorm contains over 75 Wi-Fi access points, all of which can be optimized in connection to each other and to the users accessing them. Finally, the users are active on the network over a long period of time-up to 12 hours a day - creating a long usage profile, which is the key to our analysis and learning process.

Each Wi-Fi Access Point (AP) is updated with a thin agent that reports specific key performance indicator (KPIs) to a cloud-based server. In this instance, the cloud server was hosted in a public cloud space. The cloud server collects KPIs from the APs on performance parameters such as throughput, signal to noise ratio (SINR), bit error rates, neighbor APs, channel characteristics, and user device statistics. The bulk (95%) of the system intelligence and algorithmic logic is executed within the cloud server. The APs themselves can also undertake certain actions, under policies controlled and distributed by the cloud server. By analyzing the stats that the AP agents have collected, the cloud server works to determine what triggers to pull enhance overall Wi-Fi and individual user performance.

5.2 Dataset

The datasets for training and testing the machine learning models were generated from Access points located in ETS (école de technologie supérieure) smart residence. Metrics collected from these access points logs in a cloud storage with an interval of every one minutes — these data including different tables. Tables then merged to have a concrete dataset of handovers within this network. The network covers the area of the ETS residency (Phase 3,4), which consists of 3615-meter square. There are more than 360 clients who use the network to access different services via the Internet such as video streaming, voice over IP (VOIP), file sharing, web pages, etc. The user connection information is monitored and stored in the central controller. The log files are in CSV format and include:

- AlarmDetails: CSV file with the details of each alarm generated by any of the managed APs.
- apDashboardClientsDetails: CSV file with client information reports sent by all the managed APs for all connected clients to the server.
- apDashboardRadiosDetails: CSV file with the raw radio information reports sent by all the managed APs for all radios to the server.
- apLog: CSV file with the AP log information extracted by the server from logs files uploaded to the server by all managed APs every fifteen minutes.
- apPosition: CSV file with information about the location of the APs.
- channelChange: CSV file with the details of the conditions before and after for each channel change that occured for any of the managed APs.
- channelUtil: CSV file with the channel utilization statistics throughput, channel switches, noise level, etc.

- clientSteering: CSV file including details about steering of moving users between APs and Bands.
- clusterinfor: CSV file with details configuration of the server once per hour.
- combinedPerformanceReport: CSV file with statistics per clients in the AP dashboard clients details CSV.
- eventDetails: CSV file with information about details happend in the network.
- hn-group: CSV file including details about profile of each APs, their address and the topology that they are using.
- hn-profile: CSV file including niformation about air time percentage in different band connectivity.
- neighbourInfo: CSV with summary of all detected neighbors of managed APs.

5.3 QoE Degradation Analysis

The statistics of our experiments comes from clientSteering CSV file, where There are several identification information such as Client MAC address and managed Access point MAC address when they are connected and Mac address of target AP which each client is going to connect to after steering. There are also details such as RSSI of source AP and target AP as well as steering status and the time which steering action issued on it. To have a rich source of raw data, tables of 442 days merged, then all the analysis performed on several months. There are several reasons for steering, such as 5G preferences, Low signal strength, and congestion for each record in the data. Some fault in logged data observed during analysis that we preferred to don't use steering type reason which already exists on the logged data to infer between Band-to-Band steering and AP-to-AP steering. Instead, we applied our definition for these terms based on the most basic information such as "AP Mac Address" in data. We aim to find two types of steering, AP-to-AP when Mac address of an AP before and after steering is different (Figure 5.1) and Band-to-Band steering when Clients steered on the same AP but through different interfaces. In this case, we considered the "BSSID" of APs (Figure 5.2), which only the two last characters are different. In the end, we came up with several new tables, including full data and details of each client and distribution of band (2.4G or 5G) connectivity after steering.

Figure 5.1 In access point to access point you can see different MAC address at two different APs.

We addressed Band steering and client steering in the data-set and tried to study the performance of the system before and after each steering. We divided our analysis into AP-to-AP steering and Band-to-Band steering. Each client can either connect to 2.4GHz or 5GHz; we investigate the connectivity distribution of clients on two 5GHz and 2.4GHz band for Band-to-Band steering. A client with a signal strength below the idle client steering signal threshold and the Nonidle client steering signal threshold are candidates for client steering. The first threshold (idle threshold) is used when client UL throughput is below the level defined in the Nonidle client UL throughput threshold. The wait interval defines the amount of time the cloud logic will wait after a client steering before assessing the results. This is used for client steering tracking only. The number of successful steering varies from month to month due to different number of users in seasons such as summer session which the number of students is less than other sessions. In Figure

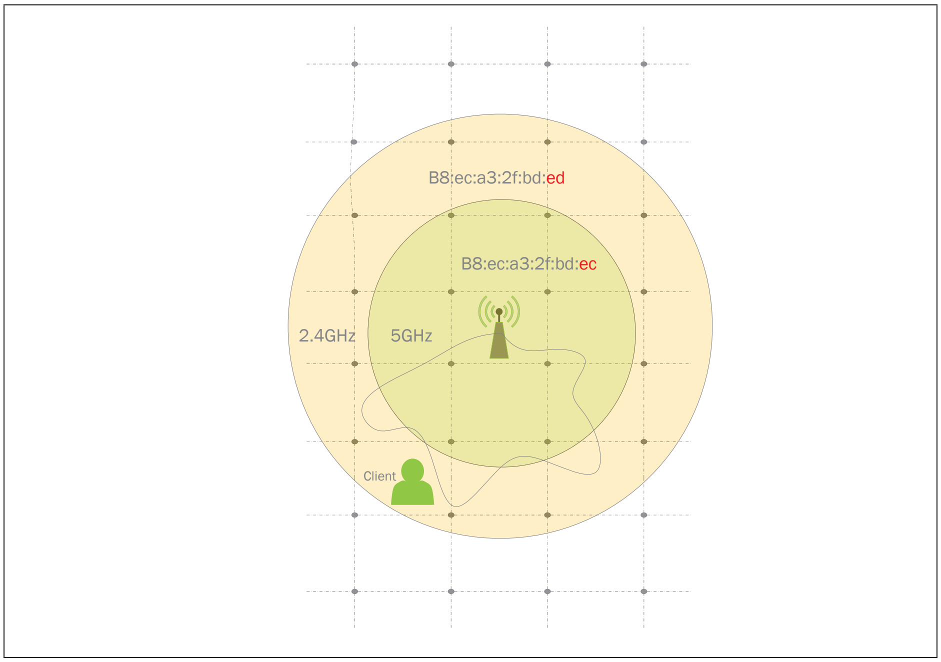

Figure 5.2 Two BSSID at the same AP (Only two characters of BSSID are different)

5.3 demonstrate the number of Band-to-Band steering and AP-to-AP steering. The number of Band-to-Band steering is much more than AP-to-AP steering. In figure 5.4, we can see the degradation and improvement of QoS after steering in both AP-to-AP and Band-to-Band steering. In Band-to-Band steering we observed the number of degradation are much more than improvement. But as for AP-to-AP steering, we find out the improvement are three times more than degradation, which even here the number of degradation is considerable. For the rest of the report, we chose month April as representative of our analysis.

5.3.1 AP-to-AP Steering Analysis

Our analysis coming from clientSteering CSV file. In this file information of each steering logged and saved with the time of each steering. We took the average of RSSI one minutes before and after each client steering and tried to study the improvement and degradation of QoS after each steering. We grouped the data based on each client and omitted the client with records less than 10. The value of RSSI in our data is a negative value ranging from -1 to

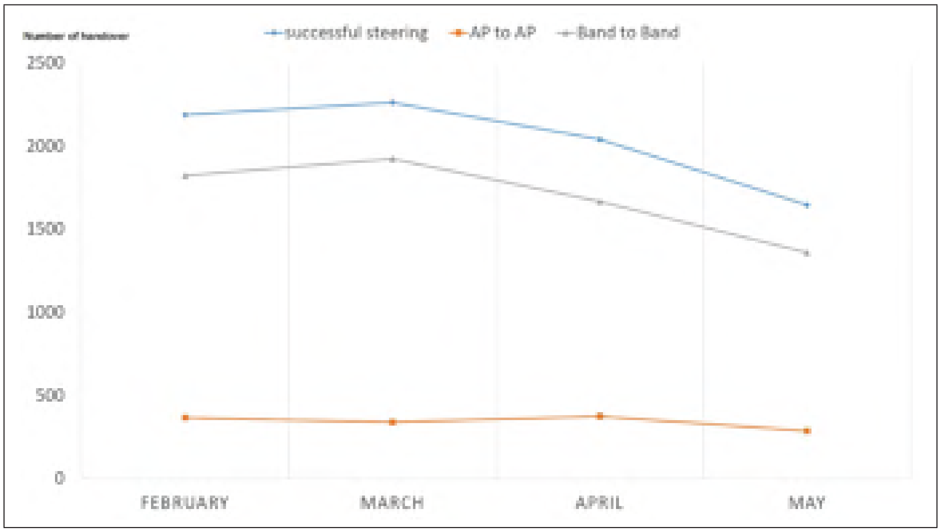

Figure 5.3 All steering, Band-to-Band steering, AP-to-AP steering. imp stands for improvement, deg stands for degradation and no change for maintaining.

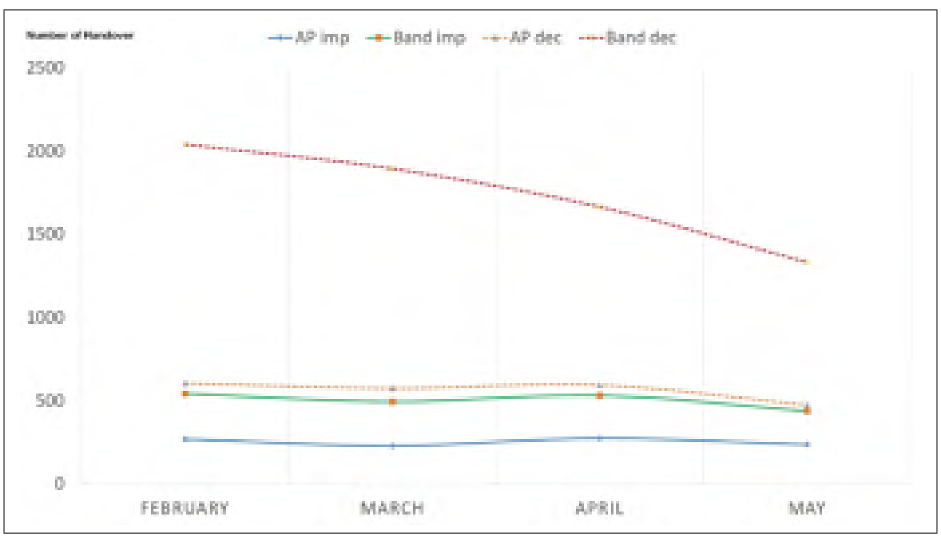

Figure 5.4 Band-to-Band improvement and degradation, AP-to-AP improvement and degradation. imp stands for improvement, deg stands for degradation and no change for maintaining.

-100. As for the improvement and degradation, we calculate the difference of RSSI before and after each steering, if the difference value is between 0 to 10, we assigned the corresponding steering to "Maintaining" category. If the difference value is between 10 to 60, we assigned it to "Degradation." For the "Improvement" category if the RSSI improved for value more than 0, we considered that steering as our improvement. Figure 5.5 show the number of successful AP-to-AP steering with status of QoE.

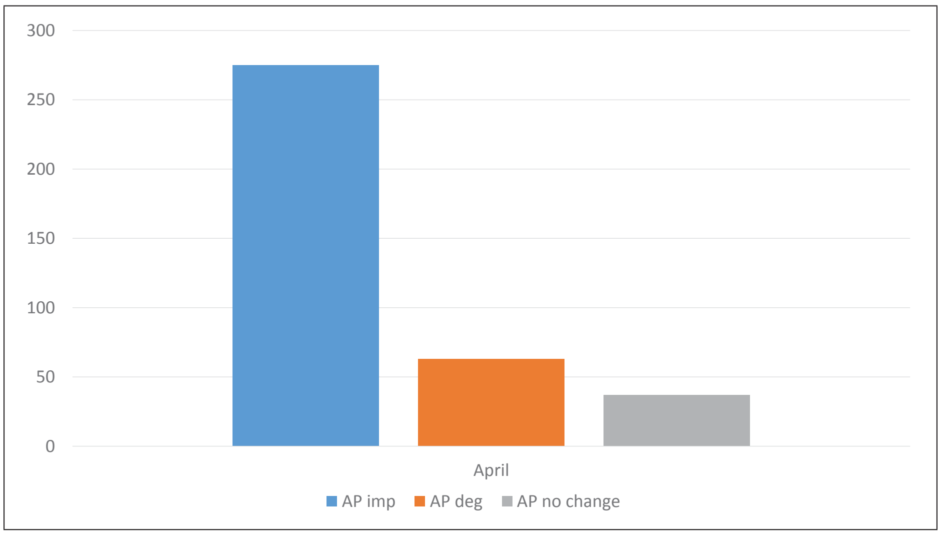

Figure 5.5 Number of successful AP-to-AP steering which QoE of client drop or improved or maintained after steering. imp stands for improvement, deg stands for degradation and no change for maintaining.

5.3.2 Band-to-Band Steering Analysis

We divided our Band-to-Band steering into two sections to study the effect of steering from 5GHz to 2.4GHz and vice versa. When a client is connected to 2.4GHz before steering, the same client will connect to 5GHz after Band-to-Band steering. We assigned this steering to "2.4GHz-to-5GHz". In some cases Client was connected to 2.4GHz, after steering by "5G Preferences" the band connectivity is still 2.4GHz. We observed the same situation for 5GHz too, which mean the band before and after steering are the same, so we eliminate this cases which can be happened due to some fault in logging data and could bias our analysis. We repeat the same categorization as AP-to-AP for improvement and degradation in Band-to-Band steering.

Our analysis shows the distribution of band connectivity after Band-to-Band steering that 77% of the time, on average clients connected to 5GHz after steering. 23% of the time clients connected to 2.4GHz. 5GHz preference leads clients for 77% connectivity to 5GHz.

Number of steering from 5GHz to 2.4GHz compared to 2.4GHz to 5GHz is smaller, and that is basically because of 5GHz preferences which are somehow forcing the users to connect to the 5GHz band to have a better quality of service. We observed that almost all the steering from 5G to 2.4GHz had improvement, and this can be a prove for the fact that, 5GHz preferences is not always improving the performance. From figure 7, we can see the majority of clients steered from 2.4GHz to 5GHz based on ’5G Preferences’ faced RSSI degradation.

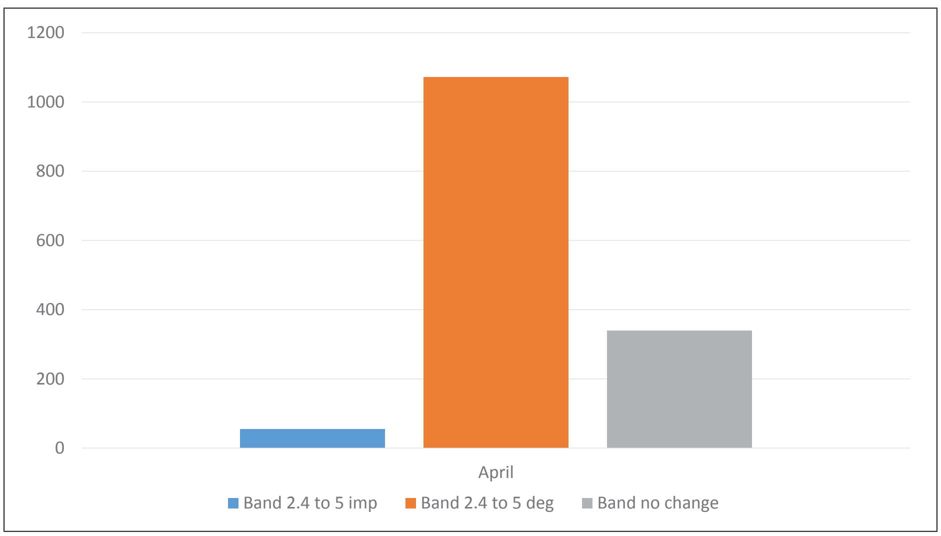

Figure 5.6 Number of successful Band-to-Band steering which QoS of client drop or improved or did not change after steering. imp stands for improvement, deg stands for degradation and no change for maintaining.

5.3.3 Throughput Analysis

As we find out the degradation of signal strength in 5G preference is a natural rule, we tried to address the degradation in other metrics such as download and upload throughput, transmit and receive bit rate. To this aim, we considered the value of each metric 1 minutes before and after each band change. We take the difference of before and after value, then count the number of degradation and improvement after each steering.

Table 5.1 2.4GHz to 5GHz Throughput Analysis within one month (March). Number of improvement and degradation after each steering.

Month	No change	Improvement	Degradation
Dlthroughput	552	2627	973
Ulthroughput	585	2772	878
Txbitrate	4	4636	191
Rxbitrate	3	4739	91

Table 5.2 5GHz to 2.4GHz Throughput Analysis within one month (March). Number of improvement and degradation after each steering.

Month	No change	Improvement	Degradation
Dlthroughput	719	1403	1576
Ulthroughput	814	1172	2400
Txbitrate	3	314	4513
Rxbitrate	10	191	4615

It's obvious that the value of bit rate from 2.4G to 5G will increase vice verse but its important to see what happened to throughput. Results of comparison can be found in table 5.1 and 5.2.

5.4 Parameter Selection and model Evaluation

To fine-tune our model and obtain the best parameters for support vector regression and to evaluate the model, we calculated the root mean square error for the test data in three groups. for each group of parameters (C, σ, ϵ). We considered groups as follow, $C(1, 2, 5)$, $\sigma(0.01, 0.001, 0.000001)$ and $\epsilon(0.1, 0.03, 0.05)$. At the end, the group which could be minimized the average root mean square error was chosen as the parameter for our SVR model. (Table 5.3 shows the effect of different parameters in our model.) To do so the following objective function solved:

$$min_{C,\epsilon,\sigma}(\sum_{i=1}^{n} RMSE_i) \tag{5.1}$$

where n is the number of sections, $RMSE$ is the root mean square error calculated from the actual value y_j and the predicted value $\hat{y}_j$ is given by:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{N}(y_j - \hat{y}_j)^2} \tag{5.2}$$

Table 5.3 Parameter table

ϵ	C	σ	RMSE Rate
0.1	2	0.000001	0.11
0.05	5	0.000001	0.06
0.05	1	0.001	0.05
0.03	1	0.01	0.04

5.4.1 Experiment and Analysis

5.4.1.1 Model Evaluation

To evaluate the capabilities of the prediction models, we carried out a set of experiments all of these experiments performed on a real-life data-set from ETS smart residence. Our model could fit the data with an accuracy score of 88.91% using r2-score. We will do another experiment with cross-validation to furthermore evaluate performance of our model.

We used 70% of data for our training or basically to fit the parameters and 30% percent for the data to measure the performance of the model. To avoid over-fitting, we did use K-Folds cross-validation. With k value of 10, this method will divide the entire data randomly in 10 fold and each time algorithm will fit the model on the $K-1$ (K minus 1) folds and validate on the $k_t h$ fold, then repeat this process until every K-fold being as test set. Figure 5.8 shows the results of our cross-validation accuracy over different folds. At the end we could reach the accuracy of 87% in overall.

Table 5.4 Predictor Performance

Measure	Percentage Accuracy
r2-Score	88.91

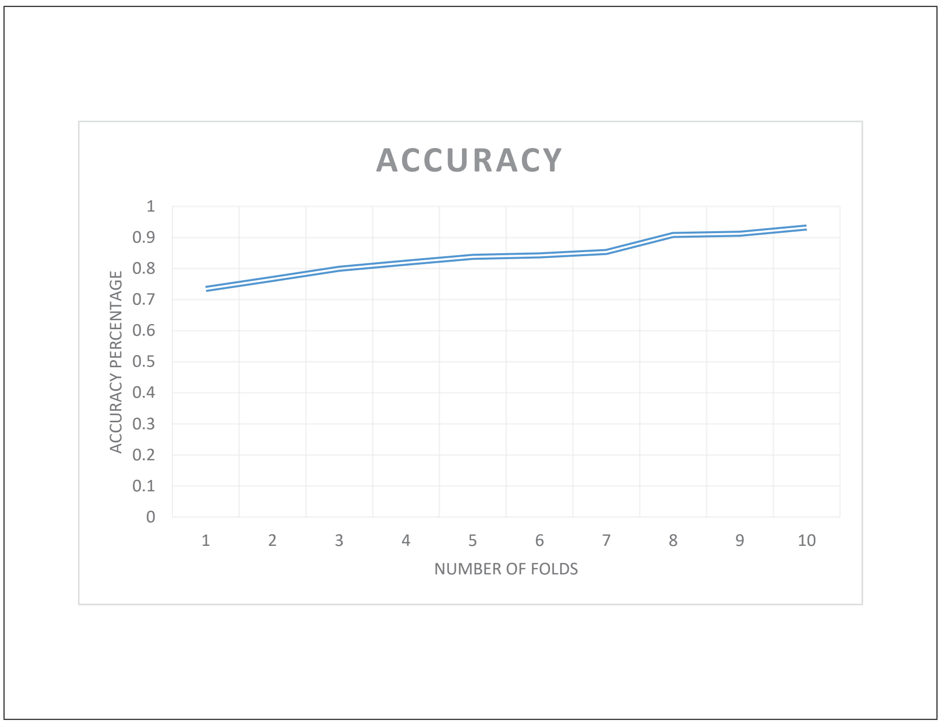

Figure 5.7 Experimental Results using the dataset and 10-fold Cross validation.

We observed, if we apply our policy on the system, we can improve the average of the throughput significantly. Unnecessary handovers with low score shouldn't take place in the system. For our baseline, we will use a real-world case study, which is XcellAir system, a Wi-Fi service provider for our use-case and we will compare the average throughput of the XcellAir system with our simulated average throughput to see the effect of the proposed method on the overall system. Figure 5.9 shows the average throughput of the baseline and the proposed method. The proposed method can increase the average of throughput by 25.13% in download throughput and upload throughput by 26.05% percent accordingly.

Quality of experience for user in our problem is whether to download or upload using their devices. Therefore we chose throughput as our evaluation metrics of proposed method. Such a way we simulate numbers of access points then apply the proposed method to find the performance of our method. To this aim we defined 60 agent to act as our access points. handover parameters will pass through these agents, each agent is only allowed to activate the handover mechanism when throughput of user will be maintaining or improving. This means every handover mechanism that can cause degradation in throughput is not allowed to trigger by the agent. Our predicted handover score is between 0 to 1. Agent will predicted handover score then filter it based on the defined spans. Handover with score under 0.3 count as degradation and it's not acceptable for our method. This handover should not take place. Agent won't trigger this handover to avoid throughput degradation for users. Scores between 0.3 to 0.6 count as maintaining means handover will take place to maintain the throughput of users. Finally scores from 0.6 to 1 are improvement which will improve the throughput of the users.

We calculated the number of handover and compare it with the number of handover Figure 5.8 that passed by our agents. It's obvious that our method could reduce the number of unnecessary handover by 38%. Here unnecessary handover defined as a handover that cause degradation in QoE of users. Table 5.5 show the number of improvement, Maintaining handover and Degradation. Degradation counts as our unnecessary handover and should not take place by our agent.

Table 5.5 Distribution of handovers over agents.

Handover Category	Number of Handovers
Improvement	931
Maintaining	615
Degradation	973

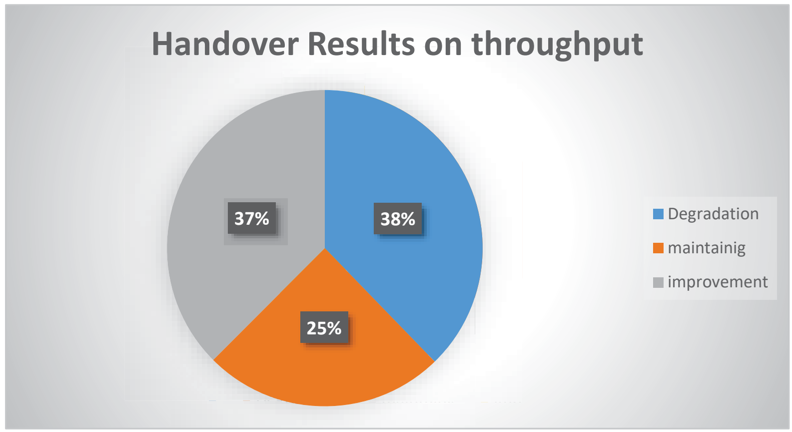

Figure 5.8 Results of handover on number of time that throughput faced improvement, degradation or maitaning.

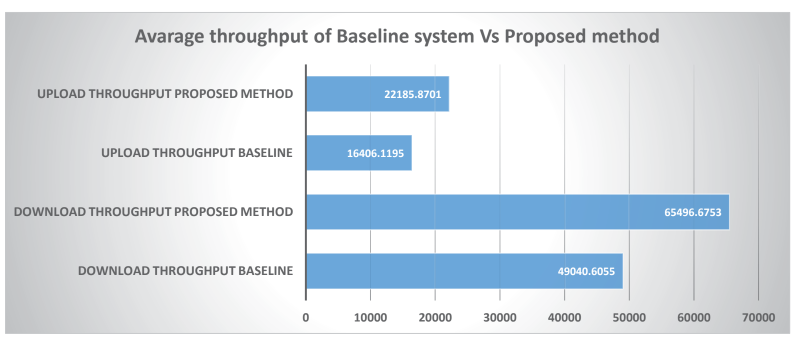

Figure 5.9 Results of handover on number of time that throughput faced improvement, degradation or maitaning.

As for the trade-off the proposed system, It's not reasonable to take to account the energy consumption of the access points. One of the main reason is that access points are often connected to AC power supply and they are independent of any battery. On the other hand, when the throughput of the access points is higher, there is not much change in power consumption of an access point. Therefore we neglect this from our studies.

The proposed method could reduce the number of expected handovers up to 38% percent, such a way our method can reduce the workload of the system by reducing the number of unnecessary handovers.

CONCLUSION AND RECOMMENDATIONS

In this thesis, we presented a novel framework to predict the best handover parameters to maintain the Quality of service of connected users. We also developed a new decision-making algorithm which is taking advantage of our prediction results to guarantee the Quality of experience for users inside Wi-Fi enterprise network.

We proposed a prediction method based on Support Vector Regression. We used handcrafted features such as download throughput, upload throughput, transmit and receive bit rate as our features as well as received signal strength. By using these features, we then performed model selection to fine-tune our prediction model to predict the QoE score that we defined earlier.

While the handover mechanism proven to be reliable on maintaining connectivity of users, QoS degradation and being de-active are still challenging issues. Therefore, we aimed at solving both these issues by introducing a pro-active method in centralized Wi-Fi networks. It was empirically shown that predicting the handover parameters helped to obtain higher throughput. This also helped to the proposed approach to achieve less number of handovers and void ping-pong effect. A comparison with the baseline system revealed that the pro-active approach performs at-par or better. We also showed that the overhead of the system is less with reducing the number of handovers.

6.1 Limitation and Recommendation

The research works presented in this thesis addressed in the initial attempts to solve performance degradation within a handover mechanism in Wi-Fi networks. However, there is still more room for improvement. Below we summarize the potential path to continue this research work.

One of the significant limitations of the work for handover prediction is that we don't have the precise time of the handover event, due to the delay of the controller in logging (collecting) the data and logging it separately from a service provider point of view. We aim to look at the analysis from user point of view to guarantee the quality of experience. Thus if we could have a precise time-series data set, modeling the prediction with deep learning, with the current rate of their popularity, can lead us to achieve a better result.

Our proposed method for handover prediction can be improved in several ways. Defining a new Quality of experience score based on new parameters can give a better understanding of QoE for handover mechanism. It is also reasonable to expand our method beyond the only access point handover mechanism to band handover mechanism as well. Finally, upgrading our algorithm in a way that can find the QoE parameter for handover between two access points can further promote our approach to its best version.

6.2 Summary of Contribution

Below, we briefly highlight the major contribution of this thesis.

- An automatic pipeline proposed to detect the handover event from user data.
- A score function defined which evaluates the handover event according to the user QoE (Quality of Experience)
- A predictive handover approach proposed in which the system can make its decision in more appropriate time. This is achieved using a machine-learning algorithm to predict the appropriate handover score.
- This study performed on a real-world test-bed, in "École de Technologie supérieure" Smart Residence, located in Montreal, Canada.

6.3 Publication in peer reviewed international conferences

- Sadegh Aghabozorgi, Abdolkhalegh Bayati, Kim-khoa Nguyen, Charles Despins, Mohamed Cheriet. "Toward Predictive Handover Mechanism in Software-Defined Enterprise Wi-Fi Networks" In IEEE Sustainable ICT, 2019.

APPENDIX I

APPENDIX EXAMPLE

1. Toward Predictive Handover Mechanism in Software-Defined Enterprise Wi-Fi Networks

Toward Predictive Handover Mechanism in Software-Defined Enterprise Wi-Fi Networks

Sadegh Aghabozorgi, Abdolkhalegh Bayati, Kim-khoa Nguyen, Charles Despins, Mohamed Cheriet
École de Technologie Supérieure, University of Quebec, Canada
Email: {sadeghaghabozorgi@yahoo.com, bayati.khalegh@gmail.com,
Kim-Khoa.Nguyen@etsmtl.ca, charles.despins@etsmtl.ca, mohamed.cheriet@etsmtl.ca}

***Abstract*—In an enterprise Wi-Fi network, Mobile users may be covered by multiple enterprise access points (APs). To optimize resource allocation, a soft handover is require in which the user's device is seamlessly transferred from one AP to another, and this decision made centrally by a Wi-Fi network controller. Unfortunately, state-of-the-art soft handover mechanisms are often designed to optimize resources from the network provider's point of view and do not take into account user's real-time behaviours, which may affect user's Quality of Experience (QoE). In this paper, a new machine learning (ML)-based method presented to find an optimal handover mechanism. This method allows to predict whether the handover that is going to happen will maintain QoE when users are moving inside a building. Our proposed method improves 34% of user throughput compared to state-of-the-art algorithms.**

***Index Terms*—Wi-Fi network, Handover prediction, Wi-Fi Handover, Machine Learning, Prediction**

I. Introduction

INDOOR Wi-Fi networks are established in enterprise buildings and residence area based on the IEEE 802.11 standard. The user is connected to an AP which provides the highest QoS (usually the AP with shortest distance). When the user moves inside the building, the connection is switched from an AP to another AP to ensure the QoS. This process is called the *handover* mechanism. In some cases it is referred as steering. There are two forms of handover mechanism:(i) soft handover in which the source and target APs are in the same network, and (ii) hard handover in which the source AP and destination AP belong to different networks. Handover is actually a recommendation to the client device. The client device can ignore the request or change to another AP based on its own internal algorithm. Many studies have been conducted in the wireless community to improve the soft-handover mechanism. Those studies focused on reducing the transfer time at physical and software layers and decreasing the effects of handover on the QoS [1], selecting the optimal AP [2], etc.

A principal issue in handover mechanism is to find the appropriate moment for triggering the process (i.e., switch the AP). For example, consider the scenario in fig. 1. A user is connected to AP1 and moves toward AP2. As the users moves along the path, the connection QoS drops because the distance to AP1 increases. At some point, the connection must be transferred to AP2. The handover is done by the network, and it is transparent to the user (the user does not need to take any action). The connection information (e.g., signal strength, throughput) is periodically monitored and logged by the network controller. The network controller frequently checks the connection and triggers the handover process when the status of connection meets the handover conditions which is defined based on threshold. In general, the thresholds are constant and manually configured by the network operator. This approach is reactive and does not ensure the real-time QoE. In other words, the handover mechanism should be triggered in advance before the connection encounters QoS degradation. Moreover, as the space inside the building changes (e.g., the decoration, the locations of objects, obstacles, and walls) the threshold values need to be adjusted manually by the operator which is not effective. Another issue is the ping pong effect which happens when the user connection is frequently transferred between APs in a short-period of time.

In this work, an approach for the handover mechanism presented which aims to maintain the connection QoE during the process. Unlike the existing approaches which are based on the fixed threshold values, our approach relies on a machine learning algorithm to find the optimal moment to trigger the handover. The connection QoS is measured before and after the handover process. A continuous handover score is defined which scales from 0 to 1 to evaluate the handover event according to the connection QoS (before and after the transfer). In the proposed approach, the handover score is predicted (before the transfer occurs) using a machine learning algorithm. The connection data (e.g., the signal strength, connection throughput, and the number of previous handover events) are employed as the features in the prediction algorithm. The predicted score determines whether the handover is able to preserve the connection QoS.

Algorithm applied to the real data collected from a Wi-Fi network in a university residence area. There are 70 APs and 360 users in the network. The historical data collected during 1 year has been utilized in our analysis. A database of handover events created, Each handover event

Figure-A I-1 Published paper on IEEE SustainableICT Conference 2019

2. Data-Set Example of Records

Table-A I-1 Description of the AlarmDetails table, collected from centralized system (Controller)

Header	Description	Example of record
AP (Access point)	Identical Mac Address	b8:ec:a3:2f:bd:57
AP ID	Identification Number of AP	8108662df1b8b8eca32fbd57
AP Name	Access point name	AP-4237
Entity	Entity specification of AP	b8:ec:a3:2f:bd:57/zombie
Category	Category of Access point	Zombie
Message	Message of Alarm	Interface 5G OutOfSync (1) MINOR
Createdtime	Time That alarm have been submitted	1555867941072

Table-A I-2 Description of the apDashboardClientsDetails table, collected from centralized system (Controller)

Header	Description	Example of record
Client	Mac Address of User Device	cc:20:e8:24:de:ec
AP	Mac Address of Access point of AP	b8:ec:a3:2f:bd:ff
Band	Band which user is connected to	2.4GHz or 5GHz
Date time	Time that this record logged	2019-04-21 15:30:01
txBitrate	Transmit bit rate value	142
rxBitrate	Receive bit rate value	87
ulthroughput	Upload throughput value	6768
dlthroughput	Download throughput value	4624
channelFreq	Frequency of channel	2462,...,5765
txPower	Transmit Power of AP	20
signalStrength	RSSI (Recieved Signal Strength Indicator) to AP	-45
time-axis	Time Axis that when this record logged	1555875001201
bssid	AP Mac Address (different in Band)	b8:ec:a3:2f:bd:fc
ssid	Identical name of AP in controller	APT_4538

Table-A I-3   Description of the apDashboardRadiosDetails table, collected from centralized system (Controller)

Header	**Description**	**Example of record**
AP	Mac Address of Access point of AP	b8:ec:a3:2f:bd:ff
Band	Band which user is connected to	2.4GHz or 5GHz
NumStations	Number of connected user to AP	1,...,n
Date time	Time that this record logged	2019-04-21 15:30:01
channelUsage	Percentage of channel usage	34
obssChannelUsage	Percentage of overlapped channel usage	31
txBitrate	transmit bit rate value	142
rxBitrate	Receive bit rate value	87
ulthroughput	Upload throughput value	6768
dlthroughput	Download throughput value	4624
channelFreq	Frequency of channel	2462,...,5765
txPower	Transmit Power of AP	20
signalStrength	RSSI (Recieved Signal Strength Indicator) to AP	-45
noise	Noise value	-88

Table-A I-4   Description of the apLog table, collected from centralized system (Controller)

Header	**Description**	**Example of record**
Date Time	Time that this record logged	2019-04-21 00:01:38
Primary MAC address	Mac address of primary AP	04:bf:6d:5e:11:87
Manufacturer	Access point brand	ZyXEL$_{E}MG$2926
Date time	Time that this record logged	2019-04-21 15:30:01
Used % - Flash	Used Percentage of flash (Server)	22
Used % - RAM	Used percentage of RAM (server)	6
Uptime	Value of Uptime	41.31
CPU %	Used Percentage of CPU	17

Table-A I-5   Description of the apPosition table, collected from centralized system (Controller)

Header	**Description**	**Example of record**
Mac Address	Mac Address of APs	b8:ec:a3:2f:be:03
Latitude	Latitude value of APs	45.49311205
Longitude	longitude value of APs	-73.56329481
Altitude	Altitude value of APs	4

Table-A I-6 Description of the area-profile table,
collected from centralized system (Controller)

Header	Description	Example of record
Date Time	Time that this record logged	2019-04-21 00:01:38
id	Id profile of each AP	00000000–0000-00aggressive
Name	Profile name of APs	Default/Balanced/Conservative
rrmConfig	RRM functionality	True/False

Table-A I-7 Description of the channel-change table,
collected from centralized system (Controller)

Header	Description	Example of record
AP	AP Mac Address	04:bf:6d:5e:11:8f
Date Time	Time that this record logged	2019-04-21 00:01:38
Channel	Channel Connectiviy	149/11/6/1/...
new channel	New channel that user is connected through	149/11/6/1/...
Reason for change	Reason that RRM changed the channel for user	OBSS threshold
interference before	interference before channel change	65
interference after	interference after channel change	38
Overall before	Overall interference before channel change	65
Overall after	Overall interference after channel change	38
Noise level before	Noise level before channel change	-97
Noise level after	Noise level after channel change	-97
result	result of channel change	Channel not match/Success
Client Mac	Mac address of User	00:f4:8d:1b:b8:2f

Table-A I-8 Description of the client steering table,
collected from centralized system (Controller)

Header	Description	Example of record
Client MAC Address	User Mac Address	64:a2:f9:3b:5f:dd
Source AP	AP that user is connected before steering	b8:ec:a3:2f:bd:55
Source AP Band	Band of source AP before steering	5GHz/2.4GHz
Target AP	The AP that user is going to connect to	04:bf:6d:5e:12:99
Target AP Band	Band of target AP after steering	5GHz/2.4GHz
Source SS	Signal Strength of Source AP	-81
Target SS	Signal Strength of Target AP	-55
Steering type	Type of steering	LOW Signal Strength
Steering Status	Status about steering (After)	Missing/Not moved/Success
Steering Time	Time axis that steer happened	1555874027
Update Time	Update time of steering status	1555874207
Time Zone	Time zone of logged record	America/New-York

Table-A I-9 Description of the combinedPerformanceReport table, collected from centralized system (Controller)

Header	Description	Example of record
AP	AP Mac Address	b8:ec:a3:2f:bd:ff
Band	Band that user is connected to AP through	2.4GHz/5GHz
Start	Logging record start time	2019-04-21 15:30:01
Stop	Logging recod stop time	2019-04-21 15:30:01
MacAddress	User MAC Address	cc:20:e8:24:de:ec
bssid	AP identical mac address (for different band)	b8:ec:a3:2f:bd:fc
ssid	AP specific name on controller	APT_4538
dlthroughput	Download throughput value (AP side)	152.59527121
ulthroughput	Upload throughput value (AP side)	311.6828929068
SignalStrength	Recieved signal strength indicator	-45
txBitrate	Transmit bit rate	142
rxBitrate	Received bit rate	87

Table-A I-10 Description of the eventDetails table, collected from centralized system (Controller)

Header	Description	Example of record
MAC Adress	AP Mac Address	04:bf:6d:5e:11:8c
Prim. MAC Address	Primary MAC address of APs	04:bf:6d:5e:11:8f
Channel	Channel which AP is operating on	11 (2462 MHz)
Type	Type of Event	MODIFY AP CONFIGURATION
Module (from)	Module that create the event	RRM/SON
Module (to)	Module that would review the event	RRM/SON
Message	Description=This event is triggered ...	APT_4538

Table-A I-11 Description of the hn-groupe table, collected from centralized system (Controller)

Header	Description	Example of record
dateTime	Time that this record logged	2019-04-21 00:45:02
enabled	Feature availability	True/False
id	AP ID which been profiled	f85598b0-b2f1-4cd3
name	Name of AP in the controller	$AP - 4826$
priority	priority indicator	0/1
Topology	Topology that AP is following	'backhaulConnection': , '2.4 GHz'

REFERENCES

Aibinu, A., Onumanyi, A., Adedigba, A., Ipinyomi, M., Folorunso, T. & Salami, M. (2017). Development of hybrid artificial intelligent based handover decision algorithm. *Engineering Science and Technology, an International Journal*, 20(2), 381–390.

Ali, K. B., Zarai, F., Khdhir, R., Obaidat, M. S. & Kamoun, L. (2018). QoS aware predictive radio resource management approach based on MIH protocol. *IEEE Systems journal*, 12(2), 1862–1873.

Bao, X., Adjardjah, W., Okine, A. A., Zhang, W. & Dai, J. (2018). A QoE-maximization-based vertical handover scheme for VLC heterogeneous networks. *EURASIP Journal on Wireless Communications and Networking*, 2018(1), 269.

Çalhan, A. & Çeken, C. (2013). Artificial neural network based vertical handoff algorithm for reducing handoff latency. *Wireless personal communications*, 71(4), 2399–2415.

Cauwenberghs, G. & Poggio, T. (2001). Incremental and decremental support vector machine learning. *Advances in neural information processing systems*, pp. 409–415.

Chen, J., Wang, Y., Li, Y. & Wang, E. (2018). QoE-aware intelligent vertical handoff scheme over heterogeneous wireless access networks. *IEEE Access*, 6, 38285–38293.

Cisco. (2017). Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update. *Papers*, 81, 97–108.

Elhadj, H. B., Elias, J., Chaari, L. & Kamoun, L. (2016). Multi-attribute decision making handover algorithm for wireless body area networks. *Computer Communications*, 81, 97–108.

Farooq, H. & Imran, A. (2017). Spatiotemporal mobility prediction in proactive self-organizing cellular networks. *IEEE Communications Letters*, 21(2), 370–373.

Guo, Y., Bartlett, P. L., Shawe-Taylor, J. & Williamson, R. C. (2002). Covering numbers for support vector machines. *IEEE Transactions on Information Theory*, 48(1), 239–250.

Hasbollah, A. A., Ariffin, S. H. & Ghazali, N. E. (2017). Optimal Forwarding Probability for Vehicular Location Prediction Handover Algorithm. *Asian Simulation Conference*, pp. 369–380.

Javed, F., Chan, G. S., Savkin, A. V., Middleton, P. M., Malouf, P., Steel, E., Mackie, J. & Lovell, N. H. (2009). RBF kernel based support vector regression to estimate the blood volume and heart rate responses during hemodialysis. *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 4352–4355.

Liang, S., Zhang, Y., Fan, B. & Tian, H. (2017). Multi-attribute vertical handover decision-making algorithm in a hybrid VLC-femto system. *IEEE Communications Letters*, 21(7), 1521–1524.

Liu, J. & Zio, E. (2016). An adaptive online learning approach for Support Vector Regression: Online-SVR-FID. *Mechanical Systems and Signal Processing*, 76, 796–809.

López-Raventós, Á., Wilhelmi, F., Barrachina-Muñoz, S. & Bellalta, B. (2018). Machine learning and software defined networks for high-density wlans. *arXiv preprint arXiv:1804.05534*.

Manyika, J., Chui, M., Institute, M. G., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C. & Byers, A. (2011). *Big Data: The Next Frontier for Innovation, Competition, and Productivity*. McKinsey.

Memon, S. & Maheswaran, M. (2019). Using machine learning for handover optimization in vehicular fog computing. *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, pp. 182–190.

Sangkusolwong, W. & Apavatirut, A. (2017). Indoor WIFI Signal Prediction Using Modelized Heatmap Generator Tool. *2017 21st International Computer Science and Engineering Conference (ICSEC)*, pp. 1–5.

Sarma, A., Chakraborty, S. & Nandi, S. (2016). Deciding handover points based on context-aware load balancing in a WiFi-WiMAX heterogeneous network environment. *IEEE Transactions on Vehicular Technology*, 65(1), 348–357.

Shen, Y., Xi, N., Pei, Q., Ma, J. & Niu, Y. (2012). The wireless vertical handoff decision algorithm with parameter prediction. *2012 Fourth International Conference on Intelligent Networking and Collaborative Systems*, pp. 185–192.

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer science & business media.

Yang, B., Wu, Y., Chu, X. & Song, G. (2016). Seamless handover in software-defined satellite networking. *IEEE Communications Letters*, 20(9), 1768–1771.

Zhang, H., Qiu, Y., Chu, X., Long, K. & Leung, V. C. (2017a). Fog radio access networks: Mobility management, interference mitigation, and resource optimization. *IEEE Wireless Communications*, 24(6), 120–127.

Zhang, W., Hua, X., Yu, K., Qiu, W., Chang, X., Wu, B. & Chen, X. (2017b). Radius based domain clustering for WiFi indoor positioning. *Sensor Review*, 37(1), 54–60.

Zubow, A., Zehl, S. & Wolisz, A. (2016). BIGAP—Seamless handover in high performance enterprise IEEE 802.11 networks. *NOMS 2016-2016 IEEE/IFIP Network Operations and Management Symposium*, pp. 445–453.

LIST OF REFERENCES