# MODELLING OF PROPAGATION PATH LOSS USING ADAPTIVE HYBRID ARTIFICIAL NEURAL NETWORK APPROACH FOR OUTDOOR ENVIRONMENTS

*DOCTOR OF PHILOSOPHY*

in

*Electronic Engineering*

by

**VIRGINIA CHIKA EBHOTA**


**HOWARD COLLEGE, UNIVERSITY OF KWAZULU-NATAL,**
**DURBAN – 4041, SOUTH AFRICA**

**STUDENT NO.: 216075533**

**AUGUST 2018**

# Modelling of Propagation Path Loss using Adaptive Hybrid Artificial Neural Network Approach for Outdoor Environments

*This thesis submitted in fulfilment of the requirements*
*for the degree of Doctor of Philosophy: Electronic Engineering*
*in the*
*Howard College of Agriculture, Engineering & Science*
*University of KwaZulu-Natal, Durban - 4041*
*South Africa.*


*Student:*

Ms. Virginia Chika Ebhota

*Supervisor:*

Prof. (Dr.) Viranjay M. Srivastava

*Co-Supervisor:*

Dr. Joseph Isabona


**2018**

As the candidate's supervisors, We have approved this thesis for submission.

Prof. (Dr.) Viranjay M. Srivastava_____Date _____

Dr. J. Isabona _____ Date _____

## Declaration 1 - Plagiarism

I, **CHIKA VIRGINIA EBHOTA** with Student Number **216075533** with the thesis entitled ***MODELLING OF PROPAGATION PATH LOSS USING ADAPTIVE ARTIFICIAL NEURAL NETWORK APPROACH FOR OUTDOOR ENVIRONMENTS*** hereby declare that:

1. The research reported in this thesis, except where otherwise indicated, is my original research.
2. This thesis has not been submitted for any degree or examination at any other university.
3. This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

a. Their words have been re-written, but the general information attributed to them has been referenced.

b. Where their exact words have been used, then their writing has been placed in italics and inside quotation marks and referenced.

5. This thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the References sections.

Virginia Chika Ebhota_____ Date _____

## Declaration 2 -Publications

The under listed journal and conference articles which is either published, accepted or under review are gotten from the research work and constitutes the topics of the research work discussed in various chapters of this thesis.

## Journal Articles:

1. **Virginia C. Ebhota**, Joseph Isabona, and Viranjay M. Srivastava, "Investigation and comparison of the performance of multi-layer perceptron and radial basis function artificial neural networks for signal power loss prediction," ***Int. J. on Communications Antenna and Propagation*** (IRECAP 15330), Accepted, Aug. 2018. ***[SCOPUS]***

2. **Virginia C. Ebhota**, Joseph Isabona and Viranjay M. Srivastava, "Effect of learning rate on GRNN and MLP for the prediction of signal power loss in microcell sub-urban environment," ***Int. J. on Communications Antenna and Propagation*** (IRECAP 15329), accepted, Aug. 2018. ***[SCOPUS]***

3. **Virginia C. Ebhota**, Joseph Isabona, and Viranjay M. Srivastava, "Investigating signal power loss prediction in a metropolitan island using ADALINE and multi-layer perceptron back propagation networks," ***International Journal of Applied Engineering Research*** (IJAER 65608), accepted, 9 July 2018. ***[SCOPUS]***

4. **Virginia C. Ebhota**, Joseph Isabona and Viranjay M. Srivastava, "Base line knowledge on propagation modelling and prediction techniques in wireless communication networks," ***J. of Engineering and Applied Sciences*** *(JEAS),* vol. 13, no. 7, pp. 1919-1934, 2018. ***[SCOPUS]***

5. **Virginia C. Ebhota**, Joseph Isabona and Viranjay M. Srivastava, "Modelling, simulation and analysis of signal pathloss for 4G cellular network planning," ***J. of Engineering and Applied Sciences*** *(JEAS),* vol. 13, no. 7, pp. 1907-1918, 2018. ***[SCOPUS]***

6. **Virginia C. Ebhota**, Joseph Isabona, and Viranjay M. Srivastava, "Improved adaptive signal power loss prediction using combined vector statistics based smoothing and neural network approach," ***Int. J. on Progress in Electromagnetics Research C*** *(PIER C*), vol. 82, pp. 155-169, March 2018. ***[SCOPUS]***

7. **Virginia C. Ebhota**, Joseph Isabona and Viranjay M. Srivastava, "Bayesian regularization in multi-layer perceptron artificial neural network model to predict signal power loss using measurement points," ***Int. J. on Communications Antenna and Propagation*** *(IRECAP_15054)*, under review, April 2018. ***[SCOPUS]***

8. **Virginia C. Ebhota**, Joseph Isabona, and Viranjay M. Srivastava, "Environment-adaptation based hybrid neural network predictor for signal propagation loss prediction in cluttered and open urban microcells," ***Wireless Personal Communication****, (WIRE-D-17-00917,* under review, June 2017. ***[SCOPUS]***

**Conference Article:**

9. **Virginia C. Ebhota**, Joseph Isabona and Viranjay M. Srivastava, "Signal power loss prediction based on artificial neural networks in microcell environment," ***3rd IEEE Int. Conf. on Electro-Technology for National Development*** (*IEEE NIGERCON)*, Owerri, Nigeria, 7-10 Nov. 2017, pp. 250-257. ***[IEEE Xplore]***

## DEDICATION

To Almighty God – who made this research work possible.

## ACKNOWLEDGEMENT

To my dear husband, my mentor and my teacher, Dr. Williams Ebhota, I owe it all to you. Many thanks.

To my supervisor Prof (Dr.) Viranjay M. Srivastava: He diligently and painstakingly followed the research progress at every phase, I am indebted to you, sir. My sincere gratitude goes to my co-supervisor, Dr. Joseph Isabona, for his teaching, advice and invaluable contributions to this research work. To Prof. Ageh and Prof. Afullo, thank you for your unfailing support and advice.

To Dr. P. Chukwu, National Board for Technology Incubation, (NBTI), Southeast zonal manager, Dr. B. Ilechukwu and all my esteemed colleagues at the Nnewi Technology Incubation Centre, I am sincerely grateful for all the supports. To Dr. Mohammed Jibrin, the Director General, NBTI, thank you for the opportunity given to me to enrol for this Ph.D. study. To Mr. AKilu, Engr. Amonye and Dr. Ogbobe; because you believed in me, you made this Ph.D. study a dream come true, I will ever remain grateful.

I am sincerely grateful to my lovely parents and siblings, my mother-in-law, sisters-in-law, brothers-in-law, for their prayers, encouragements and supports. To Pius Urama and family, Dr. Richard Urama and family, Oyiga and family, Michael and family, Okoromi Ehighamhan and family, Okeke and family (Mama yard), Auntie Nkoli, Jude Eigbiremhon and family, Dr. Samuel and family, Christian Iduma and family, Makinde and family, Blessing Ogbe and family, Engr. Sam and family, Godstime and family, Felix and family, Chidiebere Chukwumere and family, Dr. Eloka-Eboka and family, Dr. Andrew and family, Tunji and family, Engr. Matthew and family, Engr. Ademola and

family, Amos, Nnalue, and numerous other friends who have supported me along the way, I say thank you.

With special mention to my UKZN friends, Chidi, Grace, Yuwa, Emmanuel, Efe, Sunday, Kenny, Ebenezer, Tunde, Shegu, Kolawale and numerous others, it was fantastic having you guys around me to work with. Thank you for all the encouragements.

# ABSTRACT

Prediction of signal power loss between transmitter and receiver with minimal error is an important issue in telecommunication network planning and optimization process. Some of the basic available conventional models in literature for signal power loss prediction includes the Free space, Lee, COST 234 Hata, Hata, Walficsh- Bertoni, Walficsh-Ikegami, dominant path and ITU models. But, due to poor prediction accuracy and lack of computational efficiency of these traditional models with propagated signal data in different cellular network environments, many researchers have shifted their focus to the domain of Artificial Neural Networks (ANNs) models. Different neural network architectures and models exist in literature, but the most popular one among them is the Multi-Layer Perceptron (MLP) ANN which can be attributed to its superb architecture and comparably clear algorithm. Though standard MLP networks have been employed to model and predict different signal data, they suffer due to the following fundamental drawbacks. Firstly, conventional MLP networks perform poorly in handling noisy data. Also, MLP networks lack capabilities in dealing with incoherence datasets which contracts with smoothness.

Firstly, in this work, an adaptive neural network predictor which combines MLP and Adaptive Linear Element (ADALINE) is developed for enhanced signal power prediction. This is followed with a resourceful predictive model, built on MLP network with vector order statistic filter based pre-processing technique for improved prediction of measured signal power loss in different micro-cellular urban environments. The prediction accuracy of the proposed hybrid adaptive neural network predictor has been tested and evaluated using experimental field strength data acquired from Long Term Evolution (LTE) radio network environment with mixed residential, commercial and cluttered building structures. By means of first order statistical performance evaluation metrics using Correlation Coefficient, Root Mean Squared Error, Standard Deviation and Mean Absolute Error, the proposed adaptive hybrid approach provides a better prediction accuracy compared to the conventional MLP ANN prediction approach. The superior performance of the hybrid neural predictor can be attributed to its capability to learn, adaptively respond and predict the fluctuating patterns of the reference propagation loss data during training.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| ADALINE | Adaptive Linear Element |
| AMPS | Advance Mobile Phone Services |
| ANN | Artificial Neural Network |
| AT&T | American Telephone & Telegraph |
| BP | Back propagation |
| BR | Bayesian Regularization |
| BS | Base Station |
| CAPEX | CApital EXpenditure |
| CCIR | Comité Consultatif International des Radiocommunications |
| CDMA | Code Division Multiple Access |
| CEPT | Conference of Postal and Telecommunications Administrations |
| ECC | Electronic Communication Committee |
| EUROCOSTR | EUROpean Co-Operative for Scientific and Technical Research |
| FSPL | Free-Space Path Loss |
| GPS | Global Positioning System |
| GRNN | Generalized Regression Neural Network |
| GTD | Geometric Theory of Diffraction |
| IMT | International Mobile Telecommunication |
| ITU | International Telecommunication Union |
| LM | Levenberg-Marquardt |
| LOS | Line of Sight |
| LTE | Long Term Evolution |
| MATLAB | Matrix Laboratory |
| MAE | Mean Absolute Error |

| | |
|---|---|
| MMSE | Minimum Mean Square Error |
| ML | Machine Learning |
| MLP | Multi-Layer Perceptron |
| MS | Mobile Station |
| NLOS | Non-Line of Sight |
| NN | Neural Network |
| NMT | Normadic Mobile Telephone |
| PL | Path Loss |
| QoS | Quality of Service |
| RBF | Radial Basis Function |
| r | Correlation Coefficient |
| RMSE | Root Mean Square Error |
| RF | Radio Frequency |
| SD | Standard Deviation |
| SGD | Stochastic Gradient Descent |
| SUI | Standard University Interim |
| TEMS | Test Mobile System |
| UE | User Equipment |
| UHF | Ultra High Frequency |
| VHF | Very High Frequency |
| VGD | Vanilla Plain Gradient Descent |
| V*L*F-MLP | Vector *L* Filter Multi-Layer Perceptron |
| VMF-MLP | Vector Median Filter Multi-Layer Perceptron |
| WI | Walfisch-Ikegami |
| 1G | First Generation |
| 2G | Second Generation |

| | |
|---|---|
| 3G | Third Generation |
| 4G | Fourth Generation |
| 5G | Fifth Generation |

# CHAPTER 1

# GENERAL INTRODUCTION

## 1.1. Introduction

Propagation Path Loss (PL) prediction has been a crucial task in the designing and planning of mobile communication networks. This is as a result of various physical mechanisms such as reflection, diffraction and scattering of the signal as they propagates, as well as multipath phenomenon and the constant movement of users of mobile communication [1]. Therefore, predicting electromagnetic signal power loss with minimal error during signal propagation from the transmitter to the receiver is an important issue in telecommunication network planning and optimization process. When a new mobile technology such as Long Term Evolution (LTE) network is to be deployed with a goal to increase network capacity and speed, with wider coverage and strong Quality of Service (QoS), it is important to identify these factors which affect the quality and capacity of the mobile communication network.

A major objective of planning radio coverage is for an efficient use of the allocated frequency. From this perspective, network coverage is an influential factor in network planning as it allows the network engineers to carry out different configuration tests before physical implementation of the changes. However, propagation PL prediction for radio coverage is a complex task, hence, the need for an accurate and computational efficient prediction tool. Accurate and reliable models are vital for the prediction of radio channel parameters in the area for deployment of cellular mobile radio system. Traditional PL models such as empirical, deterministic and semi-empirical models have been used over the years.

Path Loss models are generally the empirical mathematical formulation of the signal propagation behaviour of an environment [2, 3]. Statistically, empirical models basically explain the relationship between the environment and Path Loss. They do not require much computational effort and are easy to implement, however, they are less sensitive to the physical and geometrical configuration of the environment. The deterministic models on the

other hand, calculate the field strength using the Geometrical Theory of Diffraction (GTD) and it is acquired as the super-position of direct, reflected and diffracted rays at the area of interest. Deterministic models are very sensitive to the environmental physical configuration, making them accurate, however, they require detailed information on the environment under coverage in comparison to empirical models and thus much computational efforts are required. Considering that propagation signal encounter different atmospheric conditions, obstacles and multi-path phenomenon, it is difficult to make use of a particular model in the formulation of the precise loss in communication network [4].

Therefore, for a signal propagation model that accurately reveals the characteristics of the communication network environment with less computational efforts using real world measured data from LTE network, our research work has developed an adaptive Artificial Neural Network (ANN) prediction modelling techniques. This caters for the stochastic signal attenuation phenomenon and the in-homogeneity of the spatial propagation channels in outdoor environments. This approach for propagation PL prediction in typical outdoor environments has been realized by developing a hybrid ANN model that combines an existing ANN model and an adaptive neural modelling capability.

## 1.2. Problem Statement

*Need for an accurate and computational efficient model with minimal prediction error using an adaptive artificial neural network to enhance the prediction of propagation Path Loss during electromagnetic signal transmission from transmitter to receiver.*

A well-designed cellular network structure is essential for effective roll out of cellular mobile communication system. This is because a physical layer has to support the cellular design for every type of radio access technology [5]. The choice of the prediction tool used for modelling propagation PL directly relates to the prediction accuracy achieved [6]. Signal propagation study is majorly concerned with what happens in-between transmitter and receiver. Transmitter produces an electromagnetic signal that is modulated onto the carrier frequency and the signal reacts with several obstacles on its way to receiver where it is then induced on the receiver's antenna for demodulation.

The environmental obstacles cause signal to be reflected, diffracted or scattered which in turn attenuates the signal power through absorption. The three basic propagation mechanisms: diffraction, reflection and scattering are essential key for the analysis of radio propagation modelling-based study. Therefore, signal propagation model which supports configurable parameters are ideal as they permit adaptation of the model to different environments thereby improving the calculated coverage prediction accuracy [7, 8].

When considering propagation in outdoor environments, our interest is in three types of areas: rural, sub-urban and urban areas. The terrain profile of any area must be considered as there may be variation from a simple curved earth to high mountain regions. Presence of trees, buildings, moving cars and other environmental obstacles must be considered. Interference conditions often prevent the best use of the cellular system because simple mathematical models do not correspond with reality. Increasing power to conceal dead spots creates other problems [9, 10].

## 1.3. Background to the Research Work

There is need for an appropriate radio system design, determination of Base Station (BS) arrangements and appropriate operating frequency during an upgrade or deployment of new communication network such as LTE to ensure enhanced QoS. However, understanding and building these networks centre on the knowledge of signal propagation over a distance in pragmatic environments [11].

Propagation PL is one of the major importance characteristics of communication networks. It is the local average received signal power relative to transmit power [12]. This fall in electromagnetic signal power density as it transmits in the space may be as a result of diffraction, reflection, scattering, etc. and it is subjective to varied environments [13]. Factors such as topography, town planning, population density, rainfall, vegetation, etc. contribute to PL. Transmitter and receiver antenna height variations also create losses [5]. As a result of these occurrences, the signal strength received, experiences both attenuation and fluctuation. Accurate estimation of PL offers a good basis for proper BS location and adequate determination of frequency plan. Also, there is need by network engineers on appropriate method for mapping the extent of coverage of both existing and planned networks [14, 15].

The effectiveness of different estimation methods of PL solemnly depends on the predictive power of the underlying Path Loss model.

A propagation model is a set of mathematical expressions, diagrams and algorithms that is used to express the radio characteristics of a given environment [16]. Propagation model can either be empirical, also known as statistical model or theoretical also known as deterministic model or a combination of these two models. The empirical models are established on measurements while the theoretical models are based on the fundamental principles of radio wave propagation phenomena. All environmental influences are wholly considered in empirical models; thus, the accuracy of these models is dependent not only on the accurateness of the measurement but also the similarities between the investigated environment and the environment where the measurements were carried out. The computational efficiency of these models is usually satisfactory. The deterministic models on the other hand, are based on the principles of radio wave propagation phenomena and therefore can be applied in any environment without effect on the accurateness. Practically, enormous database of the environmental characteristics is usually required, which most time is impractical. Also, the algorithms employed using deterministic models are quite complex which is not computational efficient [17].

Recent studies have shown the successful application of ANN models in propagation PL prediction in rural, sub-urban and urban areas [18-20]. They make use of the gains of deterministic and empirical models as they are robust, competent and proficient in showing changes in wireless signal performance, thereby overcoming the inaccuracies in propagation PL modelling using traditional methods. The ANN models can adapt to diverse environments as a result of their flexibility and are capable of processing large size of data due to their high processing speed [21]. It can also be train to act well in environments similar to where the training data were collected. However, there are different techniques used in creating and training ANNs to ensure a network that generalizes well without over-fitting and with minimal error prediction. The ANN architecture, the type of training sets, the type of training algorithms, transfer functions applied, etc. play important roles in the predictive ability of ANN models.

## 1.4. Aim and Objectives of the Research Work

The aim of the research work is to develop an adaptive hybrid ANN models for outdoor propagation PL prediction in LTE communication network for enhanced QoS.

The objectives are:

i. To review baseline knowledge on traditional propagation modelling and prediction techniques in wireless communication networks.
ii. To simulate and analyse propagation PL for LTE cellular network planning using traditional propagation PL models.
iii. To analyse the performance of different backpropagation ANN training algorithms on a known feedforward backpropagation Multi-Layer Perceptron (MLP) ANN for outdoor propagation PL prediction using real world data from LTE communication network.
iv. To compare the generalization abilities of Radial Basis Function (RBF) and MLP ANN models for outdoor propagation PL prediction using different training approaches: variation of the number of neurons in hidden layer, early stopping and Bayesian Regularization approaches.
v. To evaluate the effect of learning rate parameter on two ANN models: Generalized Regression Neural Network (GRNN) and MLP ANN models for outdoor propagation PL prediction.
vi. To evaluate the performance of two ANN models: Adaptive Linear Neural Element (ADALINE) and MLP ANN, examining the impact of variation of learning rate parameter and the effect of input delay in ADALINE.
vii. To optimize the performance of ANN by developing an adaptive hybrid ANN that combines ADALINE system predictor and MLP model for enhanced prediction accuracy with minimal prediction error.
viii. To develop a model that combines vector order statistics-based filter smoothing technique and MLP ANN model for enhanced prediction accuracy by measured signal (data set) smoothening prior to the ANN training.

## 1.5. Significance of the Research Work

The significances of the research work are as follow:

i. Developing a model for an outdoor prediction of propagation PL using adaptive hybrid ANN techniques with an enhanced prediction performance that is more accurate and computationally efficient than the traditional PL models to ensure reduction in losses during electromagnetic signal transmission from a transmitter to a receiver.
ii. Prediction of propagation PL with real world measured data from LTE cellular network, employing the developed adaptive hybrid ANN models and comparison of their predictive performances with conventional MLP ANN models.
iii. Optimization of prediction accuracy by ensuring decrease in the error margin between ANN actual output and the desired output through adequate training of the adaptive hybrid ANN models using proper training parameters and training techniques.

## 1.6. Scope of the Research Work

The scopes of the research work are:

i. Extensive review of baseline knowledge on traditional propagation PL prediction models, simulation and analysis of propagation PL using traditional models with real world measured data from LTE cellular network.
ii. Simulation and analysis of propagation PL with conventional ANN models, adopting different backpropagation training algorithms and different training techniques and comparing their predictive abilities using real world measured data from LTE cellular network.
iii. Developing adaptive hybrid ANN models for enhanced prediction performances with less error margin between ANN actual output and desired output in comparison with the conventional ANN models.

## 1.7. Thesis layout

This research work is structured and organized in seven chapters. Chapter 1 deals with the introductory part of the research work. It presents the problem statement and discusses the background of the research work. It also presents objectives, significances and scope of the research work.

**Chapter 2** reviews the baseline knowledge on traditional propagation modelling techniques in wireless communication networks. Different empirical and deterministic propagation modelling techniques have been reviewed. Their characteristics, potencies and limitations are highlighted. Several key propagation models reviewed are the Hata model, Standard University Interim (SUI) model, Walfisch-Ikegami model, Walficsh-Bertoni model, Lee model, International Telecommunication Union (ITU) model, etc. The concepts of ANNs, the evolution of ANNs, their capabilities, knowledge representation in ANNs and types of ANN learning have been highlighted.

**Chapter 3** employs some of the traditional empirical and deterministic models for simulation and analysis of propagation PL using real world measured data for LTE cellular network planning. The effect of different parameter such as link distance between transmitter and receiver, transmitter antenna height, receiver antenna height, operating frequency on PL have been analysed.

**Chapter 4** analyses the performance of various backpropagation ANN training algorithms on a feedforward backpropagation MLP ANN for outdoor propagation PL prediction. Real world measured data from LTE cellular network has been used in training the ANN. The performances of *ten* ANN training functions of *three* different training algorithms: The Gradient descent-based algorithms, the Conjugate gradient-based algorithms and the Quasi-Newton based algorithms have been examined. The error prediction accuracy of the ANN training functions has been measured by means of four statistical performance metrics: The Root Mean Squared Error (RMSE), the Mean Absolute Error (MAE), the Standard Deviation (SD) and the Correlation Coefficient (r).

Part I of **chapter 5** presents a comparison of the generalization capabilities of RBF and MLP ANN models for outdoor prediction of propagation PL by applying different ANN training approaches: variation of the number of neurons in hidden layer, early stopping and Bayesian Regularization approaches with real world measured data from LTE cellular network. This chapter also includes the concepts of ANN generalization ability and Bayesian Regularization backpropagation training algorithm. The RMSE, MAE, SD and r are used to measure error prediction.

Part II of chapter 5 presents evaluation of two ANN models: ADALINE and MLP ANNs, examining the effect of variation of learning rate parameter, momentum parameter and different combination of transfer functions on the two ANNs. Also, the effect of input delay parameter has been examined in ADALINE. The chapter also discusses the concept of backpropagation technique used for ANN training. Real world measured data from LTE cellular network has been used to train the ANN models and RMSE, MAE, SD and r statistical performance metrics are used for ANN output error prediction.

Part III of chapter 5 deals with the evaluation of the effect of learning rate parameter on two different ANN models: GRNN and MLP ANN models for outdoor prediction of propagation PL employing real world measured data from LTE cellular network. Artificial neural network learning, learning archetype and learning rate have been discussed. The prediction error has been assessed by means of statistical performance metrics: the RMSE, MAE, SD and r.

Part I of **chapter 6** develops an adaptive hybrid ANN that combines ADALINE system predictor and MLP model for an optimized performance of ANN. This enhances prediction accuracy with minimal prediction error. Additional resourceful predictive model built on MLP network with vector order statistic filter based pre-processing technique for improved outdoor prediction of propagation PL has been developed and presents in part II of this chapter. Training of the ANN has been carried out using real world measured data from LTE cellular network and the statistical performance metrics: RMSE, MAE, SD and r has been used to measure the error between actual output and desired output.

**Chapter 7** concludes the thesis research work and highlight on future research work.

## 1.8. Contribution to Knowledge

Although many propagation prediction models are obtainable in the literature (like Free space, Okumura, SUI, Lee, Walficsh-Bertoni, Walficsh-Ikegami models, etc.), they are limited in one way or the other. For example, it is reported in [22, 23], that there is built-in-error in the propagation models applied for macro-cell mobile systems (the standard deviation is as large as *7 dB-10 dB*, which in signal power is a factor of *ten*). One basic reason of the large built-in-error limitation of the existing models is due to dissimilar assumptions and different radio propagation environmental scenarios with which many of the models were developed. Any reduction in the above-mentioned quantity of error will positively impact PL prediction accuracy and the general cellular network coverage performance. Many of the models need building geometry or terrain geometry, and this makes their implementation cumbersome during network planning.

One robust technique that addresses the limitations of the existing models is our developed adaptive hybrid ANN propagation PL model that caters for stochastic signal attenuation phenomenon and the heterogeneity of the spatial propagation channels in different outdoor environments. The developed adaptive hybrid ANN model of ADALINE system predictor built on MLP model enhances the prediction performance of the ANN with minimal prediction error. Furthermore, a novel predictive model of vector order statistic filter based pre-processing technique built on MLP network has been developed for improved outdoor prediction of propagation PL. This technique employs data set pre-processing by means of filtering noisy signal (data set) before the ANN training. There is considerable reduction in performance error in comparison to training conventional ANNs without data smoothening or training existing traditional propagation PL models.

## 1.9. Chapter Summary

This thesis research work adopts different ANN training parameters and techniques for optimized prediction accuracy by ensuring a very minimal error margin between actual output and desired output. All ANN training has been carried out using real world measured data collected from different BS from LTE cellular network and statistical performance metrics assimilated in MATLAB program have been used as a measure for error prediction. The

performance metrics are RMSE, MAE, SD and r and have been utilised to measure error prediction accuracy between actual output and desired output.

# CHAPTER 2

# LITERATURE REVIEW

The chapter's objective is to review the baseline knowledge of propagation Path Loss (PL), its models and modelling techniques. The traditional empirical and deterministic models as well as modelling of propagation PL using the concept of Artificial Neural Networks (ANNs) have been highlighted. The chapter also reviews previous works on propagation PL.

## 2.1. Traditional Radio Propagation Models- An overview

Propagation models are set of algorithms, mathematical equations and diagrams that are used to convey the radio properties of a given environment, which are essential in carrying out interference investigation in the course of deployment [16]. In general, a relationship exists between a propagation model and the type of environment most suitable for the application of such model. The traditional propagation model may be empirical, deterministic, or a combination of the two.

Table 2.1. Basic Path loss models [4, 11]

| Model | Characteristics |
|---|---|
| Deterministic Model | It is site specific and includes massive number of geometric data of the site. It needs high computational effort, is highly complicated but has high accuracy. |
| Empirical Model | It is based on measurement data, uses statistical properties, well adapted to environment of any size, computational efficient and simple. However, it has low accuracy. |
| Semi-Empirical Model | It is established on combination of both empirical and deterministic models. |

## 2.2. Deterministic Models

These are analytical models derived from electromagnetic propagation idealized theory that have been widely applied in network simulators due to their usefulness in the computation of complex tasks with minimum losses [24]. Every propagation situation depends on random surveillance which is described in a pre-defined method by these models. This gives rise to a comprehensive Path Loss prediction that has nearly all propagation

phenomena. The environmental properties such as obstacle positions or their materials have to be accurate for prediction to be accurate. Some of the deterministic models are:

### *2.2.1. Free Space Model*

Electromagnetic wave signal strength loss due to Line of Sight (LOS) path through free space is termed as Free-Space Path Loss (FSPL). Loss by two isotropic radiators in free space is represented by a power ratio. It is the loss in the strength of the electromagnetic signal due to LOS path via free space. All radio signals are subject to FSPL i.e. the description of the geometric property that there is drop in energy signal as the distance from a radio transmission increases which is the square of the distance. In this model, the existence of single path between transmitter and receiver without barriers is assumed. A power ratio of *1.0 dB* or *0 dB* for the antenna gain is assumed and losses related to hardware imperfections or the effects of antenna gains are not included.

$$FSPL = \left(\frac{4\pi d}{\lambda}\right)^2 = \left(\frac{4\pi df}{c}\right)^2 \tag{2.1}$$

where $\lambda$ and $f$ are the signal wave length ($m$) and the signal frequency ($MHz$), $d$ and c are the distance from transmitter ($m$) and the speed of light in a vacuum (2.99792458) x $10^8$(m/s) respectively.

Path Loss in decibel (dB) are given as:

$$PL_{free\ space}(dB) = 10\log_{10}\left(\frac{4\pi df}{c}\right)^2 = 20\log_{10}\left(\frac{4\pi df}{c}\right) \tag{2.2}$$

$$PL_{free\ space}(dB) = 20\log 10(d) + 20\log(f) + 20\log 10\left(\frac{4\pi}{c}\right) \tag{2.3}$$

where distance ($d$) and frequency ($f$) are measured in ($m$) and ($MHz$) respectively for a typical radio application. Therefore:

$$PL_{free\ space}(dB) = 20\log 10(d) + 20\log(f) + 92.45 \tag{2.4}$$

For $d$ ($m$) and $f$ ($MHz$), the constant is 32.45 and -27.55 [25, 26].

However, the free space model is not often used alone but as part of *Friis* transmission equation which include antenna gain [27]. A mathematical expression is proposed by *Friis* for transmission loss due to free space which defines the ratio relating to received power $P_{rx}$ and transmitted power $P_{tx}$ with respect to effective area of the BS antenna $A_{tx}$, mobile antenna $A_{rx}$, distance $d$ (m) and the carrier wavelength $\lambda$. In this model, received power is a function of transmitted power, antenna gain and transmitter-receiver distance [11]:

$$\text{Transmission loss} = \frac{P_{rx}}{P_{tx}} = \frac{A_{rx}\,A_{tx}}{d^2\,\lambda^2} \tag{2.5}$$

The above formula can be evaluated further for an ideal isotropic antenna:

$$\frac{P_{rx}}{P_{tx}} = \left(\frac{\lambda}{\pi d}\right)^2 \tag{2.6}$$

For $d$ (m), $f$ ($MHz$) and from the linear domain power unit ($W$) to log domain power unit ($dBm$), Eq.( 2.6) gives Eq.(2.7) and Eq.(2.8) respectively [28]:

$$P_{rx} = P_{tx} 20\log_{10} d + 20\log f + 32.45 \tag{2.7}$$

$$P_{dBm} = 10\log_{10}\left(P_{mW}\right) \tag{2.8}$$

### *2.2.2. Two - Ray Ground Reflection Model*

This model assumes the existence of two paths between transmitter and receiver for most propagation cases: (i) direct path and (ii) reflected path [29-31]. Predictions using Ray tracing method are good when the detailed information of the area is accessible, however the predicted result may not be applied to other locations. This makes the model site specific and mostly typical indoor channel do not have only two paths, therefore making the model just a theoretical model. At a point, the receive power does not oscillate round the FSPL any further, thereby falling of stronger, a break distance is established.

The break distance is calculated as [32]:

$$d_c = \frac{4\pi h_{tx} h_{rx}}{\lambda} \tag{2.9}$$

where $h_{tx}$ and $h_{rx}$ are transmitting and receiving antenna height *(m)* respectively. Friis's equation is applied when the distance is shorter than this break distance and the modified path loss expression is used [33]:

$$P_r = \frac{P_{tx} h_{tx}^2 h_{rx}^2}{d^4} \tag{2.10}$$

### *2.2.3. Ikegami Model*

This model predicts fields' strength at definite points by applying detailed map of building shapes, positions, and heights. The limitations of trace ray paths as a result of single reflection from the wall accounts for diffraction calculation by application of single edge estimation and assumption of constant value for the wall reflection. The two ray (reflected and diffracted) are power summed as [34]:

$$L_E = 10\log_{10} f_c + 10\log_{10}(\sin\Phi) + 20\log_{10}(h_o - h_m) - 10\log_{10} w - 10\log_{10}\left[1 + \frac{3}{L_r^3}\right] - 5.8 \tag{2.11}$$

where $\Phi$ and $L_r$ are the angle amid the street-mobile direct line and the loss due to reflection (0.25) respectively. Losses are underrated by Ikegami model at large distance and frequency variation and in comparison with measurement [35].

## 2.3. Empirical Models

The empirical models are close-fitted formulas of measurement data that give a general description of channel behaviour in the environment where the measurements are obtained [36]. They predict PL between transmitter and receiver as a function of distance, considering a single path. The attenuation due to dipole antennas (*dB*) is given as [16]:

$$PL_{dB}(d) = 10\alpha\log_{10}(d) + c \tag{2.12}$$

where $\alpha$, $d$ and $c$ are the path loss exponent, distance ($m$) between transmitter and receiver, and a constant (speed of light in a vacuum) which is dependent on parameters such as antenna type and frequency, respectively. Empirical models do not compel the knowledge of the exact environmental geometry and as a result are simple to use, though they are not highly accurate [37]. Some of the empirical models are as follows:

### *2.3.1. Okumura Model*

This was developed by *Okumura*, a Japanese radio scientist, based on the measurements he obtained in and around Japan on environment clutter and irregular terrain [38]. He discovered that simple power law is related to a good PL profile with exponent $\mu$ as a function of antenna gain and frequency. The model was further classified into three models: for rural, sub-urban and urban areas and selects different modes of computation depending on the complexity of the environment (in relation to population density).

*Okumura* model is suitable for urban area with many structures but with few tall structures and valid for frequency range of *150 MHz* to *1920 MHz* over distance of *1km-100 km* and transmitter antenna height of *30m-100m*. However, it can be extrapolated up to *3000 MHz* [10, 35, 37]. The model is stated as:

$$\mathrm{PL}_{50\%}(dB) = LF + A_{mu}(f,d) - G(h_{te}) - G(h_{re}) - G_{area} \tag{2.13}$$

where $PL_{50\%}$ is the $50^{th}$ percentile loss in propagation (median) value, $LF$ and $A_{mu}$ are the loss due to free space and the median attenuation relative to the free space, $G(h_{te})$, $G(h_{re})$ and $G_{area}$ are the gain factor for transmitter antenna height and the gain factor for receiver antenna height and the environment type gain respectively. Variation of $G(h_{te})$ and $G(h_{re})$ with height:

i. For height less than *3 m* (variation at the rate of *20 dB*/ decade):

$$G(h_{te}) = 20\log_{10}(\mathrm{hte}/200)\,1000m > hte > 30m$$

ii. For height less than *3 m*, (variation at the rate of *10 dB*/decades):

$$G(h_{re}) = (10\log_{10}/3)\,hre <= 3m;\; G(h_{re}) = 20\log_{10}(hre/3)\,10m > hre > 3m$$

*Okumura's* model is known to be one of the simplest and best models when it comes to PL cellular radio systems predictions. It is based entirely on measured data with no analytical explanation. However, its slow response to fast changes in terrain profile is a major demerit of this model.

### *2.3.2. Electronic Communication Committee (ECC)-33 Model*

This is an extension of *Okumura* model formulated by Electronic Communication Committee in the European Conference of Postal and Telecommunications Administrations (CEPT) [39] and the most widely used model based on *Okumura* model [40]. Originally, the experimental data for *Okumura* model was obtained from the outskirts of Tokyo, the developers segmented urban area into large and medium size cities and correction factors given for rural and sub-urban areas [26, 41]. Given that a highly built up area like Tokyo is relatively different from what is obtainable in a standard European sub-urban area, the segmented urban area model for medium size city was suggested for European cities. Although the *Hata-Okumura* model is broadly used for Ultra High Frequency (UHF) bands, it is uncertainly accurate for higher frequencies [40, 42, 43]. Hence, the COST-231 Hata model extends the frequency range up to *2000 MHz*, however, it was designed for mobile systems with Omni-directional receiver antennas sited less than *3 m* above ground level. A different approach was considered in ECC-33 model which extrapolated the novel measurements by *Okumura* and modified the assumptions to represent a closely wireless system. It is extensively used for urban settings, large and medium size cities. ECC-33 PL model is expressed as follows [4]:

$$PL = A_{fs} + A_{bm} - G_t - G_r \tag{2.14}$$

where $A_{fs}$ and $A_{bm}$ are the attenuation due to free space and the media Path Loss, $G_t$ and $G_r$ are transmitter antenna height gain factor and receiver antenna height gain factor respectively.

These are independently defined as:

$$A_{fs} = 92.4 + 201og_{10}(d) + 201og_{10}(f) \tag{2.15}$$

$$A_{bm} = 20.41 + 9.831og_{10}(d) + 7.891og_{10}(f) + 9.56[1og_{10}(f)]^2 \quad (2.16)$$

$$G_t = 1og_{10}\left(\frac{h_t}{200}\right)\left\{13.955.8\left[1og_{10}(\mathrm{d})\right]^2\right\} \quad (2.17)$$

For medium environment,

$$G_r = \left[42.57 + 13.71og_{10}\right]\left[1og_{10}(h_r) - 0.585\right] \quad (2.18)$$

For large city,

$$G_r = 0.759\ h_r - 1.862 \quad (2.19)$$

where *f* and *d* are the frequency (*GHz*) and the distance between transmitting and receiving antenna (*m*), $h_t$ and $h_r$ are the transmitter antenna height (m) and receiver antenna height (*m*).

### *2.3.3. Stanford University Interim (SUI) Model*

The SUI Model formulated by IEEE 802.16 broadband wireless access working group in Stanford University, is proposed for frequency band below *11 GHz* containing the channel model. It is an expansion of Hata model with frequency greater than *1900 MHz* used for PL prediction in rural, sub-urban and urban environments. The model is categorized into three different groups: A, B and C, with each group having its own characteristics.

i. Group A is associated with a hilly environment that has moderate-to-heavy foliage densities and has the maximum PL. It is suitable for compact populated urban area.
ii. Group B is associated with hilly environment with rare trees or flat environment with heavy or moderate tree densities. It is suitable for sub-urban area.
iii. Group C is associated with flat environment with light tree densities and has the minimum PL [44]. It is suitable for rural area.

Typically, the different groups are generally described as:

Cells $< 10\ km$ in radius, receiver antenna height is *2 m -10 m*, BS antenna height is *15 m- 40 m* and high coverage requirement *(80-90%).*

The median path for the basic SUI model is expressed as [33]:

$$PL = A + 10\gamma 1og_{10}\left[\frac{d}{d_0}\right] X_f + X_h + \ S \tag{2.20}$$

$$A = 201og_{10}\left(\frac{4\pi d_0}{\lambda}\right), \gamma = a - bh_t + \frac{c}{h_t}, d_0 = 100m, 10 < m < h_b < 80m\ , 8.2dB < s < 10.6dB$$

where $d$ and $\lambda$ are the distance between transmitter and receiver antenna ($m$) and the wave length *(m),* ( $f \le 2000MHz$ , $\lambda$ is path loss exponent) $h_t$, $\gamma$ and $S$ are the height of transmitter antenna ($m$), free space PL and a long normally distributed factor respectively. a, b, c are constants and $h_t$ determines the path loss exponent for environment type.

Table 2.2. Geometric values for SUI Model Parameters [4, 45]

| **Parameters** | **Environment A** | **Environment B** | **Environment C** |
|---|---|---|---|
| A | 4.6 | 4 | 3.6 |
| b(m-1) | 0.0075 | 0.0065 | 0.005 |
| c(m) | 12.6 | 17.1 | 20 |

In-use frequency correction factors and transmitter antenna height are stated as follows:

$$X_f = 6.01og_{10}\left(\frac{f}{2000}\right) \tag{2.21}$$

$$X_h = -10.81og_{10}\left(\frac{h_r}{2000}\right)\ \left(type\ A\ and\ B\ environment\right) \tag{2.22}$$

$$X_h = -20.0\ 1og_{10}\left(\frac{h_r}{2000}\right)\ \left(type\ C\ environment\right) \tag{2.23}$$

$$S = 0.65\left(1og_{10}\,f\right)^2 - 1.31og_{10}\left(f\right) + \alpha \tag{2.24}$$

A is *5.2* for type A and B environment and *6.6 dB* for type C environment.

### *2.3.4. Flat-Edge Model*

This model as proposed by *Saunders* and *Bonar* computes estimated knife-edge diffraction losses as a result of uniformly spaced buildings. This proffers a way out of the concept of propagation in built-up areas by assumption of equal building height and spacing [29, 30]. The model assumes transmitter above or beneath series of obstacles that are of stable size and spacing and receiver underneath the building top. Average values for the area under consideration are used or the values computed separately when there is significant variation in urbanization. The total PL for flat-edge model is expressed as:

$$PL = L_n(t) + L_{FS} + L_E \tag{2.25}$$

where $L_E$ $L_n$ and $L_{FS}$ are the single edge diffraction above the last building, multiple diffraction above the remaining *(n-1)* buildings and the free space loss respectively. $L_n$ is a function of $t$ expressed as:

$$t = -\alpha\sqrt{\frac{\pi b}{\lambda}} \tag{2.26}$$

where $\alpha$ is in (radians), $b$ and $\lambda$ are (meters)

If $1 \le n \le 1000$ and $-1 \le t < 0$, then $L_n$ can be calculated by following the approximate formula [46].

$$L_n(t) = -20\log_{10}A_n(t) = -(C_1 + C_2\log_{10}n)\log_{10}(w) - 10\log_{10}\left(1+\frac{3}{L_r^2}\right) - 5.8 \tag{2.27}$$

where $C_1$, $C_2$, $C_3$, and $C_4$ are 3.29, 9.9, 0.77 and 0.26, respectively.

Diffraction for final building is computed from Ikegami model [47].

$$L_E = 1\log_{10}f + 10\log_{10}(\sin\phi) + 20\log_{10}(h_o - h_m) - 10\log_{10}(w) - 10\log_{10}\left(1+\frac{3}{L_r^2}\right) - 5.8 \tag{2.28}$$

where $\phi$ is the transmitter-receiver street-direct line angle and $L_r$ is loss due to reflection (0.25).

For large buildings in flat-edge model, there is approximately the same PL exponent with measurements [48].

### *2.3.5. Erceg-Greenstein Model*

Path Loss model for frequencies around *1.9 GHz* was presented by *Erceg et. al.*[39] based on measurement using substantial set of data gathered by *AT&T* in the sub-urban areas of New Jersey. This model combines both median PL and randomly distributed variation at some distance. It is expressed as:

$$PL = A + 10\left(a - b * h_{tx} + \left(\frac{c}{h_{rx}}\right)1og_{10}\left(\frac{d}{d_0}\right) + x1og_{10}\left(\frac{d}{d_0}\right) + y\mu_\sigma + yz\sigma_\sigma\right) \tag{2.29}$$

where A is marginal PL due to free space at some reference distance, *x*, *y* and *z* are random variables positioned between *-2* and *2* (*x* lies between *-1.5- 1.5*). The parameters for *a*, *b*, *c*, $\mu_\sigma$, $\sigma_\sigma$ are fitted for the three type of environments: type A is suitable for hilly environment with in-substantial tree densities, type B is suitable for flat environment with moderate-to-intense tree densities and type C is suitable for flat environment with insubstantial tree densities.

### *2.3.6. Lee Model*

Lee model as proposed by *Lee* [11] in 1982 is one the extensively used PL models due to its simplicity and reasonable prediction accuracy. The model was originally derived for frequency in the region of *900 MHz*, but was later extended to *2 GHz* for distance ranges greater than *1.6 km* [49]. Lee model is used in prediction of area to area PL and it specifies different parameters for varying type of environments. The model gives PL relative to reference condition and is expressed as [50]:

$$PL_{Lee} = P_{LO} + \beta 1og_{10}\left(\frac{r}{1.6km}\right) + 101og_{10} + 101og_{10}\left(\frac{f}{900MHz}\right) - \alpha_0 \tag{2.30}$$

$$\alpha_0 = \alpha_1\,\alpha_2\alpha_3\,\alpha_4\alpha_5$$

where:

$$\alpha_1 = \left( \frac{new transmitter\ antenna\ height\ (m)}{30.48m} \right)^2 ; \alpha_2 = \left( \frac{new reciver\ antenna\ height\ (m)}{3m} \right)^{\xi};$$

$$\alpha_3 = \left( \frac{new transmitter\ power}{10W} \right)^2;$$

$$\alpha_4 = \left( \frac{new transmitting\ antenna\ gain\ with\ respect\ to\ \lambda_c / 2\ dipole}{4} \right)$$

$\alpha_5$ is the receiver antenna gain correction factors $n$, and $\zeta$ are based on empirical data with the values:

$n$ is 2.0 for $f_c$<*450MHz* (for urban areas), $\zeta$ is 2 for transmitter antenna height >*10m* and 3 for transmitter antenna height < *3m*.

Table 2. 3. Parameters for Lee path loss for various environments at *900MHz* [26, 51]

| Environment | $L_o$ | $\beta$ |
|---|---|---|
| Free space | 80 | 20 |
| New American Sub-urban | 89 | 4.35 |
| North American sub-urban | 101.7 | 3.85 |
| North American Urban | 104 | 4.31 |
| Japanese Urban | 124 | 3.05 |

## 2.4. Semi-Empirical Models

Semi-empirical models are based on the combination of (i) empirical and (ii) deterministic models. It has the characteristics of both types of models [33]. One of the commonly used semi-empirical models is:

### *2.4.1. Walfisch-Bertoni Model*

This model was developed by Walfisch-Bertoni and takes into account the effect of diffraction from the top of roof and building [52]. It is suitable for an environment with uniform buildings height and spacing and assumes an elevated transmitter antenna realized by vertical plane wave approximation to compute buildings influence on the signal by elevated

antenna [51, 53]. Average signal strength using diffraction is predicted at street level using Walfisch–Bertoni model. The model considered PL resulting from the following three factors:

i. Free space loss is given as [44]:

$$PL_{fs} = -101og_{10}\left(\frac{\lambda}{4\pi r}\right)^2 \tag{2.31}$$

ii. Diffraction and scatter loss from the roof top down the street are given as:

$$PL_{down} = \frac{\lambda\rho_1}{2\pi^2\left(H_b - h_r\right)} \tag{2.32}$$

iii. Diffraction from the roof tops is given as:

$$PL_{rooftops} = P(g)^2 - \left(0.1\left[sin\delta\sqrt{\frac{d}{\lambda}}\right]^{0.9}\right)^2 \tag{2.33}$$

$$\sin\delta = \frac{h_t - H_b}{R}$$

The total loss can then be expressed as [33]:

$$PL_{total} = \frac{5.51}{32\pi 4}\frac{(h_r - H_b)^{1.8}\rho_1 d^{0.9}}{(H_b - h_r)^2}\frac{\lambda^{21}}{E^{3.8}} \tag{2.34}$$

$$PL_t\left(dB\right) = P_T + G_T + G_R - L_T - L_R - RSS \qquad \text{(Measured)} \tag{2.35}$$

$$PL_{total} = 89.5 - 101og_{10}\left(\frac{\rho_1 d^{0.9}}{(H_b - h_r)^2}\right) + 211og_{10}f - 181og_{10}\left(h_t - H_b\right) + 381og_{10}R_K \tag{2.36}$$

where $\rho_1$ is $\sqrt{\left(\frac{d}{2}\right)^2 + (H_b + h_r)^2}$, $f$, $h_t$, $H_b$ and $h_r$ are frequency ($MHz$), height of transmitter antenna ($m$), building height and receiver antenna height ($m$) respectively. $d$ and $R$ are the buildings space ($m$) and the distance between transmitter and receiver.

## 2.5. Terrain Models

Terrain is described as natural geographical characteristics of the land where electromagnetic signal propagates. Terrain models compute losses due to diffraction along LOS path as a result of obstruction such as buildings or the terrain itself [54]. The terrain features drastically affects the propagation of electromagnetic waves, even over moderate distances. Different terrain produces diffuse multi-path, diffraction loss, shadowing and blockage. Median PL is provided as a function of distance and terrain roughness by these models. Variations in media as a result of other effects are treated separately [55].

### *2.5.1. International Telecommunication Union (ITU) Terrain Model*

The model is simple and computes PL as a product of free space with a single diffraction due to terrain [40, 41]. Developed on the basis of theory of diffraction, PL is predicted using ITU terrain model as a function of the blockage height and the first Fresnel zone. The model describes any impediment in-between telecommunication link and thus is fit to be used in cities and in rural areas and valid in any terrain. Coverage frequency and distance are expressed as [38]:

$$A = 10 - 20C_N \tag{2.37}$$

$$C_N = \frac{h}{F_I}, h = h_L - h_0 \quad \text{and} \quad F_I = 17.3\sqrt{\frac{d_1 d_2}{f_d}}$$

where $A$ and $C_N$ are the additional loss due to diffraction (in excess of free space loss) ($dB$) and the normalized terrain clearance. $h$, $h_L$, $h_0$, and $F_1$ are the difference in height ($m$) (it is negative in the case of LOS path being completely obscured), the line-of–sight link height ($m$), the obstruction height ($m$) and the first Fresnel zone radius. $d_1$, $d_2$, $f$ and $d$ are the

obstruction distance from one terminal (*m*), the obstruction distance from the other terminal (*m*), the transmission frequency (*GHz*) and the distance between the transmitter and receiver (*m*).

The ITU terrain model calculate extra losses in every obstructed path, these are added together to the predicted PL for LOS and then computed using Friis transmission equation or an equivalent empirical or theoretical model. The model is considered suitable for losses above *15 dB* and could be suitable for losses as low as *6 dB*. It recommends the discard of a negative loss as a result of the blockage (which in reality is a gain) or any loss that is less than *6 dB*. To correct the loss due to assumption of free space, the additional maximum loss is utilized.

### *2.5.2. Egli Model*

The model predicts point-to-point link total PL and is applied in outdoor LOS propagation while presenting PL as a single quantity [56]. It is typically appropriate for cellular settings that have a fixed and a mobile antenna and applicable to settings where the propagation goes over an irregular terrain. Nevertheless, it does not consider travel through vegetative obstruction like shrubbery. Egli model is usually appropriate for UHF and Very High Frequency (VHF) spectrum transmissions. The model is given as [57]:

$$P_{R50} = G_T G_R \left( \frac{h_t h_r}{d^2} \right)^2 \left( \frac{40}{f} \right)^2 P_T \tag{2.38}$$

where $P_{R50}$ is the $50^{th}$ percentile receive power (*w*), $P_T$ $G_T$ and $G_R$, are the transmit power, the total gain of transmitter antenna (*dB*) and the total gain of receiver antenna (*dB*) respectively, $h_t$, $h_r$ $d$ and $f$ are the height of transmitter antenna (*m*), the height of receiver antenna (*m*), the distance from transmitter antenna (*m*) and the frequency of transmission (*MHz*) respectively.

Path Loss however, is predicted as a whole using Egli model and there is no further division of the loss into losses due to free space and other losses [55].

### *2.5.3. Longley-Rice (LR) Model / Irregular Terrain Model (ITM)*

This model predicts radio signal attenuation at *20 MHz* to *20 GHz* frequency range [37]. It is implemented in two configurations: (i) prediction over an area and (ii) point-to-point link prediction. It offers a simplification of the received signal power, however, there is no detailed channel characterization. Statistical resources are used to recompense for the channel characterization, this is dependent on the variable from each environment and situation. Signal variation is ascertained using the model in accordance to free space, atmospheric and topographical changes. Statistical estimates are used to describe these variations that contribute to the overall signal attenuation. The statistical estimates variables of the prediction model vary with situation, time and location. The reference attenuation is determined as a function of distance, urban area factor and attenuation variables [55].

## 2.6. Artificial Neural Network (ANN) Models

To determine the loss in signal power during electromagnetic signal propagation from one point to another point or to multiple points in wireless communication environment as seen in most cases, there is need to comprehend the pattern of Radio Frequency (RF) propagation in that environment. The prediction of signal power loss during propagation in outdoor environment is a difficult and complex task as a result of different physical mechanisms such as reflection, refraction, scattering and multipath phenomenon [1, 34].

Apart from the traditional empirical and deterministic prediction models, ANN models have successfully been used as better prediction alternatives [58]. Artificial neural networks can be defined as adaptive statistical tools which models almost the same way as the biological nervous system. Like human, ANNs are capable of learning by examples such as representation of a given process by mimicking related examples. This is as a result of its accurateness, simplicity and litheness in adapting to different environments, with distinguished characteristics such as ability to learn from data, generalization of patterns in data and ability to model non-linear functions [59, 60]. Employing ANN algorithms for problem solving is on the basis of their stochastic and evolutionary methods in finding out the relationship among physical parameters of problems [61, 62]. They fundamentally find out

the impact of different mechanisms of propagation phenomenon on the signal power without any necessary and complex mathematical calculations.

Artificial neural networks are made up of neurons arranged in layers with weighted connections. The ANN architecture can be linear or non-linear with an input layer, hidden layer (one or more) and an output layer. The data sets are fed to the ANN through input layer which transfers it to hidden layer for non-linear ANNs and an output is received through output layer. The input data are combined using suitable weights and are passed through defined transfer functions [63]. Training of the ANN is conducted using data sets and the ability of ANN to predict an unknown situation depends on the type of ANN architecture, type of training sets, training parameters and training processes [64]. The task of ANN is to ascertain the most exceptional functional fit for a set of input-output pairs. It also interpolates and extrapolates unknown data sets. Different ANN learning algorithms are employed for different purposes; thus, the choice of a given learning algorithm is dependent on the job required of the ANN to carry out. Artificial neural network can be applied for different task such as function approximation, pattern association, control, pattern recognition, filtering, etc. [65].

### *2.6.1. Evolution of Artificial Neural Networks*

Artificial neural networks started in the early *1940's* with *McCulloch* and *Pitts's* introduction of models of neurological networks [66]. In *1947*, a practical area of application in recognition of spatial patterns by neural networks was shown by *McCulloch* and *Pitts* which was omitted in their *1943* work. The classical Hebbian rule was formulated by *Hebb* [67] in *1949* which generally represents the basis of almost all neural learning procedures. The rule connotes that connection between two neurons is reinforced when the two neurons are active at same time. However, *Hebb* [67] was not able to verify this rule as a result of absence of neurological research and this necessitated the neuropsychologist, *Lashley* [68] in *1950* to defend the study that brain information storage is comprehended as a distribution system. *Marvin* [69] in *1951* developed and practically implemented in his study the neuro-computer Snark which previously has the capability to automatically adjust weight.

The first supporters of Artificial Intelligence (AI) desired that simulation of capabilities should be by means of software while the Neural Network (NN) supporters desired to realize

systems behaviour by means of the imitation of the smallest part of the brain system such as the neurons [70]. *Rosenblatt et.al.* [69] at Massachusetts Institute of Technology (MIT) developed the earliest successful neuro-computer known as the Mark I perceptron that is capable of recognizing simple numeric. In *1959*, they formulated and demonstrated perceptron convergence theorem and among other things, a learning rule by adjusting the connecting weights [69]. In *1960, Widrow* and *Hoff* [71] launched the adaptive linear neuron which is a fast and accurate adaptive learning system and has advantage over original perceptron because of its adaptivity. It becomes the first extensive saleable used neural network. *Steinbunch* [72] in *1961* launched and described the concepts for neural techniques with analysis of their possibilities and limitations. A precise mathematical analysis was published by *Marvin* and *Seymour* [66] in *1969* showing the incapability of the perceptron model to represent various important problems, therefore bringing to an end the over-estimation and the popularity of perceptron model. In *1972* a model of linear associator was introduced by *Kohonen* [71] after a long silence and slow reconstruction as a result of prove of incapability of the perceptron model in representing many important problems. However, such model as linear associator was represented independently and from the neuro-physiological point of view.

*Malsburg* [71] in *1974* made use of neurons which are non-linear and more motivated biologically, where he developed a learning procedure known as backpropagation of error. Between *1976* and *1980*, *Grossberg* [73] made presentation of various papers where several models were analysed mathematically with dedication of keeping a neural network that is capable of learning without the destruction of already learned association. This led to the model of adaptive resonance theory under the co-operation of *Carpenter* [73] in 1982. The field of neural networks also slowly regained importance as a result of the invention of Hopfields networks by *Hopfield* [63] which are motivated by the laws of magnetism in physics. *1983* witnessed the introduction of neural model of the neo-cognitron by *Fukushima* [68] which was capable of recognizing handwritten characters. The backpropagation of error learning procedure as a generalization of the delta rule was developed separately in *1986* and published widely by parallel distributed MLP, whereas the *Marvin's* negative evaluations were disproven.

### *2.6.2. Concept of Artificial Neural Networks*

Artificial neural networks are generally defined as interconnected neurons which are pre-arranged in various layers. These neurons are simple processor which operates on the inputs they are fed through the connections. The processors are vast parallel-spread out and comprises of simple processing units with natural tendency of storing and making available exponential knowledge. The correlations between ANNs and human brain are summarized as [67]:

i. Artificial neural networks acquire knowledge from the environment by means of learning process.
ii. The acquired knowledge is stored in the network by inter-neuron connection strength called synaptic weights.

The method employed to carry out the process of learning by ANNs is known as learning algorithm. This learning algorithm logically modifies the network synaptic weight to achieve the desired objective. The synaptic weight modification presents a conventional technique for ANNs design [74].

Artificial neural networks have assorted family of networks with the functionality of each type of network determined by the architecture, training algorithm adopted, neuron characteristics etc. It derives its computing power from its architecture and its learning ability to generalize [75]. The use of ANNs offer the following capabilities and properties:

i. Non-linearity- Artificial neural network can either be linear or non-linear network.
ii. Input-Output mapping- When input data which is desired output is presented to the ANN, there is modification of network synaptic weight to reduce the difference between desired and actual output to minimal according to the appropriate statistical principles. The network training is repeatedly carried out until a steady state where no observed changes in the synaptic weight adjustment are seen. Thus, the network learns by constructing input-output mapping for the considered problem.
iii. Adaptivity- Artificial neural networks possess in-built capacity of their synaptic weight adaptation to changes in proximate environments. However, its adaptive capability may not always result to robustness.

iv. Contextual information- The architectural design and ANNs activation state is used in knowledge representation. Each neuron in network architecture is probably affected by global activities of other network neurons.
v. Fault tolerance- Artificial neural network which is implemented in hardware form has the capability of robust computation.
vi. Analysis design and uniformity- The universality of the ANNs processing unit i.e. the neuron, permit it to share theories and training algorithm in various ANNs applications.

### *2.6.3. Components of Artificial Neural Networks*

The major component of ANN is the neuron. The neuron model is made up of three major parts [74]:

i. A set of connecting links known as synapses with each characterize by having a weight.
ii. A summer for summation of input signal weighted by the neuron individual synapses.
iii. An activation function used for limiting the neuron output amplitude. It defines the neuron output in terms of induced local field. There are different types of activation which ranges from linear to non-linear activation function, these include:

a. Threshold activation Function-Neurons that have a threshold activation function is describe as McCulloch-Pits model where the neuron output takes the value of *1* if the local field of the neuron that is induced is not negative and the value of *0* for the reverse with its function being non-differentiable.
b. Piece-wise linear activation function-A linear combiner results in cases where linear region of operation is kept without proceeding into saturation.
c. Sigmoid transfer Function-These are non-linear transfer functions which are known for their severely increasing functions, smoothness and are asymptotically limited. They are of two different types; the logistic functions and the hyperbolic tangent function.

### *2.6.4. Representation of knowledge by Artificial Neural networks*

Knowledge is known as information or model stored by a person or a machine to translate, predict and adequately respond to the real world. Artificial Neural Networks have the task of learning the environmental model in which they operate and to maintain the model satisfactorily, independent of any changes in the environment. Knowledge can be in form of a leading knowledge of the operating environment or measurement from the real world. These measurements are used to train the ANNs, however, most time, the measurements are noisy because of measurement system imperfections. Knowledge representation in different architectural design of ANN is defined by the values of the network weights and biases [76].

### *2.6.5. Process of Learning by Artificial Neural Networks*

The ANNs learn about their environmental behaviour by interactively adjusting their synaptic weights and biases based on the measurement from the environment. Artificial neural network learning is a process at which the network free parameters are adapted via a process of environmental stimulation at which the network is embedded [77]. The learning type is determined by the way a parameter change. A defined rule that offers solution to learning problem is known as learning/training algorithm. The different learning algorithms vary from each other based on their weight adjustment formulation. Also, during learning, the way the inter-connected neurons of ANN relate to the environment is considered. There are different learning processes adopted by ANNs.

i. Error correction learning - The cost function minimization using this method results to a learning rule usually referred as delta rule. The delta rule state that the adjustment carried out by the neuron synaptic weight is proportional to the error signal product and the inputs of the considered synapses [78].
ii. Memory based learning - Most of the previous learning experiences by the ANNs are overtly stored in huge memory of correctly classified input-output instances [79].
iii. Hebbian learning - The application of the input signal repeatedly results to increase in output and thus, an exponential growth which finally drives the weight connections to saturation. Then, information is stored in the synapses and there is lost in selectivity [80].

iv. Competitive learning - The neurons output of the ANN competes for activeness among them. However, only a single neuron is active at a particular time unlike in Hebbian learning where many neurons output may be active at the same time [79].

The learning process can either be supervised or unsupervised learning.

**Supervised Learning** - In supervised learning, a training set that consists of training patterns and a corresponding correct (desired) output values are known while expecting an actual output after training of ANN. The output value generated by network while it is still training is known as teaching input.

**Unsupervised learning** - There is no teaching input with unsupervised learning, thus, no desired output is provided to the ANN for training vector. Alternatively, there is a provision made for identifying a measure of the representation quality required by ANN to learn and the network free parameters are optimized based on the measure [63]. A grouping of the provided training input to the network is achieved at the end of network training based on similarity measure imposed by network.

### *2.6.6. Artificial Neural Network Architectures*

Artificial neural networks like human brain, learn by example, by means of a learning process and are designed for specific uses such as system prediction, pattern recognition, function approximation, etc. The ANNs helps in cases where the formulation of exact algorithm solution to a problem is difficult. Different ANN architectures have been used as alternative models for the prediction of propagation losses. The ANN architecture is made up of many processing elements known as artificial neurons. The processing element is made up of inputs, transfer function and output [81, 82]. There are different types of ANNs with different architectures; however, they are all described via a transfer function which is used by processing element by method of training with a learning rule known as algorithms. Artificial neural network architectures comprise of either a single layer or multiple layer neurons. Different ANN architectures such as Perceptron model, Adaptive Linear Element (ADALINE), Multi-Layer Percepton (MLP) network, Radial Basis Function (RBF) network, Generalized Regression Neural Network (GRRNN), analysed in course of this research work have been discussed in the proceeding chapters.

## 2.7. Evolution of Cellular Technology justifying the use of LTE Data

Mobile phone networks are dependent on evolving cellular technology which is subject to change with time. It comprises the evolution of cellular technology in the field of 1G, 2G, 3G, 4G, and upcoming mobile network 5G. As time changes, latest network technologies have been designed for an increase in cellular networks as this increase speed capabilities [83]. Radio access technology selection is of great important in the global market as each cellular operator struggle to take dominant position to provide excellent services to their various users.

Electromagnetic spectrum is being used by mobile operators in the provision of their services. This spectrum is usually shared between the broadcasting cellular communication and other purposes such as armed forces. Sequel to the inception of mobile network, capacity is increased by the division of frequency over channels. Thus, there is reduction in the overall availability of bandwidth which directly have influence on the mobile network [84]. In *1970*, geographical area division instead of frequency division offers the use of more efficient electromagnetic spectrum. Over the years, there has been evolution of the cellular network based on factors such as quality, cost, quantity, availability etc.

First Generation Network (1G) – The advancement of 1G started in the *1970*'s by Japan who took the first step in the cellular technology development followed by Normadic Mobile Telephones (NMT) in Europe and in America where it was developed as Advanced Mobile Phone Services (AMPS) [85]. The 1G also known as first generation of analogue cellular network has various cells that permits the same frequency usage severally resulting in system capacity increment. Its limitations include unreliable handoff, low capacity, poor voice link and no data security as analogue signal does not permit advance method of encryption.

In *1990*'s, an improvement on the 1G network known as Second Generation Network (2G) was introduced by the European Commission, set by standard made by the cellular telecommunication that are functioned and maintained by International Telecommunication Union (ITU). The 2G cellular Network include Global System for Mobile Communication (GSM) and Code Division Multiple Access (CDMA). It has advantage of increased coverage capacity, improved version of security system and quality [86]. Though the 2G Network can

handle more calls per amount of given bandwidth and provide services such as emails and SMS, its limitations include weaker digital signal and reduced sound range, since GSM has a fixed range of 35km after which there occurs technical limitations.

The Third Generation Network known as the Generation Mobile Telecommunication (3G) introduced in *2001* was established for faster and easier wireless communication from standard set by International Telecommunication Union (ITU) known as International Mobile Telecommunication-*2000* (IMT-*2000*). The improvement permits high network quality that supports improved services such as video streaming, speed browsing and video calling. It makes use of W-CDMA as a crucial standard [87]. However, its limitations include high power consumption, increased cost of cellular data among others.

The Fourth Generation Network (4G) also known as Long Term Evolution Network (LTE) rolled out in *2010* but still significantly inaccessible in many cities has the capacity to offer safe and secure internet protocol solution at which multi-media, voice and data can be accessible to users at each time and anywhere. The major aim of LTE network is to offer high data quality services with high data transfer speed of about *100mbps*. It has speed greater than *20mbps* and has possibility of roaming around different network and technology [88].

## 2.8. Review of Past Works on Prediction of Propagation Path Loss

A lot of research works have been carried out on the effectiveness of different PL models using traditional empirical and deterministic models. The authors usually make an assessment based on the theoretical analysis using electromagnetic propagation idealized theories or by assessing the model that fits in through measurement data collected from environment of interest or the combination of the *two* traditional models. Practical, [89, 90] presented lower bounds on the PL prediction accuracy using *thirty* propagation models. Measurement was carried out in different urban and rural environments; however, it was concluded at the end that there is no considered PL model that consistently predicted PL. A comparative evaluation of *five* different PL models using collected data from urban and sub-urban area at *910 MHz* was presented [46] with no conclusion on the particular model that offers the best result. Different PL models for fixed wireless access system were compared [46]. This was based on the measurement carried out in Cambridge with the COST-231

model, ECC-33 model and the SUI model showing the most guaranteed result. Studies on PL at UHF/VHF bands were carried out in Southern India where field strength measurement was taken at *200 MHz*, *400 MHz* and *450 MHz* and Hata model showed superiority over other considered models in all cases [47]. Propagation models for Global System for Mobile communication (GSM) *900*MHz and *1800 MHz* using modified Okumura and COST 231 Hata models were developed for Enugu and Port Harcourt in Nigeria, with the models fully adapted in the cities among other considerations and also made provision for rain attenuation and distinct features of the cities [15].

Models representing the propagation characteristics in NLOS situations with up to *three* intermediate vehicles were considered [91]. The effectiveness of the models was verified by comparison of results from calculations made with the measured received power as a function of the height of the receiving antenna. In [38], *five* propagation scenarios representing rural, sub-urban and urban environments were investigated comparing the radio propagation characteristics at *700 MHz* and *2500 MHz* relating to macro-cellular coverage. The result shows the mean PL with advantage at *700 MHz* against *2500 MHz* and ranges approximately from *11 dB*-*14 dB* except for forested hilly terrain with the difference of about *18 dB*. Optimized PL empirical model by means of proposed least squared method was introduced in [12]. The outdoor measurement taken in Cyberjaya, Malaysia was used in PL comparison with other considered models. The optimized Hata model offered a better performance as its relative error is lowest in comparison to other models. Three propagation models were presented for sub-urban area revealing the least PL with Okumura model and the highest PL with COST-231 model for a particular transmission distance [92]. The SUI model was used in PL calculation in *three* different terrains (rural, sub-urban and urban areas) and parameters from different terrain analysed [27].

The PL behaviour of propagation models was presented in [93], proposing a better prediction using semi-empirical model (Walfisch-Ikegami). Path loss was estimated using *five* different models: Hata Okumura model, ECC-33 model, COST-231 Hata model, SUI model and the Ericsson model [94]. ECC-33 shows a better prediction result for sub-urban area over other models. Analysis of the performance of various PL models were carried out in different environments for wireless network [17]. It suggested the use of SUI model as a preferred model because of lesser PL value with *10%* difference at reduced receiver antenna height for

rural and sub-urban areas when compared to other model reviewed in the work with reference to free space estimated value. However, the author concluded that the use of a particular model for PL estimation at various antenna heights in all areas is not ideal. The efficiency of Okumura-Hata model was investigated using GSM base station operating at *900 MHz* in a sub-urban area of the Northern part of Nigeria [95]. On comparison of the measured results from the field with the Okumura-Hata model for rural and sub-urban area, the result obtained shows the least variation with Okumura-Hata model for sub-urban areas. Okumura Hata model was optimized for outdoor propagation coverage in urban southern part of Nigeria using Code Division Multiple Access system (CDMA) at *800 MHz* operating frequency [14]. This was developed by comparison of calculated PL and collected measurements data using Hata, SUI, Egli and Lee models within applicable CDMA frequency range. Based on small mean error and closest PL exponent, Hata model have a preference as a reference for PL optimization when the measured path losses were compared. The application of the optimized model in Nigeria CDMA system shows more reliability for urban PL calculation at *800 MHz* frequency band. A novel PL model to tackle propagation delay in Long Term Evolution (LTE) network was introduced [96].

Different correlation factors were considered using the propagation algorithm for both transmitter and receiver antenna heights. On comparison with the Friis model, simulation result of the proposed propagation model for both uplink and downlink shows a decrease in propagation delay. Artificial neural network models for the prediction of PL in urban area was employed in [1] while a hybrid model that combines a traditional model, COST Walfisch-Ikegami model with adaptive neural component was used in [62]. The influence of training set selection in ANN based PL propagation predictions was presented in [97] and a GRNN model for the prediction of PL at *900 MHz* for Jos city in Nigeria was presented in [98].

A novel ANN model for PL prediction in different environment of propagation medium was presented in [60] while [99] used ANN for macro-cell prediction. A new micro-cell prediction model was introduced using a feedforward ANN to overcome the limitations of the traditional models [34]. Measurements of electric field strength were obtained in Belgrade from four separate transmitter locations in [100] and the mathematical computing models for supporting decision of instalment of new base station and the selection of their configuration in order to obtain the trade-off between coverage maximization and cost minimization was

investigated. A GRNN model for path loss prediction at *900 MHz* in Jos, Nigeria was studied in [98]. A single neuron ANN model and MLP ANN models was evaluated for macro-cell PL prediction via measurement data from a CDMA mobile network in a rural Australia for the neural network training [101]. A hybrid model that combines a MLP ANN and an empirical based log-distance model was studied in [62]. A learning rate of *0.01* was employed to predict the accuracy of a system, however the major drawback of the work was poorly trained data and high complex system with only a *37.5%* accuracy obtained [102]. A learning rate of *0.2%* was used in a hybridized ANN of linear regression and ANN with a prediction accuracy of *90.3%* [103]. The major drawback of the work is too many features and use of polynomial data [104]. The effect of learning rate between *0.1* to *0.8* was investigated on the prediction vector on ANN. The fluctuations gave a low prediction of *80.65%* with *0.1%* learning rate and high prediction of *90.3%* with *0.8%* learning rate.

## 2.9. Present Research Work

An adaptive neural network predictor that combines MLP ANN and Adaptive Linear Element (ADALINE) is developed for enhanced signal power prediction. This is followed by a resourceful predictive model, built on MLP network with a vector order statistic filter based pre-processing technique for improved prediction of measured signal power loss in different micro-cellular urban environments using real world measure data from LTE cellular network.

## 2.10. Chapter Summary

Propagation models are valuable tools and algorithms for the prediction of signal propagation loss between transmitter and receiver in locations where the wireless communication systems network is to be deployed. This chapter presents a theoretical background of our research work and a detailed baseline survey of different types of propagation models in cellular communication networks. Traditional modelling techniques using empirical and deterministic models have been discussed. Each of the models has its own peculiar characteristics and limitations and suitable for use in various radio propagation environment. This study would be of help to Radio Frequency (RF) engineers in choosing the right propagation model suitable for a given environment.

Artificial neural networks, which are alternative to prediction of propagation PL have been introduced. The evolution and concept of ANN and the components of ANN have been discussed. The chapter also includes a comprehensive review of past works on prediction of PL using different traditional empirical and deterministic models as well as ANN models and brief evolution of cellular technology justifying the use of LTE data. Finally, our present research work and the novelty are introduced.

# CHAPTER 3

# PROPAGATION PATH LOSS MODELLING USING TRADITIONAL EMPIRICAL AND DETERMINISTIC MODELS

This chapter demonstrates the influence of link distance, operating frequency, transmitter antenna height and receiver antenna height on Path Loss (PL) using four different traditional propagation PL models: (i) Hata model, (ii) COST 231 model, (iii) Walfisch-Ikegami model and (iv) Comité Consultatif International des Radiocommunications (CCIR) model. The performances of these models have been assessed via a written program used for model simulation using real world measured data from a micro-cellular urban area. Different traditional empirical and deterministic propagation models have been used over the years with emphasis on their characteristics, potencies and limitations. The main objective of this chapter is performance assessment of some traditional models using real world measured data for Long Term Evolution (LTE) network planning. The network parameters considered during simulations are: operating frequency, transmitter antenna height, receiver antenna height, building height, building separation, street width, street orientation angle, LTE cell link distance.

## 3.1. Introduction

According to report, the number of mobile connected devices are expected to exceed the population of the world by the year 2020 [105]. The Fourth Generation (4G) mobile systems based on LTE radio access technology are currently being deployed by mobile network operators to meet the ever increasing subscriber traffic growth and demand for higher data rates [106]. It offers high speed for mobile phones data access and may increase the speed and network capacity by the use of different radio interface.

Long term evolution technology is developed to provide access to wider range of communication services, which include advanced mobile services that is supported by mobile and fixed networks [107]. Radio Frequency (RF) network engineers are often confronted with

the difficulties that come with deployment of new technology. A new radio network interface technology like the LTE needs different set of distinctive planning tools and algorithm for its effective operation in different propagation terrains. Of vital importance to the LTE system network deployment is the ability to predict accurate strength of radio signals from different transmitters in the system. The coverage planning of a cell is imperative in finding the best locations for Base Stations (BS) to build continous coverage according to the planning requirements.

Path Loss models are specific mathematical algorithms and models for appraising the radio signal path attenuation loss and coverage area of a BS transmitter [108]. A reliable and accurate prediction model helps in optimizing the coverage area, transmitter power and also gets rid of interference problems of the radio transmitters. Appropriate models must therefore be selected for performance assessment of the field strength and PL as well as other parameters. This helps the network engineers and planners in optimizing the coverage area and in using the correct transmitting powers. Path Loss is also calculated to analyse the link establishment in the telecommunication system.

Some of the basic propagation PL models: (i) Hata model, (ii) COST 231 Hata model, (iii) Walfisch-Ikegami model and (iv) CCIR Model have been analysed to re-assess their performances in the prediction of propagation PL for proper planning of LTE networks in different radio signal outdoor propagation environments.

The models have been analysed and compared at variation of transmitter-receiver (T-R) distance, heights of Mobile Station (MS) and BS antennas and transmitting frequency via computer simulations using real world measured data.

## 3.2. Propagation Loss Modelling

In the course of radio wave propagation, signal undergoes various losses as a result of obstacles between transmitter and reciever. Considering transmitter and reciever seperation, received transmitted power is given as [33]:

$$\mathrm{P_r(d)} = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L} \tag{3.1}$$

where $P_t$, $G_t$ , and $G_r$ are transmitter power, transmitter antenna gain, and receiver antenna gain respectively. The $L$ and $d$ are the system loss factor and link distance between transmitter and reciever *(m)*, respectively.

In logarithm form, the expression in Eq. (3.1) can be rewritten as [46]:

$$PL(dB) = -10log_{10}\left(\frac{\lambda^2}{(4\lambda)^2 d}\right) \tag{3.2}$$

In more general form, Eq. (3.2) can be expressed as [14]:

$$\mathrm{PL}(dB) = PL(d) + 10log_{10}\left(\frac{d}{d_o}\right) \tag{3.3}$$

where *PL(dB)*, $d$ and $d_o$ are path loss exponent, link distance between transmitter and receiver, and the close-in-reference distance, respectively. These are standard formulas used to derive the power gain, transmitter-receiver separation and PL exponent that are embedded in the wireless technologies [12]. The traditional empirical and deterministic models employed in this work are described in the following sections.

## 3.3. Propagation Path Loss Models

These are different mathematical expressions or algorithms employ in expression of different environmental radio characteristics [16]. They are either established on measurements from an area or on fundamental principles of radio wave phenomenon. The four different propagation PL models employed in this work are as follows:

### *3.3.1. Hata Model*

This model is an advanced version of Okumura model, also known as Okumura-Hata model and commonly applied for PL prediction for cellular transmission in built-up areas. It integrates data from Okumura model and advances it to capture the impact of scattering, reflection, and diffraction caused by structures in urban, sub-urban and rural areas. The model is suitable for frequencies from *150 MHz* to *1500 MHz*; distance of *1 Km* to *20 Km* from

transmitter to receiver; transmitter antenna height of *30 m* to *200 m* and receiver antenna height of *1 m* to *10 m*. The PL models for urban, sub-urban and rural areas are expressed as [50, 109]:

Urban area;

$$PL_U\left(Urban\right)=69.55+26.16\log_{10}(f)-13.82\log_{10}\left(h_t\right)-a\left(h_r\right)+44.9-6.55\log_{10}\left(h_t\right)\log_{10}\left(d\right) \tag{3.4}$$

where $PL_U$, $h_t$, $h_r$ and $a(h_r)$ are urban areas PL *(dB)*, height of transmitter antenna *(m)*, height of receiver antenna, and correction factor for receiver antenna height, respectively.

Sub-urban area [50]:

$$PL_{50}\left(dB\right)=PL_{50}\left(urban\right)-2\log_{10}\left[\left(f_c\ /28\right)\right]^2-5.4 \tag{3.5}$$

Rural area [11]:

$$PL_{50}\left(dB\right)=PL_{50}\left(urban\right)-4.78\left(\log_{10}\ f_c\right)^2-18.33\log_{10}\ f_c-40.98 \tag{3.6}$$

The correction factor for receiver antenna height [17]:

$$a\left(h_r\right)=\left(1.1\log_{10}\ f_c-0.7\right)h_r-\left(1.56\log_{10}\ f_c-0.8\right)dB \tag{3.7}$$

For large city, the receiver antenna correction factor is [17]:

$$a\left(h_r\right)=8.29\left(\log_{10}1.54h_r\right)^2-1.1,\ f<300MHz \tag{3.8}$$

$$a\left(h_r\right)=3.2\left(\log_{10}11.75h_r\right)^2-4.97,\ f\geq300MHz \tag{3.9}$$

Hata model is not suitable for frequencies from *1800 MHz* to *1920 MHz* micro cell planning when antenna is below the height of the roof and does not offer coverage outside *1500 MHz* frequencies [110].

### *3.3.2. COST 231 Hata model*

This model also known as Hata model PCS extension was formulated by the EUROpean Co-Operative for Scientific and Technical Research (EUROCOSTR). It is an off-shoot of Hata model that has its origin from Okumura model, covers frequency from *1500 MHz* to *2000 MHz* and is suitable for urban areas with the following characteristics: receiver antenna height of *1 m* to *10 m*; transmitter antenna height of *30 m* to *200 m*; and link distance of *1 km* to *20 km*. The COST231 Hata Model for PL is stated as follows [17]:

$$PL_{50}\left(urban\right) = 46.3 + 33.9\log_{10}\left(f_c\right)13.82\log_{10}\left(h_t\right) - a\left(h_r\right) + \left(44.9 - 6.55\log_{10} h_t\right)\log_{10} d + C_m \tag{3.10}$$

where $f_c$ $h_t$ and $d$ are transmission frequency *(MHz)*, transmitter antenna height *(m)*, link distance between base and mobile station *(km)*, respectively. $C_m$ is *0 dB* for sub-urban and rural environment and *3 dB* for urban environment and $a(h_r)$ is receiver antenna height correction factor, a function of the size of the area of coverage.

For small and medium sized environment, $a(h_r)$ *(dB)* is given as [91]:

$$a\left(h_r\right) = \left(1.1log_{10} f_c - 0.7\right)h_r - \left(1.561log_{10} f_c - 0.8\right) \tag{3.11}$$

For urban environment; $a\left(h_r\right)$ *(dB)* is given as [17]:

$$a\left(h_r\right) = 3.2\left(1log_{10}\left(11.75\right)\right)^2 - 4.97; > 400MHz \tag{3.12}$$

$h_r$ is receiver antenna height *(m)*.

### *3.3.3. CCIR MODEL*

This is an empirical model which combines the effects of PL due to free space and path loss generated by terrain. It is made available by Comité Consultatif International des Radiocommunications (CCIR), also known as ITU-R. The CCIR PL is expressed as [111]:

$$PL(dB) = 69.55 + 26.161log_{10}(f) - 13.821log_{10}(h_t) - a(h_r) + (44.9 - 6.551log_{10}h_t)1log_{10}d - B \tag{3.13}$$

where $h_t$ and *hr* are the height of transmitter antenna and height of the receiver antenna *(m)*, *d* is the link distance *(km)* and *f* is the frequency *(MHz)* respectively

$$a(h_r) = (1.11log_{10}(f) - 0.7)h_r - (1.561log_{10}(f) - 0.8) \tag{3.14}$$

$$B = 30 - 251og_{10} \quad \text{\% area of building coverage} \tag{3.15}$$

Eq. (3.13) is for Hata model's sub-urban and rural area propagation conditions, supplemented by correction factor $B$. *B is 0* for urban area with building coverage of up to *15%*.

$$B = 30 - 25\log_{10} 20 = 2.5dB \text{ for 20\% building coverage in urban area} \tag{3.16}$$

The CCIR differs from Hata model in two ways: (i) the effect of coverage area and (ii) receiver antenna height [33].

### *3.3.4 Walfisch-Ikegami Model*

Walfisch-Ikegami model also known as COST Walfisch-Ikegami, is a combination of Walfisch-Bertoni and Ikegami models developed by COST-231 project [55]. It considers only buildings in the vertical plane between transmitter and receiver and differentiates between two states; (i) Line of sight (LOS) and (ii) Non-Line of Sight (NLOS) and each of them are calculated differently. The model formulation defining the PL equation for LOS situation is expressed as [15]:

$$PL_{los} = 42.6 + 261og_{10}R + 201og_{10}f;\ for\ R \geq 20m \tag{3.17}$$

The path loss for NLOS is defined as [50]:

$$PL_{NLOS} = \begin{cases} L_{FS} + L_{rts} + L_{msd} \\ L_{FS} \qquad if\ L_{rts} + L_{msd} > 0 \end{cases} \tag{3.18}$$

where $L_{FS}$, $L_{rt}$, and *Lmsd* are loss due to free space, roof top to street diffraction, and loss due to multi-screen diffraction respectively.

The grouping of propagating signal along the multi-screen path into the street of mobile location is designated by $L_{rts}$ [50]:

$$L_{rts} = \{-16.9 - 10\log_{10} w + 10\log_{10} f + 20\log_{10} \Delta h_r + L_{ori} \tag{3.19}$$

where $h_{roof} > h_r; if\ L_{rts} < 0$ and $\Delta h_t = h_t - h_{roof}$ and $\Delta h_r = h_{roof} - h_r$ (where $h_t$ and $h_r$ are transmitter antenna height and receiver antenna height respectively).

$L_{ori}$ is defined as [33]:

$$L_{ori} = \begin{cases} -10 + 0.354\ \phi, & for\ 0 \le \phi < 35 \\ 2.5 + 0.075(\ \phi - 35), & for\ 35 \le \phi < 55 \\ 4 - 0.114(\ \phi - 55), & for\ 55 \le \phi < 90 \end{cases} \tag{3.20}$$

The multi-screen diffraction loss $L_{msd}$ is an integral, approximated by Walfisch-Betoni model and an answer to cases where transmitter antenna height is taller than the average roof top [33]. This was then extended by COST 231 to cases where transmitter antenna height is shorter than the average roof top by the inclusion of empirical formula. The model formulations are expressed as [9, 33]:

$$L_{msd} = L_{bsk} + K_a + k_b 10\log_{10} R + K_f 10\log_{10} f - 9 10\log_{10} b - 9 10\log_{10} f \tag{3.21}$$

Where $$L_{bsk} = \begin{cases} -18\log_{10}(1 + \Delta h_t); & for\ h_t > h_{roof} \\ 0\ ; & for\ h_t \le h_{roof} \end{cases} \tag{3.22}$$

$$K_d = \begin{cases} 18; & for\ h_t > h_{roof} \\ 18 - 15\dfrac{\Delta h_t}{h_{roof}}; & for\ h_t \le h_{roof} \end{cases} \tag{3.23}$$

$k_a$ is increase in path loss for transmitter antenna shorter than the roof top of adjacent building [33].

$$k_a = \begin{cases} 54; & for\ h_t > h_{roof} \\ 54 - 0.8\Delta h_t\ ; & for\ d \geq 0.5km\ , h_t \leq h_{roof} \\ 54 - 0.8\Delta h_t \left(\frac{d}{0.5}\right); & for\ d < 0.5km\ , h_t \leq h_{roof} \end{cases} \quad (3.24)$$

where $k_d$ and $k_f$ are control multi-screen diffraction loss against distance and frequency, respectively.

$k_f$ for sub-urban with moderate-tree density and urban area are given as [17]:

$$k_f = -4 + 0.7\left(\frac{f}{925} - 1\right) \quad (3.25)$$

$$k_f = -4 + 1.5\left(\frac{f}{925} - 1\right) \quad (3.26)$$

where *d*, *f*, *R*, *w*, and $\phi$ are the distance between transmitter and receiver *(km)*, frequency *(MHz)*, distance between buildings *(m)*, street width *(m)*, and direct path street incidence angle (degrees), respectively.

Walfisch-Ikegami model is limited to urban environment and only inserts a characteristic value [48].

Real world measured data used for the research work has been collected from a micro-cellular outdoor environment with built-up terrain, clusters of residential, heavy industrial and moderate commercial buildings. The map of the metropolitan area, Port-Harcourt, Nigeria is presented in appendix I. Data collection are done between JAN-DEC 2017 via a drive test and from different sectors of BS transmitting at *1900 MHz* operating frequency. A total of one thousand, nine hundred and seventy (1970) sample data has been collected at different distances between the transmitter and the receiver. The setup for data measurement includes a laptop installed with appropriate interface: Test Mobile System (TEMS) software, Sonny Ericson mobile handset equipped with TEMS software, Global Positioning System (GPS), compass, power inverter, test cables, vehicle, and digital maps of the area.

## 3.4. Graphical Results and Discussions

Programs are written for the different propagation models used for simulations to ascertain their performances at the variation of link distance between transmitter (Tx) and receiver (Rx), operating frequency, transmitter antenna height and receiver antenna height. The simulation parameters applied in the analysis are presented in Table 3.1.

Table 3.1. Simulation parameters

| S/N | Parameter | Value |
|---|---|---|
| 1. | Operating frequency(MHz) | 1800, 1900, 2600 |
| 2. | Transmitter antenna height (Tx)(m) | 30, 40, 50, 60, 70 |
| 3. | Receiver antenna height (Rx) (m) | 1.5, 3, 5 |
| 4. | Building height ($H_B$)(m) | 15 |
| 5. | Building separation (m) | 5 |
| 6. | Street width (m) | 20 |
| 7. | Street orientation angle (degree) | 30 for urban area, 40 for suburban area |
| 8. | LTE cell link distance (km) | 1, 2, 3, 4, 5 |

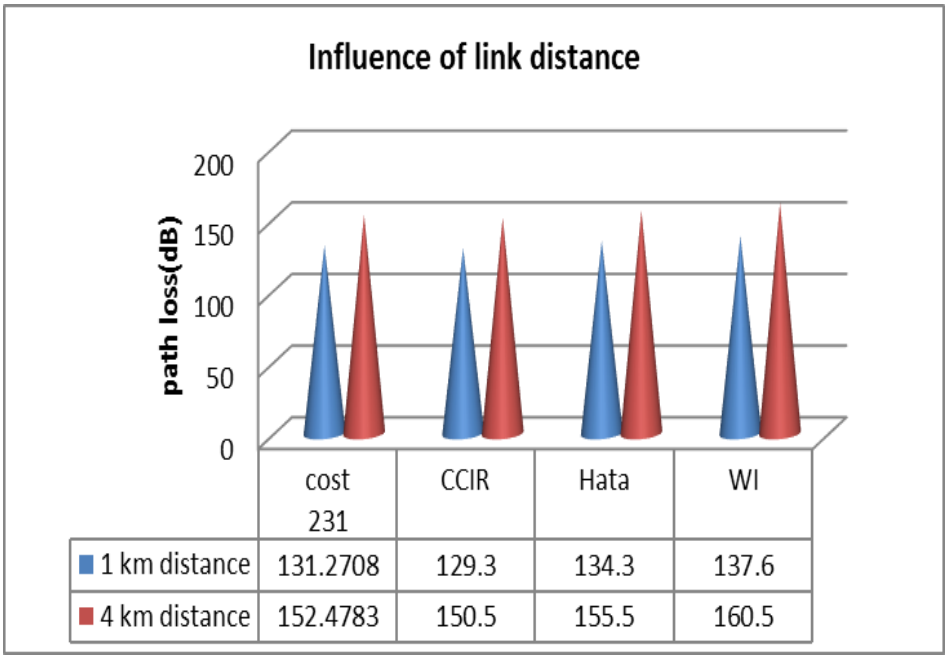
Based on these programs, we have done the following analysis:

i. Influence of link distance on Path loss
ii. Influence of operating frequencies on Path loss
iii. Influence of transmitter antenna height on Path loss
iv. Influence of receiver antenna height on Path loss

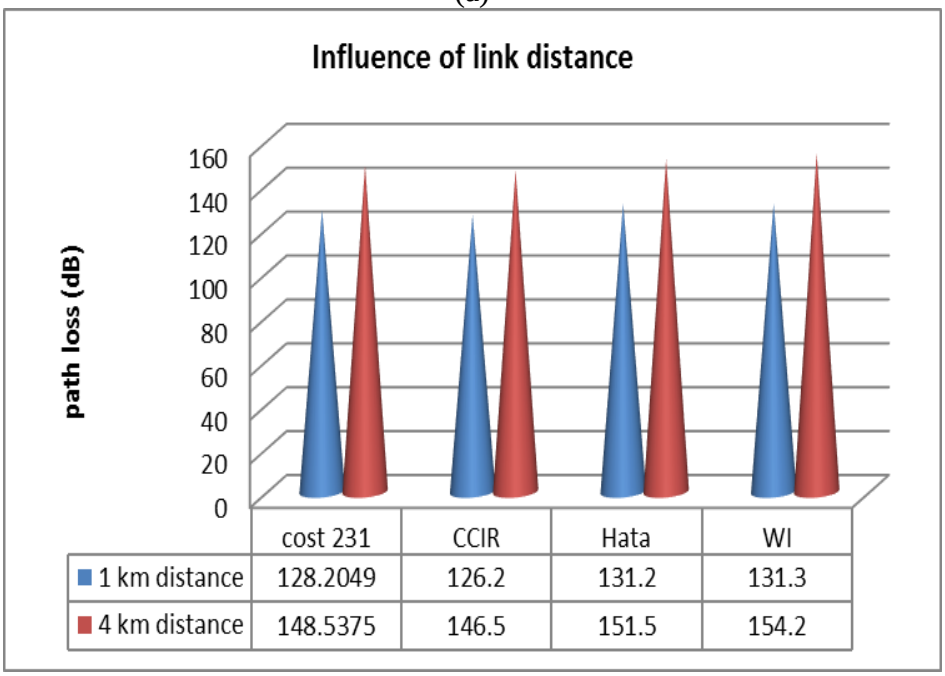
These details are given on following pages:

### *3.4.1. Influence of link distance on Path loss*

Propagation PL increases as link distance between transmitter and receiver increases. This is as a result of electromagnetic energy spreading in the free space which is explained by the inverse square law [34].




(a)

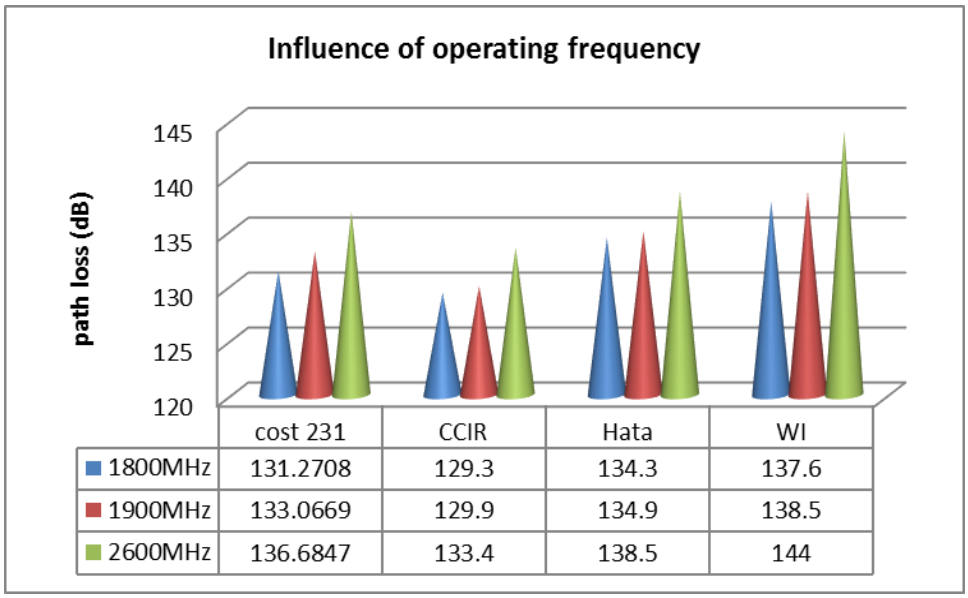



(b)

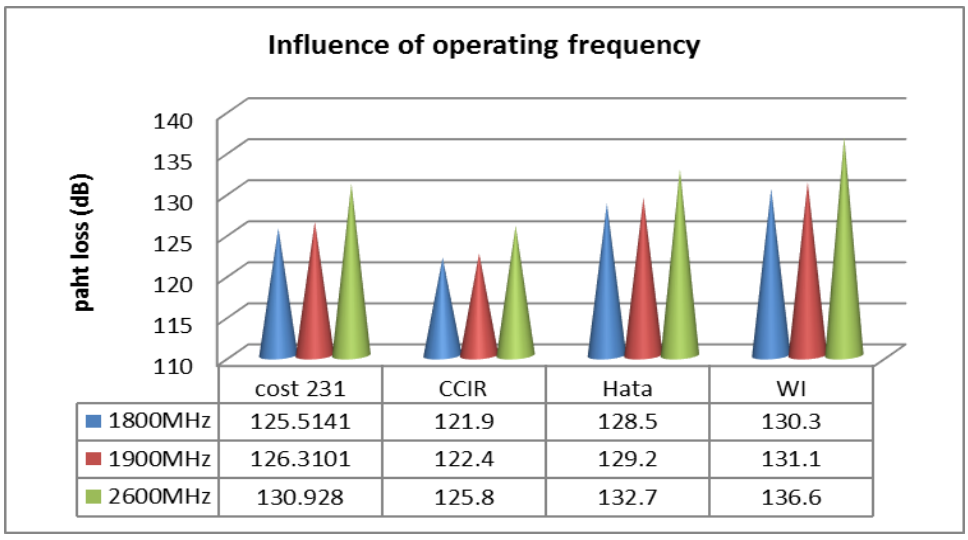
Figure 3.1. Path loss model performance at *1 km* and *4 km* link distances, *1800 MHz* operating frequency, *1.5 m* receiver antenna height for the transmitter antenna height of (a) *30 m* and (b) *50 m.*

### *3.4.2. Influence of operating frequencies on Path loss*

Path loss dependency on operating frequency is because of frequency dependency of the receiver antenna's aperture with fixed antenna gain. The extent electromagnetic power is picked up by antenna is determined by the antenna aperture [112].




(a)

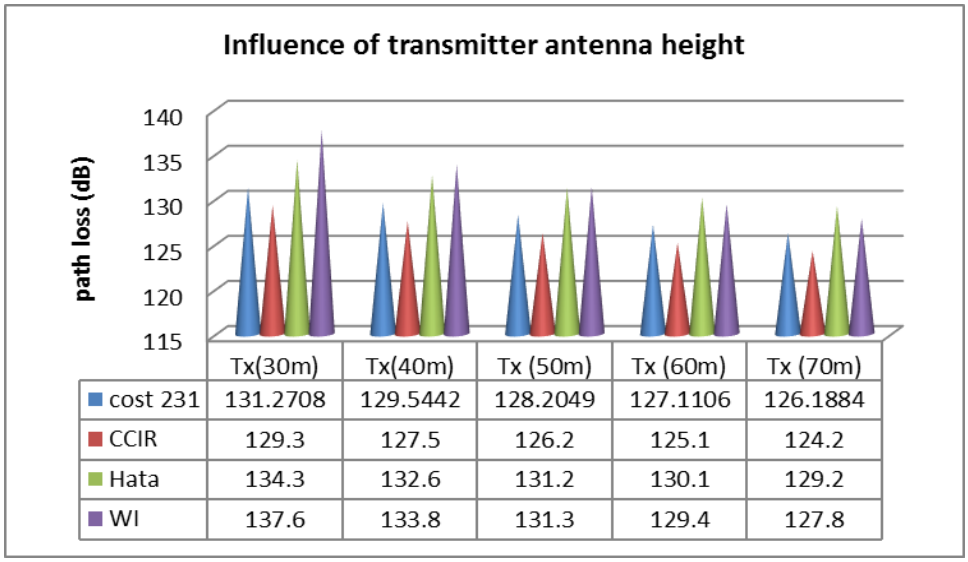



(b)

Figure 3.2. Performance of Path loss models at different operating frequencies, *1 km* link distance for (a) *30 m* & *1.5 m* and (b) *50 m* & *3 m* of transmitter & receiver antenna height respectively.

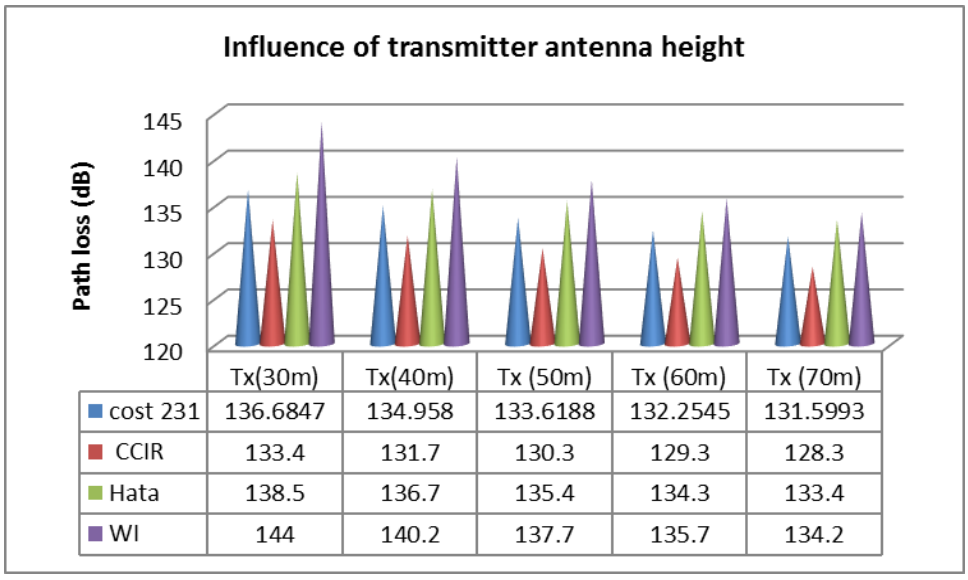
### *3.4.3. Influence of transmitter antenna height on Path loss*

Transmitter antenna height is the foundation of BS coverage area and increase of transmitter antenna height result to progressive extension of the distance at which the propagation path loss starts. Plane earth model has demonstrated that when the antenna height is doubled, it results to *6 dB* gain [113].




| | Tx(30m) | Tx(40m) | Tx (50m) | Tx (60m) | Tx (70m) |
|---|---|---|---|---|---|
| cost 231 | 131.2708 | 129.5442 | 128.2049 | 127.1106 | 126.1884 |
| CCIR | 129.3 | 127.5 | 126.2 | 125.1 | 124.2 |
| Hata | 134.3 | 132.6 | 131.2 | 130.1 | 129.2 |
| WI | 137.6 | 133.8 | 131.3 | 129.4 | 127.8 |

(a)



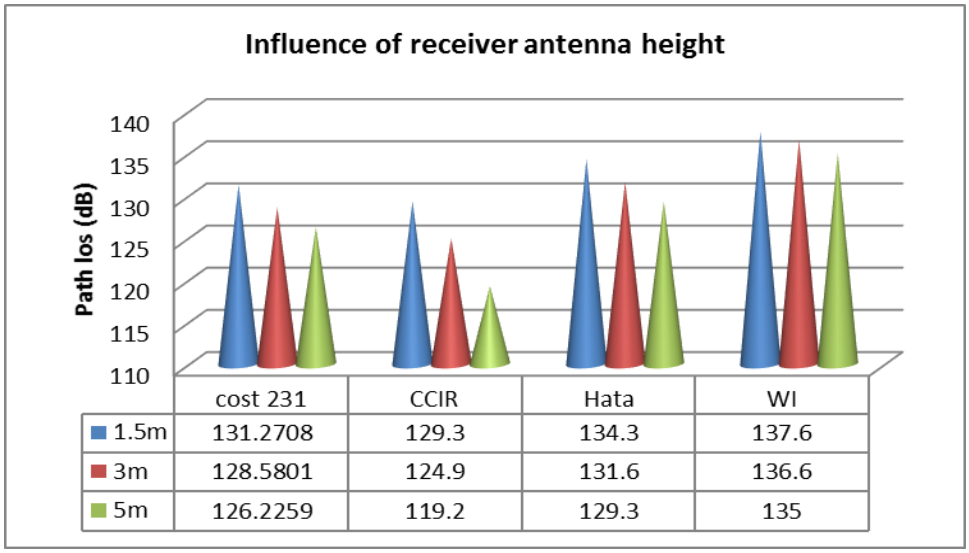

| | Tx(30m) | Tx(40m) | Tx (50m) | Tx (60m) | Tx (70m) |
|---|---|---|---|---|---|
| cost 231 | 136.6847 | 134.958 | 133.6188 | 132.2545 | 131.5993 |
| CCIR | 133.4 | 131.7 | 130.3 | 129.3 | 128.3 |
| Hata | 138.5 | 136.7 | 135.4 | 134.3 | 133.4 |
| WI | 144 | 140.2 | 137.7 | 135.7 | 134.2 |

(b)

Figure 3.3. Performance of Path loss models at different transmitter antenna height, 1km link distance, *1.5 m* receiver antenna height and operating frequency of (a) *1800 MHz* and (b) *2600 MHz*

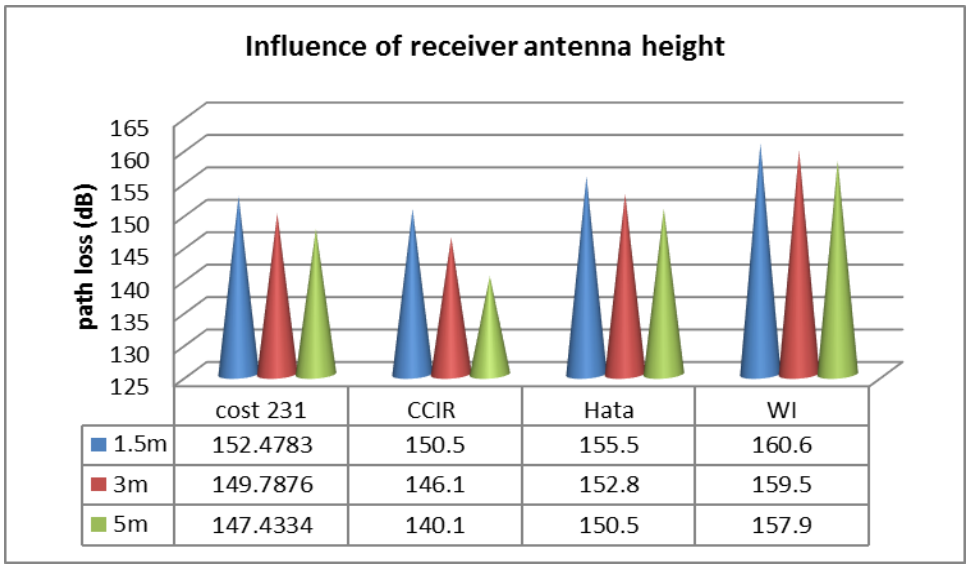
### *3.4.4. Influence of receiver antenna height on Path loss*

Loss as a result of rooftop to street diffraction is expected to reduce at higher receiving antenna height resulting to decrease in PL [91].




| | cost 231 | CCIR | Hata | WI |
|---|---|---|---|---|
| 1.5m | 131.2708 | 129.3 | 134.3 | 137.6 |
| 3m | 128.5801 | 124.9 | 131.6 | 136.6 |
| 5m | 126.2259 | 119.2 | 129.3 | 135 |

(a)




| | cost 231 | CCIR | Hata | WI |
|---|---|---|---|---|
| 1.5m | 152.4783 | 150.5 | 155.5 | 160.6 |
| 3m | 149.7876 | 146.1 | 152.8 | 159.5 |
| 5m | 147.4334 | 140.1 | 150.5 | 157.9 |

(b)

Figure 3.4. Performance of Path loss models at different receiver antenna heights, *30 m* transmitter antenna height and *1800 MHz* operating frequency (a) *1 km* link distance and (b) *4 km* link

## 3.5. Discussions

The simulation results of the different parameters variation to ascertain their effects on propagation path loss during electromagnetic signal propagation have been analysed in the following section:

### *3.5.1. Influence of link distance between the transmitter and the receiver*

The graphs presented in figure 3.1 show an increase in PL as the distance between transmitter and receiver increases. The CCIR model in both link distances of *1 km* and *4 km* for varied operating frequency and transmitter antenna height gives the lowest PL prediction values and Walfisch-Ikegami model gives the highest PL prediction values. This is because CCIR model addresses the effect of free space and PL generated by terrain while Walfisch-Ikegami model considers only buildings in the vertical plane between the transmitter and receiver, thus the high PL prediction value in outdoor environment that encounters different propagation phenomenon. This trend was also recorded at increase of transmitter antenna height to *40 m*, *50 m* and *60 m* respectively.

### *3.5.2. Influence of operating frequency*

As frequency increases, the need to ensure that receiver antenna gain is intact leads to less energy capture by receiver antenna which results to increase in PL. Graphs of figure 3.2 show lowest PL prediction values by the four considered models at *1800 MHz* operating frequency and the highest PL at *2600 MHz* operating frequency. The CCIR model gives the lowest PL prediction values while Walfisch-Ikegami model gives the highest PL prediction values.

### *3.5.3. Influence of transmitter antenna height*

At the increase of transmitter antenna height, there is decrease in PL as shown in graphs of figure 3.3. There is minimal PL as the signal propagates for all the models at *70 m* transmitter antenna height and maximum loss at *30 m* transmitter antenna height. The CCIR model gives minimal PL prediction values while Walfisch-Ikegami model gives highest PL prediction values.

### *3.5.4. Influence of receiver antenna height*

Path loss decreased with increase in receiver antenna height as shown in graphs of figure 3.4 with the various considered models. The lowest PL prediction value are recorded by CCIR model while Walfisch-Ikegami gives the highest PL prediction value.

## 3.6. Chapter Summary

This chapter demonstrates through simulation using real world measured data, the performance of four traditional propagation PL models: (i) Hata model, (ii) COST 231 Hata model, (iii) Walfisch-Ikegami model and (iv) CCIR model. The influence of link distance, operating frequency, transmitter antenna height and receiver antenna height on path loss have been analysed. Programs has been written for the simulation of the various models using the data set. The lowest PL prediction values are recorded using CCIR model at different link distance while Walfisch-Ikegami model gives the highest PL prediction values. This is because CCIR model considers the effect of free space and generated PL by terrain while Walfisch-Ikegami model only considers buildings in the vertical plane between transmitter and receiver operating frequency.

As the operating frequency increases from *1800 MHz* to *2600 MHz*, there is increase in PL because of less energy capture by receiver antenna to ensure that receiver antenna gain is intact. Increase in transmitter antenna height results to decrease in Path Loss. This is because it results to advancement in the distance of commencement of propagation PL as transmitter antenna height is the foundation of BS coverage area. As receiver antenna height increases, there is decrease in PL because of expected reduction in the loss due to rooftop to street diffraction. The obtained simulation results are in line with what are obtainable in literature.

# CHAPTER 4

# ARTIFICIAL NEURAL NETWORKS AND TRAINING ALGORITHMS

The major objectives of this chapter are to understand the architectural complexity of the simplest attainable Artificial Neural Network (ANN) model (the perceptron network) and further, the Multi-Layer Perceptron (MLP) network. The concept of backpropagation in feedforward ANNs are discussed and the performance of different backpropagation ANN training algorithms assessed on MLP ANN using real world measured data from a Long Term Evolution (LTE) cellular network in a sub-urban and urban areas. The predictive abilities of various ANN training algorithms are measured in terms of Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Standard Deviation (SD) and Correlation Coefficient (r). These performance metrics for the prediction of propagation Path Loss (PL) measures and offers quantitative calculated information to calculate the error difference between actual output and desired output.

## 4.1. Introduction

This chapter introduces ANN models: the perceptron ANN model, its characteristics and limitations, and the MLP ANN model, its characteristics, potencies and limitations. Other theories of ANN models have been discussed. The training of the MLP ANN with different training algorithms is organized in two parts:

Part I trained the MLP ANN using nine training functions of five different training algorithms with measured data from an outdoor micro-cellular sub-urban area that lies within latitude of $4^{o}45^{'}N$ and $4^{o}\ 60^{'}N$ and longitudes of $6^{o}50^{'}E$ and $8^{o}00^{'}E$ of the area.The area covers $260km^{2}$ with a population of four hundred and sixty-four thousand, seven hundred and eighty-nine and a tropical wet climate.

Part II re-assesses further the performance of ten training functions of different training algorithms on MLP ANN using data (measured) from a micro-cellular outdoor environment

with the experimental location lying within the latitude of $4^0 49'27'' N$ and longitude of $7^0 2'1'' E$ of the urban area. The area is a built-up terrain with clusters of residential, heavy industrial and commercial buildings with a population of over *two million*. It has a tropical wet climate characterize by heavy and lengthy rainy seasons, short dry season, a land mass of $360\ km^2$ and water of $9\ km^2$.

The predictive abilities of these training functions have been assessed in terms of different performance metrics. Electromagnetic signal propagation is strongly affected by obstacles such as buildings and propagation phenomenon. Thus, there is need for adequate planning of communication network to provide reliable service by optimizing the network performance through dynamic analysis, prediction and regulation of transmission signal behaviour over the network.

### *4.1.1. Perceptron Network*

Perceptron is the simplest possible neural network with computational model of single neuron invented in *1957* by *Rosenblatt* [114]. It is a unit that has weighted inputs which produces binary output based on threshold. Perceptron networks are trained by simple learning algorithms known as least square method or delta rule. This calculates error between the output of sample data and the network output and makes use of it to create an adjustment to the weights thereby implementing sort of gradient descent. The most fundamental type of an activation function is a binary function with only two possible results.

$$f(x) = \begin{cases} 1 if w^* x + b > 0 \end{cases} \tag{4.1}$$

Otherwise, $w$ is a vector of the weights which is real value, $w^* x$ is the point product $\sum_{i=1}^{n} x_i w_i$, where $n$ is the inputs number and $b$ is the bias. The function from Eq. (4.1) returns 1 if the input is *0* or positive and returns *0* for an input that is negative.

A neuron with the above type of activation function is known as perceptron. The perceptron network is also known as the single layer perceptron network and the simplest of the feedforward neural network. It is able to learn only linearly separable patterns as shown by *Minsky* and *Papert* in *1969* [63]. Its algorithm is summarized as follows:

i. For every input, it is multiplied by its weight
ii. All the weighted inputs are summed
iii. The output of the perceptron is computed based on the sum that is passed through an activation function.

However, single layer perceptron network are not capable of solving problems that are not linearly separable [115]. Also, because a perceptron network is linear, it will not be able to reach a state with all the input vectors appropriately trained with non-linearly separable training set i.e. if a hyper plane cannot be used to separate positive examples from negative examples. This will lead to non-convergence of the perceptron network. By connection of network neurons in multi-layers with non-linear activation function, non-linear decision boundaries which permit the solving of problems that are more complicated and non-linearly separable are created [71].

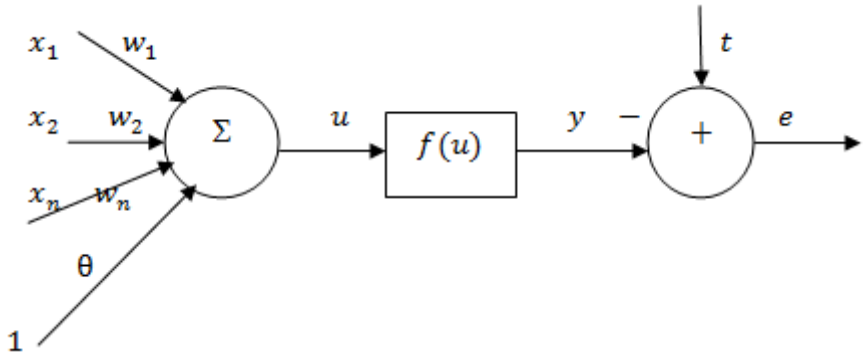



Figure 4.1. Simple neuron model [54]

The input signals are given as:

$$X = \left(x_1\, x_2 \quad \ldots . x_n 1\right)^T \tag{4.2}$$

Output of Eq. (4.2) gives:

$$u = W^{(T)} X \tag{4.3}$$

where $(.)^T$ represents the transpose and the weights of neuron $W$ and is given as:

$$W = \left(w_1\, w_2 \ldots\ldots . w_n \theta\right)^T \tag{4.4}$$

From figure 4.1, the output error of the neuron *(e)* is computed by subtraction of the transfer function *f(u)* from the target value*(t)*, $e = t - y$, where *y* is = *f(u)*.

To present the possibility for shifting the activation function *f(.)*, a scalar bias parameter $\theta$ is added to the weight to either right or left. Generally used transfer function can be referred as the activation function and is defined as [63]:

$$f(\mathrm{u}) = \frac{1}{1+\exp^{-u}} \tag{4.5}$$

### *4.1.2. Multi-Layer Perceptron (MLP) Network*

Multilayer-layer perceptron neural network is a system of inter-connected neurons that models a system of non-linear mapping between input and output vector [71]. It is a feedforward ANN created by *Rosenblatt* [114] in *1958* using the perceptron as the building blocks for a bigger and more practical network structure to cater for the limitations of the mapping ability of the single perceptron network. A standard MLP network structure is made up of source neurons which form the input layer, one or more hidden layers which are the computational neurons and an output layer. The input signal propagates layer by layer through the network. Each of the neurons in network contains a differentiable non-linear activation function and weights and output signal that are function of the input neurons modified by a non-linear activation function, fed through synaptic connection to subsequent layers. By the selection of an appropriate connection weights and activation functions, MLP network has been shown to approximate a smooth and measurable function between input and output vectors [116].

The network learns by means of training using sets of data made-up of input with an associated output vector. During the MLP network training, the network is trained repeatedly with the data using backpropagation algorithm while adjusting network weights to achieve expected input-output mapping. The expected MLP network output after training may not be equivalent to a known output thereby resulting in error signal (the difference between real output and expected output). The error magnitude is utilized during the training for further adjustment of the weight to ensure reduction in the global error of the network.

Different backpropagation training algorithms are employed in the training of a MLP network. Once the network is trained with an appropriate representative training data using a suitable training algorithm, the MLP can generalize well to a new input data. One of the problems associated with the use of MLP network is to determine the architecture of network, i.e. adequate number of layers and neurons [116]. No specific rule is in existence on the method of layer and neuron selection for solving a problem. Instead, the selection depends on the existing problem and its complexity. Importantly, the reason of training a MLP network is to attain a good generalization on input data such as in prediction applications. Too many neurons in hidden layer of MLP network may result to poor generalization while few neurons may lead to non-convergence of the network.

Multi-layer perceptron networks use different learning techniques, the most common being supervised learning backpropagation technique [80]. In supervised learning backpropagation technique, output values are compared with expected results by calculating the value of some error function that is pre-defined. The error is feedback through the network while the weight of each connection is adjusted by the algorithm to reduce error function value by certain amount. After sufficient training cycles, the network converges at a certain state with small calculation error. By this, the network has learned a clear target function. Therefore, learning occurs by varying of connection weights after processing of each data based on the output error in comparison to the expected result. This is referred as backpropagation of error.

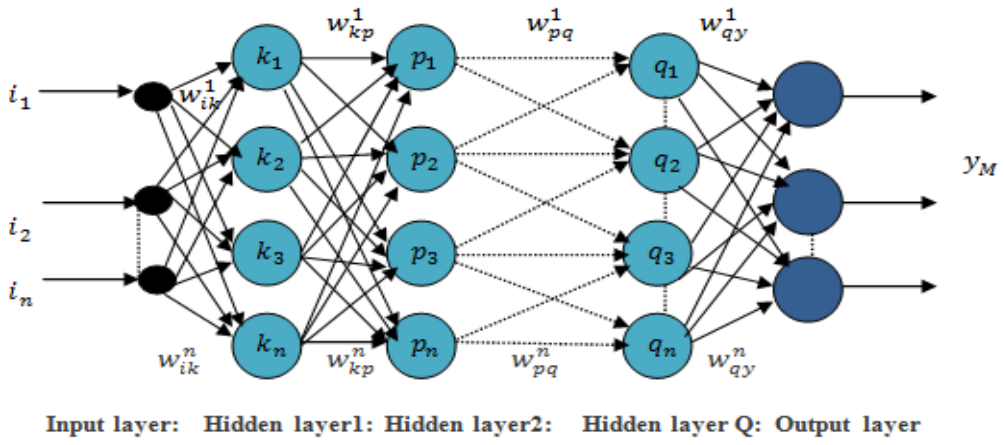



Figure 4.2. Architecture of a multi-layer neural network with n-hidden layer [117].

Assuming input layer with $i_n$ neurons, $i= (i_0, i_1,i_2,...i_n)$ as shown in figure 4.2, the network output at first hidden layer $k$ is computed as:

$$k_1^n = f\left(\sum_{k=1}^{n} w_{ik} i_1^n\right) \tag{4.6}$$

where $n=1,2,3...$ and $w_{ik}^n$ is the weight between the $i$ and $k$ neurons

This becomes the input to the second hidden layer $p$ and the network output at the second hidden layer becomes:

$$p_1^n = f\left(\sum_{p=1}^{n} w_{kp}^n k_1^n\right) \tag{4.7}$$

This becomes an input to the $Q$ hidden layer and the output of the $Q$ hidden layer becomes:

$$q_1^n = f\left(\sum_{q=1}^{n} w_{pq}^n p_1^n\right) \tag{4.8}$$

Assuming a sigmoid activation function is used, thus:

$$f(x) = \frac{1}{1+e^{-x}} \tag{4.9}$$

The network overall output $y_M$ is computed as:

$$y_M = f\left(\sum_{y=1}^{n} w_{qy}^n q_1^n\right) \tag{4.10}$$

where $y_M = y_1 + y_2 ..... \; y_n = f(w,i)$ and $f$ and $w$ are the transfer function and the matrix weight defined as:

$$w = \left(w_0, ...w_{ik}^{\ n}, ....w_{kp}^{\ n}, ....w_{pq}^{\ n}, ....w_{qy}^{\ n}\right) \in \Re \tag{4.11}$$

Using multiple layers of hidden neurons in the hidden layer facilitates better processing power and flexibility of the system [65]. However, too many hidden neurons normally result to over specification of the system thereby making it incapable of generalization, while few hidden neurons result to improper fitting of the input data by the system thereby reducing system robustness.

Architectural definitions of MLP networks are very important as lack of adequate choice of layers and neurons for connection can prevent the network from solving problems by inadequate adjustment of weight parameters. Architectural optimization of the connection of hidden layers and neurons for establishing ANN that effectively solve a given problem remains one of the tasks yet to be solved in many research areas[118].

## 4.2. Concept of Back Propagation (BP)

Backpropagation determines the gradient of loss function and is applied in gradient descent optimization algorithm for weight adjustment of every neuron that contributed to the process of training [119, 120]. For every supervised learning algorithm, the aim is to discover function which maps appropriately set of inputs to the corresponding right output. The essence of backpropagation is to train network for adequate learning of internal representations that permit learning of any random input to output mapping [121].

Before the network training commences, the weights are randomly selected, and neurons learn from the training examples. If $x_{i1}....x_{in}$, $o_t$ are the training examples with $x_{i1}....x_{in}$, being the network inputs and $o_t$ being the correct output which is expected from the network given the inputs, then $x_{i1}....x_{in}$ will likely compute an output $y_M$ different from the expected or desired output $o_t$ considering random weights. To measure the discrepancy of the desired output $o_t$ and the actual output $y_M$, squared error measure is applied as:

$$E = \left(0_t - y_M\right)^2 \tag{4.12}$$

where $E$ is the error or discrepancy. The difficulty of inputs to outputs mapping can therefore be cut down to an optimization problem by locating a function which will give minimal error as shown in figure 4.3.

$$y_M = w_{i1}, \ldots\ldots w_{in} \tag{4.13}$$

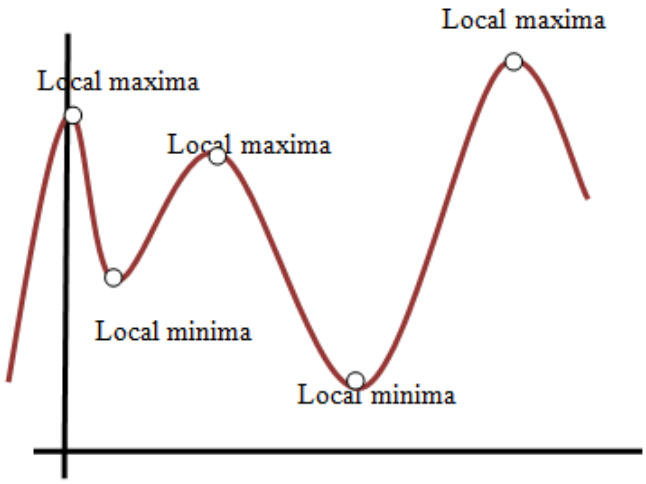



Figure 4.3. A sketch of a function with local minima and local maxima [69].

The error is also dependent on the neuron weights which eventually need to be modified in the network to permit learning. A conventional algorithm used in finding weights set which minimizes the error is the gradient descent algorithm while backpropagation is applied in the calculation of the direction of the steepest descent [122].

### *4.2.1. Gradient Descent*

Neuron weight permits ANNs to learn by updating after the forward passes of data through the network. The reason for weight adjustment is to ensure reconciliation of the difference between desired value and actual value ensuing forward passes [65]. Error is an important measure to ascertain the differences, and the respective error of each neuron send backward through the network to aid the process of update, i.e. backpropagation of error.

Cost function for error determination based on neuron weights as shown in figure 4.4 can be applied and the lowest point on the cost function known as the optimal value i.e. local minima where rate of function change equals zero can be ascertain. Conceptual use of slope of the angle of the cost function at the present location reveals the direction of the slope [123]. Algebraically, a negative slope indicates a downward movement while a positive slope indicates an overshoot i.e. movement beyond the optimal. The slope is determined using gradient descent.

Gradient is the rate at which a function changes while descent implies exploring at the base of the cost function with changing gradient. It takes into account a total of forward pass data set and calculates cost and thereafter propagates the errors backward to the neurons through the network [78]. There are two type of gradient descents: (i) vanilla plain and (ii) stochastic gradient descents [124]. All data weights are repeatedly adjusted when applying Vanilla Plain Gradient Descent (VGD) while Stochastic Gradient Descent (SGD) samples data randomly. Learning can be speed up by data random sampling for an improve prediction result. Gradient descent shows vulnerability to local minima if all data instance is applied in weight adjustment determination and may be made less vulnerable to extremes and outliers by considering the data en bloc; however, this is undesirable when in search for global minima.

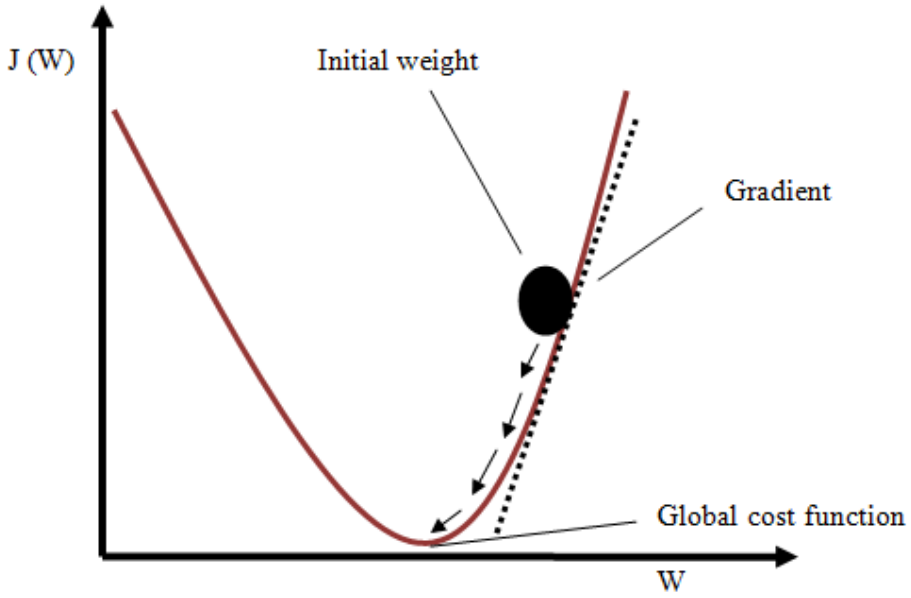



Figure 4.4. A sketch of cost function [125].

The method of gradient descent comprises the calculation of derivative of squared error function with respect to network weights. This is achieved using backpropagation. Squared error function is expressed (considering an output neuron) as:

$$E = \frac{1}{2}\left(0_t - y_M\right)^2 \tag{4.14}$$

where $E$ and $o_t$ are squared error and expected output respectively, $y_M$ is actual output and ½ cancels the exponent during differentiating. For each neuron $i$, the output $o_k$ is defined as:

$$o_k = \varphi\left(net_i\right) = \varphi\left(\sum_{k=1}^{n} w_{ik} o_k\right) \tag{4.15}$$

The input $net_i$ is the weighted sum $o_k$ of output preceding neurons, if $net_i$ is in first hidden layer, then $o_k$ is inputs $x_i$ to the network. $n$ is the neuron input units number, $w_{ik}$ is the weights between $i$ and $k$ neurons and φ is the activation function which is non-linear and differentiable such as logistic function [126]:

$$\varphi\left(z\right) = \frac{1}{1+e^{-z}} \tag{4.16}$$

The calculation of partial derivative of the error $E$ with respect to the weights between neurons $w_{ik}$ is performed using chain rule twice as:

$$\frac{\partial E}{\partial w_{ik}} = \frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial net_i} \frac{\partial net_i}{\partial w_{ik}} \tag{4.17}$$

where a single term in the sum $net_i$ is dependent on $w_{ik}$, therefore:

$$\frac{\partial net_i}{\partial w_{ik}} = \frac{\partial}{\partial w_{ik}} \left(\sum_{k=1}^{n} w_{ik} o_k\right) = \frac{\partial}{\partial w_{ik}} w_{ik} o_i = o_i \tag{4.18}$$

$o_i = x_i$ for first layer neurons.

Assuming the use of a logistic function, output derivative of neuron $k$ w.r.t.to input is described as:

$$\frac{\partial o_k}{\partial net_i} = \frac{\partial}{\partial net_i} \varphi\left(net_i\right)\left(1-\varphi\left(net_i\right)\right) \tag{4.19}$$

For this purpose, backpropagation requires an activation function to be differentiable. For output layer neuron, then the evaluation is straightforward as:

$$o_k = y_M \text{ and } \frac{\partial E}{\partial o_k} = \frac{\partial E}{\partial y_M}\frac{\partial}{\partial y_M} = \frac{1}{2}\left(o_t - y_M\right)^2 = y_M - o_t \tag{4.20}$$

However, if $k$ is inner layer of the network, it is less obvious finding the derivative $E$ with respect to $o_k$. In view of $E$ being function of inputs of all neurons $N=p, q,\ldots r$ receiving input from the neuron $k$, then:

$$\frac{\partial E\left(o_k\right)}{\partial o_k} = \frac{\partial E\left(net_p, net_q, \ldots . net_r\right)}{\partial o_k} \tag{4.21}$$

Taking the sum derivative in light of $o_k$, recursive example is obtained for the derivative as [118]:

$$\frac{\partial E}{\partial o_k} = \sum_{i\in N}\left(\frac{\partial E}{\partial net_i}\frac{\partial net_i}{\partial o_k}\right) = \sum_{I\in N}\left(\frac{\partial E}{\partial o_i}\frac{\partial o_i}{\partial net_i}w_{ik}\right) \tag{4.22}$$

This can be a calculation of derivative w.r.t.to $o_k$ if every derivative with respect to the next layer output $o_i$, the derivative nearer to the output neuron is known. Their summation gives [124]:

$$\frac{\partial E}{\partial w_{ik}} o_i \delta_k \tag{4.23}$$

$$\delta_k = \frac{\partial E}{\partial o_k}\frac{\partial o_k}{\partial net_i}\begin{cases}\left(o_k - o_t\right)o_k\left(1-o_k\right) & for\ k = output\ neuron \\ \sum_{\iota\in N}\delta_\iota w_{ik})\,\mathrm{o}_k\left(1-o_k\right) & for\ k = inner\ neuron\end{cases}$$

However, gradient descent algorithm with backpropagation is not a guarantee to achieving the global minimum, it merely guarantees a local minimum and display problem of crossing plateau in the scenery of error function [125]. These problems of non-convergence of error function might limits the performance of gradient descent with backpropagation including its non-requirement of input vector normalization because normalization improves network performance [127, 128].

## 4.3. Learning Algorithms

The key attribute of artificial neural network is their ability to get familiar to problems by means of training and thereafter be able to solve an unknown problem of similar class [125]. Learning rules are known as algorithms. There are various algorithms, so to select the appropriate learning algorithm for ANN training is critical as the selection of the learning algorithm is dependent on many factors. The aim of training (using different ANN algorithms) is to establish a weight combination that gives the smallest error.

A neural network learns from different phenomenon and a learning system changes to adapt to the situation e.g. environmental changes [79]. It certainly requires initial questions such as: where the learning input will come from and in what form, how to modify weights to ensure fast and reliable learning, how to measure objectively the success of learning and determine the best learning procedure, how to ensure that a learning procedure reaches optimal state, how to store the learned pattern in a network etc [23, 129]. They are used in finding appropriate weights or other required network parameters.

The algorithms that have been used in training the ANN comprise of both first order and second order approximation algorithms. The algorithm with the least prediction error has been studied and employed for further ANN training in this research work. The basic objective of training pattern is for global error reduction by weight and bias adjustments. Some of the examined training functions under different training algorithms are described. They have been re-examined to ascertain their prediction capabilities and to check their correlation with what is obtainable in literature.

### *4.3.1. Gradient descent algorithm*

This training algorithm updates weight and bias values in the negative direction of gradient for performance function [130].The different gradient descent algorithm training functions employed are:

i. Gradient descent*(traingd)*- The training function updates values of weight and bias in the direction of negative gradient of the performance function, shows instability with large learning rate and delay in convergence with small learning rate [130].

ii. Gradient descent with momentum *(traingdm)*- Momentum makes ANN network ignore small feature in error and slide through small local minimum. It trains network as long as the weight, net input, and transfer functions contains derivative functions [131].
iii. Gradient descent with adaptive learning *(traingda)*- This is the optimization of the method of gradient descent with the aim of minimizing network error using gradient of function and network parameters [68].

### *4.3.2. Conjugate gradient algorithm*

The search direction in conjugate gradient algorithms is carried out along the conjugate direction for fast convergence [132]. Standard re-set point occurs when the number of weights and biases are equal. However, the efficiency of training is improved using different re-set methods. The various conjugate gradient algorithm training functions used are:

i. Scaled conjugate gradient algorithm *(trainscg)*- Scaled conjugate gradient algorithm eliminates the line search at every learning iteration using step size scaling mechanism [133, 134]. The direction of new search is described as:

$$E_k^{''} = \frac{E^{'}\left(w_k + \sigma_k p_k\right) - E^{'}\left(w_k\right)}{\sigma_k} + \lambda_k \rho_k \tag{4.24}$$

ii. Conjugate gradient with Powell-Beale restarts *(traincgb)*- This method of rest uses the technique of restarting in case of little orthogonality between previous and current gradient [132].
iii. Conjugate gradient with Fletcher-Reeves updates *(traincgf)*- This training function is the ratio of the norm squared of present gradient with respect to the norm squared of previous gradient [131].
iv. Conjugate gradient with Polak-Ribiére updates *(traincgp)*- This training function is the inner product of previous change in gradient with present gradient divided by the norm squared of previous gradient [135].

### *4.3.3. Resilient backpropagation algorithm (trainrp)*

This is a first order backpropagation algorithm that eliminates the negative effects of magnitude of partial derivative. The learning rate is given as [136]:

$$\begin{cases} \eta^{+} * \Delta_{ij}^{t-1}, & if \ \frac{\delta E^{t}}{\delta w_{ij}} * \frac{\delta E^{t}}{\delta w_{ij}} > o \\ \eta^{-} * \Delta_{ij}^{t-1}, & if \ \frac{\delta E^{t}}{\delta w_{ij}} * \frac{\delta E^{t}}{\delta w_{ij}} < 0 \\ \Delta_{ij}^{t-1} & \end{cases} \tag{4.25}$$

where $0<\eta^{-}<1<\eta^{+}$. $w_{ij}$ and $\Delta_{ij}$ are the weight and change in weight, respectively, $\eta$ and $E$ are the learning rate factor and the partial derivative of error function respectively.

### *4.3.4. Quasi-Newton algorithm*

Quasi-Newton algorithm (based on Newton's method) does not require the calculation of second derivative, and they update an approximate Hessian matrix in every iteration of the algorithm [65]. This algorithm is used for better and fast optimization and approximates the inverse Hessian by a different matrix G, applying the first partial derivatives of loss function [137, 138]. The algorithm is defined as:

$$W_{i+1} = W_i - (G_i.g_i).\eta_i = 0,1,.... \tag{4.26}$$

where $\eta$ and $G$ are the learning rate and inverse Hessian approximation respectively. The applied Quasi-Newton training function is described as Broyden–Fletcher–Goldfarb–Shanno (BFGS) *(trainbfg)*. The training function updates weights and biases according to Newton's method, a group of optimization method that searches for a stationary point of function. A required condition for optimality is at zero gradient. The algorithm converges in less iteration; however, it requires more computation and storage than using the conjugate gradient method. It has effective training function for smaller network and good performance for smooth optimization [139].

### *4.3.5. Levenberg-Marquardt algorithm (trainlm)*

Levenberg-Marquardt method was designed to enhance the second order training speed without the need of calculating or approximating the Hessian matrix as in Newton algorithm or Quasi-Newton algorithm [140]. It is an iterative method that ensures the reduction of performance function in each iteration. Because of this feature, it is a fast training algorithm for moderate size networks. However, because of gradient and approximated Hessian matrix calculations, it has a problem of memory and computational overhead. The parameter improvement using Levenberg-Marquardt algorithm is updated as [141]:

$$W_{i+1} = W_i - \left( J_i^T . J_i + \lambda_i I \right)^{-1} . \left( 2 J_i^T . e_i \right), i = 0,1... \tag{4.27}$$

where $\lambda$ and $I$ are the damping factor and the identity matrix, $e$ and $J$ are the vector of error terms and the Jacobian matrix, respectively.

### *4.3.6. Bayesian Regularization algorithm (trainbr)*

Bayesian regularization minimizes squared errors and weight combination and determines accurate combination to ensure a generalize network [142]. This algorithm updates the values of weight and bias according to Levenberg-Marquardt optimization. It reduces the squared error and weights and determines the right combination to produce a well generalized network. The modified performance function is defined as [143]:

$$F_{mp} = \beta SSE + \alpha SSW \tag{4.28}$$

where $SSE = \sum_{q=1}^{n} e_q^{\,n}(x)$, $SSW = \sum_{j=1}^{n} w_j^{\,n}$, $n$, α and β are the total number of weights and biases $w_j$ in the network, training rate and decay rate, respectively.

## Learning by Bayesian Regularization (BR) backpropagation algorithm

Bayesian Regularization (BR) backpropagation algorithm updates weight and bias variables in accordance with Levenberg-Marquardt (LM) optimization [144, 145]. The algorithm determines the right combination to give a proper generalized network by

minimizing linear permutation of squared error and weight variables. It adjusts the linear combination to ensure network with good generalization qualities at the end of network training. The algorithm occurs within the LM algorithm and makes use of Jacobian for computations. However, Jacobian assumes a performance that is mean otherwise sum of squared errors, therefore network trained employing Bayesian Regularization ought to use Mean Squared Error (MSE) or Sum of Squared Error (SSE) performance function. Jacobian is computed using backpropagation and all the variables modified in accordance with a simple and robust LM algorithm function approximation technique as:

$$\left(J^t J + \lambda I\right)\delta = J^t E \tag{4.29}$$

where $J$, $\delta$ and $E$ are Jacobian matrix, unknown weight update vector, and error, $\lambda$ and $J^tJ$ are the damping factor of Levenberg and the approximated Hessian, respectively. The modification of the damping factor for process optimization is at all iteration.

The first order partial derivative matrix of Jacobian is established by taking all output partial derivatives of all the weights as expressed [146]:

$$J = \begin{pmatrix} \frac{\partial F\left(x_i, w\right)}{\partial w_j} & \cdots & \frac{\partial F\left(x_i, w\right)}{\partial w_W} \\ \vdots & \ddots & \vdots \\ \frac{\partial F\left(x_n, w\right)}{\partial w_i} & \cdots & \frac{\partial F\left(w_n, w\right)}{\partial w_W} \end{pmatrix} \tag{4.30}$$

$F(x_i,w)$ is the network function, $w_j$ is $j^{th}$ of the weight vector $w$ of the network.

The Hessian in general does not require to be computed for least square problems, instead an approximation can be done using Jacobian matrix as:

$$H \approx J^t J \tag{4.31}$$

However, this is an excellent Hessian approximation with small solution residual error, without which the approach may give rise to slow convergence with residual error that is not sufficiently small.

The LM algorithm is extremely receptive to the initial weights and the data outliers are not considered, thus may result to poor generalization. Also, LM algorithm enormously depends on parameters preliminary guess, based on network's preliminary weight values, there may be a convergence of the algorithm at the local minimal or no convergence at all.

To prevent poor generalization, a technique known as regularization is employed. Bayesian regularization framework allows the approximation of the effective number of network weights that are really required in solving a specific problem [144]. There is expansion of cost function to search for smallest error making use of the smallest weights. This works by the introduction of two Bayesian hyper-parameters: (i) alpha and (ii) beta, to notify the direction the learning process must seek (minimal error or minimal weights). The cost function becomes [146]:

$$F = \beta E_d + \alpha E_s \tag{4.32}$$

where $E_d$ and $E_s$ are sum of squared error and sum of squared weights, respectively.

By adding BR to LM, small overhead is added up to the process as a Hessian approximation already existing.

## 4.4. Performance Metrics

The Mean Absolute Error, Root Mean Squared Error, Standard Deviation, and Correlation Coefficient have been steadily used in model assessment and research studies [147]. The RMSE is a measure of how far residuals (prediction data) are from the regression data points i.e. extent the residual data concentrate just about the line of best fit. It measures the difference between measured data values and predicted values by the model [148]. The MAE is a measure of the average magnitude of error in a prediction set. It averages the test samples of the differences between measured data values and prediction data values without the consideration of their directions. Standard deviation measures the dispersion of data set from its mean. It is used to calculate the amount of dispersion of data set values. Data points that are further from the mean show a high deviation within the set of data while data points closer to the mean indicates a low standard deviation [149]. Correlation coefficient measure

the linear relationship in terms of strength and direction between measured data values and predicted data values. It ranges from *+1* to *-1*, which indicates a positive or negative correlation coefficient. A correlation coefficient of *+1* indicates a relationship with a perfect positive fit while a correlation of *-1* indicates a relationship with a perfect negative fit [150]. In summary, the computation of the performance metrics are given as:

The RMSE measures the differnce between measured data and predicted data values [148]:

$$RMSE = \sqrt{\frac{1}{N_{\exp}} \sum_{d=1}^{N_{\exp}} \left[ l(d) - y_0(d) \right]^2} \tag{4.33}$$

The MAE measures closeness of predicted values to measured data values[147]:

$$MAE = \frac{1}{N_{\exp}} \sum_{d=1}^{N_{\exp}} \left[ l(d) - y_0(d) \right] \tag{4.34}$$

The SD measures the amount of variation between measured data and predicted data values. Low standard deviation shows closeness of the data points while high standard deviation shows the reverse [151].

$$SD = \sqrt{\frac{1}{N_{\exp}} \left( \sum_{d=1}^{N_{\exp}} | l_d - y_d | - \mathrm{MAE} \right)^2} \tag{4.35}$$

The Correlation Coeffient (r) measures the statistical relationship between measured data and predicted data values. It returns a value between -1 and 1. +1 signifying a strong positive connection and -1 signifying a negative connection between measured and prediction data values [152].

$$r = \frac{N_{\exp} \sum l(p) - y_0(p) - \left( \sum l(p) \left( \sum y_0 \right) \right)}{\sqrt{N_{\exp} \left( \sum l(p) \right)^2} - \left( \sum l(p) \right)^2 \sqrt{N_{\exp} \left( \sum y_0(p) \right)^2}} \tag{4.36}$$

where $N_{exp}$ and $l(p)$ are the number of the measured data and values of measured signal power respectively, $p^{th}$ is the input pattern and $y_0(p)$ is the neural network output.

## 4.5. Results and Discussions

The simulation results obtained using different ANN learning algorithms in training a MLP ANN are discussed.

### *4.5.1. Assessment of the Performance of Different Artificial Nural Network Training Algorithms*

Real world measured data was collected via a drive test during JAN-DEC 2017 from a sub-urban area that has moderate commercial bulidings and foliages. Data measurement setup are: a laptop and two Samsung Galaxy Mobile Handsets (Model-SY *4*) installed with Test Mobile System (TEMS, *15.1* version) software, network scanner and other acessories such as Global Positioning System (GPS), power inverters, test cables,compass, digital map of the area and a vehicle for the test drive. Drive test started from choosen Base Station (BS) transmitting at *1900 MHz* operating frequency while capturing the signal power at various distances from the LTE cellular network. The training set considered are *1,970* measured signal power across different distances from the BS.

A MATLAB program for one hidden layer MLP ANN model is written with measured data and serves as inputs to the ANN. For the learning process, *90%* of data are used for training while remaining *10%* are used for validation. Training of MLP ANN are done using *five* training algorithms comprising of *nine* training functions which are run in ANN training tool box *(nntraintool)* in MATLAB *2013a* and the performance of the training algorithms are assessed in terms of RMSE, MAE, SD and r.

The processes of training are made more efficient and the speed increased by normalizing the input and desired output values to lie about *zero* mean and *unity* standard deviation. The measured data is normalized using excel spread sheet with the expression [153]:

$$v_n = \frac{v_i - v_{\min}}{v_{\max} - v_{\min}} \tag{4.37}$$

where $v_n$, and $v_i$, are the normalize value and the initial parameter value, $v_{min}$ is the minimum parameter value and $v_{max}$ is the maximum parameter value, respectively.

The task of an accurate assessment of the different training algorithms that mostly predict the signal power with minimal error has been a demanding task as a result of different factors such as complexity of environment of data collection, number of training data, quality of training data, network weights and biases etc.

Sigmoid transfer function is used in the MLP ANN hidden layer and the basic training parameters are fixed for all the training functions. The ANN network has been trained with each of the training function for an average of *ten* runs to ensure the network learns the pattern that signal propagates after several passes on training set. The training values with the least errors are taken for each of training function. The parameters for comparison are the number of epochs at the end of training (E), time of training, regression on training, regression on validation, RMSE, MAE, SD and r. All these parameters are checked for *10*, *20* and *30* neuron numbers in the hidden layer. The results are grouped in three different tables for three different algorithms used.

Table 4.1. Comparison of *3* training functions of Gradient descent algorithm

| **Training functions** | **Parameters for comparison** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | No of neurons | Epoch (E) (1000) | Training time (s) | RMSE | MAE | SD | R | Regression on training | Regression on validation |
| **traingd** | 10 | 6 | 0 | 17.3494 | 13.5048 | 10.8914 | 0.4202 | 0.4429 | 0.2108 |
| | 20 | 6 | 0 | 14.4900 | 12.2472 | 7.74900 | 0.4570 | 0.4840 | 0.3444 |
| | 30 | 6 | 0 | 18.0274 | 14.1806 | 11.1309 | 0.3129 | 0.3372 | 0.1504 |
| **traingdm** | 10 | 6 | 0 | 15.9239 | 13.1845 | 8.9295 | 0.5042 | 0.5060 | 0.4756 |
| | 20 | 6 | 0 | 14.0825 | 11.4637 | 8.1792 | 0.4631 | 0.5155 | 0.4517 |
| | 30 | 6 | 0 | 15.2804 | 12.1649 | 9.2470 | 0.4561 | 0.4381 | 0.5850 |
| **traingda** | 10 | 125 | 0 | 3.2579 | 2.5661 | 2.0072 | 0.9046 | 0.9161 | 0.8928 |
| | 20 | 207 | 0 | 3.4851 | 2.7775 | 2.1051 | 0.8911 | 0.9019 | 0.8630 |
| | 30 | 353 | 0 | 3.9701 | 3.0643 | 2.5241 | 0.8576 | 0.8566 | 0.8387 |

Table 4.2. Comparison of *3* training functions of Conjugate gradient algorithm

| **Training functions** | **Parameters for comparison** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | No of neurons | Epoch (E) (1000) | Training time (s) | RMSE | MAE | SD | r | Regression on training | Regression on validation |
| **traincgb** | 10 | 36 | 0 | 3.0902 | 2.4603 | 1.8699 | 0.9114 | 0.8965 | 0.9525 |
| | 20 | 27 | 0 | 2.9898 | 2.3553 | 1.8416 | 0.9171 | 0.9109 | 0.9292 |
| | 30 | 117 | 0 | 2.7519 | 2.0341 | 1.8535 | 0.9309 | 0.9466 | 0.9240 |
| **traincgf** | 10 | 23 | 0 | 3.0866 | 2.4662 | 1.8560 | 0.9114 | 0.9065 | 0.9023 |
| | 20 | 96 | 0 | 2.9332 | 2.2853 | 1.8387 | 0.9204 | 0.9427 | 0.8832 |
| | 30 | 41 | 0 | 3.0849 | 2.4543 | 1.8190 | 0.0147 | 0.9255 | 0.7962 |
| **traincgp** | 10 | 31 | 0 | 3.0984 | 2.4400 | 1.9096 | 0.9112 | 0.9272 | 0.9033 |
| | 20 | 35 | 0 | 2.9806 | 2.3663 | 1.8124 | 0.9278 | 0.9105 | 0.9436 |
| | 30 | 80 | 0 | 2.9948 | 2.2799 | 1.9418 | 0.9177 | 0.9366 | 0.8829 |

Table 4.3. Comparison of Quasi-Newton, Levenberg-Marquardt and Bayesian regularization algorithms

| **Training functions** | **Parameters for comparison** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | No of neurons | Epoch (E) (1000) | Training time (s) | RMSE | MAE | SD | R | Regression on training | Regression on validation |
| **trainbfg** | 10 | 24 | 0 | 3.0694 | 2.4610 | 1.8342 | 0.9132 | 0.9171 | 0.9373 |
| | 20 | 722 | 0.5 | 3.0255 | 2.3980 | 1.8447 | 0.9160 | 0.9275 | 0.9414 |
| | 30 | 950 | 0.8 | 3.2859 | 2.5207 | 2.1079 | 0.9009 | 0.9260 | 0.8214 |
| **trainlm** | 10 | 28 | 0 | 2.9920 | 2.3014 | 1.9120 | 0.9173 | 0.9180 | 0.9354 |
| | 20 | 28 | 0 | 2.6044 | 2.0031 | 1.6645 | 0.9384 | 0.9426 | 0.9399 |
| | 30 | 14 | 0 | 2.7626 | 2.0790 | 1.8193 | 0.9305 | 0.9428 | 0.9366 |
| **trainbr** | 10 | 1000 | 22 | 3.0485 | 2.3687 | 1.9191 | 0.9146 | 0.9272 | 0.8881 |
| | 20 | 1000 | 22 | 2.4904 | 1.8546 | 1.6620 | 0.9435 | 0.9465 | 0.9381 |
| | 30 | 1000 | 22 | 2.3928 | 1.8002 | 1.5764 | 0.9477 | 0.9596 | 0.8946 |

The results of the MLP ANN training using *nine* different training functions from *five* different training algorithms are shown in Tables 4.1, 4.2 and 4.3. The input to the ANN is signal power over range of distances. The performance of the *nine* training functions are measured in terms of RMSE, SD, MAE, and r. Regression on training and regression on validation have also been examined. There is variation of the neuron number in MLP hidden layer from *10, 20* and *30* during the training of network to ascertain their impact in network predictive abilities.

From Table 4.3, using BR algorithm (trainbr) gives minimal errors in comparison with other training algorithms considered. However, it requires more training epoch and time to converge. The LM algorithm (trainlm) converges with less iteration compared to BR algorithm (trainbr) and within lesser time. However, the two algorithms have slight differences in their prediction errors with LM algorithm showing higher error than the BR algorithm. In terms of speed, LM algorithm (trainlm) demonstrates the fastest prediction algorithm with minimal error while BR algorithm (trainbr) demonstrates the most accurate prediction algorithm with minimal error. Effort to train the ANN for further error reduction results in network over-fitting for all training functions.

This is in-line with previous studies carried out on comparison of backpropagation algorithm in ANN based identification of power system [154],comparison of neural network training functions for Hematoma classification in brain Computed Tomography (CT) images [131],and in [155] where *trainbr* showed more accurate predictive abilities in a comparative empirical study on social data. However, relative studies of the performance abilities of different ANN training functions in prediction of signal power loss as electromagnetic signal propagates using measured data from LTE cellular network is analysed in this research work.

In terms of neuron variation in the hidden layer, two gradient descent algorithm training functions show better prediction abilities with *20* neurons in the hidden layer. Only gradient descent with adaptive learning rate *(traingda)* shows better performance with *10* neurons in the hidden layer. Further, increase in the number of neurons leads to increase in prediction error.

The Conjugate gradient algorithm training functions demonstrate better predictive abilities with *20* neurons in the hidden layer while the Quasi-Newton algorithm training functions also gives better prediction with *20* neurons in the hidden layer. The BR algorithm *(trainbr)* gives least error with *30* neurons in the hidden layer showing its ability to train a complex network.

The graphical representations showing the prediction abilities of best performed training function in terms of accuracy with minimal errors and worst performed training function with the highest prediction errors are shown. figure 4.5 and figure 4.6 show the

prediction performance of the lowest and the highest number of hidden layer neurons for the BR algorithm and figure 4.7 shows the prediction performance of gradient descent algorithm with *30* neurons in the hidden layer.

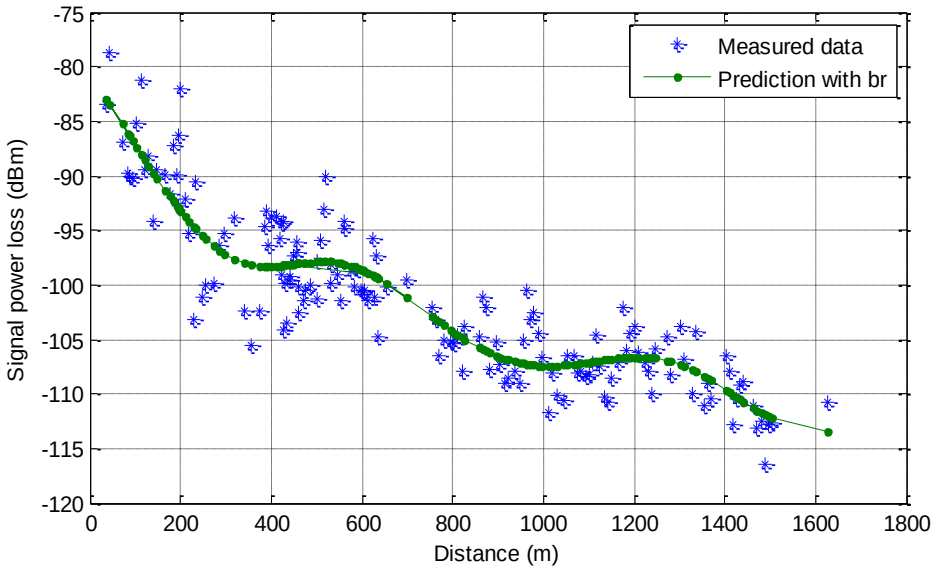



Figure 4.5. Training performance of Bayesian regularization algorithm with 10 neurons in hidden layer

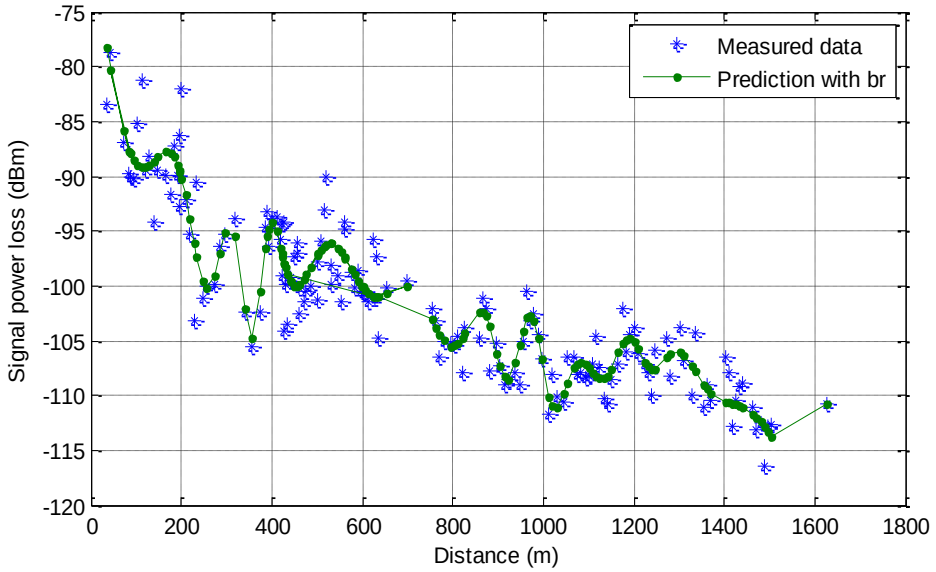



Figure 4.6. Training performance of Bayesian regularization algorithm with 30 neurons in hidden layer

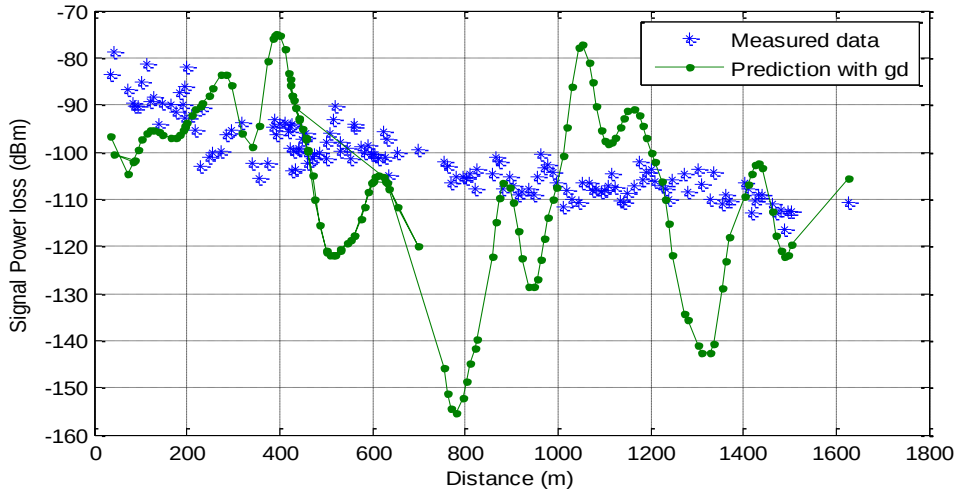



Figure 4.7. Training performance of Gradient descent algorithm with 30 neurons in hidden layer

The regression analysis compares MLP ANN actual outputs and equivalent desired output i.e. the target. Figures 4.8 and 4.9 show regression analysis for worst and best training functions i.e. gradient descent *(traingd)* and BR *(trainbr)* algorithms. The regression analysis returns the correlation coefficient between measured and prediction data, and slope and intercept of the equation with the best linear fit.

Figures 4.5 to 4.7 demonstrate a decrease in signal power as distance increases i.e. the fall in the power density of an electromagnetic signal as it transmits in the space [13]. This is because of refraction, reflection, scattering etc. as it is particular to different environment.

### *4.5.2. Performance Analysis of Different MLP-ANN Training Functions*

Real world measured data from LTE cellular network in a built-up urban area with with clusters of residential, heavy industrail and commercial buildings are collected via a drive test. The area has a triopical wet climate characterize by heavy lengthy rainy season and short dry season.The data measurement setup are a laptop and *two* Samsung Galaxy mobile handsets (Model-SY 4) installed with TEMS software (*15.1* version), network scanner, digital map of the area and other assessories as employed for data collection in part I. All measurements are conducted at different sectors of the BS which transmits at *1900 MHz* frequency band.

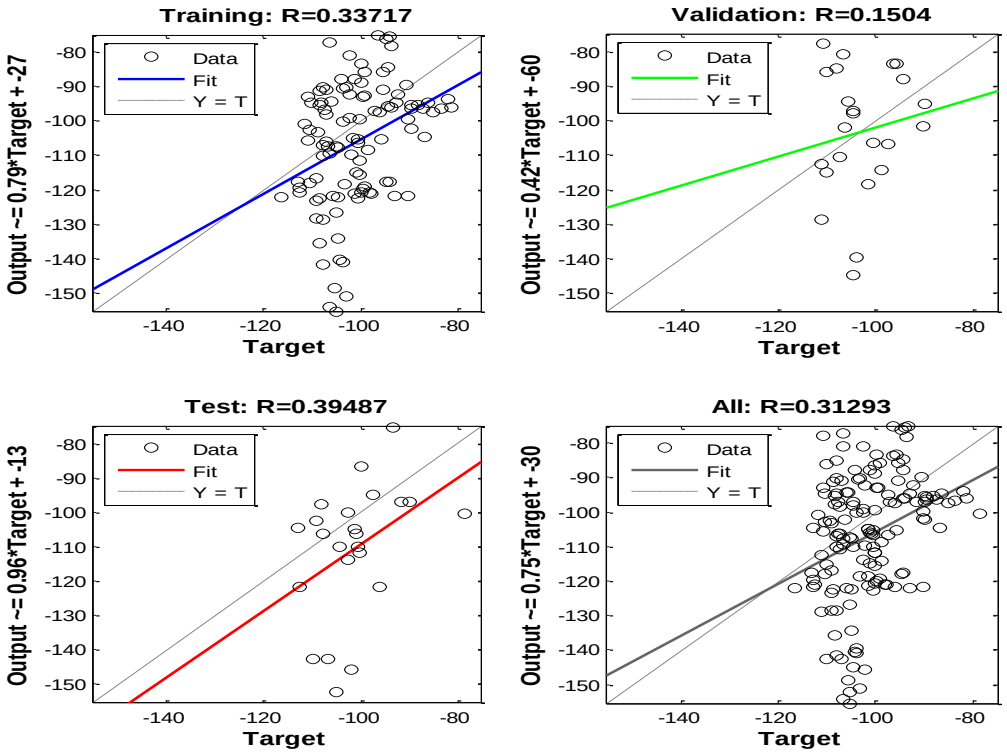



Figure 4.8. Regression analysis for Gradient descent algorithm with 30 neurons in hidden layer

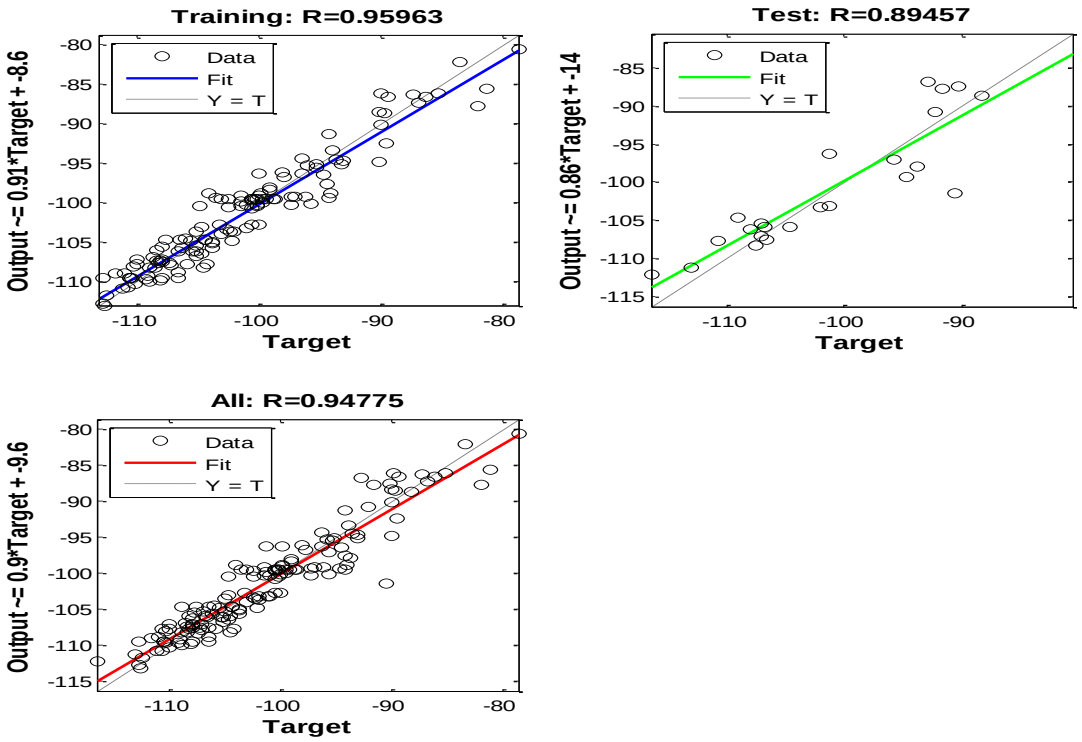



Figure 4.9. Regression analysis for Bayesian regularization algorithm with 30 neurons in the hidden layer

The signal power computed at various range of distances are used as an input to MLP ANN. A total of *1,051* measured signal power over range of distances are recoreded and used as input to MLP ANN. *100%* of measured data has been used as training set and *ten* different training functions from various algorithms applied in network training using ANN training tool box *(nntraintool)* in MATLAB *2013a*. The performances of *ten* different training algorithms used for network training are analysed in terms of RMSE, MAE, SD and r. The measured data are normalized using excel spread sheet as described in Eqn. (4.37) of part I.

The basic training parameters are fixed for all training functions and sigmoid transfer function used in hidden layer of MLP ANN. Training of network is repeatedly carried out for *three* runs and an average result considered. The list of considered training functions and the training results are shown.

The overall results of Table 4.5 and Table 4.6 show that training with BR algorithm *(trainbr)* give a minimal RMSE of *2.5664*, MAE of *1.9360*, SD of *1.6667*, and the highest *r* of *0.9398*. The LM algorithm *(trainlm)* gives RMSE of *2.7532*, MAE of *2.0954*, SD of *1.7851* and r of *0.9303*. Training with gradient descent algorithm *(traingd)* gives the highest error with RMSE of *24.0963*, MAE of *21.8294*, SD error of *10.1327* and the least *r* of *0.6099*. This validates the performances of training functions of various training algorithms assessed in part I. The training performance of the BR, LM and gradient descent algorithms measured in terms of mean squared error (MSE) are shown in figure 4.10, figure 4.11 and figure 4.12 respectively.

Table 4.4. List of assesed training functions

| S/NO | Algorithm Abbreviation | Algorithm | Description |
|---|---|---|---|
| 1 | BFG | trainbfg | BFG-Quasi-Newton |
| 2 | RP | trainrp | Resilient Back propagation |
| 3 | LM | trainlm | Laverberg-Marquardt |
| 4 | SCG | trainscg | Scaled Conjugate Gradient |
| 5 | BR | trainbr | Bayesian Regularization |
| 6 | GD | traingd | Gradient descent |
| 7 | OSS | trainoss | One step secant |
| 8 | GDX | traingdx | Variable learning Rate Back propagation |
| 9 | CGF | traincgf | Fletcher-Powell Conjugate Gradient |
| 10 | CGP | traincgp | Polak-Ribiere Conjugate Gradient |

Table 4.5. Performance results with five training functions

| Performance metrics | Error measurement | Training algorithms | | | | |
|---|---|---|---|---|---|---|
| | | trainbfg | trainrp | trainlm | trainscg | trainbr |
| **RMSE** | Run 1 | 3.0885 | 3.0929 | 2.7424 | 3.2706 | 2.5004 |
| | Run 2 | 3.0537 | 3.2578 | 2.7542 | 3.1921 | 2.6319 |
| | Run3 | 3.0847 | 3.3375 | 2.7630 | 3.1664 | 2.5669 |
| | Average | 3.0756 | 3.2294 | 2.7532 | 3.2097 | 2.5664 |
| **MAE** | Run 1 | 2.4492 | 2.5040 | 2.0892 | 2.5978 | 1.8940 |
| | Run 2 | 2.3539 | 2.6015 | 2.0582 | 2.5543 | 1.9852 |
| | Run 3 | 2.4651 | 2.6296 | 2.1389 | 2.5159 | 1.9288 |
| | Average | 2.4227 | 2.5784 | 2.0954 | 2.5560 | 1.9360 |
| **SD** | Run 1 | 1.8814 | 1.8156 | 1.7765 | 1.9871 | 1.6324 |
| | Run2 | 1.9453 | 1.9610 | 1.8300 | 1.9144 | 1.7280 |
| | Run 3 | 1.8545 | 2.0552 | 1.7490 | 1.9226 | 1.6398 |
| | Average | 1.8937 | 1.9439 | 1.7851 | 1.9414 | 1.6667 |
| **r** | Run 1 | 0.9119 | 0.9110 | 0.9307 | 0.9005 | 0.9429 |
| | Run 2 | 0.9138 | 0.9017 | 0.9303 | 0.9056 | 0.9369 |
| | Run 3 | 0.9117 | 0.8976 | 0.9301 | 0.9066 | 0.9398 |
| | Average | 0.9126 | 0.9034 | 0.9303 | 0.9042 | 0.9398 |

Table 4.6. Performance results with five training functions

| Performance metrics | Error measurement | Training algorithms | | | | |
|---|---|---|---|---|---|---|
| | | trainbfg | trainrp | trainlm | Trainscg | trainbr |
| **RMSE** | Run 1 | 3.0885 | 3.0929 | 2.7424 | 3.2706 | 2.5004 |
| | Run 2 | 3.0537 | 3.2578 | 2.7542 | 3.1921 | 2.6319 |
| | Run3 | 3.0847 | 3.3375 | 2.7630 | 3.1664 | 2.5669 |
| | Average | 3.0756 | 3.2294 | 2.7532 | 3.2097 | 2.5664 |
| **MAE** | Run 1 | 2.4492 | 2.5040 | 2.0892 | 2.5978 | 1.8940 |
| | Run 2 | 2.3539 | 2.6015 | 2.0582 | 2.5543 | 1.9852 |
| | Run 3 | 2.4651 | 2.6296 | 2.1389 | 2.5159 | 1.9288 |
| | Average | 2.4227 | 2.5784 | 2.0954 | 2.5560 | 1.9360 |
| **SD** | Run 1 | | | | | |
| | Run2 | 1.8814 | 1.8156 | 1.7765 | 1.9871 | 1.6324 |
| | Run 3 | 1.9453 | 1.9610 | 1.8300 | 1.9144 | 1.7280 |
| | | 1.8545 | 2.0552 | 1.7490 | 1.9226 | 1.6398 |
| | Average | 1.8937 | 1.9439 | 1.7851 | 1.9414 | 1.6667 |
| **r** | Run 1 | 0.9119 | 0.9110 | 0.9307 | 0.9005 | 0.9429 |
| | Run 2 | 0.9138 | 0.9017 | 0.9303 | 0.9056 | 0.9369 |
| | Run 3 | 0.9117 | 0.8976 | 0.9301 | 0.9066 | 0.9398 |
| | Average | 0.9126 | 0.9034 | 0.9303 | 0.9042 | 0.9398 |

The training performances of BR, LM and gradient descent algorithms measured in terms of MSE are shown in figure 4.10, figure 4.11 and figure 4.12 respectively, with a near *zero* horizontal slope for BR algorithm which is close to the local minima where minimum error is obtained. At epoch *1000*, the best training performance has approximately *6.26* MSE which is very minimal in comparison with LM and gradient descent algorithms. The LM algorithm trained at *36* epochs and validates the training set with *13.56* MSE while gradient descent algorithm shows positive uphill slope far from the local minima and at *zero* epoch validates the training performance at 672.32 MSE.

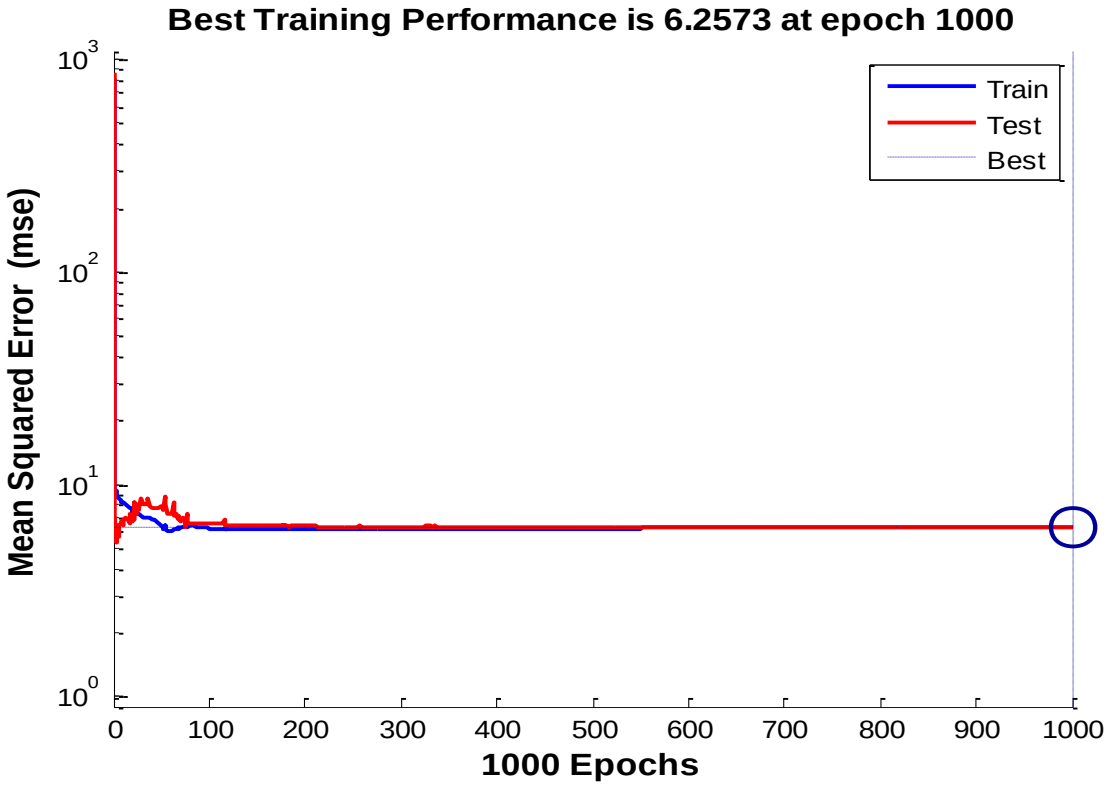



Figure 4.10. Training performance of Bayesian regularization algorithm

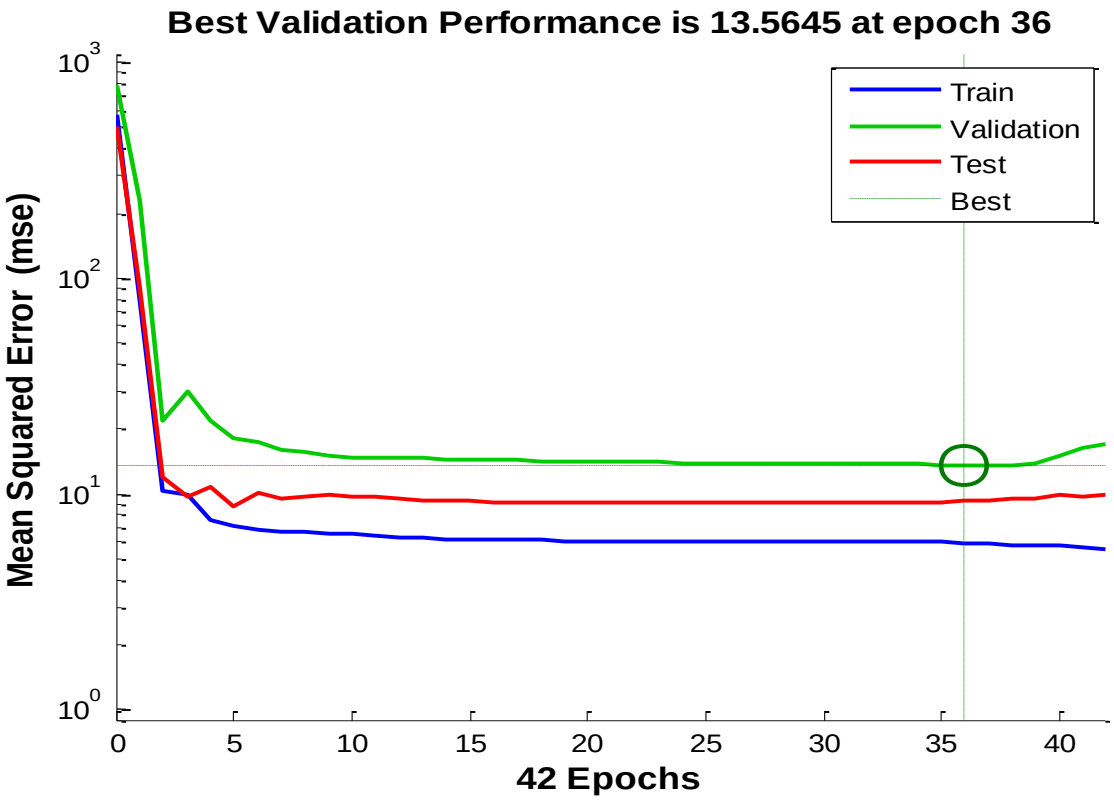



Figure 4.11. Training performance of Levenberg-Marquardt algorithm

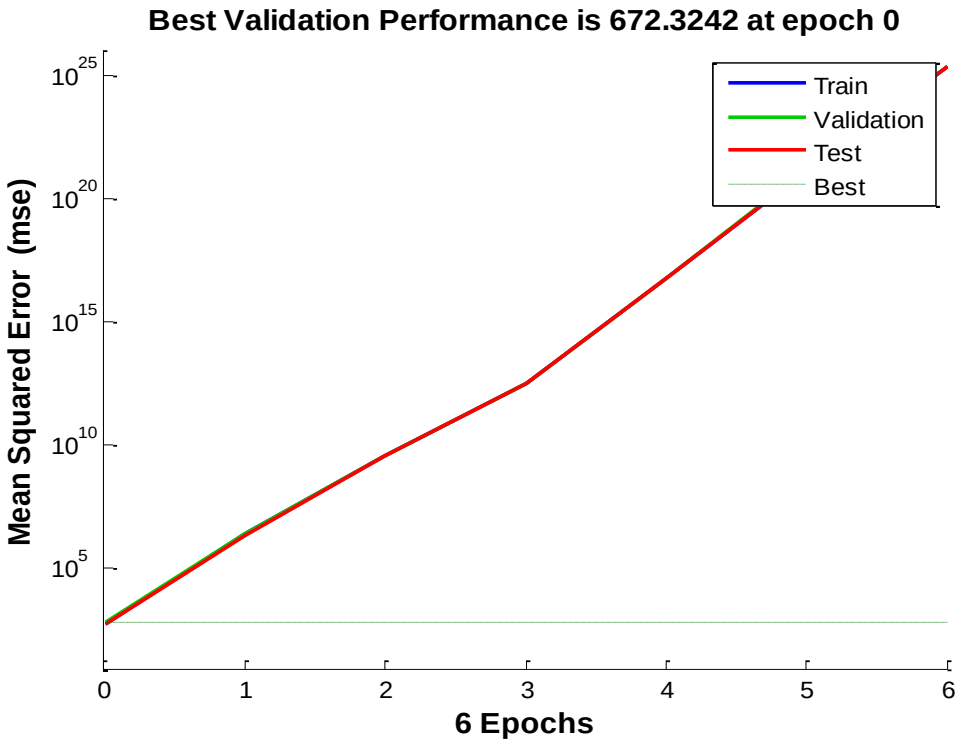



Figure 4.12. Training performance of Gradient descent algorithm

## 4.6. Chapter Summary

This chapter analyses the prediction performances of different ANN training algorithms used in training MLP ANN with real world measured data from a LTE micro-cellular sub-urban and urban areas. The architectural composition of a simple possible ANN model, perceptron network, MLP network and concept of backpropagation feedforward ANNs are discussed. The performances of the ANN algorithms have been measured in terms of RMSE, MAE, SD and r. These are statistical performance metrics applied in calculation of the error difference between actual output from the ANN model and desired output.

The chapter is divided into two parts- Part I deals with training MLP ANN with *nine* training functions of *five* different ANN algorithms using real world measured data from LTE network in a sub-urban area. The performance result of the ANN training algorithms measured in terms of RMSE, MAE, SD and r demonstrates least error prediction using BR algorithm in comparison with other ANN algorithms employed in MLP ANN training. However, the BR algorithm requires longer training time in comparison with LM algorithm which has closer prediction values. It is ascertained that in terms of accuracy in prediction, BR demonstrates a better prediction algorithm while LM demonstrates fastness in prediction i.e. in learning and prediction of the pattern the signal propagates and with considerable minimal error very close to that of the BR algorithm.

The neuron numbers in the hidden layer of MLP ANN has been varied from *10*, *20*, and *30* during network training to ascertain their effect in prediction abilities of the ANN training algorithms. Most of the algorithms give the least error with *20* neurons in MLP ANN hidden layer but gradient descent with adaptive learning rate training function gives the least error with *10* neurons in the hidden layer, thus, its ability to adapt fast during the network training in comparison with two other Gradient descent training functions that show very high prediction errors i.e. they were incapable of learning the pattern the signal used as input data to MLP ANN propagates. The BR algorithm shows better performance with *30* neurons in hidden layer which demonstrates its capability in training a complex network with expected minimal error.

Part II of the chapter employs *ten* different training functions of different ANN training algorithms to re-examine the performance of the training algorithms in prediction of signal power loss using real world data from LTE network in an urban area. Their performances are also analysed in terms of RMSE, MAE, SD and r. The result validates prediction with BR algorithm as it gives minimal prediction error of RMSE, MAE, SD and r. The closest training algorithm with prediction error closer to BR algorithm is also LM algorithm.

Lastly, the current work of this chapter is in-line with previous studies on ANN presented in [34, 60, 98-101] in terms of trend in PL, however, their prediction accuracies are different as LTE network offers data when trained, has error prediction advantage over data from previous generations of telecommunication networks.

# CHAPTER 5

# APPLICATION OF DIFFERENT ANN ARCHITECTURES IN MODELLING OF PROPAGATION PATH LOSS

The chapter's objectives centres on modelling of propagation Path Loss (PL) using three different Artificial Neural Network (ANN) architectures: (i) Adaptive Linear Element (ADALINE), (ii) Radial Basis Function (RBF), and (iii) Generalized Regression Neural Network (GRNN). The performances of each of the ANN models in prediction of signal power loss using real world measured data from different Long Term Evolution (LTE) micro-cellular environments has been analysed in comparison with the performance of Multi-Layer Perceptron (MLP) ANN.

This is to realistically ascertain the peculiarities and limitations of these models in training and prediction of signal power loss. Their prediction performances are computed in terms of Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Standard Deviation (SD) and Correlation Coefficient (r). Bayesian Regularization (BR) algorithm has been employed as the training algorithm. The different techniques for training ANN, the effect of learning rate parameter, and the generalization abilities of these ANN architectures in harmony with the underlying physical complexity of the problem of signal power loss prediction with the data sets has been analysed.

## 5.1. Introduction

Different approaches to avoid poor generalization and overcome the tendencies of over-fitting during ANN training have been analysed in RBF and MLP ANNs, the performance of ADALINE with input delay of *1:5* in comparison to MLP ANN with different variation of learning rate has been studied and the effect of learning rate in GRNN and MLP ANNs analysed. The concept of ANN learning and other parameters of learning during network training have been highlighted. The chapter is organized in three parts.

Part I analyses and presents three different approaches to avoid poor generalization and overcome the tendencies of over-fitting in RBF ANN and MLP ANN. The essence is for an enhanced prediction accuracy of signal power loss during electromagnetic signal propagation in a metropolitan area. Real world measured data from LTE cellular network are used as a training set to the ANNs. The different approaches adopted are variation of hidden layer neurons, early stopping and Bayesian regularization techniques. The data set is from a metropolitan area with a population of *one hundred and ninety thousand, eight hundred and eighty four thausand* covering an area of *138 km$^2$* and lies within the latitde *4$^0$47'31.2''N* and longitude of *7$^0$7'12''E* of the area. The area has a tropical wet climate with two major seasons: the rain and the dry season with networks of creeks that span the rivers stretching into the altlantic ocean. The vegetation is characterized by mangrove and thick forest with arable land.

The effect of learning rate parameter on the performance of two ANN architectures is examined in Part II using ADALINE and MLP network in predicting the signal power loss using measured data from LTE cellular network. Learning rate parameter is utilized to check the prediction effectiveness of the two network models while considering prediction error between actual measurement and prediction values. The gradient and momentum parameters of the two networks are also analysed at different variation of learning rate. The data set is from an Island that covers 645.*60km$^2$* with a population of *two hundred and fourteen thousand, nine hundred and eighty-three thousand* and has coordinates of *4$^o$ 26'N 7$^o$ 10'E*. It comprises territorial areas, virgin lands and water taxis.

The effect of spread factor in GRNN is explored in Part III using data set from Base Station (BS)*1* while the effect of learning rate parameter in GRNN and MLP ANNs are re-examined using data set from BS*2* in the prediction of signal power loss in a rural area with data set from LTE cellular network. The area is a low land topography with land mass of *376.5 m$^2$* and an estimated population of around *six hundred thousand* and lies between the latitude of *4.990833$^o$N* and longitude of *7.054444$^o$E* with a tropical wet climate.

### *5.1.1. Adaptive Linear Element (ADALINE)*

Adaptive linear element is a linear single layer ANN based on McCulloch Pitts neurons that makes use of linear activation function [156]. It comprises of a neuron weight, a bias and

a summation function with multiple nodes and each of the nodes takes multiple inputs and gives an output. If $x$ is an input vector to the neural network and $n$ is the number of inputs, $w$ and $\theta$ are the weight vector and a constant respectively and $y$ is the model output, then, the output of ADALINE neural network is given as [68]:

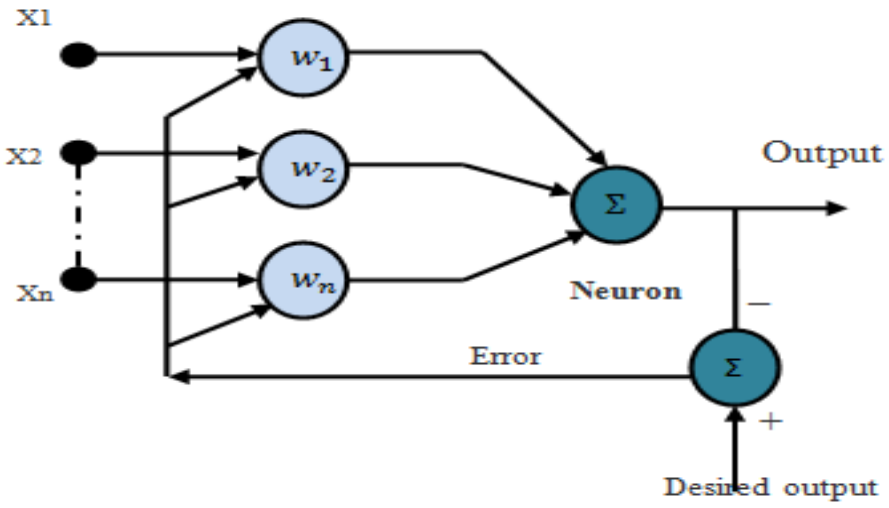



Figure 5.1. Architecture of an ADALINE [69].

$$y = \sum_{i=1}^{n} x_i w_i + \theta \tag{5.1}$$

If $x_0$ and $w_0$ are $1$ and $\theta$ respectively, then, the output will reduce to:

$$y = \sum_{i=0}^{n} x_i w_i \tag{5.2}$$

If the learning rate $\eta$ is a positive constant, $y$ and $0_T$ is the output of the model and the target output respectively, the weight is then updated as:

$$w \leftarrow w + \eta\left(o_T - y\right)x \tag{5.3}$$

The ADALINE converges at $E = \left(0_T - y\right)^2$ which is the least square error. The supervised learning of ADALINE has some similarities with the multi-layer perceptron learning algorithm. However, they have significant differences as shown in Table 5.1.

Table 5.1. Differences between ADALINE and Multi-Layer Perceptron Artificial Neural Network [114, 157, 158].

| Differences | ADALINE neural network | Multi-layer perceptron (MLP) |
|---|---|---|
| **Learning algorithm** | Based on the adjustment of the neuron weight for each weighted summation of the network inputs. The neuron in the network takes more than a single input but generates one output. | Uses perceptron rule as the learning algorithm where the neuron weights and biases are trained in a way that a correct target is produced on entering of the inputs. |
| **Checks during network training** | After every iteration, there are checks if the weight works for all the input patterns. | Weight update after every new input pattern, no further checks are carried out. |
| **Accuracy** | It gives information on the suitability of the neuron for a given set of input pattern as the weights are continuously updated. | Does not give information on the suitability of the neuron as the neurons are trained and there is no continuous update. |
| **Learning method** | Does not use derivative of transfer function. Instead, it makes use of continuous values predicted from the network input to learn model coefficient. This notifies the degree of correctness or incorrectness. | Uses derivative of transfer functions in the computation of changes in weight or class labels in the learning of model coefficient. |
| **Transfer function** | Uses linear transfer function like purelin. | Uses non-linear transfer function like logistic sigmoid or hyperbolic tangent. |
| **Network architecture** | A single layer of artificial neurons with linear decision boundaries, solves problems that are linearly separable. | Network of multiple artificial neurons over multiple layer which creates complex non-linear decision boundaries that permits solving of problems that are not linearly separable. |

### *5.1.2. Radial Basis Function (RBF) Network*

Radial basis function networks are generally made up of fixed three-layer architecture [159]. These are the input layer which have one or more predictor variables with each variable associated with a separate neuron, hidden layer which contain many numbers of neurons, and output layer. The input layer supplies the network with inputs, the input data is re-mapped by the hidden layer to ensure that they are linearly divisible, and output layer carries out the linear division [24]. This unique architecture of RBF network establishes that the design is organized in three stages: (i) finding the proper size of the network, (ii) finding the proper initial parameters, and (iii) finally training the network. Each of the hidden layer neuron has a radial basis function centred at a point. This depends on the dimensionality of the input-output predictor variables [160]. Also, each of the neuron has a weight, hidden layer values are multiplied by associated weight and transmits to the summation that sums the weighted values, thereafter hands it over to the network output.

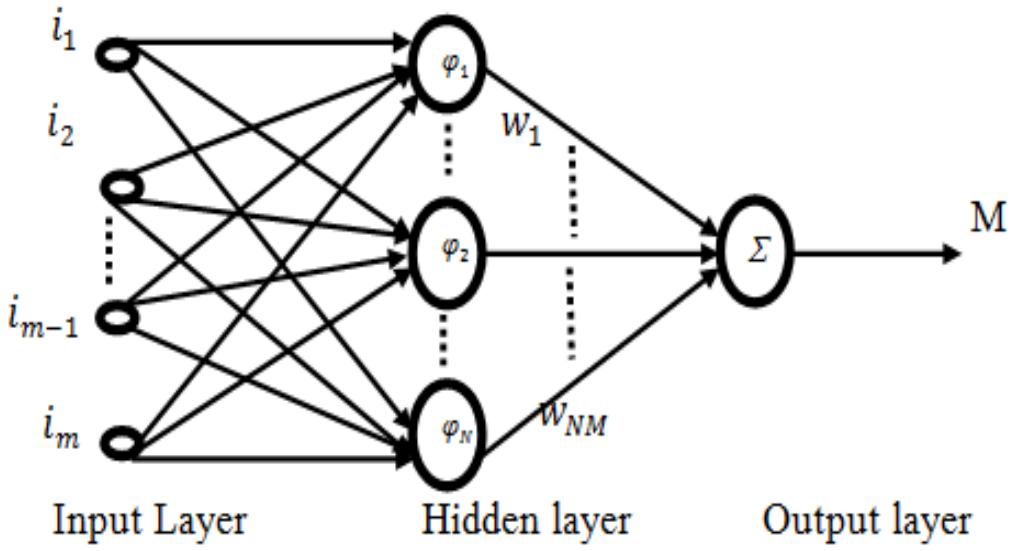



Figure 5.2 .Architecture of a Radial Basis Function Network [161].

On the input to the hidden unit *N*, input weights $w^h$ is weighted by input vector *i* [162]:

$$S_N = \left[ i_1 w_1^{\,h}, {}_N\, i_2 w_2^{\,h}, {}_N \,..i_{m-1} w^h{}_{m-1,N-1} .. i_m w^h{}_{m,N} \right] \tag{5.4}$$

where *m* and *N* are the input index and the hidden units index respectively, $i_m$ is the $m^{th}$ input, $w^h{}_{m,N}$ is the input weight between *m* and the hidden unit *N*.

The hidden unit output is computed as [68]:

$$\varphi_N \left( S_N = \exp\left[ \frac{\| \mathrm{S_N} \| - C_N \|^2}{\sigma_N} \right] \right) \tag{5.5}$$

where $\varphi_N$, *CN* and $\sigma_N$ are the activation function of hidden unit *N* (usually selected as Gaussian function), the centre of hidden unit *N* and the width of hidden unit *N*, respectively. The output unit or index *M* is given as:

$$O_M = \sum_{n=1}^{N} \varphi_N \left( S_N \right) W^0{}_{N,M} + W^0{}_{0,M} \tag{5.6}$$

where $W^0{}_{N,M}$ and $W^0{}_{0,M}$ are the output index, the output weight between hidden and output unit and the bias weight of output unit, respectively.

From the Eq.(5.6), the key parameters are: input weight matrix $W^h$, output weight matrix$W^0$, centre matrix $c$ and the vector width $\sigma$. The input weights are usually set as *1*. The weights of output can be adjusted using simple linear least square technique which also works for non-linear cases. The non-linear performance of output unit is iteratively improved by the use of linear square method [163].The output weights, the widths and the centres can be adjusted during the training process using first order gradient methods. However, they require longer convergence time with limited search ability [161].

### *5.1.3 Generalized Regression Neural Network (GRNN)*

Generalized regression neural network is a distinctive case of RBN with two hidden layers: (i) radial basis layer and (ii) unique linear layer [164]. These are known as the pattern and the summation layers. There is no specific requirement for iterative training as required by backpropagation ANN models, every arbitrary function between input and output vector is approximated, thereby estimating the function directly from training data [165, 166]. The GRNN neural network is a representation of an improvement on the technique in neural networks based on non-parametric regression. The essence is to have all training samples represent a mean to the radial basis neuron [164]. The architecture of GRNN is made up of input, pattern, summation and output layer as shown in figure 5.3.

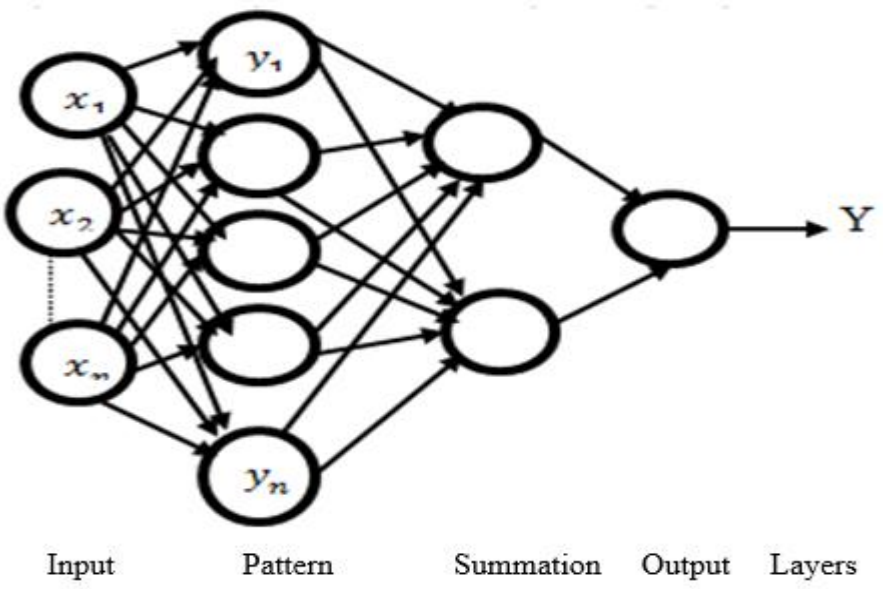



Figure 5.3. Architecture of a Generalized Regression Neural Network [98].

The neurons are fed to pattern layer through input layer where the Euclidean distance and activation function are calculated. These are fed to summation layer which is made up of two sub-divisions: (i) denominator and (ii) numerator. The numerator controls the summation of the multiplication of output training data while the denominator sums all activation functions [167]. The summation layer feeds output layer which computes network output by division of numerator of summation layer by the denominator. The GRNN also has a parameter known as "spread factor" which is input vector distance from the neuron weight [98].

$$Y(x) = \frac{\sum_{i=1}^{n} y_i e^{-\left(\frac{d_i^2}{2\sigma^2}\right)}}{\sum_{i=1}^{n} e^{-\left(\frac{d_i^2}{2\sigma^2}\right)}} \tag{5.7}$$

$$d_i^2 = \left(x - x_i\right)^T \left(x - x_i\right) \tag{5.8}$$

where $x$, $x_i$ $y_i$ and $n$ are input sample, training sample, output of training sample and number of training data, $d_i^2$, $e^{-(di2/2\sigma 2)}$ and $Y(x)$ are Euclidean distance, activation function (which is theoretically the weight of input) and prediction value of input samples, respectively.

## 5.2. Artificial Neural Network Learning

Artificial neural networks usually learn by recognizing the pattern or trend of data set during the training phase. Once it learns, it proceeds to production phase where it gives an independent output result. They learn by means of different learning paradigms such as learning algorithms and learning guidelines. When the learning and production phases are discrete, ANN is known as discrete network but network with continuous learning to production phase is known as dynamic network system [168].

Also, in all node of a constructed neural network model, a transfer function is required. There are different types of transfer functions and the choice of an appropriate transfer function for a given model is essential, however, there are no strict rules to the choice of a

transfer function. The neural network architecture, the task to be performed by the neural network and the learning paradigm are crucial in the choice of a good transfer function [169] .

### *5.2.1. Learning archetypes*

Artificial neural network learning can be supervised, unsupervised or combination of the two (hybrid learning) [170]. A learning rule known as model that addresses the type of technique to be applied during network training of the system to achieve expected output is adopted during neural network training. Every learning rule provides diversity of possible learning algorithm to be adopted. Learning algorithm is a mathematical method applied in updating weights during the training iteration. The process of creating a learning algorithm requires a well described training and validation protocol so as to ensure an accurate and robust system prediction [104, 171]. The rate of learning during network training known as learning rate caters for the problem that arises because of excess training samples which causes neural network's inability to give a valuable generalization. This also occurs when neural network architecture is made up of too many neurons that lead to tedious volume of computation which exceeds the vector space dimensionality.

Learning rate is parameter for neural network training that controls size of weight and bias changes during training. It is simply the fastness of network in abandoning old belief for a new one. Generally, there is need for a learning rate that is small enough for useful convergence of network and high enough that training of network doesn't take so long [172].

Therefore, in updating the weight $w_{ik}$ by gradient descent, a learning rate η must be selected. The change in weight added to the old weight is equivalent to the product of the learning rate and the gradient descent multiply by *-1*.

$$\Delta w_{ik} = \eta \frac{\partial E}{\partial w_{ik}} \eta O_k \delta_k \tag{5.9}$$

The essence of the *-1* is for updating in the minimum direction of error function and not the maximum. Hence, to prevent oscillation such as connection weight alternation inside the network and improve convergence rate, an adaptive learning rate is used as a modification of gradient descent backpropagation [173].

### *5.2.2. Momentum Parameter*

The momentum parameter is applied in preventing a neural network system from converging to a saddle point or local minima. It simply adds a fraction of previous weight to the present weight. High momentum parameter helps to increase the convergence speed of the network system, however, momentum that is too high, creates the risk of system overshooting the minima thereby causing instability of the system. On the other hand, very low momentum slows down the training of the network system and cannot dependable prevent local minima [174].

In applying momentum (also known as a variable inertia term), gradient descent and last changes in weight can be weighted in a way that the adjustment of the weight in addition is dependent on the preceding change. A '*0'* momentum results from alteration of the gradient while a momentum of *'1'*depends on the last weight change.

$$\Delta w_{ik}\left(t+1\right)=\left(1-\alpha\right)\eta\Delta w_{ik}\left(t\right) \tag{5.10}$$

where $w_{ik}$ *(t+1)* is the connection of *k* and *i* neurons at time *(t+1)*, $\Delta w_{ik}$ *(t+1)* is the change in weight, $\eta$, $\delta_k$, $0_i$ and $\alpha$ are learning rate, error signal of neuron *k*, neuron *i* output and the inertial term.

Momentum is dependent on present weight change *(t+1)*, together with present gradient of error function and changes in weight from the previous point in time. It resolves the problem of getting stuck in steep ravines and flat plateau by deceleration of gradient descent as it gets small in flat plateau to ensure quicker escape.

### *5.2.3. Delay Parameter*

The delay parameter enhances the recognition of pattern place i.e. the time invariant by storing older activation and connection values of feature elements [175]. This is carried out by re-copying of feature elements with every out going connection in every time step before the update of the original elements. The totality of the number of time-steps saved by the procedure is known as delay.

### *5.2.4. Activation Functions*

The input data set to the ANN are transformed using activation functions to ensure that data are within a controllable range [176]. Activation functions are located only in hidden and output neurons as inputs are already scaled, thus no transformation is required in the layer. Once input data are transferred to hidden layers, multiplied and summed by weights, they tend to be off the initial scale. The activation function ensures they are within acceptable and useful range to be transferred to the neurons of the next layer. There are linear and non-linear activation functions, however, non-linearity of activation function permits a universal approximation of neural network as the function becomes continuously differentiable which is vital for optimization methods based on gradient descent [177]. This permits an effective backpropagation of errors all through the network. Different types of activation functions are available; however, this research work examines the performance of two common non-linear and one linear activation functions: the hyperbolic tangent, logistic sigmoid and purelin activation functions.

## 5.3. Precept of Artificial Neural Network

The ability of ANN weights to converge at a point for an adequate operation on a data sets is known as generalization [74]. The training instances, network configuration and complexity of the problem to be solved usually determine the ability of network to generalize well. The architecture and the size of network training set also contribute to network generalization performance. The network architecture is expected to be in harmony with underlying physical complexity of the problem to impact on the training procedure. In view of a given training set size, the neuron numbers for realization of the training data should be equivalent to training instances. An over-sized network results to memorization of training data and poor generalization as network must be complex enough to draw decision boundaries to solve the problem. Hence, keeping the network size as low as possible reduces transmission overhead. Over learning in course of network training also give rise to poor generalization, however, this can be minimize by the use of different training techniques [67, 178], such as early stopping mechanism, choosing the appropriate hidden layer neurons and Bayesian regularization approach.

In application of early stopping technique, the data is divided into three sets: training, testing and validation sets and the course of training is stopped as soon as there is emergence of over-training signs which starts the worsening of network prediction accuracy. In utilizing the method of selection of appropriate hidden layer neurons, cross validation scheme are used while observing the ANN mapping accuracy as the number of neurons are varied [74]. Bayesian regularization approach reduces the problem of poor generalization or over-fitting by considering network architecture as well as goodness-of-fit. It entails modification function of network which caters for better network generalization. While early stopping method reduces the variance, it leads to increase in bias. However, both the variance and the bias can be reduced using Bayesian regularization approach [179]. This is as a result of expansion of the objective function by adding $E_s$ with the sum squares of neuron weights in the network to penalize huge weight that might be introduce so as to get an even mapping.

$$F = \beta E_d + \alpha E_s \tag{5.11}$$

where β and α are enhanced in the framework of Bayesian regularization [180]. α is the decay rate, $\alpha << \beta$ results to algorithm yielding smaller error while $\alpha >> \beta$ emphasizes reduction in size of the weight at the cost of the network error thereby giving rise to network with an even response [155, 181]. $E_d$ is a decisive factor.

## 5.4. Results and Discussions of this Work

The simulation results of this work obtained from training the different ANN architectures in comparison to MLP architectural network are discussed:

### *5.4.1. Analysis of Different Training Approaches*

The measured data are collected through a drive test from a metropolitan area. The measurement setups are a laptop and two Samsung Galaxy mobile handsets (Model-SY *4*) installed with Test Mobile System (TEMS, *15.1* version) software, network scanner, Global Positioning System (GPS), power inverters, test cables,compass, digital map of the area and a vehicle for the drive test. Drive test started from choosen base station transmitting at *1900 MHz* operating frequency and captures the signal power at various distances from LTE

cellular network. Transmitter antenna and reciever antenna parameters used for the experiments are as follows: transmitter power *(43 dB)*, transmitter antenna height*(34 m)* reciever antenna height*(1.5 m)* reciever antenna gain*(18 dBi)*. Training set are made up of *1,176* data which are collected at various measurement of signal power across different distances from the choosen BS.

MATLAB programs are written and run in ANN training tool box (*nntraintool*) in MATLAB *2013a* for RBF ANN and MLP ANN with one hidden layer using the measured data as input to the ANNs. In course of network training, using the approach of variation of neuron number in hidden layer, *100%* of the data are used for training and the neurons varied from *10*, *20*, *30*, *40*, *50*, *60* and *70*. Applying early stopping approach, *80%* of the data are used for training, 10% are used for testing and *10%* are used for validation and the neuron number with the smallest error during variation of neuron in hidden layer used. Employing the third approach of Bayesian regularization, *90%* of the data are used for training and *10%* for validation. The neural network is trained for an average of ten runs and the result with the least error taken. This is to ensure the network has learnt the pattern the signal propagates after many runs on training set and a predictive ability developed.

Sigmoid transfer function is used with Bayesian regularization training algorithm and the training process made more effective by normalizing the input i.e. data set to lie about *zero* mean and unity standard deviation using excel spread sheet as state in Eq. (4.37) of chapter 4. The results from the three different ANN training techniques are measured and analysed in terms of RMSE, MAE, SD and r. The results are presented in Table 5.1, 5.2 5.3 and 5.4 respectively. The values with the highest prediction accuracy are highlighted in bold and black colour in Table 5.1 and 5.2.

From the network performance results in Table 5.1, *40* neurons in hidden layer of MLP network shows the least RMSE, MAE, SD and the highest r.As the neurons increases above *40*, there is re-occurrence and increase in prediction error as observed with neurons fewer than *40* in hidden layer. Therefore, MLP network performs more effectively with moderate hidden layer neurons, fewer and many neurons result to poor network generalization. Different observation are made with RBF network as presented in Table 5.2.

Table 5.1. Results of neuron variation in MLP hidden layer

| **Training function** | **Parameters for comparison** | | | | | | |
|---|---|---|---|---|---|---|---|
| | No of neurons | Epoch (E) (1000) | Training time (s) | RMSE | MAE | SD | r |
| **(trainbr)** | 10 | 429 | 10 | 2.7988 | 2.2255 | 1.6972 | 0.9783 |
| | 20 | 892 | 23 | 2.3202 | 1.8514 | 1.3984 | 0.9854 |
| | 30 | 1000 | 30 | 2.2010 | 1.7291 | 1.3618 | 0.9867 |
| | **40** | **1000** | **34** | **1.9148** | **1.4875** | **1.2058** | **0.9900** |
| | 50 | 1000 | 41 | 2.1490 | 1.3584 | 1.4798 | 0.9873 |
| | 60 | 1000 | 49 | 2.1957 | 1.3833 | 1.7060 | 0.9867 |
| | 70 | 1000 | 145 | 2.3580 | 1.4607 | 1.8511 | 0.9849 |

Table 5.2. Results of neuron variation in RBF hidden layer

| **Training function** | **Parameters for comparison** | | | | | | |
|---|---|---|---|---|---|---|---|
| | No of neurons | Epoch (E) (1000) | Training time (s) | RMSE | MAE | SD | r |
| **(trainbr)** | 10 | 644 | 12 | 7.4142 | 5.8336 | 4.5760 | 0.8356 |
| | 20 | 1000 | 22 | 4.6320 | 3.6137 | 2.8976 | 0.9393 |
| | 30 | 1000 | 25 | 2.7550 | 2.2073 | 1.6486 | 0.9789 |
| | 40 | 1000 | 29 | 2.3878 | 1.9169 | 1.4237 | 0.9842 |
| | 50 | 1000 | 33 | 2.1454 | 1.7002 | 1.3085 | 0.9873 |
| | 60 | 1000 | 38 | 1.9095 | 1.4920 | 1.1917 | 0.9899 |
| | **70** | **1000** | **44** | **1.6921** | **1.3059** | **1.0760** | **0.9921** |

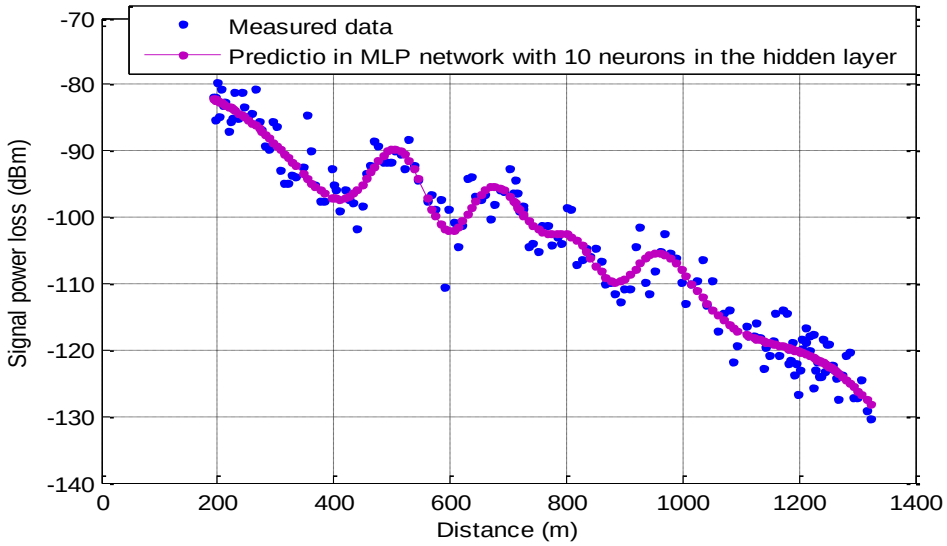
Table 5.3. Results MLP and RBF network training using early stopping approach

| **Parameters for comparison** | **Training function (trainbr)** | |
|---|---|---|
| | MLP | RBF |
| Neuron number in the hidden layer with best prediction | 40 | 70 |
| Epoch (E) (1000) | 1000 | 1000 |
| Training time (s) | 28 | 44 |
| RMSE | 1.7868 | 1.6921 |
| MAE | 1.3658 | 1.3059 |
| SD | 1.1520 | 1.0760 |
| r | 0.9912 | 0.9921 |

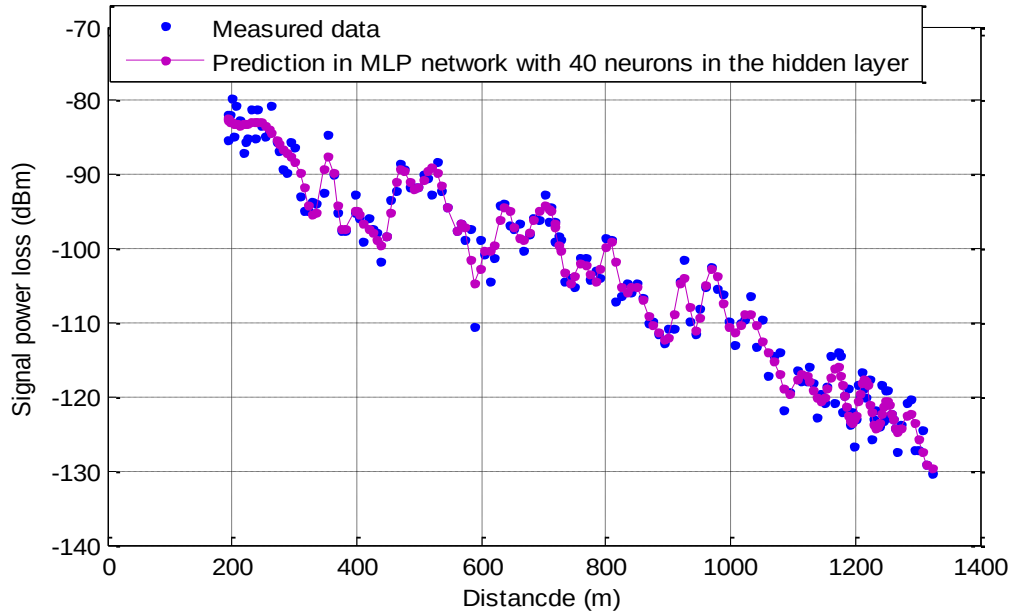
Table 5.4. Results of MLP and RBF network training using Bayesian Regularization approach

| **Parameters for comparison** | **Training Function (trainbr)** | |
|---|---|---|
| | MLP | RBF |
| Neuron number in the hidden layer with best prediction | 40 | 70 |
| Epoch (E) (1000) | 1000 | 1000 |
| Training time (s) | 28 | 44 |
| RMSE | 1.8027 | 1.6921 |
| MAE | 1.4196 | 1.3059 |
| SD | 1.1111 | 1.0760 |
| r | 0.9911 | 0.9921 |

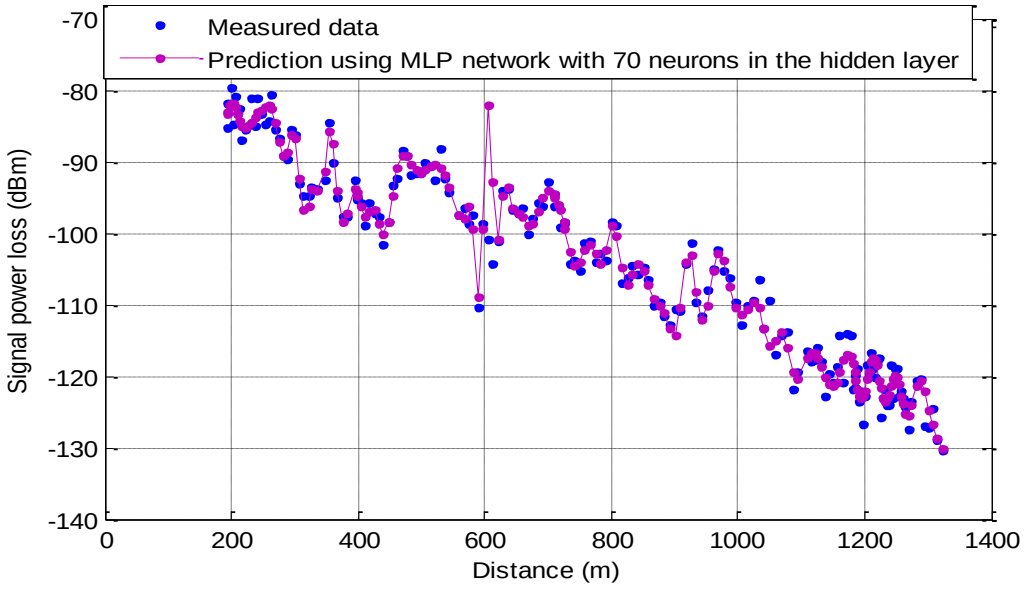
The RBF network predicted the system effectively with large number of hidden layer neurons. As the network gets complex, the prediction accuracy increases and RMSE, MAE, SD reduces while r increases. The best prediction are recorded at *70* neurons in hidden layer and RBF network also demostrates faster learning of the pattern the signal propagates without several runs before the stabilization of such learning. However, there is significant increase in training time to train network as the number of neurons in hidden layer increases in comparison to MLP ANN. Figures 5.4(a,b&c) show MLP training output results of *10*, *40* and *70* neurons in hidden layer which are the best and worst prediction performances while figures 5.5(a,b&c) show the RBF network training with *10*, *40* and *70* neurons in hidden layer.




(a)

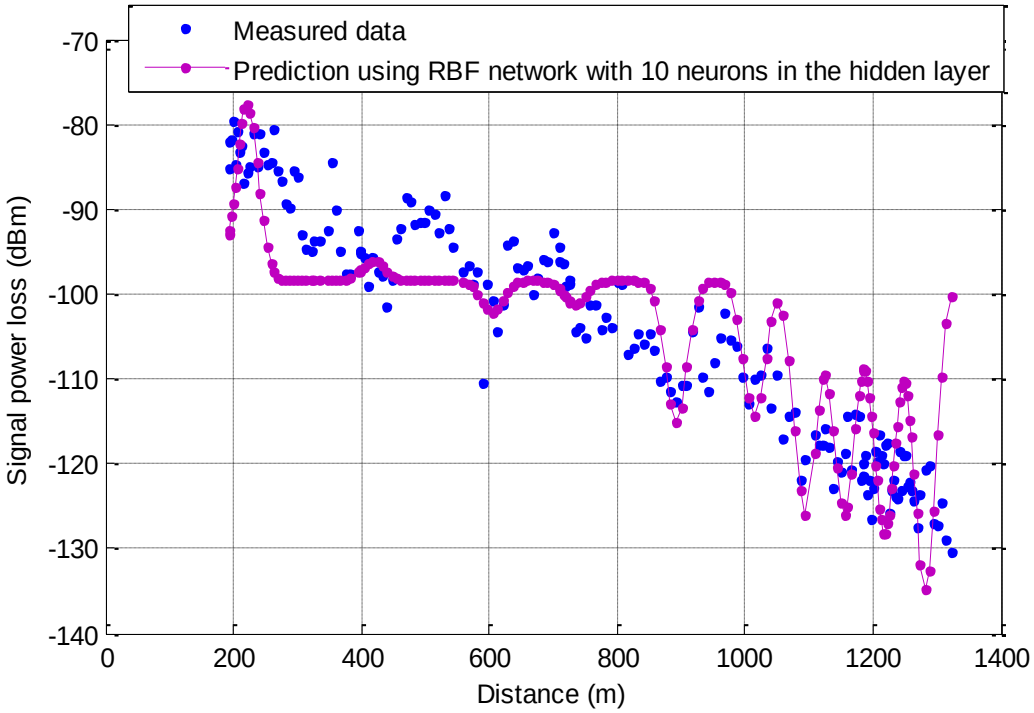



(b)

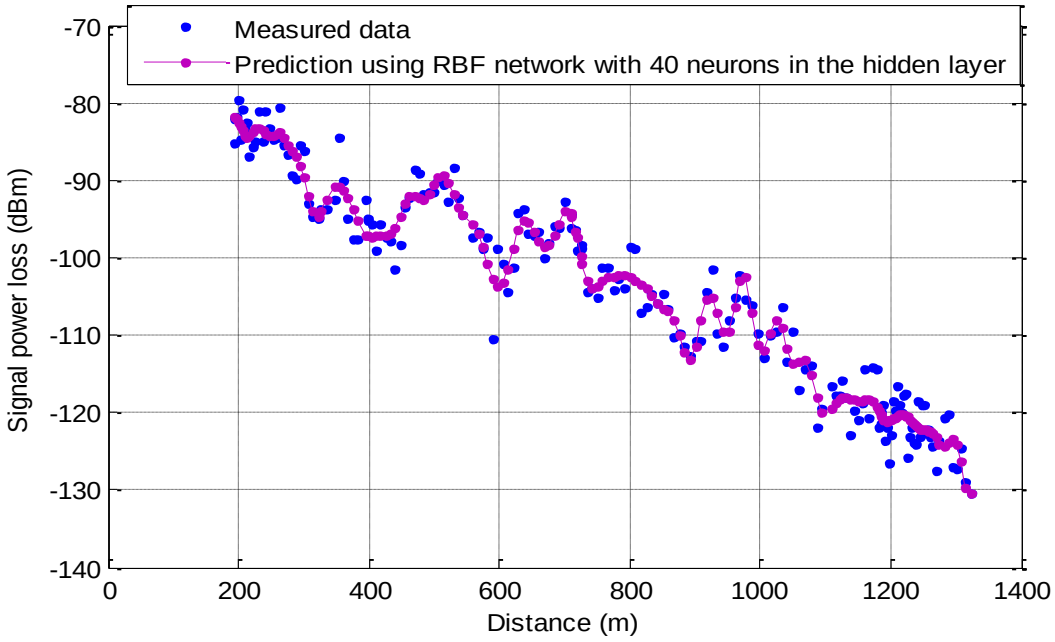



(c)

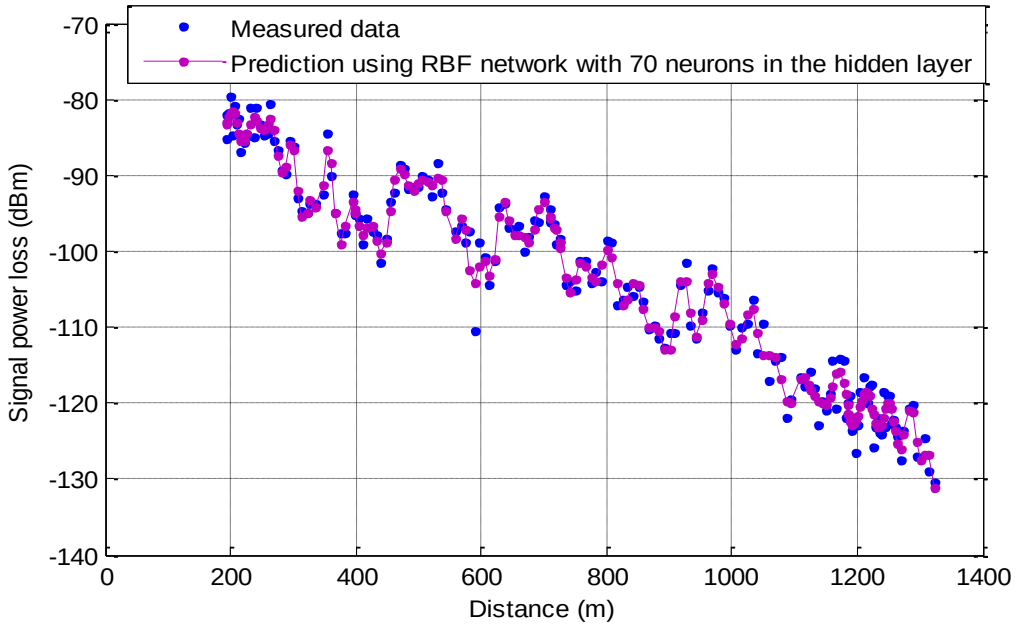
Figure 5.4. Prediction with Bayesian regularization algorithm in MLP network (a) 10; (b) 40; and (c) 70 neurons in hidden layer


Measured data
Prediction using RBF network with 10 neurons in the hidden layer
Signal power loss (dBm)
Distance (m)
-70
-80
-90
-100
-110
-120
-130
-140
0
200
400
600
800
1000
1200
1400

(a)


Measured data
Prediction using RBF network with 40 neurons in the hidden layer
Signal power loss (dBm)
Distance (m)
-70
-80
-90
-100
-110
-120
-130
-140
0
200
400
600
800
1000
1200
1400

(b)




(c)

Figure 5.5. Prediction with Bayesian regularization algorithm in RBF network (a) 10; (b); and 40 (c) 70 neurons in the hidden layer

In applying the early stopping approach during the network training, the neuron with the least prediction errors during neuron variation in hidden layers of MLP and RBF ANN are used: *40* and *70* neurons in hidden layer of MLP and RBF networks respectively. Radial basis function network gives the best prediction output with the least RMSE, MAE, SD and highesr r than the MLP network. However, it took significantly longer training time than the MLP network. Figures 5.6 and 5.7 show the graphical results obtained using early stopping approach.

Application Bayesian regularization approach shows no clear difference in the errors seen from the RBF network in comparison to that obtained using early stopping approach. However, there is slight error difference in comparison to early stopping approach in MLP network.

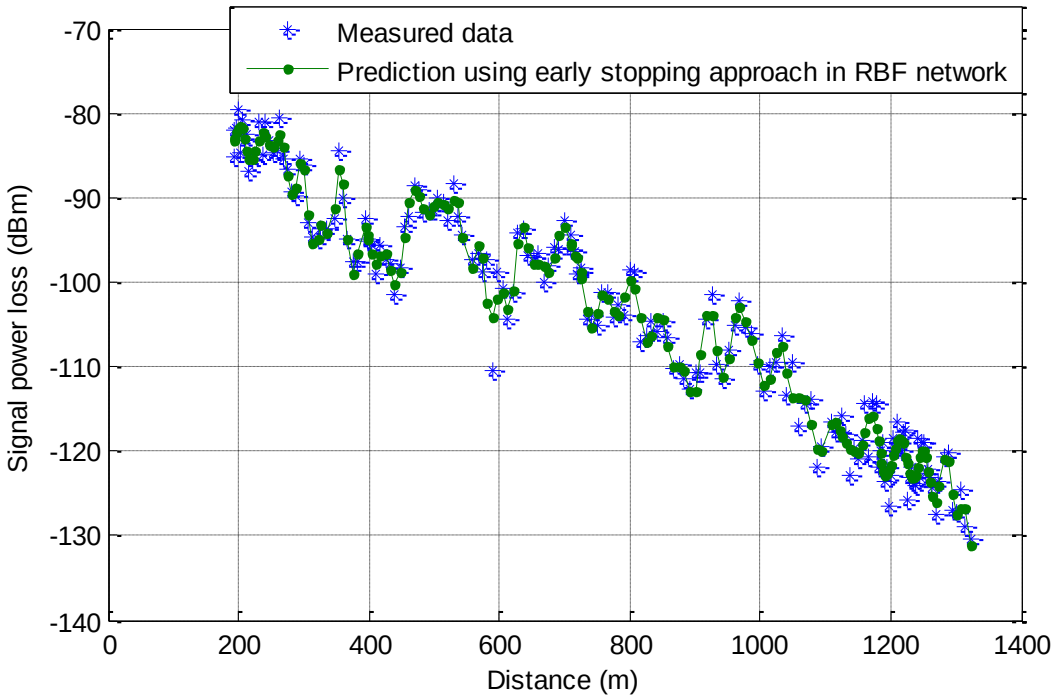



Figure 5.6. Prediction using early stopping approach in RBF network

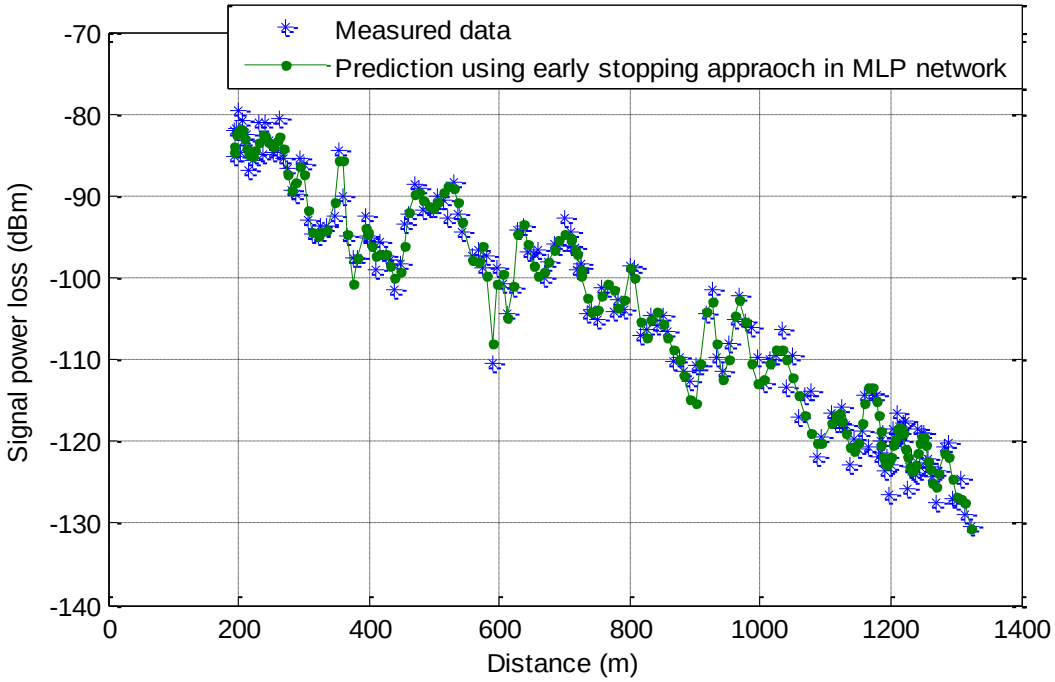



Figure 5.7. Prediction using early stopping approach in MLP network

### *5.4.2. Analysis of Effect of Learning Rate*

The data set are collected via a drive test from an Island comprising of territorial areas, virgin lands and water taxis. The measurement setups are as stated in *first* section of this chapter. The training set are made up of *1,036* data collected at various measurement of signal power across different distances from the choosen BS. The neural network architecture of ADALINE and MLP networks has been trained using Bayesian regularization algorithm with written MATLAB programs of ADALINE and MLP and implemented using ANN toolbox (*nntraintool*) in MATLAB *2013a*. To ensure there is no bias in the order of presentation of data pattern to ANNs, measured data has been normalized in excel spread sheet. Applying Bayesian regularization approach, *90%* of the data are used for training while the remaining *10%* are used for validation. The performance of the ADALINE and MLP models in the prediction of the signal power has been measured in terms of RMSE, SD, MAE and r. These performance metrics are used to measure error between measured values and predicted values using different values of learning rate on the MLP and ADALINE network with ADALINE having an input delay of *1:5*. The gradient and momentum parameter for ADALINE and MLP ANNs have also been analysed at different learning rate. Finally, the performance of different combination of linear and non-linear transfer function in hidden and output layers of ADALINE and MLP ANNs are analysed. The results are presented in Table 5.5 to Table 5.8.

A small learning rate of *0.1%* with ADALINE network gives minimal error than the MLP network as shown in Table 5.5, however, the gradient is *0.66707* with momentum of *0.05* as shown in Table 5.6, this is slope movement beyond the optimal that may result to network overshoot. These are graphically shown in figures 5.8 and 5.9 respectively. At *0.1%* learning rate, the gradient of MLP network is *0.31436* with a momentum of *0.005* which is a slope movement towards the optimal i.e. the local minima. Better momentum value as seen with MLP network helps to increase convergence speed of the network. Increased learning rate to *0.9%* leads to a better prediction ability of MLP network than that of ADALINE. However, the gradient increases to *0.44194* with a momentum of *0.05* while the gradient of ADALINE reduces to *0.074406* with a momentum of *0.05*. These are graphically shown in figures 5.10 and 5.11 respectively. Therefore, with increase in learning rate, high gradient parameter as seen will result to fast network convergence, however, this may also lead to

system overshoot. Gradient descent shows an effective use in determining optimal weights by assisting in search of optimal value of a cost function.

Table 5.5. Training Result of MLP Network and ADALINE with 1:5 input delay

| **Artificial neural networks parameters for comparison** | **ADALINE with input delay (Learning rate %)** | | | | | **Multi-Layer Perceptron with no input delay (Learning rate %)** | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 | 0.1 | 0.3 | 0.5 | 0.7 | 0.9 |
| Correlation co-efficient (r) | 0.9905 | 0.9910 | 0.9894 | 0.9883 | 0.9877 | 0.9884 | 0.9896 | 0.9907 | 0.9911 | 0.9915 |
| RMSE | 1.8610 | 1.8121 | 1.9660 | 2.0610 | 2.1184 | 2.0537 | 1.9399 | 1.8627 | 1.8055 | 1.7525 |
| MAE | 1.4746 | 1.3771 | 1.5270 | 1.5868 | 1.6771 | 1.4473 | 1.5197 | 1.3946 | 1.3722 | 1.3900 |
| SD | 1.1354 | 1.1778 | 1.2383 | 1.3152 | 1.2941 | 1.4571 | 1.2057 | 1.2348 | 1.1733 | 1.0673 |
| Training Time | 28 | 28 | 30 | 30 | 28 | 29 | 28 | 28 | 28 | 28 |
| Epoch Iteration (1000) | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 | 1000 |

Table 5.6. Gradient and Momentum parameters for ADALINE and MLP at different learning rate

| **Learning rate (%)** | **Gradient** | | **Momentum Parameter** | |
|---|---|---|---|---|
| | **ADALINE** | **Multi-Layer Perceptron** | **ADALINE** | **Multi-Layer Perceptron** |
| 0.1 | 0.66807 | 0.31436 | 0.050 | 0.005 |
| 0.3 | 0.06772 | 0.38942 | 0.050 | 0.005 |
| 0.5 | 0.69185 | 0.40904 | 0.050 | 0.050 |
| 0.7 | 0.05764 | 0.42981 | 0.050 | 0.050 |
| 0.9 | 0.07441 | 0.44194 | 0.050 | 0.050 |

Table 5.7. Performance of the MLP network with different combination of transfer functions

| **Multi-Layer Perceptron Network** | | | | |
|---|---|---|---|---|
| **Transfer Function** | **RMSE** | **SD** | **MAE** | **r** |
| logsig,logsig | 2.0244 | 1.4199 | 1.4430 | 0.9887 |
| logsig,tansig | 1.9760 | 1.2377 | 1.5402 | 0.9892 |
| tansig, logsig | 1.8671 | 1.1275 | 1.4882 | 0.9904 |
| logsig,purelin | 2.2009 | 1.5020 | 1.6087 | 0.9866 |
| tansig,purelin | 2.1007 | 1.4353 | 1.5340 | 0.9878 |

The first term denotes hidden layer transfer function while the second term denotes output layer transfer function.

Table 5.8. Performance of ADALINE with linear and non-linear transfer functions

| ADALINE | | | | |
|---|---|---|---|---|
| **Transfer Function** | **RMSE** | **SD** | **MAE** | **r** |
| logsig | 2.1457 | 1.3746 | 1.6475 | 0.9874 |
| tansig | 2.3167 | 1.5182 | 1.7499 | 0.9853 |
| purelin | 1.8846 | 1.2139 | 1.4416 | 0.9903 |

The performance of ADALINE and MLP network with different non-linear and linear transfer functions during network training has been shown in Tables 5.8 and 5.9. Purelin transfer function in ADALINE shows the least errors because it is a linear transfer function which is most appropriate for a linear neural network such as ADALINE. The combination of different transfer functions in hidden and output layer of MLP network show the least performance error with combination of hyperbolic tangent and logistic sigmoid in hidden and output layers respectively.

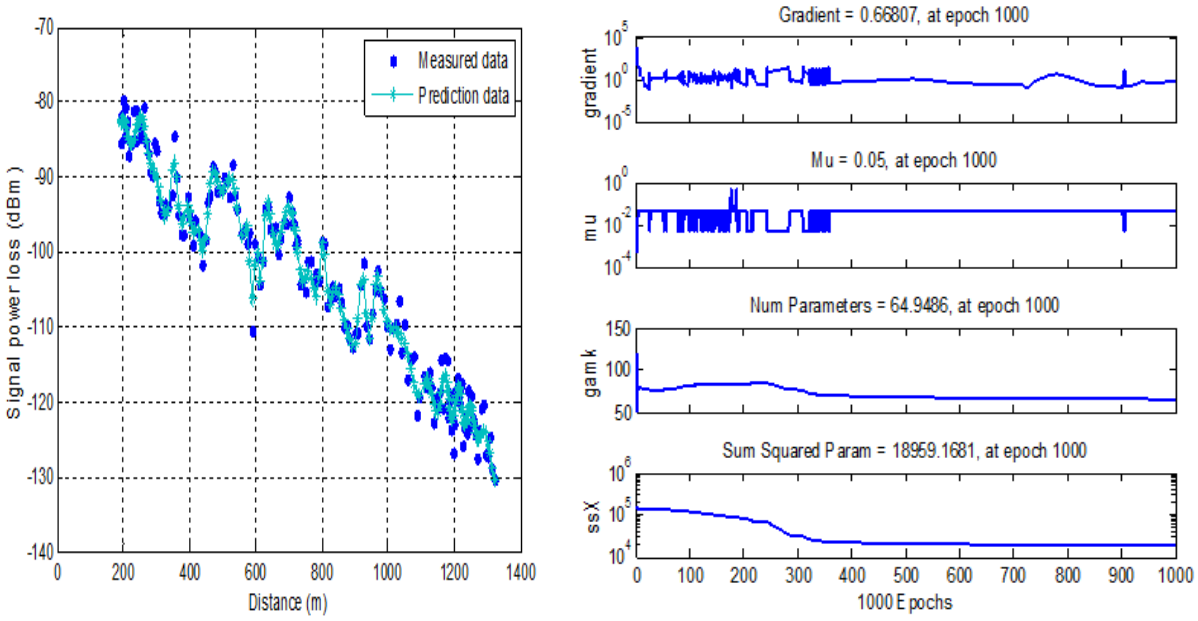



Figure 5.8. Prediction graph and training state of ADALINE at 0.1% learning rate

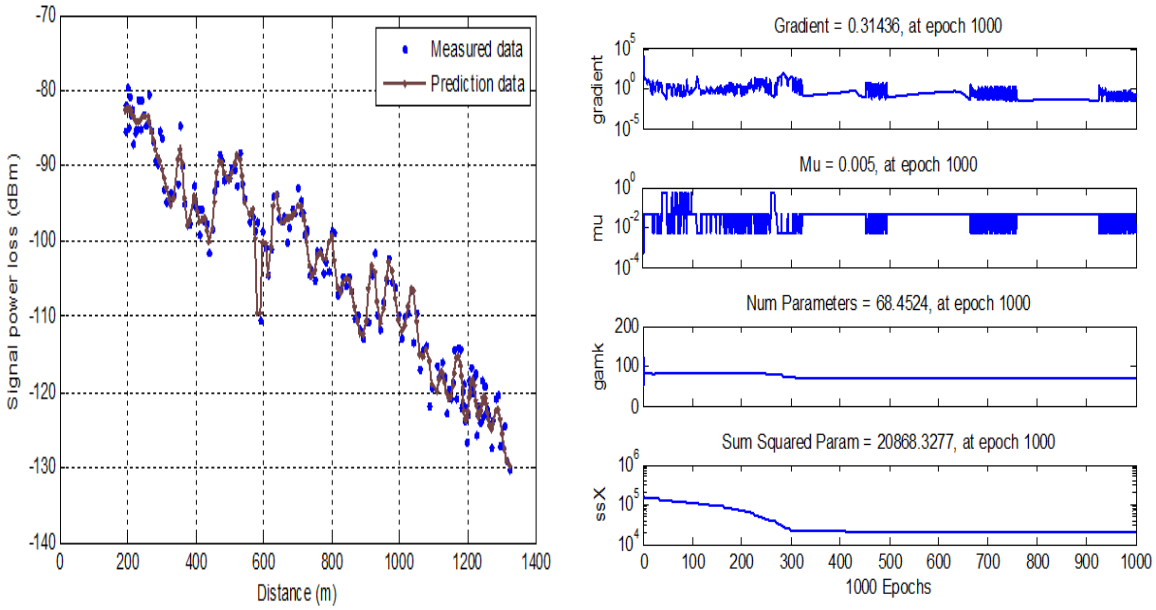



Figure 5.9. Prediction graph and training state of MLP Network at 0.1% learning rate.

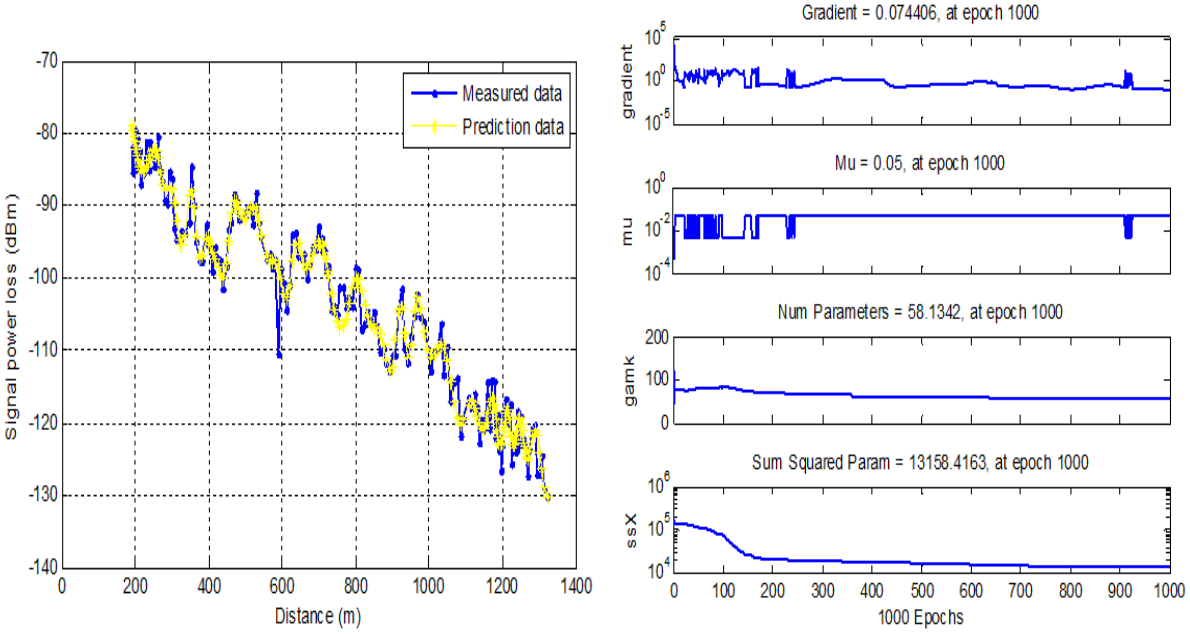



Figure 5.10. Prediction and training state of ADALINE at 0.9% learning rate

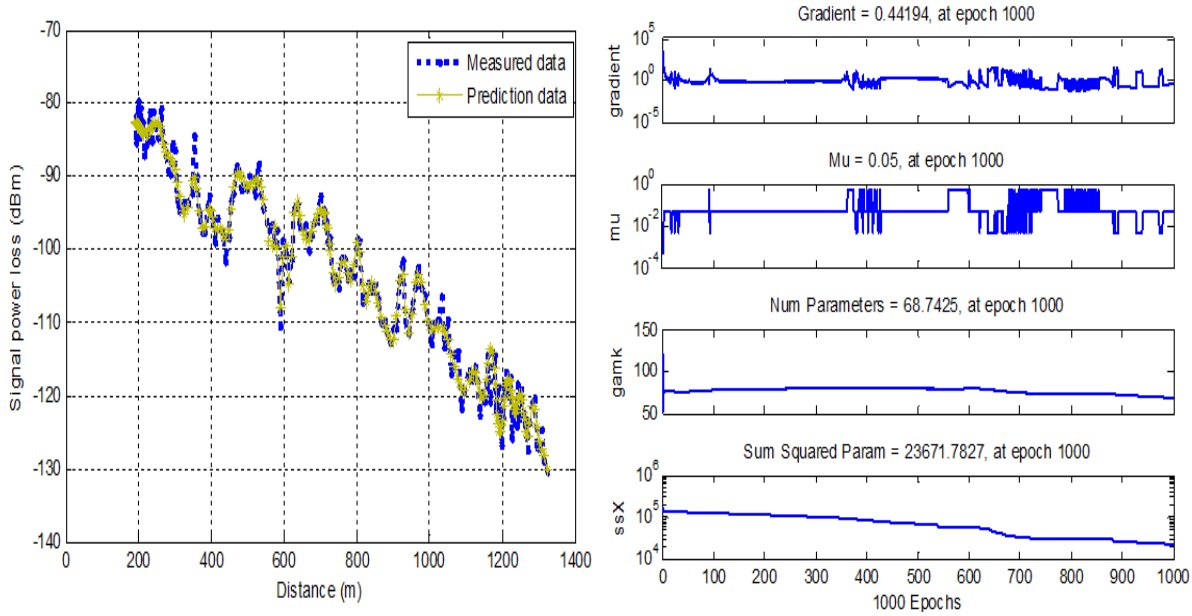



Figure 5.11. Prediction and training state of MLP Network at 0.9% learning rate.

### *5.4.3. Analysis of Effect of Spread Factor in GRNN and Effect of Learning Rate in GRNN and MLP ANN*

Data are collected from two base stations, BS1 and BS2 from a micro-cell LTE cellular network in a sub-urban environment with a low land topography that has a land mass of 376.5 $m^2$ and an estimated population of around two million. The area has a tropical wet climate and lies between the latitude of 4.990833$^o$N and longitude of 7.054444$^o$E. Data are collected via a drive test and the measurement setup is as described in first section of this chapter with a digital map of the area used during the drive test. All the measurements are conducted from different sector of BS that transmits at *1900 MHz*. The information on the cell file is provided by the mobile network provider. The measured signal power gotten at different distances serves as data set and input to the GRNN and MLP network during the network training.

A MATLAB program for GRNN and MLP ANNs are written and implemented in MATLAB 2013a using ANN toolbox (*nntraintool*) with seven *1,020* and *1,541* real world measured data from BS*1* and BS*2* respectively. Early stopping approach has been used during the learning process with *80%* of the data set used for training, *10%* for testing and the remaining *10%* for validation. The GRNN and MLP neural networks are trained using Bayesian regularization training algorithm. The performance metrics: RMSE, MAE, SD and r

are used to analyse the prediction performance of GRNN and MLP ANN at different variation of learning rate parameter. The effect of spread factor in GRNN is also analysed using measured data from BS*1* while the effect of learning rate on ANNs are re-examined on GRNN and MLP ANNs using measured data from BS*2*. The results are presented in Tables 5.9, 5.10 and 5.11 respectively.

Table 5.9. GRNN results using different spread factor

| Spread factor variation in GRNN | RMSE | SD | MAE | r |
|---|---|---|---|---|
| 0.5 | 0.1559 | 0.1500 | 0.0426 | 0.9997 |
| 1.0 | 0.6302 | 0.5954 | 0.2064 | 0.9954 |
| 2.0 | 1.2179 | 1.0468 | 0.6226 | 0.9829 |
| 3.0 | 1.6684 | 1.2343 | 1.1225 | 0.9678 |
| 4.0 | 2.0104 | 1.4042 | 1.4388 | 0.9528 |
| 5.0 | 2.2459 | 1.5413 | 1.6336 | 0.9406 |
| 6.0 | 2.4328 | 1.6541 | 1.7840 | 0.9299 |
| 7.0 | 2.5980 | 1.7670 | 1.9045 | 0.9196 |
| 8.0 | 2.7458 | 1.8686 | 2.0119 | 0.9096 |
| 9.0 | 2.8745 | 1.9544 | 2.1079 | 0.9003 |
| 10.0 | 2.9833 | 2.0272 | 2.1887 | 0.8918 |
| 11.0 | 3.0732 | 2.0816 | 2.2608 | 0.8848 |
| 12.0 | 3.1465 | 2.1267 | 2.3289 | 0.8787 |
| 13.0 | 3.2059 | 2.1647 | 2.3647 | 0.8736 |
| 14.0 | 3.2542 | 2.1929 | 2.4048 | 0.8694 |

Table 5.10. Variation effect of learning rate parameter in GRNN

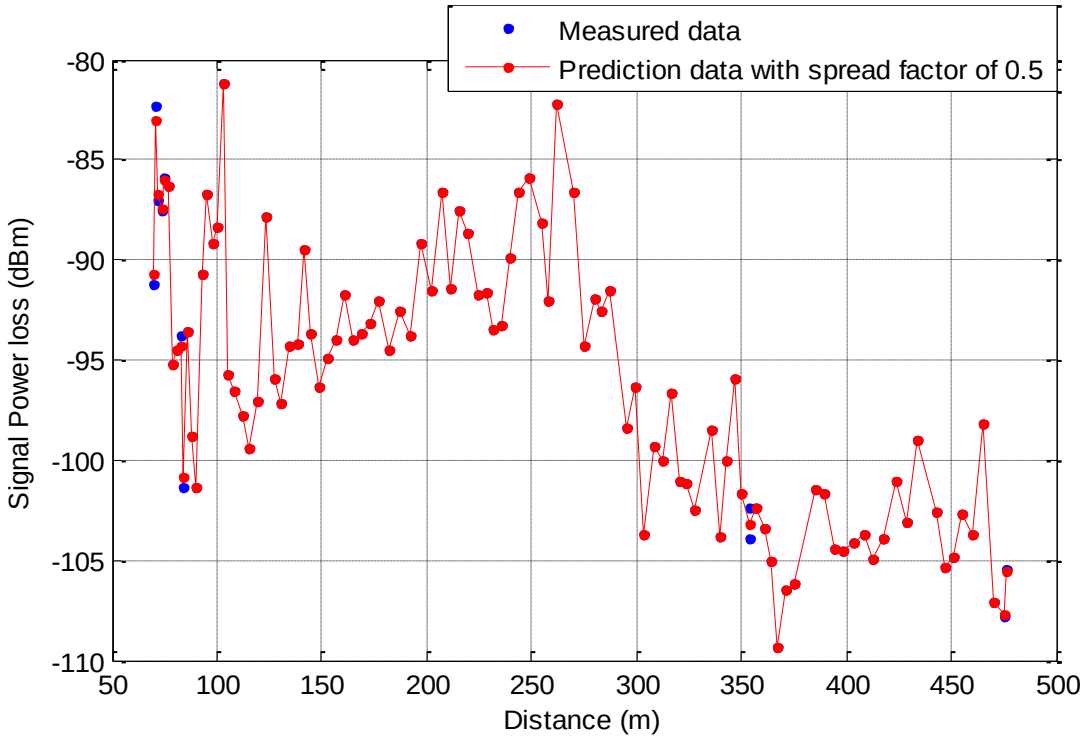
| **Generalized regression neural network (GRNN)** | | | | |
|---|---|---|---|---|
| Learning rate (%) | RMSE | MAE | SD | r |
| 0.2 | 0.5187 | 0.2234 | 0.4681 | 0.9993 |
| 0.4 | 0.9770 | 0.6463 | 0.7326 | 0.9974 |
| 0.6 | 1.4187 | 1.0890 | 0.9093 | 0.9945 |
| 0.8 | 1.7306 | 1.3534 | 1.0786 | 0.9918 |
| 1.0 | 1.9489 | 1.5329 | 1.2036 | 0.9895 |
| 1.2 | 2.1171 | 1.6714 | 1.2995 | 0.9876 |
| 1.4 | 2.2511 | 1.7876 | 1.3682 | 0.9860 |

Table 5.11. Variation of learning rate parameter in MLP network

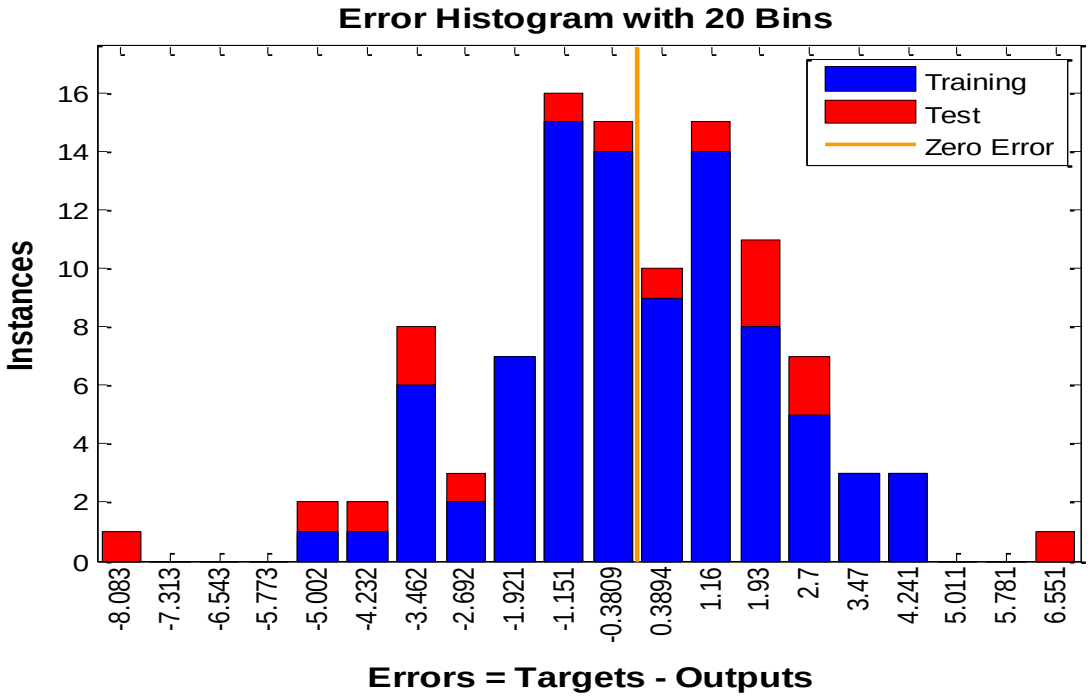
| **Multi-layer perceptron (MLP)neural network** | | | | |
|---|---|---|---|---|
| Learning rate (%) | RMSE | MAE | SD | r |
| 0.2 | 2.0222 | 1.5465 | 1.3029 | 0.9888 |
| 0.4 | 1.9374 | 1.5303 | 1.1883 | 0.9897 |
| 0.6 | 1.8844 | 1.4668 | 1.1830 | 0.9902 |
| 0.8 | 1.8654 | 1.4538 | 1.1687 | 0.9904 |
| 1.0 | 1.8267 | 1.4206 | 1.1485 | 0.9908 |
| 1.2 | 1.9231 | 1.4685 | 1.2417 | 0.9899 |
| 1.4 | 2.0248 | 1.5274 | 1.3292 | 0.9887 |

Table 5.9 shows prediction results using different values of spread factor. Large spread factors result in high prediction error even when a small learning rate is correspondingly applied, while a small spread factor results to minimal error. However, small spread factor leads to a steep radial basis function, but with increase in the spread factor, the slope of the radial basis function becomes smoother thereby making many neurons respond to an input vector as shown in the error histogram with *20* bins of figures 5.12 (b) and 5.13 (b). The prediction performance of *0.5%* spread factor and *14%* spread factor which give the minimal and the highest prediction performance errors are shown in figures 5.12 (a) and 5.13 (a) respectively.

Training GRNN with a small learning rate results to minimal prediction errors which eventually result to good network convergence and generalization as shown in Table 5.10. Measuring network performance in terms of performance metrics used, the least prediction errors with the highest correlation coefficient is seen at learning rate of *0.2%*. The prediction errors increase as learning rate increases with learning rate of *1.4%* giving the highest prediction error. These are graphically represented in figures 5.14 (a & b), 5.15 (a& b) respectively.

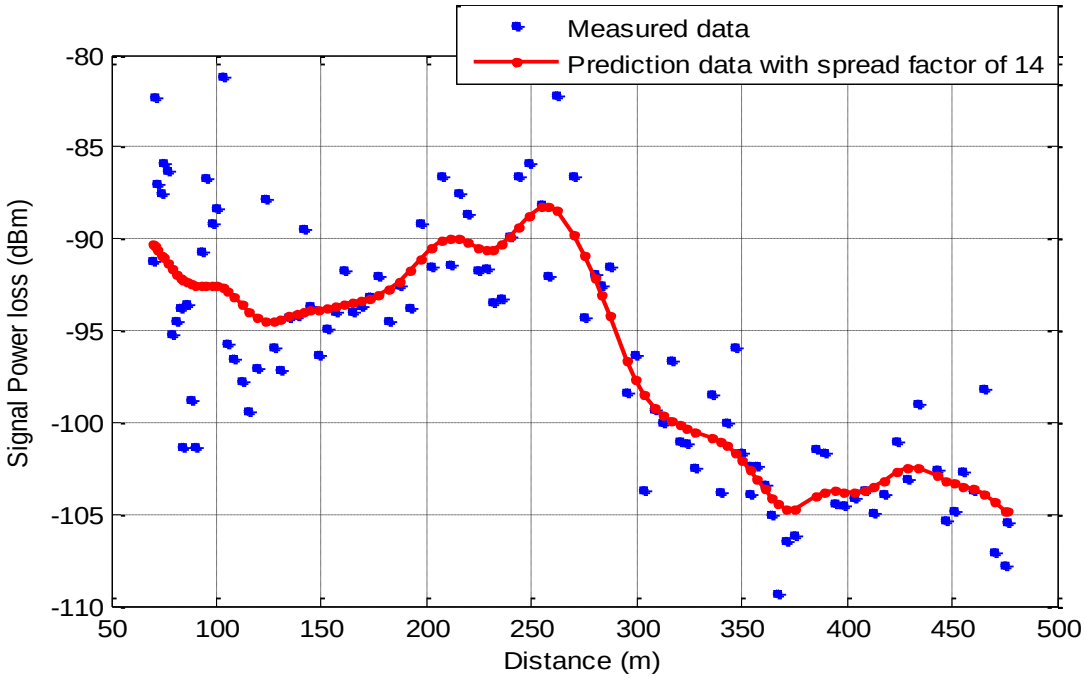



(a)

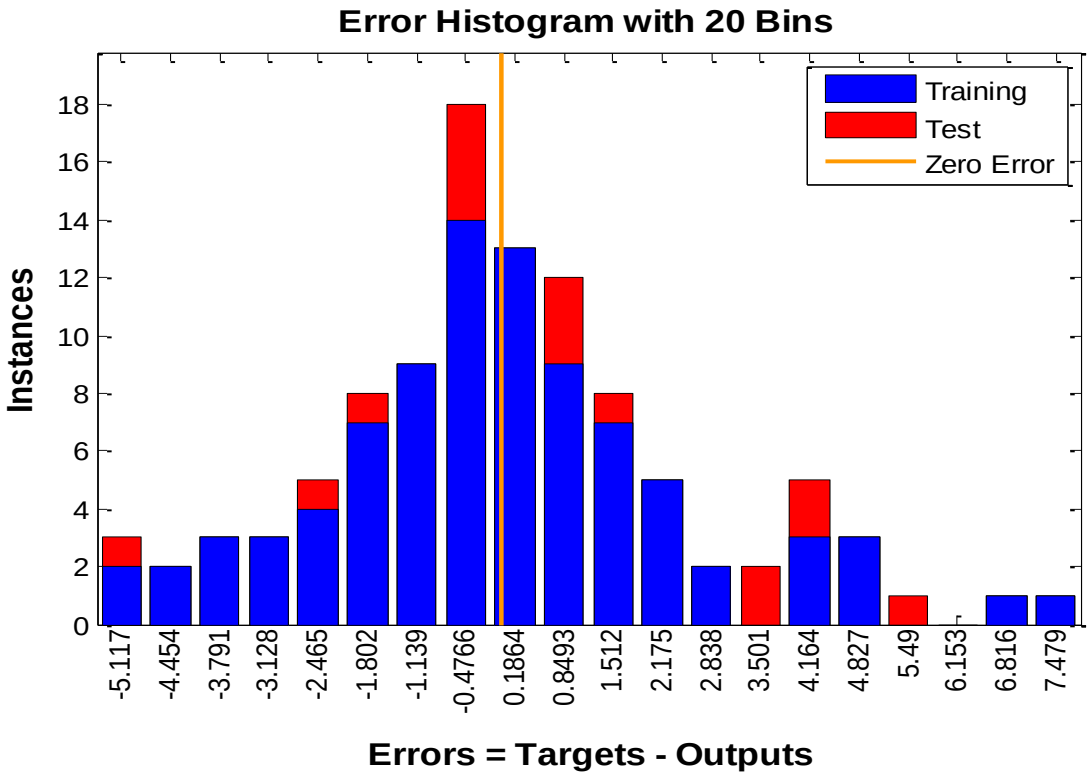



(b)

Figure 5.12. (a) Effect of spread factor of 0.5, (b) Error Histogram with 20 bins for spread factor of 0.5 in GRNN

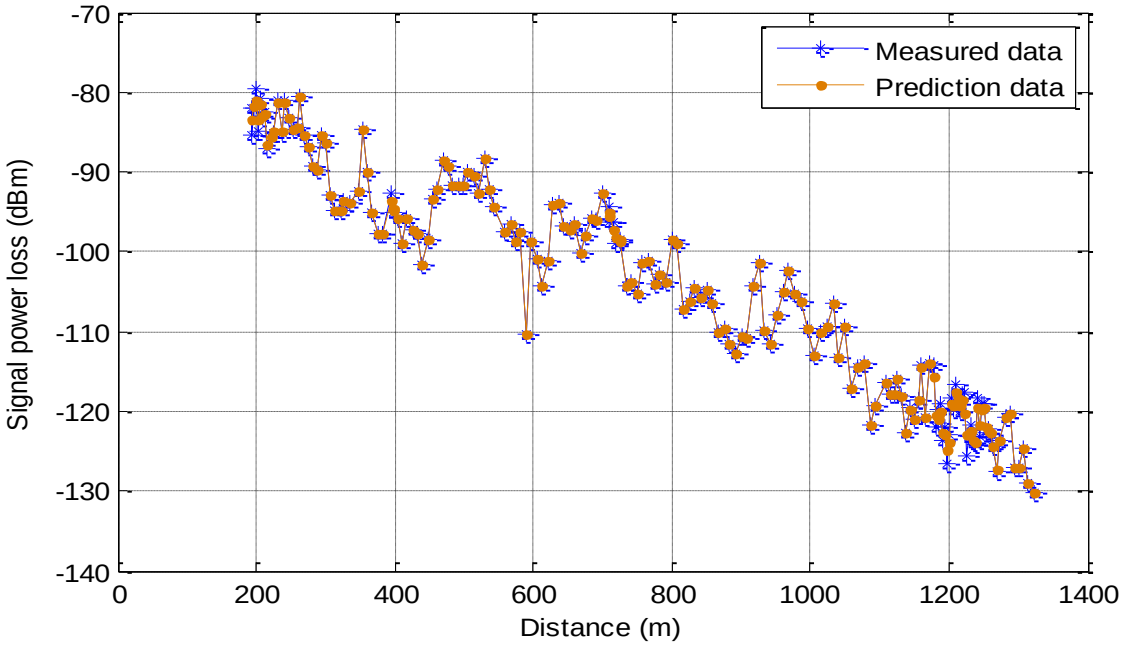



(a)

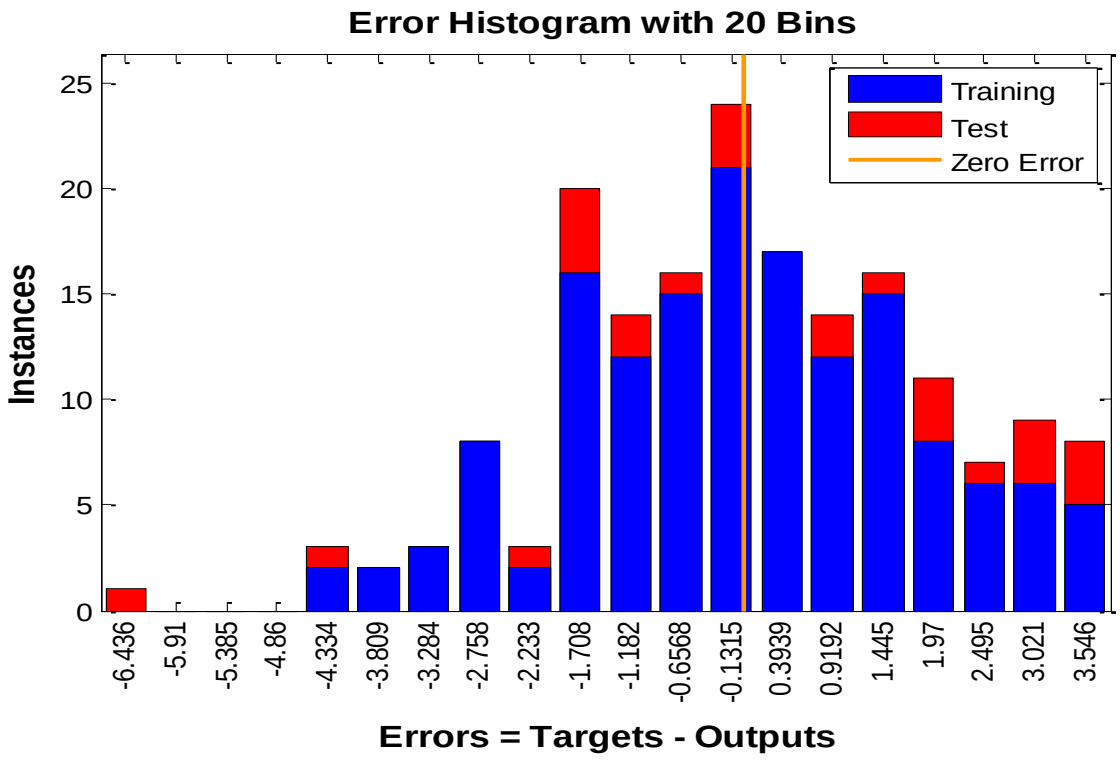



(b)

Figure 5.13. (a) Effect of spread factor of 14, (b) Error Histogram with 20 bins for spread factor of 14 in GRNN

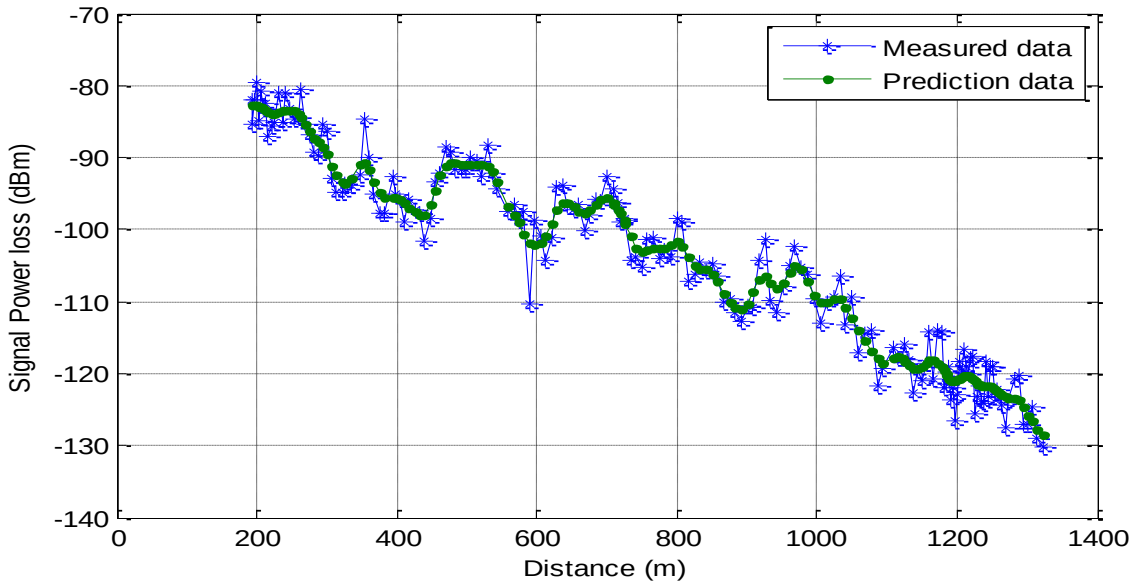



(a)

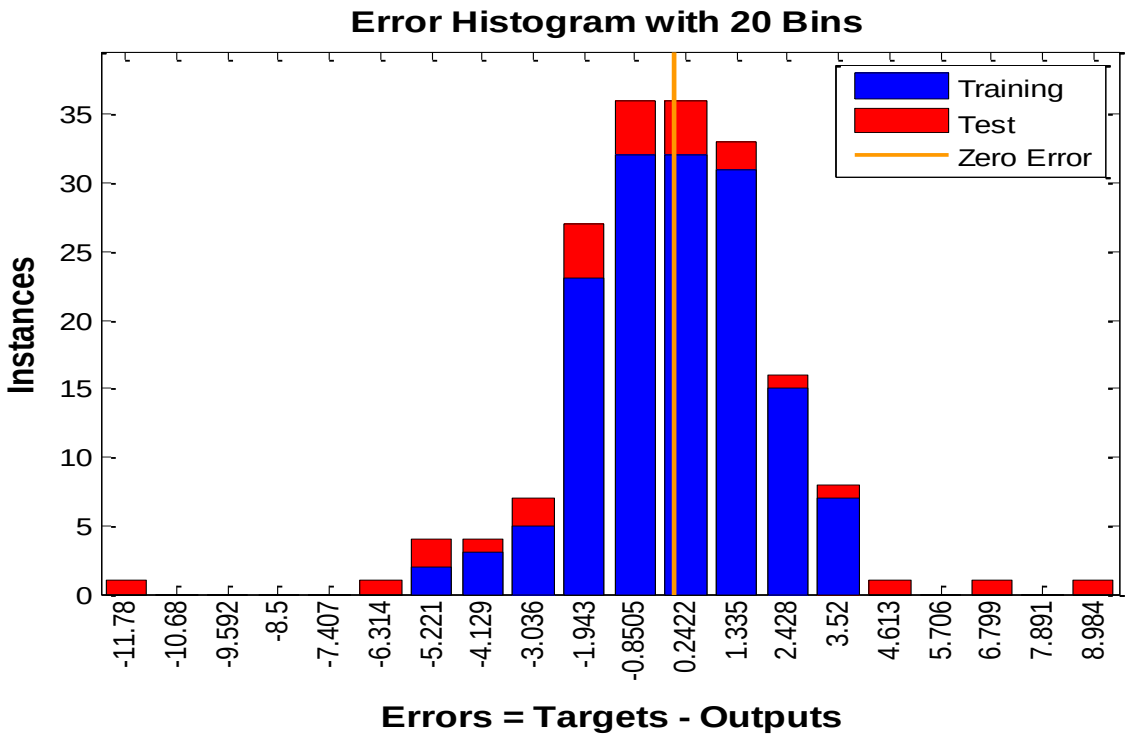



(b)

Figure 5.14. (a) Prediction at 0.2% learning rate, (b) Error Histogram of at 0.2% learning rate in GRNN

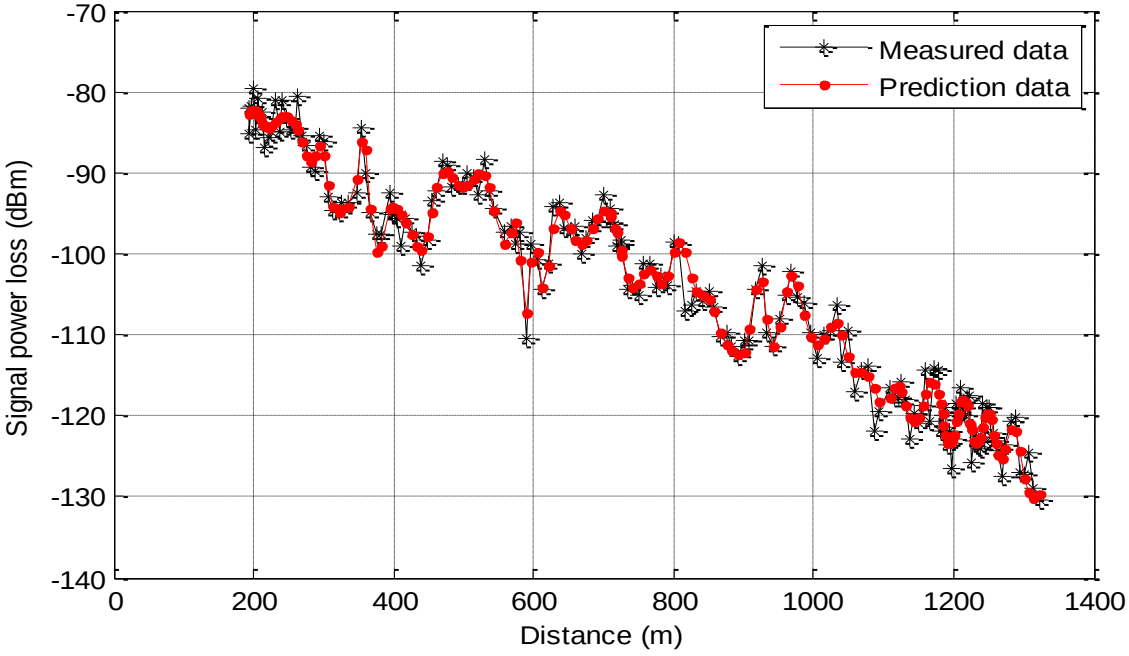



(a)

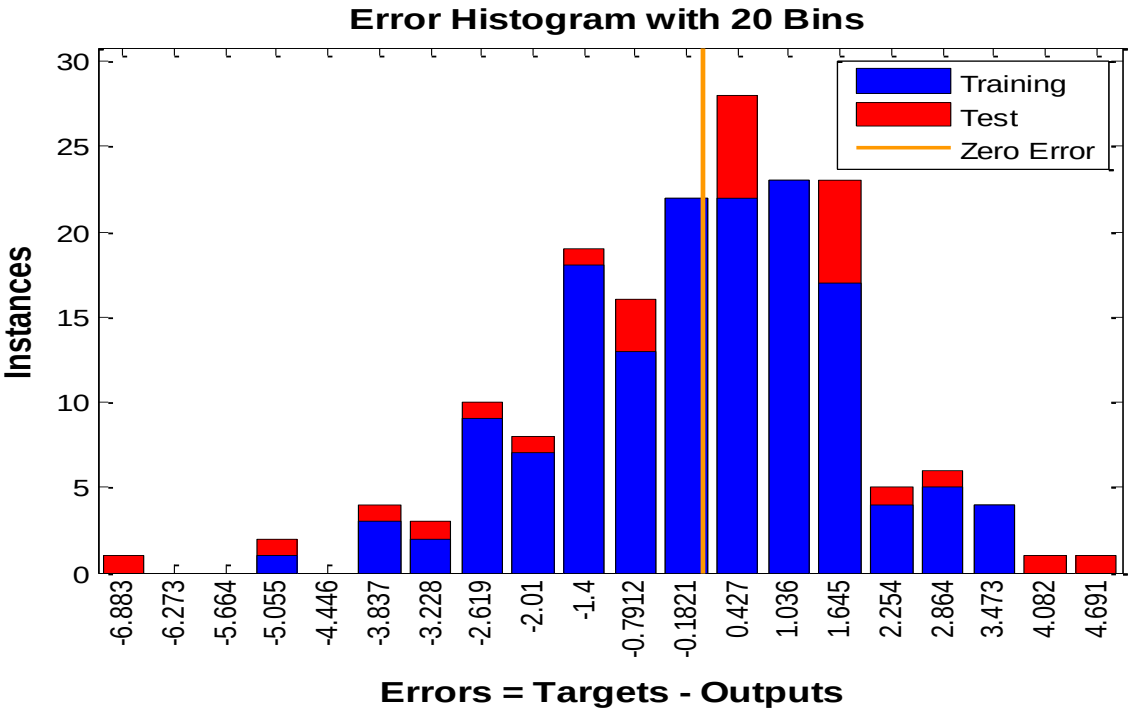



(b)

Figure 5.15. (a) Prediction at 1.4 % learning rate, (b) Error Histogram at 1.4% learning rate in GRNN




(a)




(b)

Figure 5.16. (a) Prediction at 1.0 % learning rate, (b) Error Histogram at 1.0% learning rate in MLP neural network

Figure 5.14 (a) shows a well-trained network at *0.2%* learning rate which effectively predicted the measured data in comparison to figure 5.15 (a) which is trained with a learning rate of *1.4%*. Thus, the GRNN gives minimal prediction error with learning rate of *0.2%*, training *17* out of *20* data in error histogram with *20* bins as shown in figure 5.14 (b). Only *3* out of the *17* trained data in the error histogram are not tested. Error histogram graph of figure 5.15 (b) shows trained and tested *9* out of *20* data in the error histogram with *20* bins at learning rate of 1.4%. The remaining data are either not trained or only tested.

However, the results are different with MLP neural network as very small learning rates result to poor network generalization with high prediction errors and low correlation coefficient. Learning rate of *1.0%* gives best generalized network with a minimal prediction errors and high correlation coefficient. This is shown in figures 5.16 (a). As the learning rate increases above *1.0%*, the prediction error starts to increase with display of poor network generalization just as observed with small learning rate.

Figure 5.16 (b) is the error histogram of MLP neural network at *1.0%* learning rate which gives minimal errors with highest correlation coefficient as learning rate are varied from *0.2%* to *1.4%*. The error histogram shows that out of data in the error histogram with *20* bins, *14* measurement data are trained and tested while the remaining data are either not trained or only tested.

## 5.5. Chapter Summary

The performance of three different ANN architectures: (i) RBF, (ii) ADALINE and (iii) GRNN has been analysed in comparison with MLP ANN in the prediction of signal power loss using real world measured data from LTE cellular environment from metropolitan, Island and sub-urban areas. Three different training approaches to avoid poor generalization and overcome the tendencies of over-fitting during network training are analysed in RBF ANN and MLP ANN respectively. The learning rate parameter is varied during the ANN training, while analysing the effect of the different values on ADALINE and MLP ANN architectures. Also, the effects of different values of spread factor are analysed in GRNN using data set from BS*1,* and data from BS *2* are used in re-training GRNN and MLP ANN while analysing

the effect of different values of learning rate parameters on the network architectures. The performance results are measured in terms of RMSE, MAE, SD and r.

The performance of RBF ANN and MLP ANN using different techniques during network training are analysed in the *first* section. These are variation of neuron numbers in hidden layer, early stopping technique and Bayesian regularization approach. The MLP ANN shows an effective prediction performance with *40* neurons in hidden layer while further increase in the number of neurons leads to poor network generalization. Thus, MLP network shows capability of modelling a moderate size propagation network and for more complex networks, intermediary layers or network modifications are required.

For RBF ANN, the generalization ability of the network increases as network gets more complex with *70* neurons in hidden layer giving the best prediction with minimal error. Training RBF network using early stopping approach gives better prediction with minimal errors in comparison to neuron variation in hidden layer and Bayesian regularization in MLP ANN. However, in RBF network training, there is no difference in errors obtained using early stopping approach and Bayesian regularization approach. RBF ANN models appropriately a complex network without signs of over-fitting, and, because of its fixed three-layer architecture, there is no poor generalization resulting from architectural complexity which Bayesian regularization approach tackles.

ADALINE and MLP ANN models are analysed in section *two* considering prediction error between actual measurement and prediction values. The gradient and momentum parameters of the two networks are checked at different variation of learning rate. The ADALINE shows minimal prediction error at *0.1%* learning rate but has the best gradient of 0.074406 which approximates to local minima at *0.9%* learning rate. The reverse is seen in MLP ANN as its minimal prediction error is at learning rate of *0.9%* but the gradient is farther from the local minima at 0.9% learning rate. Therefore, there is need for appropriate choice of learning rate that will be high enough to increase network convergence, but not too high to result to over-fitting. Furthermore, training the two ANN network with different combination of linear and non-linear transfer functions shows effective performance of hyperbolic tangent and logistic sigmoid in hidden and output layers of MLP ANN. Purelin transfer function performs best with ADALINE network.

Finally, section *three* analysed the effect of spread parameter in GRNN and effect of learning rate parameter in GRNN and MLP network. The GRNN and the MLP neural network are trained using Bayesian regularization training algorithm and *40* neurons in hidden layer. The data set from BS*1* are used to analyse the effect of spread factors in GRNN model during network training. The data from BS*2* are used to re-examine the performance of learning rate variation in GRNN and MLP ANN considering their prediction accuracy. The results show a good performance of GRNN with a small spread factor of *0.5*, however many neurons did not respond to the input vector at small spread factor of *0.5*, resulting to a steep radial basis function. The GRNN also demonstrates good prediction ability with small learning rate of *0.2%* which results in good network convergence that shows good generalization ability by training and testing *85%* of data in the error histogram with *20* bins, leaving only *15%* untrained, however, it took longer training time in comparison with MLP neural network. The MLP network trained *70%* of data in the error histogram with *20* bins while leaving *30%* untrained at a learning rate of *1.0%*. Nevertheless, the training time required for MLP network is very short which even shows drastic reduction as the learning rate increases.

This study reveals the need for proper optimization techniques and strategies in training ANN as it has great influence in prediction output. The different ANN architectures and parameters play vital roles in effective prediction of training set and appropriate selection of these parameters in line with the complexity of the problem ensures an improved output result.

# CHAPTER 6

# DEVELOPMENT OF ENHANCED ADAPTIVE ANN MODELS FOR THE PREDICTION OF ROPAGATION PATH LOSS

The need for proper optimization techniques to ensure error minimization using Artificial Neural Network (ANN) models for the prediction of signal power loss during electromagnetic signal transmission from transmitter to receiver has been established in the previous chapters. In this chapter, two different ANN models have been proposed, developed and analysed for predicting signal power loss between transmitter and receiver with minimal error.

Firstly**,** an adaptive neural network predictor that combines Multi-Layer Perceptron (MLP) and Adaptive Linear Element (ADALINE) ANNs is proposed for enhanced signal power prediction in micro-cellular outdoor environments. The prediction accuracy of the proposed hybrid adaptive neural network predictor has been tested and evaluated using real life measured data acquired from Long Term Evolution (LTE) radio network environment with mixed residential, commercial and cluttered building structures. The prediction accuracy of the model in comparison to standard MLP ANN is measured in terms of Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Standard Deviation (SD) and Correlation Coefficient (r).

Secondly, a resourceful predictive model, built on MLP network with vector order statistic filter based pre-processing technique for improved prediction of measured signal power loss in micro-cellular LTE network environments is proposed and developed. The predictive model is termed Vector Median Filters Multi-Layer perceptron (VMF-MLP). The prediction performances of the standard MLP network and Vector *L* filters built on MLP (VLF-MLP) in comparison to the VMF-MLP are measured in terms of RMSE, MAE, SD and r.

## 6.1. Introduction

In the design and placement of radio Base Station (BS) transmitters, accurate field signal power prediction and modelling is of critical importance. Efficient coverage planning plays an important role in reducing both OPerational EXpenditure (OPEX) and CApital EXpenditure (CAPEX) when deploying and expanding existing cellular communication systems. Particularly, it assists in providing good link and service quality at the receiving User Equipment (UE) mobile terminals. A vital phase in coverage planning concerns accurate prediction of signal coverage or signal propagation Path Loss (PL) between transmitter and receiving UE mobile terminals [182].

Consequently, the provision of an adaptive modelling tools and algorithms which could emulate the real characteristics of radio cellular network environment and predict radio signal coverage loss with less error could be of enormous assistance to radio frequency cellular network planners and network service providers as well. Given the above context, good number researchers in the field wireless cellular communication have turned their focus to the use of Machine Learning (ML) architecture and algorithms. This is with the goal of exploring the enhanced training and learning capability of ML tools to automatically examine the properties of systems under study and adapt to them.

One of this ML tools is ANNs, which possess mapping and generalization capabilities that enable it to predict new results from past outcomes. There exist many ANN architectures for solving various functional approximation problems. These include the Multi-Layer Perceptron (MLP) ANN, Radial Basis Function (RBF) ANN, Adaptive Linear Element (ADALINE), Generalized Regression Neural Network (GRNN), etc. as discussed in chapter 5. However, one of the most popular and common used neural network architecture for functional approximation problems in literature as contain in many research works is the MLP [183-186].

The MLP ANN belongs to a well-known group of neural networks structure known as the feedforward neural networks. The MLP ANN possess many advantages such as scalability and simple design, thus it has been employed in solving problems in many research areas like radio wave propagation modelling and optimization problems. However, despite possessing

these advantages, employing standard MLP ANN poses some key challenges, among which are choice of suitable number of hidden neurons and layers for learning, choice of suitable learning parameters versus learning rate and choice of training technique, algorithm and training time. Also, conventional MLP networks perform poorly in handling noisy data [187-189], and lack capabilities in dealing with incoherence datasets which contracts with smoothness [189].

The ADALINE is another suitable type of ANN employed for modelling, adaptive prediction, system selection and noise cancellation [190, 191]. Some of the notable advantages of ADALINE are fast learning and simple usage. However, ADALINE is a single layer of artificial neurons with linear decision boundaries that can solve problems that are only linearly separable [114, 157, 158].

This chapter is organized in two parts: Part I presents the proposed adaptive neural predictor that combines ADALINE and MLP ANN while Part II presents the proposed combination of non-linear data filtering-based denoising method and MLP neural network model termed VMF-MLP for improved prediction of measured micro-cellular signal power dataset.

## 6.2. Proposed Adaptive Neural Predictor

In this part of research work, a hybrid neural predictor which combines the ADALINE with MLP ANN is employed for optimal signal power prediction in micro-cellular LTE radio networks. Related approach has been employed in [191], but for signal noise filtering, reduction and prediction in Rayleigh fading channel. Particularly, while *Gao et.al.* [191] concentrated on effective noise reduction in simulated temporal signal data, our focus is combining the faster learning capability of ADALINE with MLP for optimal training and improved prediction accuracy of realistic spatial signal data. Unlike in [191], the Minimum Mean Square Error (MMSE) is considered to solve the neural network structure complexity for optimal prediction.

This research drive is to find an efficient predictive model that combines a linear predictor and a non-linear predictor for adaptive prediction of large scale signal power. Also,

to find equivalently signal propagation loss with minimal error in any radio propagation environment.

### *6.2.1. Proposed Artificial Neural Network Architecture*

The respective feedforward neural network architecture of ADALINE (the linear predictor) and MLP (the non-linear predictor) are shown in figures 6.1 and 6.2 respectively. The $z^{-1}$ designates the structural delay element to enable the afore-mentioned scale and re-sample the reference input data at different rate. Assuming that the ADALINE has $n$ input nodes $x_1$, $x_2$, $x_3$,.$x_n$ which are the signal power sample number to be predicted, the ADALINE prediction output can be expressed as:

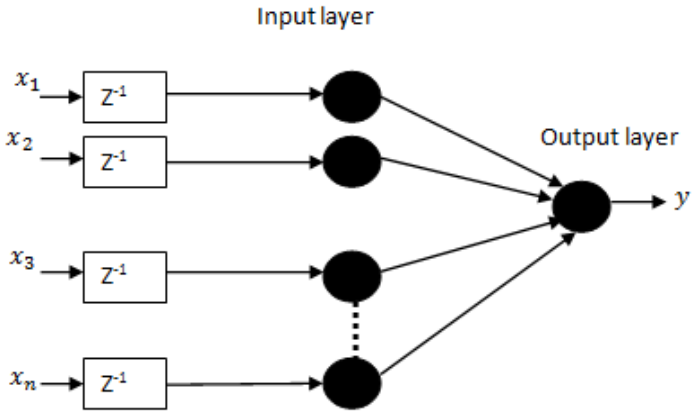



Figure 6.1. ADALINE neural-network [73].

$$y = \sum_{j=1}^{n} w_j x \tag{6.1}$$

where $x_n$ and $w_j x$ are the expected value and weights sum of the linear neural predictor. To enhance the neural training structure, the Minimum Mean Square Error (MMSE) technique is employed at the input nodes as [192]:

$$E_n = \sum_{j=0}^{n} \left[ y - \bar{y} \right]^2 = \sum_{j=0}^{n} e_n^2 \tag{6.2}$$

where $\bar{y}$ is the n$^{th}$ desired output prediction response.

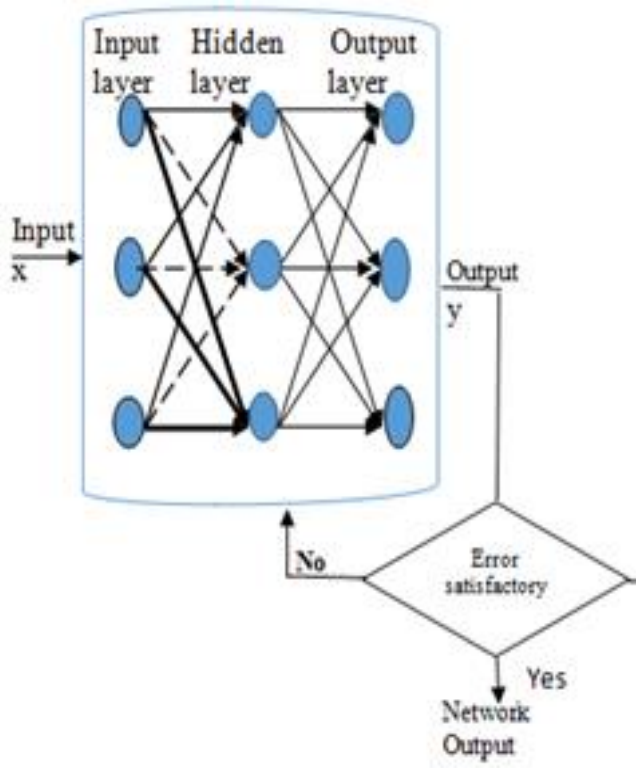



Figure 6.2. Multi-layer perceptron neural-network with one Hidden Layer [65].

The input data are transmitted through the layers in a forward direction. Figure 6.2 is a single (i.e. one) hidden layer MLP neural network architecture and output can be expressed as:

$$y_i = f_o\left(\sum_{j=0}^{n} w_{jq}\left(f_h\left(\sum_{i=0}^{n} w_{iq}x_i\right)\right)\right) \quad \text{for } q=1, 2, \ldots n \tag{6.3}$$

where $w_{iq}$ and $w_{jq}$ indicate the respective connection weight and synaptic weight vectors in relation to hidden layer neuron-inputs, and from hidden layer-output neuron. $x_i$ stands for the input vector elements, $i$ =1…. $n$, $f_o$ and $f_h$ are the respective output layer and hidden layer activation function. The MLP learning and training process entails minimizing the error function (i.e. cost function), which can be expressed as:

$$E(w) = \frac{1}{2}\sum\left(y_q - d_q\right) = \frac{1}{2}\sum_{q=1}^{N} e_q^2 \tag{6.4}$$

where $e_q = y_q - d_q$, $y_q$ and $d_q$ indicate desired target output value and actual network value, respectively.

The set-up of the proposed adaptive hybrid predictor (figure 6.3) is such that the measured signal power values are first trained with ADALINE system predictor and the resultant output fed into the MLP predictor for further training to deliver optimal prediction accuracy. The adaptive hybrid predictor is trained using three different fast backpropagation algorithms discussed in chapter *4*: (i) Resilient propagation (RP), (ii) Leverberg-Marquardt (LM), and (iii) Bayesian regularization algorithms to establish the algorithm that best train the hybrid predictor and predict the behaviour of the signal strength with most favourable convergence results and performance.

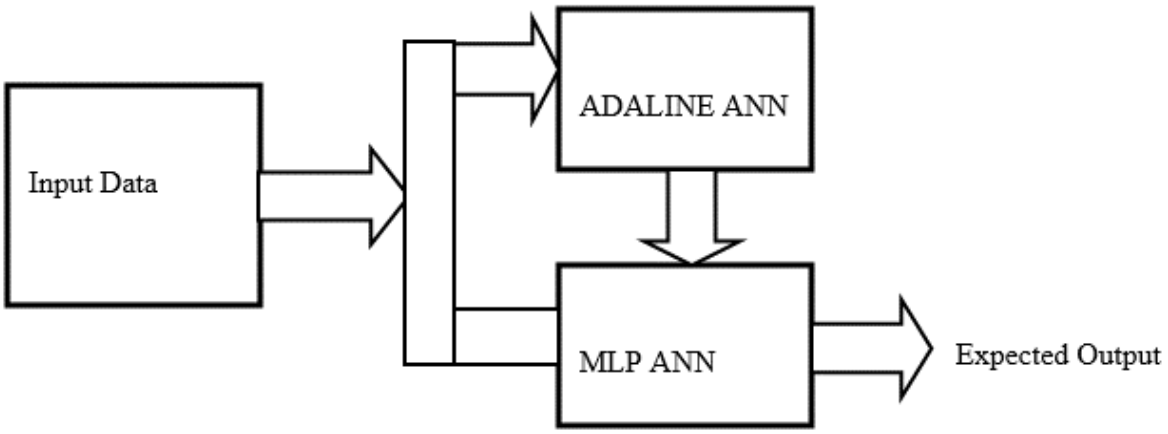



Figure 6.3. The Proposed Hybrid Adaptive Neural structure.

The field measurements are piloted using LTE cellular networks that operate at *1900 MHz* frequency band between JAN-DEC 2017. A drive-test tools and measurement technique as described in chapter *5* are used and some key computer software such as MapInfo, MATLAB *2013a* and excel spread utilized for post-processing of acquired signal testing log files and data analysis. The LTE mobile phones and the HP laptop are both engrained with Test Mobile Software (TEMS) which enables it to access, record and extract the signal data along the measurement test routes. The Global Positioning System (GPS) and compass are used for matching up the Mobile Station (MS) (i.e. user equipment) measurement locations in correspondent to field test environment and the Base Station (BS) transmitter. The main LTE signal parameter extracted from the drive test log file for ANN prediction analysis is the Reference Signal Receive Power (RSRP *dB m*). This provides signal strength information at the UE receiver terminal. The field signal power measurements are carried out in *5* selected

locations with special concentration on built-up busy urban streets, roads and open areas with mixed residential and commercial building structures. The first *4* locations, i.e. locations *1-4* are built-up urban terrains with mixed residential and cluttered building structures. The remaining *1* location is an uncluttered open urban terrain with many commercial buildings, few residential buildings, country motor parks, petrol filling stations and among others. The

All the written neural network program, simulations and computations are achieved using the MATLAB*2013a* software. To avoid over-fitting, which often reduces ANN predictive and modelling capability, the early-stopping technique has been employed. In view of that, the data-set are shared into three parts using ratio 7*5%:* 1*5%: 15%* for training, testing and validation respectively. The input data-sets are normalized to enhance the generalization of ANNs as expressed [193].

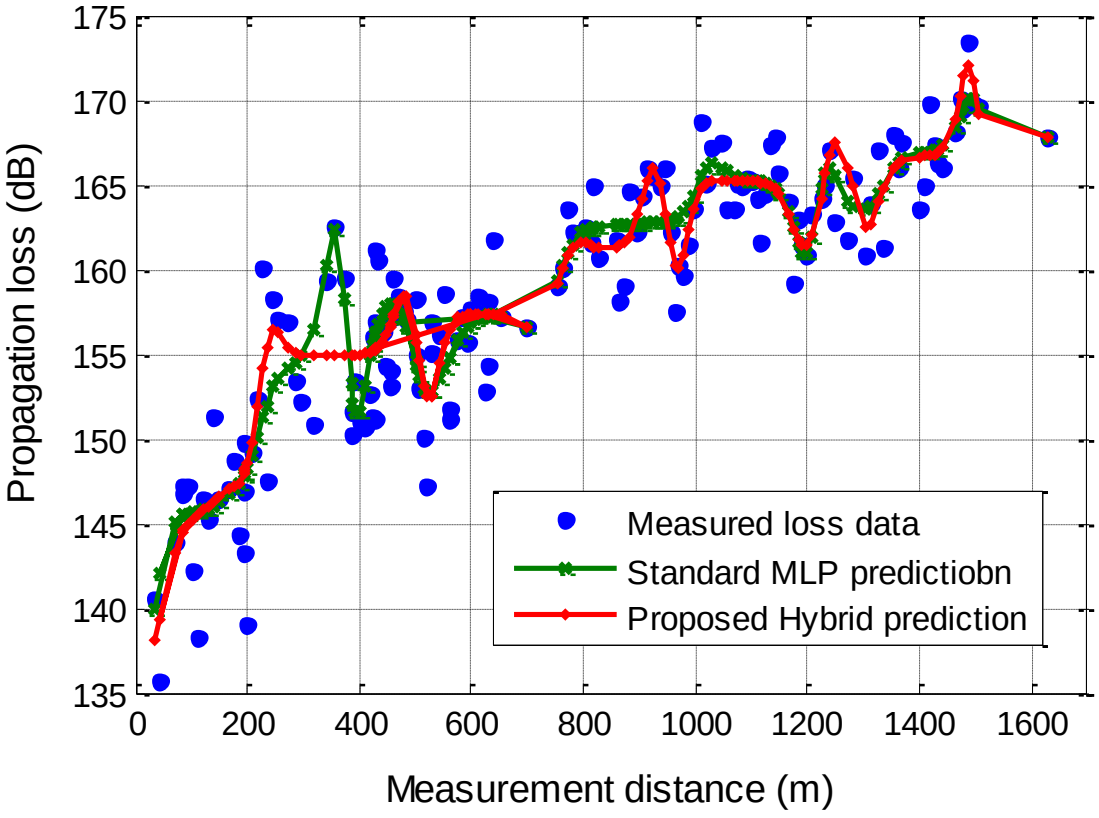
$$V_n = \frac{\left(V_o - V_{\min}\right)}{\left(V_{\max} - V_{\min}\right)} \tag{6.6}$$

where $V_{o,}$ $V_n$, $V_{max}$ and $V_{min}$ are original value of the parameter, normalized value, maximum parameter value and minimum parameter value respectively.

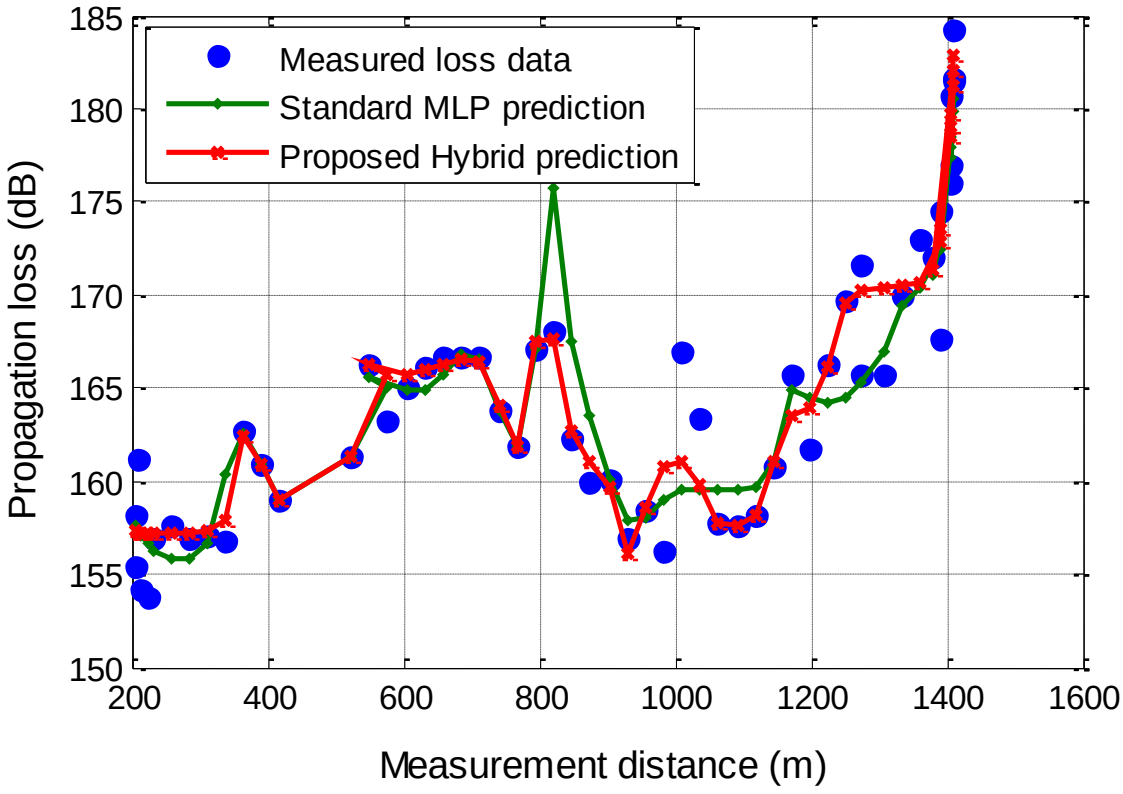
The results of ANNs prediction accuracy are measured in terms of *four* statistical performance indicators: RMSE, MAE, SD and r. Both MAE and RMSE express the mean error magnitude between actual observation and prediction result. The SD articulates the measure of dispersion between actual observation and prediction result while r measures the strength of relationship between actual observation and prediction result.

## 6.3. Results and Discussions of this Work

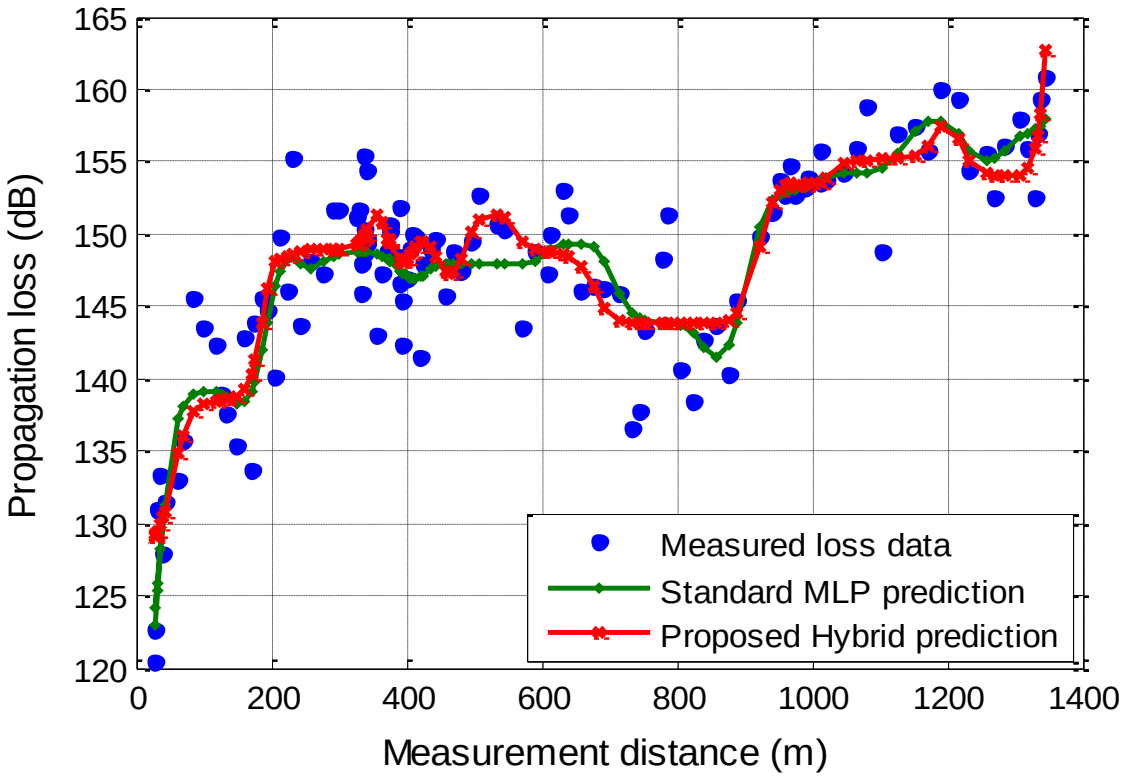
To analyse the signal prediction efficiency of the proposed neural adaptive model, five different study locations as earlier mentioned are considered. The respective results obtained for locations *1* to *5* after employing the various ANN prediction approaches are presented on the plotted graphs in figures 6.4 (a, b, c, d & e).


Propagation loss (dB)
175
170
165
160
155
150
145
140
135
0
200
400
600
800
1000
1200
1400
1600
Measurement distance (m)
Measured loss data
Standard MLP predictiobn
Proposed Hybrid prediction

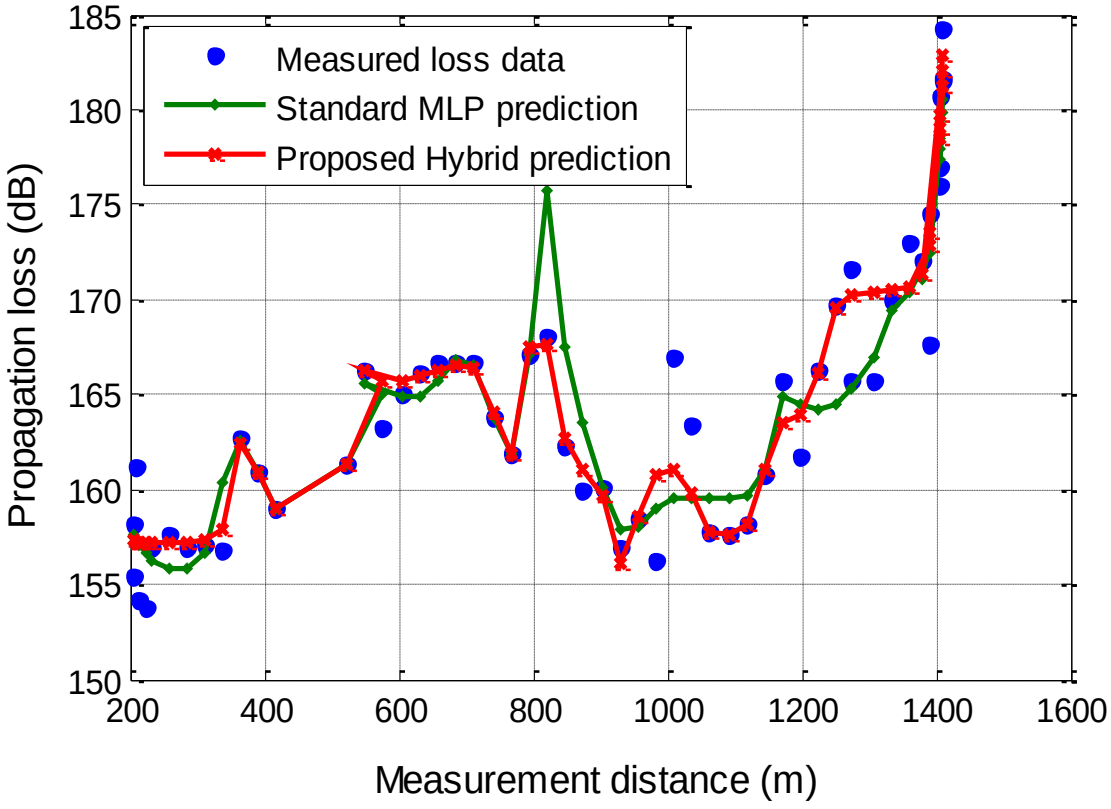
(a)


Propagation loss (dB)
185
180
175
170
165
160
155
150
200
400
600
800
1000
1200
1400
1600
Measurement distance (m)
Measured loss data
Standard MLP prediction
Proposed Hybrid prediction

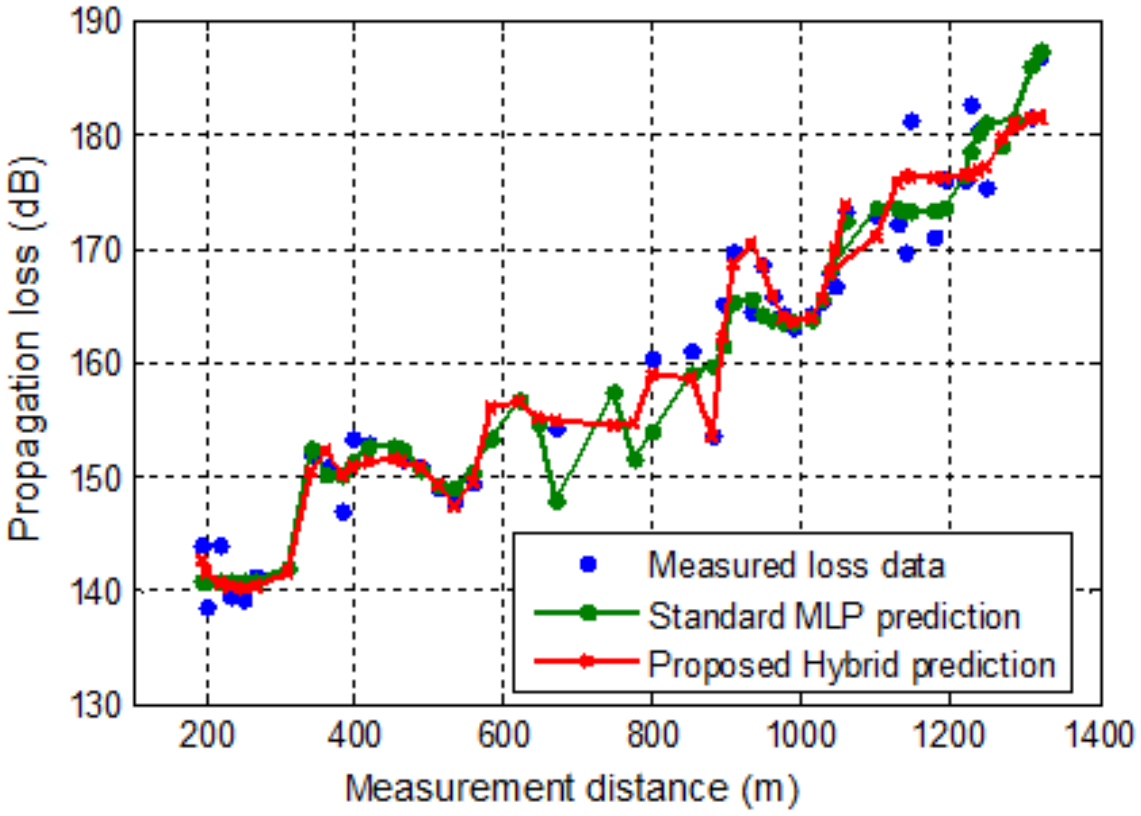
(b)


Propagation loss (dB)
Measurement distance (m)
Measured loss data
Standard MLP prediction
Proposed Hybrid prediction

(c)


Propagation loss (dB)
Measurement distance (m)
Measured loss data
Standard MLP prediction
Proposed Hybrid prediction

(d)




(e)

Figure 6.4. Proposed ANN prediction output with measurement signal data versus distance in location (a) 1; (b) 2; (c) 3; (d) 4; and (e) 5.

In the graphs, blue colour indicates measured signal power data, red and green colours indicate prediction made by the proposed ANN model and conventional MLP model respectively. The graphs show fluctuation and decreasing signal power as function of the measurement distance between UE terminal and BS transceiver, especially in locations *1* and *3*. The phenomenon can be credited to different cluttered environmental factors and multipath radio wave propagation mechanisms that act on transmitted signal power, thus resulting to huge signal path losses. More importantly, from the displayed plotted graphs in figure 6.4 (a, b, c, d & e) it is clear that the proposed adaptive ANN model yielded a better signal prediction accuracy with minimal MAE, SD, RMSE and increased r in comparison to the accustomed MLP prediction approach. Also, the tabulated result shows prediction with minimal errors using Lenvenberg-Marquardt algorithm on ANN architecture in comparison to Resilient backpropagation and Bayesian regularization algorithms.

The overall performance results are summarized in Tables 6.1 (a, b, c, d & e). For clarity and conciseness, only the prediction results using Lenvenberg-Marquardt algorithm are represented in the graphs for performance comparison between proposed adaptive hybrid neural network predictor and the standard MLP prediction approach. Other results using Resilient backpropagation and Bayesian regularization algorithms are all summarized in Tables 6.1 (a, b, c, d, & e).

Table 6.1. Computed performance metrics with MAE, RSME, SD and r for Location (a) 1, (b) 2, (c) 3, (d) 4, (e) 5.

(a)

| **Training pattern** | **MAE** | **RMSE** | **SD** | **r** |
|---|---|---|---|---|
| MLP with LM | 2.427 | 3.079 | 1.900 | 0.930 |
| MLP with BR | 1.937 | 2.546 | 1.653 | 0.941 |
| MLP with RP | 4.382 | 5.381 | 3.123 | 0.791 |
| Hybrid with LM | 2.022 | 2.730 | 1.835 | 0.952 |
| Hybrid with BR | 1.847 | 2.463 | 1.623 | 0.945 |
| Hybrid with RP | 2.550 | 3.215 | 1.956 | 0.906 |

(b)

| **Training pattern** | **MAE** | **RMSE** | **SD** | **r** |
|---|---|---|---|---|
| MLP with LM | 2.347 | 3.052 | 1.951 | 0.914 |
| MLP with BR | 2.306 | 2.945 | 1.832 | 0.918 |
| MLP with RP | 2.583 | 3.403 | 2.216 | 0.892 |
| Hybrid with LM | 2.031 | 2.724 | 1.812 | 0.932 |
| Hybrid with BR | 2.121 | 2.742 | 1.738 | 0.930 |
| Hybrid with RP | 2.128 | 2.844 | 1.886 | 0.924 |

(c)

| Training pattern | MAE | RMSE | STD | R |
|---|---|---|---|---|
| MLP with LM | 4.107 | 6.632 | 3.414 | 0.744 |
| MLP with BR | 3.902 | 5.265 | 3.522 | 0.775 |
| MLP with RP | 3.584 | 5.366 | 3.144 | 0.804 |
| Hybrid with LM | 2.493 | 3.598 | 2.593 | 0.909 |
| Hybrid with BR | 3.512 | 4.986 | 3.594 | 0.811 |
| Hybrid with RP | 3.512 | 4.867 | 3.428 | 0.811 |

(d)

| **Training pattern** | **MAE** | **RMSE** | **SD** | **r** |
|---|---|---|---|---|
| MLP with LM | 1.850 | 2.640 | 1.890 | 0.933 |
| MLP with BR | 2.410 | 2.983 | 1.756 | 0.914 |
| MLP with RP | 3.193 | 4.010 | 2.401 | 0.853 |
| Hybrid with LM | 1.379 | 1.993 | 1.439 | 0.962 |
| Hybrid with BR | 1.953 | 2.455 | 1.488 | 0.942 |
| Hybrid with RP | 1.889 | 2.840 | 2.120 | 0.926 |

(e)

| Training pattern | MAE | RMSE | SD | r |
|---|---|---|---|---|
| MLP with LM | 1.801 | 2.687 | 1.995 | 0.980 |
| MLP with BR | 2.792 | 3.392 | 1.925 | 0.967 |
| MLP with RP | 3.683 | 4.701 | 2.921 | 0.939 |
| Hybrid with LM | 1.815 | 2.597 | 1.857 | 0.981 |
| Hybrid with BR | 2.640 | 2.909 | 1.857 | 0.976 |
| Hybrid with RP | 2.156 | 2.898 | 1.936 | 0.977 |

Presented in figures 6.5 and 6.6 are plotted regression graphs using the proposed adaptive MLP and the conventional MLP prediction approach after network training, validation and testing for location *1*. The dashed line and the solid line in the two figures denote the perfect result and best linear regression fit between targets and outputs. Noticeably from the graphs, a closer value of all r to 1 (i.e. All: $r = 0.9506$) indicates how better the predicted output correlate with the targets using proposed ANN approach over the conventional MLP prediction (with All: $r = 0.9477$). The results summary in Tables 6.1 (a, b, c, d & e) for all r also confirm that the proposed approach delivered better prediction accuracy in the other *5* study locations.

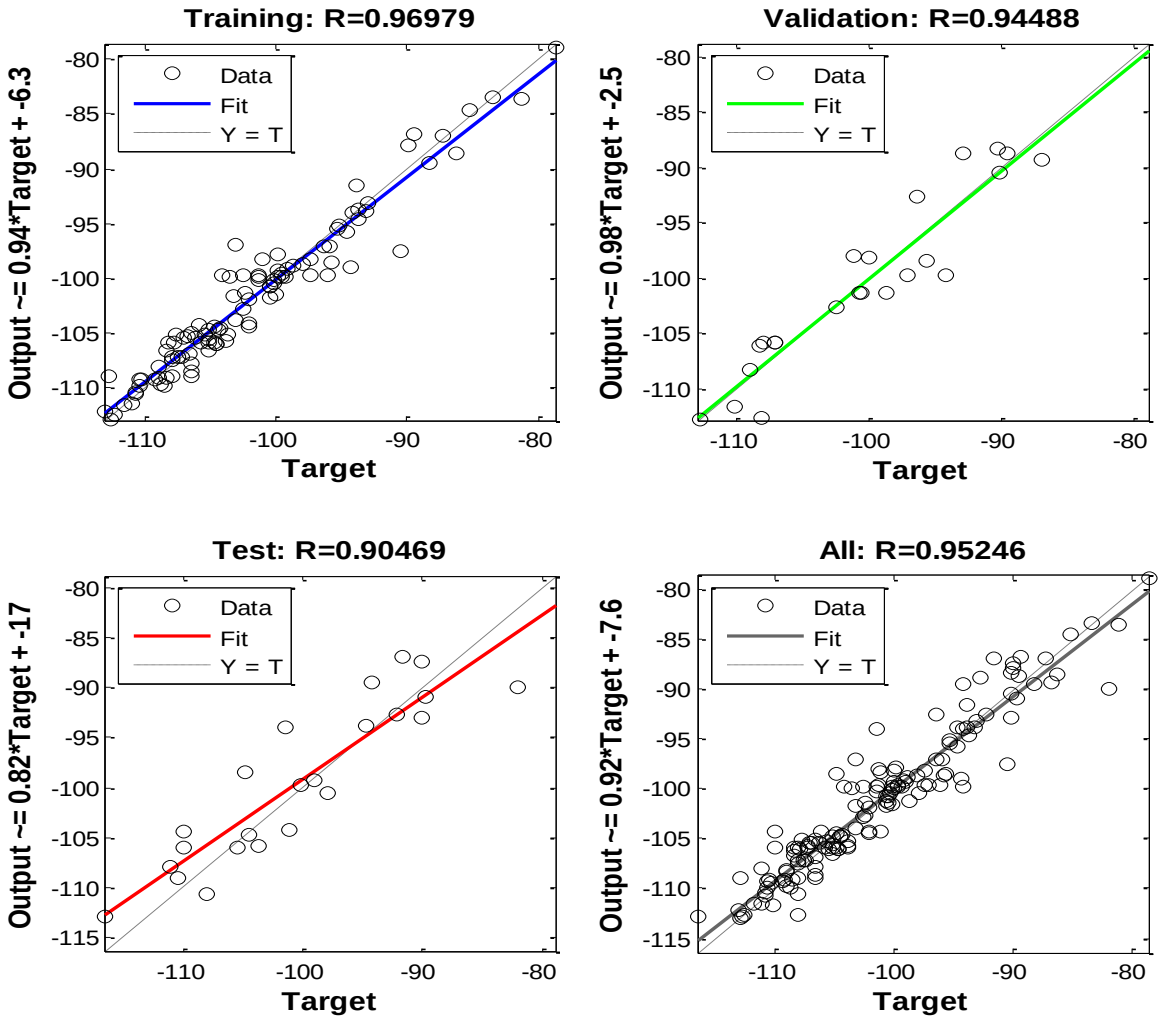



Figure 6.5. Regression coefficient outputs versus targets using the proposed adaptive MLP prediction approach in location 1

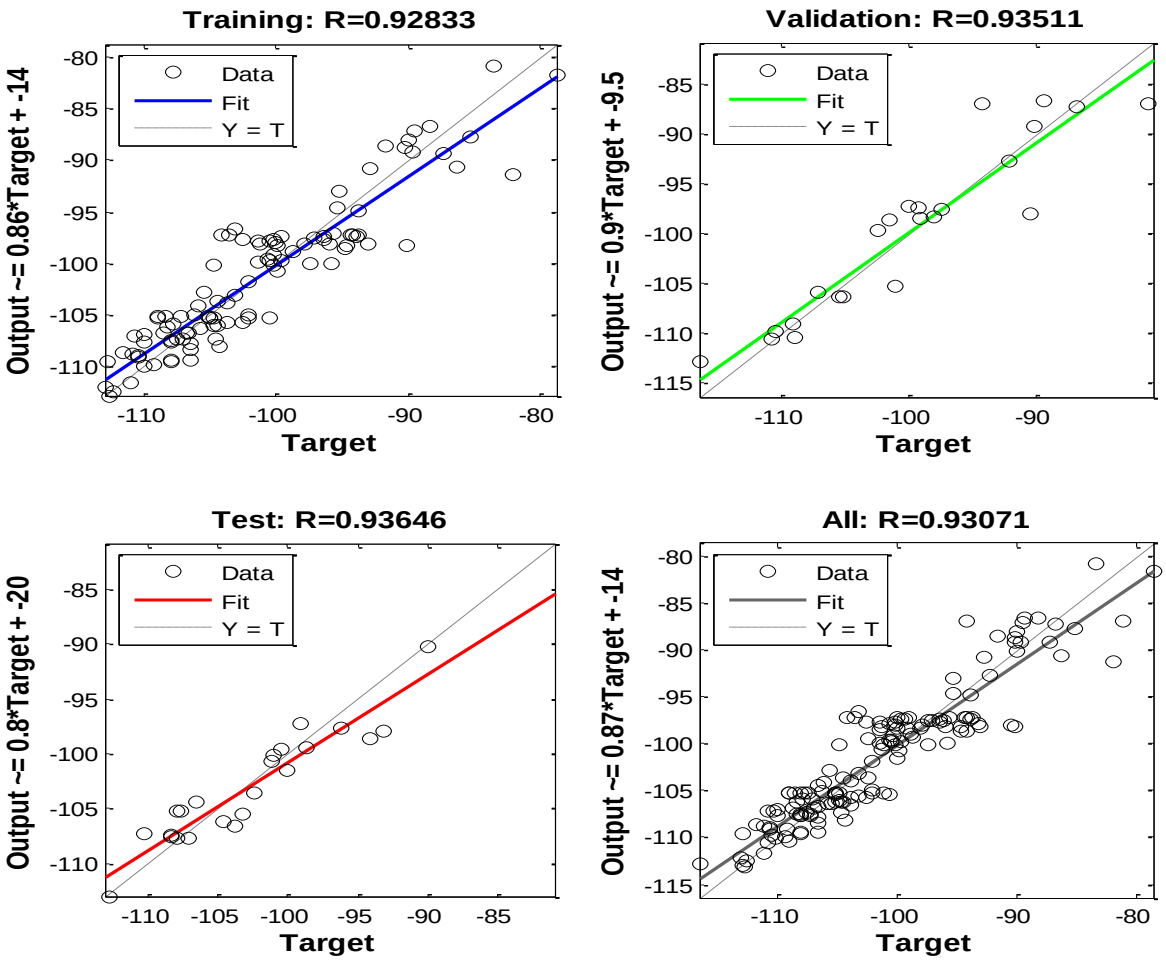



Figure 6.6. Regression coefficient outputs versus targets using the conventional MLP prediction approach in location 1.

## 6.4. Review of Vector Order Statistics Filters based pre-processing technique

Recently, some attentions have been drawn to the area of data pre-processing in boosting the training effectiveness and prediction fastness of neural network-based algorithms. Some of the key data pre-processing techniques includes: sampling (which opt for a subset representation of a bulky datasets), normalization (which standardizes data for a better access), denoising (which succors in removing noise from datasets) and transformation (which helps in manipulating raw datasets to produce a single input).

In data denoising, smoothing and transformation, different techniques such as match filtering, singular spectrum analysis, moving average smoothing, factor analysis, wavelet multi-resolution analysis, etc. have been examined in previous works [194-203]. Specifically, a combination of wavelet denoising method and neural network model were applied for trend prediction in rainfall time series dataset [195, 196, 204]. It was found that the combined wavelet neural network modelling-based prediction is more efficient than using only the ANN models. The same approach was also used in [197] and [198], but for enhanced prediction of underground water levels and earthquake data respectively.

A study by *Wu et.al* [199], presented three data pre-processing approaches involving Moving Average (MA), filtering-based Singular Spectrum Analysis (SSA) and Principal Component Analysis (PCA) with modular neural networks for improved rainfall data predictions in China and Indian. The effectiveness of dynamic filtering and convolution on accurate prediction of video and stereo data was proposed and demonstrated [200]. In [201], a data pre-processing based modelling technique was applied to examine the daily reservoir inflow and they discovered that model-driven prediction accuracy of the uneven inflow of the reservoir only improved when the pre-processed seasonal datasets were used. *Nithya et.al.* in [202] explored the combination of dynamic linear model and Kalman filters to predict missing occurrences in time series sensor data stream and they concluded that the application of the filter with linear model is a viable methodology for boosting the prediction efficiency of sensor data. A similar linear data filtering approach has been employed to analyse and predict cellular network coverage, but using CDMA2000 signal data [203].

The results and conclusion from the previous works above showed that the exploration of the information content in a dataset through pre-processing plays a major role in enhancing the model training and the prediction precision. However, the concentration was majorly on time series datasets with linear smoothing-based data filtering and standardization approach which does not capture the stochastic non-linearity in some multifaceted spatial datasets.

This part of research work proposes and employs vector order statistics filters based pre-processing technique to enhance adaptive trend prediction of stochastic noisy signal power data using ANN model. The proposed predictive approach is termed Vector Median Filters Multi-Layer Perceptron (VMF-MLP).

## 6.5. Order Statistics Filters

Noise is undesirable information that contaminates desired signal. Every measured data or signal contains some amount of noise which may have been added to the desired signal owing to thermal or natural environmental phenomena and other physical accoutrements associated with the signal generation structure and data sampling process. Filtering in the utmost universal term is a method of noise detection and extraction in dataset, to moderate the influence of errors on the succeeding input data analysis. It also helps to enhance or reveal the actual information about a quantity of interest in any given dataset.

Different filtering methods are presented in the literature for denoising signal data. Typically, the techniques can be group into two, namely: (i) linear filtering and (ii) non-linear filtering techniques. The performance of each filtering technique hinges on its ability to detect and remove the presence of noise from desired signal data. Linear filters (e.g. mean filter, wiener filter, and Gaussian filter) are known to perform poorly in the presence of non-additive or non-Gaussian signal dependent noise [205, 206]. The concept of non-linear filtering is centred on the theory of non-linearity. This research work analyses two non-linear order statistics filters namely, VMF and Vector *L* Filters (V*L*F), where *L* symbolizes the typical parameter.

### *6.5.1. Vector Median Filters (VMF)*

The use of median filter was first proposed in 1974 by *Tukey* [207], as time series-based data smoothing method. The VMF is a robust ranked order filters for signal data smoothing and are well suitable when the noise sorts and characteristics are unknown [208-210]. Given the N observations of $x_i$, $i$ = 1, 2, 3, ……$N$, the median $x_{med}$ of the dataset of $x_i$ can be expressed as:

$$f(x) = \sum_{i}^{N} |x_i - x| \tag{6.5}$$

$$f(x_{med}) \leq f(x_i) \vee x_i, x_{med} \in \{x_i, i = 1,...N\}$$

where $x_i$ defines the $k$ dimensional vectors $\left[x_i, x_{i2}, x_{i3},...x_{ik}\right]^T$

### *6.5.2. Vector L Filters (VLF)*

The V*L*F is a generalization of median filters which was first introduced by *Bovik et.al.* [210] utilizing combined linear order statistics. The covariance matrix with respect to the ordered samples are approximated using Taylor expansion defines as [211]:

$$Tn = \sum_{i}^{N} a_i x_{(i)} \text{, where } \sum_{i}^{N} a_i = 1 \tag{6.6}$$

where $a_i s$ expresses the set of weights which describes the performance of the estimators.

## 6.6. Gradient Descent Back Propagation (BP) Algorithm

The gradient descent backpropagation algorithm is one of the well-known neural network algorithms that employs the gradient descent method to minimize the cost function expressed in Eq. (6.4) and to accomplish this, we must have:

$$\Delta w = \frac{\partial E(w)}{\partial w_i} = 0 \tag{6.7}$$

The gradient descent BP algorithm update rule as:

$$\Delta w = e(w) + \nabla \mathrm{w} \tag{6.7}$$

$$\nabla w = -\eta \frac{\partial E}{\partial w} \tag{6.8}$$

and $\eta$ designates learning rate parameter.

## 6.7. Levenberg-Marquardt (LM) Algorithm

Levenberg-Marquardt algorithm is applied to minimize error function problems and it is adapted from Gauss–Newton and gradient descent methods [212]. In correspondence to cost function, Newton's method weight update is given as:

$$\Delta w = -\left[H(w)\right]^{-1} g(w) \tag{6.9}$$

where, g*(w)* and *H(w)* denote the gradient vector and Hessian matrix expressed in Eq. (6.10) and Eq. (6.11) respectively.

$$g(w) = J\ \ (w)^T e(w) \tag{6.10}$$

$$H(w) = J(w)^T J(w) + Q(w) \tag{6.11}$$

with *J(w)* being the Jacobian matrix

$$Q(w) = \sum_{i=1}^{N} \nabla^2 e(w) e(w)^T \tag{6.12}$$

If *Q(w) = 0* for the Gauss- Newton method, then, Eq. (6.9) with the gradient method would become:

$$\Delta w = -\left[J(w)^T J(w) + \mu I\right]^{-1} J^T(w) e(w) \tag{6.13}$$

The expression in Eq. (6.13) is the LM weight update, with $I$ being the identity matrix and $\mu$ the damping parameter.

## 6.8. Proposed Model: Combination of Vector Order Statistic Filters with MLP Network Model

The set-up of the proposed predictive model VMF-MLP is illustrated in figure 6.11. The prediction process is such that the measured signal power datasets are first pre-processed with order statistic filters, and then standardized, before feeding desired resultant output components to MLP network model to enable it deliver training, testing and validation effectively.

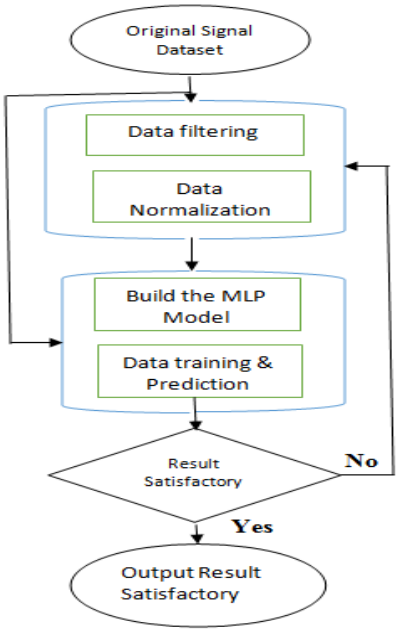



Figure 6.7. Proposed prediction approach based on Combined Vector Order Filtering and MLP neural architecture.

The field test measurements are carried out around *4* operational LTE cellular networks BS sites operating at *1900 MHz* band with concentration on built-up busy urban streets, and roads between JAN-DEC 2017. The area is a flat topography with mixed commercial and residential building edifices. The BS antenna heights range from *28 m* to *45 m*, elevated above the ground level to broadcast signals in three sectors configuration.

With the aid of drive test equipment which include the GPS, HP Laptop, two Samsung Galaxy Mobile Handsets (Model-SY 4) and network scanner, signal power measurement was conducted round the cell sites, in active mode. Both the Samsung handsets and the HP laptop are enhanced with TEMS, *15.1* version, which enable us to access, acquire and extract signal data including serving BS information after measurement. A total of *1,502* signal data points was extracted for further analysis using MapInfo and Microsoft Excel spreadsheet.

The measured signal power, which is Reference Signal Receive Power (RSRP), is related to propagation path loss by:

$$RSRP(dBm) = P_{tx} - G_t - G_r - L_t - L_r + PL \tag{6.4}$$

where $L_r$, $L_t$ *and* $PL$ are the received feeder loss, transmitter loss and path loss, $G_r$,$G_t$ and $P_{tx}$ are receiver gain, transmitter gain and BS transmit power respectively.

A *2013a* MATLAB software platform has been utilized to implement the models. For optimal neural networks learning and training with the three analysed models, the measured signal dataset was shared into three subsets as follows: a training set (*75%* of the data), testing set (*15%* of the data) and validation set (*15%* of the data). The early stopping method has been employed to cater for over-fitting during training. The training embroils the connection weights adjustments such that the network can predict the assigned value from the member training set. Levenberg–Marquardt training algorithm was utilized to ascertain the training capabilities of the investigated MLP, VMF-MLP and VLF-MLP schemes.

As the measured signal power data contains different values with different scales, adjusting and normalizing the dataset to improve the network training phase is very important. Thus, the vector normalization technique is considered and the normalizing equation given by [153]:

$$x_v = \frac{x}{\sqrt{\sum_{l=1}^{n}(x_i)}} \tag{6.15}$$

where $x_v$ and $x_i$ indicate normalized and original data values respectively.

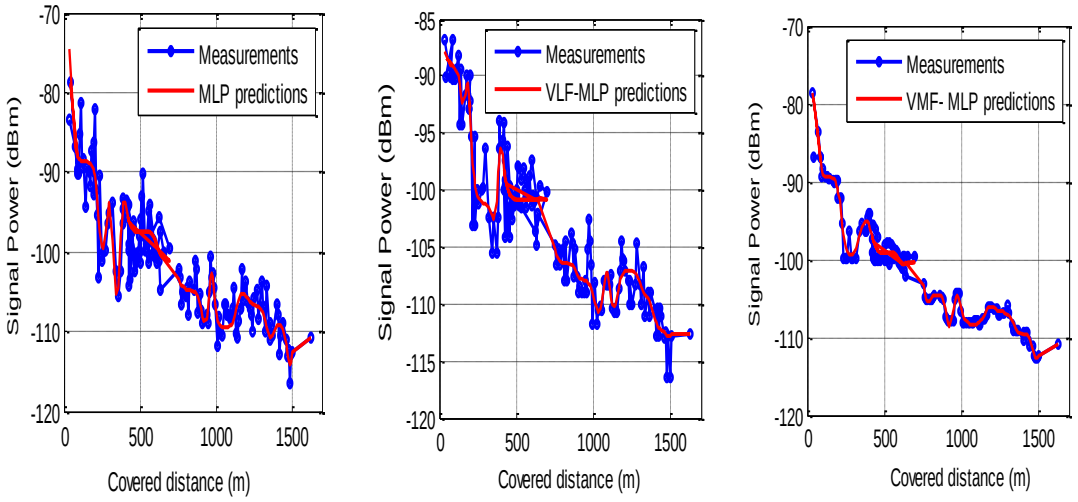
Four statistical performance indices: RMSE, MAE, SD and r have been utilized to analyse the prediction accuracy of each scheme.

## 6.9. Results and Discussion of this Work

The results of MLP, VMF-MLP and VLF-MLP network models utilized to learn and predict the measured LTE signal power are presented and discussed. Figures 6.8 (a, b, & c) to 6.11 (a, b, & c) show the measured signal power and their predictions in locations I to IV, using MLP, VLF-MLP and VMF-MLP models. Table 6.2 shows the summarized performance results of the three neural network models employed to predict measured signal power, using MAE, RSME, SD, and r indices. A closer value of r to *1* indicates better performance in predicting or fitting the actual data. On the other hand, the lower the values of MAE, SD and RMSE are, the better the neural network model prediction performance. From Table 6.2, it is established that VMF-MLP and VLF-MLP models attained the lowest MAE, RSME, SD values in the *4* study locations, when compared to MLP model. Similarly, VMF-MLP and VLF-MLP models also attained the highest prediction accuracy in terms of correlation coefficient as compared to other models in the *4* study locations.

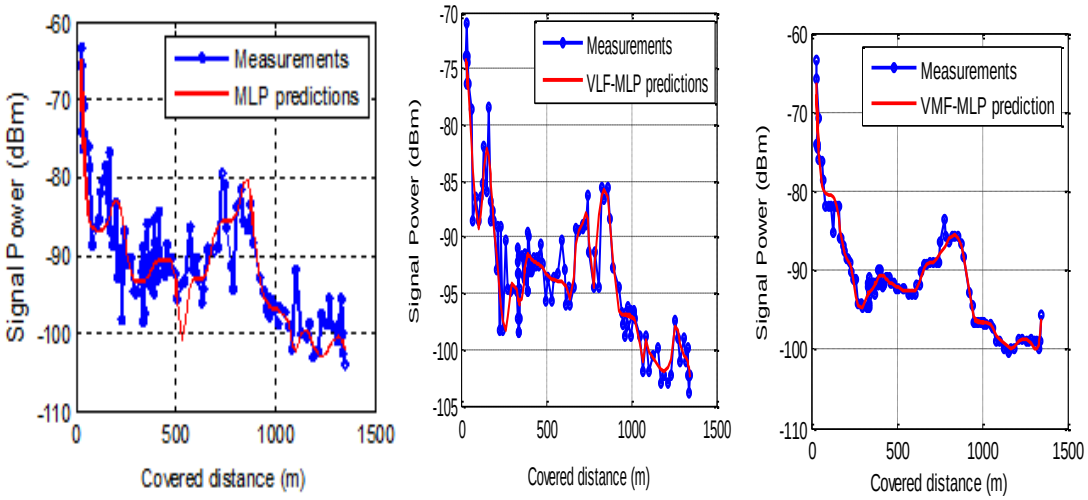
Table 6.2. Computed first order statistics with MAE, RSME, SD and r for Locations 1 to IV

| | **MLP Model Prediction** | | | |
|---|---|---|---|---|
| **Location** | **MAE** | **RMSE** | **SD** | **r** |
| I | 2.071 | 2.777 | 1.849 | 0.931 |
| II | 1.992 | 2.529 | 1.557 | 0.982 |
| III | 2.677 | 2.642 | 2.642 | 0.872 |
| IV | 1.967 | 1.779 | 1.779 | 0.934 |
| | **VMF-MLP Model Prediction** | | | |
| I | 0.696 | 1.079 | 0.825 | 0.988 |
| II | 0.867 | 1.229 | 0.870 | 0.995 |
| III | 0.784 | 1.271 | 1.001 | 0.982 |
| IV | 0.734 | 1.071 | 0.780 | 0.986 |
| | **VLF-MLP Model Prediction** | | | |
| I | 1.464 | 1.947 | 1.284 | 0.957 |
| II | 1.587 | 2.129 | 1.459 | 0.987 |
| III | 1.663 | 2.285 | 1.567 | 0.941 |
| IV | 1.504 | 2.343 | 1.796 | 0.946 |




(a) (b) (c)

Figure 6.8. Signal power loss predictions with(a) MLP; (b) VLF-MLP; and (c) VMF-MLP models versus covered distance in BS site I




(a) (b) (c)

Figure 6.9. Signal power loss predictions with (a) MLP, (b) VLF-MLP, (c) VMF-MLP models versus covered distance in BS site II

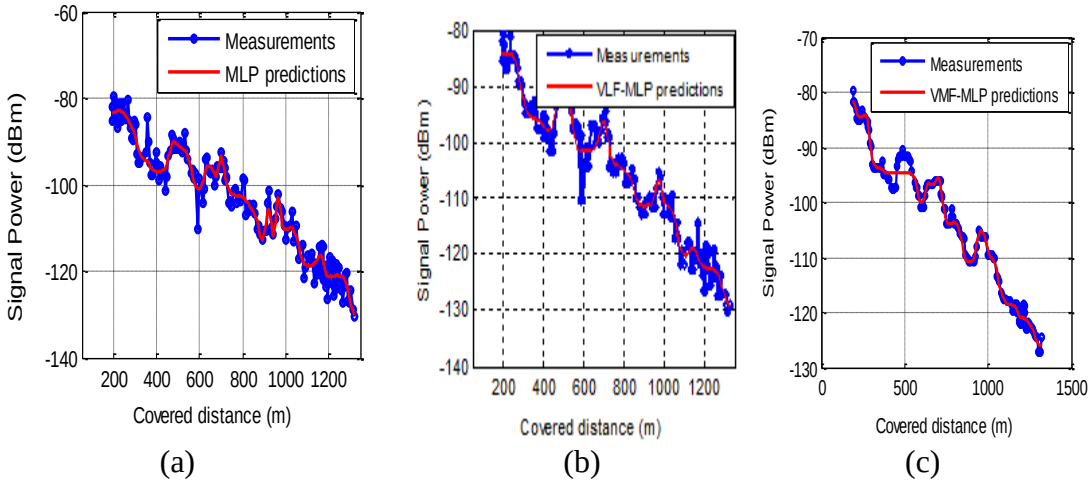



Figure 6.10. Signal power loss predictions with (a) MLP, (b) VLF-MLP, (c) VMF-MLP models versus covered distance in BS site III

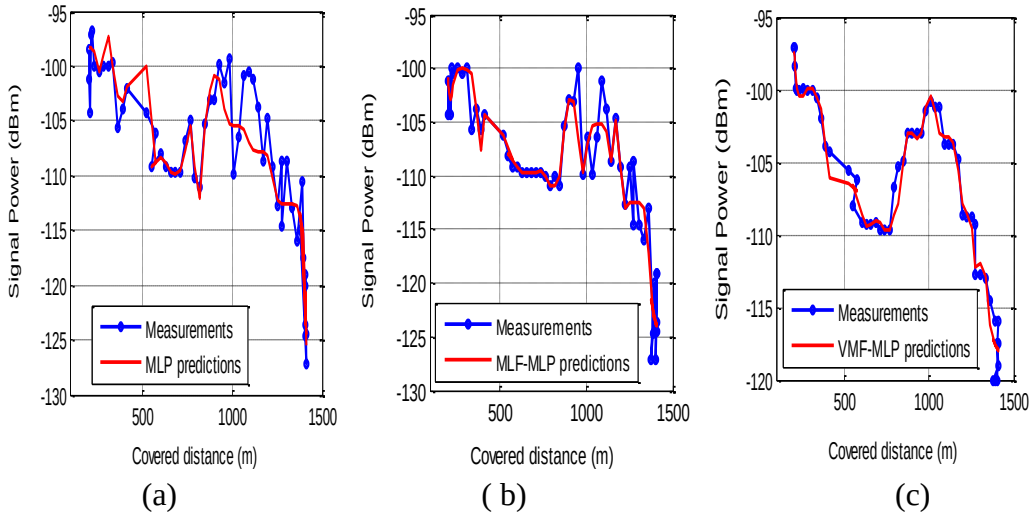



Figure 6.11. Signal power loss predictions with (a) MLP; (b) VLF-MLP; and (c) VMF-MLP models versus covered distance in BS site IV

The spreading and distribution of the mean signal prediction error along the measurement routes using MLP, VLF-MLP and VMF-MLP models during training, testing and validation period are presented in figures 6.12 to 6.15 respectively. From the graphs, minimum error spreads with VMF-MLP and VLF-MLP models along the measurement distance indicates excellent signal prediction accuracy in comparison to standard MLP model. The plotted graphs show that the prediction values from VLF-MLP and VMF-MLP models matched properly and better with measured signal values, compared to MLP model prediction with measured values.

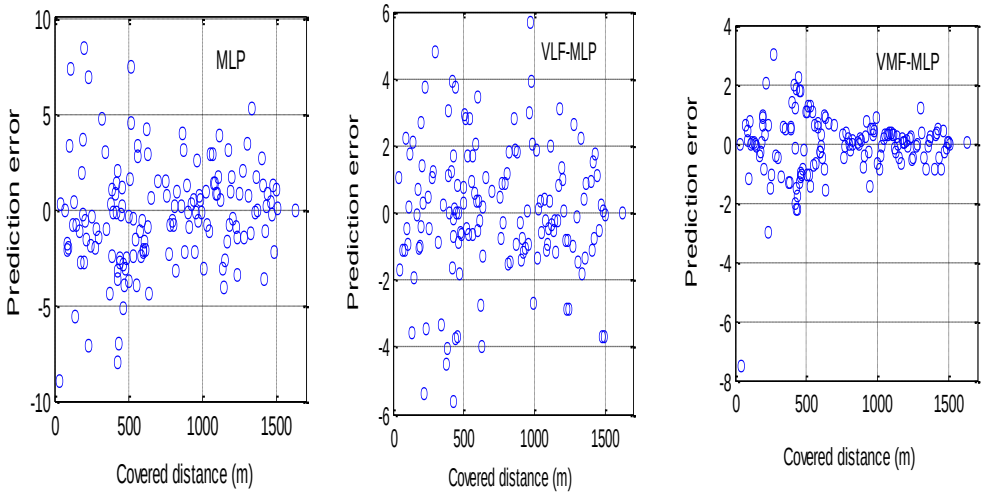



Figure 6.12 Signal power loss prediction error with (a) MLP, (b) VLF-MLP, (c) VMF-MLP models versus covered distance, in BS site I

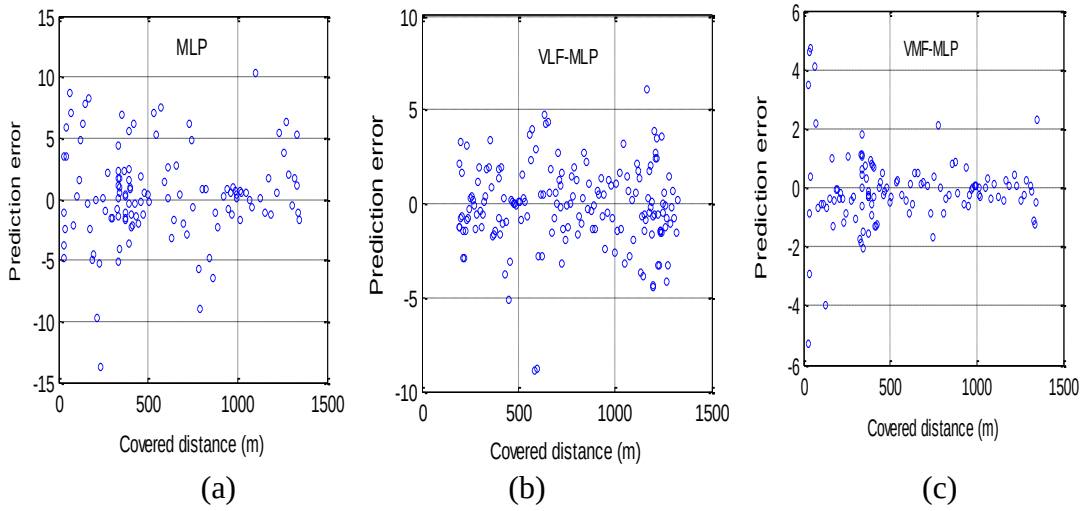



Figure 6.13. Signal power loss prediction error with (a) MLP, (b) VLF-MLP, (c) VMF-MLP models versus covered distance, in BS site II

Graphical results of the *3* analysed ANN models trained with LM algorithm at varied hidden layer sizes for comparative study in locations 1 and II are shown in figure 6.16 (a & b) while comparison of LM algorithm and the gradient descent BP algorithms performances at varied hidden layer sizes if MLP ANN is presented in figure 6.17. The results clearly demonstrate that the prediction in terms of RMSE of VMF-MLP model trained with LM algorithm was lower in comparison to that trained with gradient descent BP algorithm and the

LM algorithm has lower RMSE than the gradient descent BP algorithm in all the variation of the hidden layer sizes.

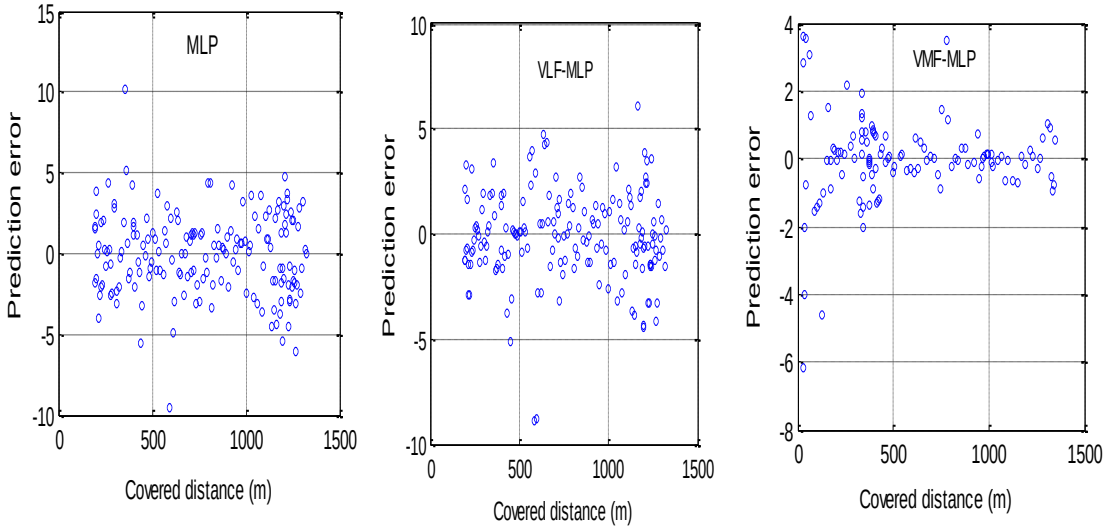



Figure 6.14. Signal power loss prediction error with (a) MLP, (b) VLF-MLP, (c) VMF-MLP model versus covered distance, in BS site III

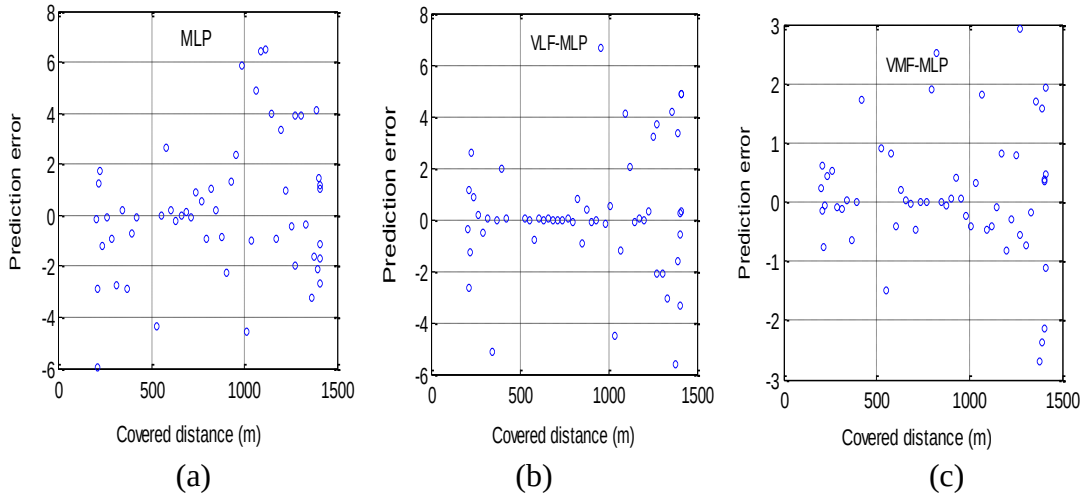



Figure 6.15. Signal power loss prediction error with (a) MLP, VLF-MLP. VMF-MLP models versus covered distance, in BS site IV

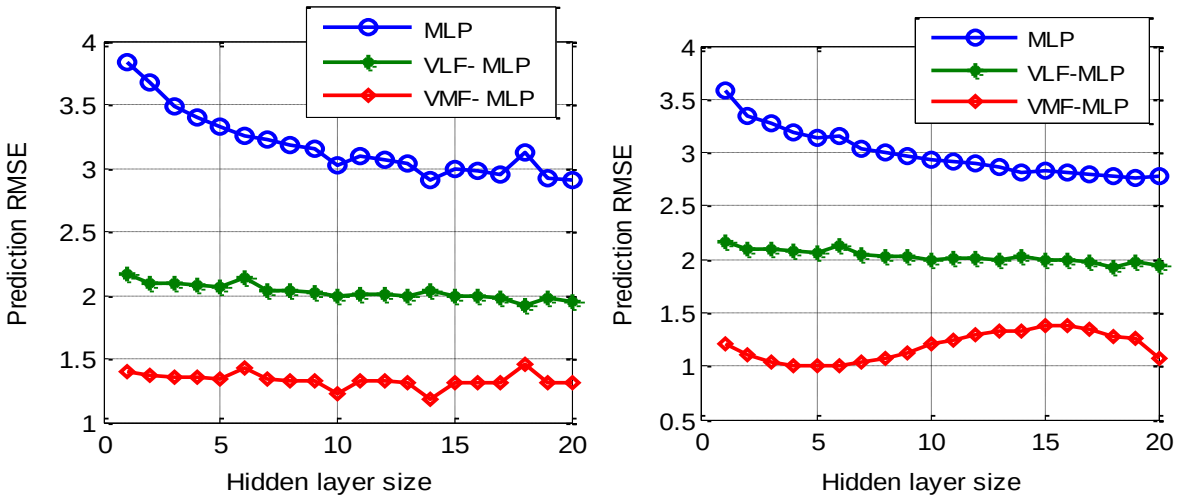



Figure 6.16. Signal power loss prediction, RMSE error using the three investigated models trained with LM algorithms at different Hidden layer size, in (a) BS site I (b) BS site II.

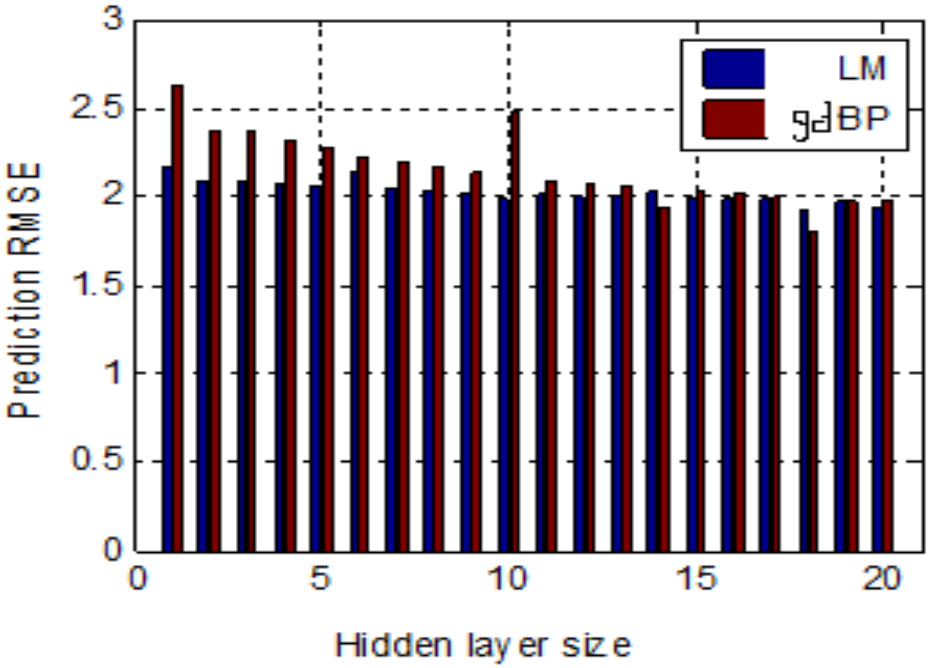



Figure 6.17. Signal power loss prediction in terms of RMSE, in MLP at different hidden layers size from comaprison of LM and gradient descent BP algorithms, in BS site III

## 6.4. Chapter Summary

This chapter analyses and utilizes the proposed adaptive hybrid neural network predictor that combines MLP ANN and ADALINE ANN for enhanced propagation PL prediction in a micro-cellular urban environment. The prediction accuracy of the hybrid predictor has been assessed using field test-based propagation loss data collected from *5* selected locations from LTE cellular networks environment with built-up busy urban streets, roads and open areas with mixed residential and commercial building structures. By means statistical performance evaluation metrics: RMSE, MAE, SD and r, the proposed adaptive hybrid approach provides a better prediction accuracy in comparison to the standard MLP ANN prediction approach. The superior performance of the hybrid neural network predictor can be attributed to its capability to learn, adaptively respond and predict the fluctuating patterns of the reference propagation loss data during network training.

Furthermore, a vector order statistics filter which is termed VMF-MLP that is based on the technique of pre-processing was tested, evaluated and utilized for an enhanced adaptive trend prediction of signal power loss using MLP ANN. The performance of the proposed model was compared with VLF-MLP built on MLP ANN and standard MLP ANN. By means of statistical performance indices: RMSE, MAE, DS and r, the adaptive prediction results on LTE signal power data collected from *4* study locations in urban terrain demonstrates that VMF-MLP model performs considerably better compared to VLF-MLP and standard MLP prediction approach. This rightly validates that the pre-processing of the information content in datasets enhances its training and prediction accuracy with neural network models.

# CHAPTER 7

# CONCLUSIONS AND FUTURE WORKS

## 7.1. Conclusion

The aim of this research work is to analyse and identify the limitations of traditional propagation Path Loss (PL) models as well as different Artificial Neural Network (ANN) models and proffer an improve hybrid adaptive ANN models that will cater for the limitations of (i) traditional PL models and (ii) conventional ANN models in modelling a micro-cellular outdoor environment using real world data from Long Term Evolution (LTE) cellular networks. The research objectives have been set to reflect the aim and to guarantee achieved set goals. As a result, this research work considered traditional propagation PL modelling techniques: the empirical and the deterministic models, prediction of PL with different traditional models and conventional ANN models: Multi-Layer Perceptron (MLP) neural network, Radial Basis Function (RBF) neural network, Generalized Regression Neural Network (GRNN) and Adaptive Linear Element (ADALINE) neural network. Different ANN training algorithms and training techniques have been analysed to ascertain their performances during the network training.

A hybrid adaptive neural network model that combines ADALINE model and MLP ANN has been developed to cater for the stochastic signal attenuation phenomenon and the heterogeneity of the spatial propagation channels in various LTE environments. Furthermore, another improved PL prediction model that combines vector order statistics based smoothing technique and a MLP ANN has been developed. These works have been successfully executed in different submitted, accepted journal and conference papers presented in this thesis.

Chapter 2 of this thesis deals with a detailed review of baseline knowledge on traditional propagation modelling techniques in wireless cellular networks. It considered and studied key empirical and deterministic models. The analysed models have their own distinctiveness and limitations. They are affected by factors such as reflection, refraction,

diffraction and multi-path phenomenon. Also, terrain profile, Base Station (BS) operating frequency, transmitter antenna height, and receiver antenna height all affects the performances of these models. Furthermore, review of some of the conventional ANN such as MLP ANN, RBF ANN, GRNN, ADALINE network has been done considering their peculiarities, characteristics and limitations. Finally, literature review of some past works on the modelling of propagation PL using different traditional empirical and deterministic models as well as conventional ANN models has been done.

Simulation and analysis of propagation PL using traditional models were carried out in chapter 3 with real world data from LTE cellular network to ascertain the effects of propagation parameters on PL. Each of the considered models has been studied and compared under the variation of different parameters such as transmitter-receiver distance, base and mobile station antenna heights and different transmitting frequencies. The simulation results validate increase in PL as the link distance between transmitter and receiver increases because of spreading and attenuation of electromagnetic energy by various propagation mechanisms. There is also increase in PL as the operating frequency increases as a result of decrease in both antenna aperture and wavelength of the radio signal. However, at the increase of both the transmitting and the receiving antenna heights, there was a drastic reduction in PL as the losses due to building roof tops to street diffraction are reduced. The research work in this chapter concludes with the need for an improved PL model that will be adaptive to any environment while combining the benefits of both the empirical and the deterministic models.

The predictions of propagation PL in different areas has been analysed in Chapter 4 using conventional ANN model: the MLP ANN. The focus was on the prediction performance of different ANN training algorithms during the network training using real world data from LTE cellular network. First order statistical performance indices: Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Standard Deviation (SD) and Correlation Coefficient (r) has been used for error measurement during network training. Comparisons of the training results show an outstanding performance with Bayesian Regularization (BR) algorithm with the least measurement errors and the highest correlation coefficients, thus the best in terms of accuracy, but require longer training time. Lavenberg-Marquardt (LM) algorithm performs best in terms of training speed. However, the error margins between Bayesian regularization algorithm and Lavenberg-Marquardt algorithm is

very minimal. Also, Bayesian regularization algorithm is an improvement on Lavenberg-Marquardt to address poor network generalization by addition of a small overhead to the Hessian approximation already in existence in Lavenberg-Marquardt algorithm.

In part I of chapter 5, different ANN training techniques to overcome the tendencies of over-fitting thereby avoiding poor network generalization during network training has been investigated and the results compared using two neural network models: the RBF ANN and the MLP ANN. These techniques include neuron variation in the hidden layer of neural network architecture, Bayesian regularization approach and early stopping approach. For MLP ANN, the network demonstrates an excellent capability of modelling a moderate size network with *40* neurons in hidden layer. Subsequent increase in neuron results to over-fitting showing the need for intermediary layers to improve network generalization. However, with RBF ANN, network generalization ability increases with increase in the number of neurons in the hidden layer. This is because of its fixed three-layer architecture, hence there is no poor network generalization resulting from architectural intricacy. Training the RBF ANN using early stopping approach shows a better PL prediction with lesser measurement errors using first order statistical performance indices in comparison to MLP ANN. Finally, training the RBF ANN shows no changes during the network training using Bayesian regularization approach. This is because of it fixed three-layer structure, hence, there is no poor generalization as a result of architectural complexity which Bayesian regularization approach addresses.

Part II of chapter 5 presented an evaluation of the effect of learning rate on two ANN models: the GRNN model and the MLP ANN model while also analysing the effect of spread factor in GRNN and the effect of different combination of non-linear and linear transfer functions in hidden layer of MLP ANN. Small spread factor leads to good network generalization in GRNN while hyperbolic tangent and logistic transfer function in hidden and output layer of MLP ANN out-performed other combination of transfer functions. Also, small spread factor in GRNN requires small learning rate for an excellent prediction with minimal errors while MLP ANN requires higher learning rate.

In part III of chapter 5, the prediction effectiveness of ADALINE and MLP ANN have been analysed using data from two base stations. Data from BS 1 is used to analyse the effect

of learning rate on the two different ANN models while data from BS 2 is used to validate the results of analysis from base station 1. The gradient and the momentum parameter of the two ANN models have been analysed at different variation of learning rates. Adaptive linear element neural network shows good prediction capability using small learning rate but at higher learning rate, the gradient is approximately zero which approximate to the local minima. The reverse was the case for MLP ANN model. However, there is need for an adequate learning rate to be selected to ensure increase convergence but not too high learning rate that will lead to over-fitting during network training. Hyperbolic tangent and logistic sigmoid perform excellently in hidden and output layers of MLP ANN while purelin transfer function performs well with ADALINE neural network.

A hybrid adaptive neural network predictor that combines ADALINE neural network and MLP ANN has been developed and presents in part I of chapter 6. The prediction accuracy of the developed model has been tested and analysed using real world data from LTE cellular network environment with varied residential, commercial and clustered buildings. Comparison of the prediction accuracy of the hybrid adaptive neural model using first order statistical performance evaluation indicators give better PL prediction accuracy than the analysed conventional ANNs. The superior performance of the hybrid adaptive neural network is a result of its adaptive response and ability to predict the fluctuating patterns of the cited propagation loss data in course of network training. Furthermore, in part II of chapter 6, a second model has been developed where a vector order filters based pre-processing method built on MLP ANN is used to enhance adaptive prediction trend of the stochastic noisy data. The developed model shows that pre-processing of signals enhances the training and prediction accuracy.

## 7.2. Future work

In the course of this research work, different areas which are worthwhile for further studies and development has been encountered. The best training technique that will be adaptive to all neural network models while enhancing the network performance still require further investigation. Details of the synergies the new developed hybrid model draws from ADALINE and MLP ANN will be further studied. Furthermore, studies and development of other probable hybrid ANN models and comparison of their characteristics and performances

with hybrid network such as Adaptive Neuro-Fuzzy Inference System (ANFIS) and the developed hybrid ANN in this work will be exploited. Also, there is need to utilize other machine learning techniques in prediction of propagation PL for prediction performance comparison between them and the developed adaptive hybrid neural network model. Further work will be built on 5G cellular network and the impact of the atmospheric and environmental influences on data samples will be critically studied and analysed.

# REFERENCE


[1] P. S. Sotirios and K. Siakavara, "Mobile radio propagation path loss prediction using artificial neural networks with optimal input information for urban environments," *International Journal of Electronics and Communications* pp. 1453-1463, 2015.

[2] H. L. Bertoni, *Radio propagation for modern wireless systems*. Upper Saddle River, New Jersey: Prentice Hall, 2000.

[3] T. S. Rappaport, *Wireless communications: Principles & practice*, 2nd ed. Upper Saddle River, New Jersey: Prentice Hall 1996.

[4] H. Andrej, K. Gorazd, and J. Tomaz, "A Survey of radio propagation modeling for tunnels," *IEEE Communications, Surveys & Tutorials* vol. 16, pp. 658-669, 2014.

[5] T. S. Rappaport, *Wireless communications: Principles & practice*, 2nd ed. Upper Saddle River, New Jersey: Prentice Hall, 2001.

[6] N. Andrea, C. Cecchetti, and A. Lipparwi, "Fast prediction of performance of wireless links by simulation trained neural networks," *Proceeding of IEEE Microwave Theory and Techniques Society Digest*, pp. 429-432, 2000.

[7] M. Yang and W. Shi, "A linear least square method of propagation model tuning for 3G radio network planning," *IEEE Fourth International Conference on Natural Computation*, pp. 150-154, 2008.

[8] D. Chhaya, P. Marehalli, and D. Pankaj, "Tuning of COST-231 Hata model for radio wave propagation predictions," *Computer Science and Information Technology* pp. 255-267, 2012.

[9] J. B. Andersen, T. S. Rappaport, and S. Yoshida, "Propagation measurements and models for wireless communications channels," *IEEE Communications Magazine*, vol. 3, pp. 42-49, 1995.

[10] K. S. Tapan, J. Zhong, K. Kyungjung, M. Abdellatif, and S.-P. Magdalena, "A survey of various propagation models for mobile communication," *IEEE Antennas and Propagation Magazine* vol. 45, pp. 51-82, 2003.

[11] P. Caleb, S. Douglas, and G. Dirk, "A survey of wireless path loss prediction and coverage mapping methods," *IEEE Communication, Survey & Tutorials*, vol. 15, pp. 255-270, 2013.

[12] R. Mardeni and K. F. Kwan, "Optimization of Hata propagation prediction model in sub-urban area in Malaysia," *Progress in Electromagnetics Research* vol. 13, pp. 91-106, 2010.

[13] P. M. Shankar, *Introduction to wireless system*, 2nd ed.: John Wiley & sons, 2002.

[14] J. Isabona and C. C. Konyeha, "Urban area path loss propagation prediction and optimisation using Hata model at 800MHz," *Journal of Applied Physics* vol. 3, pp. 08-18, 2013.

[15] J. C. Ogbulezie, M. U. Onuu, J. O. Ushie, and B. E. Usibe, "Propagation models for GSM 900 and 1800 MHz for Port Harcourt and Enugu, Nigeria," *Network and Communication Technologies*, vol. 2, pp. 1-6, 2013.

[16] N. Aleksandar, N. Natasa, and P. George, "Modern approaches in modeling of mobile radio systems propagation environment," *IEEE Communication Survey*, vol. 3, pp. 2-12, 2000.

[17] K. Imranullah, C. E. Tan, and A. K. Shakeel, "Performance analysis of various path loss models for wireless network in different environments," *International Journal of Engineering and Advanced Technology* vol. 2, pp. 255-267, 2012.

[18] Ostlin, "Macro-cell path loss prediction using artificial neural networks," *IEEE Transactions on Vehicular Technology*, vol. 59, pp. 2735-2744, 2010.

[19] G. Wolfle and F. M. Landstorfer, "Dominant paths for the field strength prediction," *48th IEEE Conference on Vehicular Technology*, vol. 1, pp. 552-556, 1998.

[20] F. a. A. Ignacio, A. R. A. Juan, and P. e. F. a. Fernado, "Influence of training set selection in artificial neural network-based propagation path loss prediction," *International Journal of Antennas and Propagation*, vol. 12, pp. 351-487, 2012.

[21] S. Garba, M. B. Mu'azu, and D. D. Dajab, *Modeling QoS in a telecommunications network using neuro-fuzzy logic: Quality of service in the GSM industries using ANFIS as modeling tool*, 2nd ed.: Lambert Academic Publishing, 2012.

[22] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, *Learning internal representations by error propagation*, 1st ed. Cambridge,Massachusetts: MIT Press, 1986.

[23] M. T. Hagan, H. B. Demuth, and M. H. Beale, *Neural network design*. Boston ,MA: PWS, 1996.

[24] M. O. Omae, E. N. Ndungu, P. L. Kibet, and H. Tarus, "Artificial intelligence approach to signal propagation modeling for outdoor to indoor wireless communication networks: A proposed study," *Proceedings of the Mechanical Engineering Conference on Sustainable Research and Innovation* vol. 4, pp. 12-16, 2012.

[25] S. Bernard, *Digital communications: Fundamentals & applications*, 2nd ed.: Bernard Goodwin, 2005.

[26] J. P. Kiran, D. N. Vishal, and G. Rajkot, "Comparative analysis of path loss propagation models in radio communication," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 3, pp. 1-4, 2015.

[27] J. Milanovic, S. R. Drlje, and K. Bejuk, "Comparison of propagation models accuracy for WiMax on 3.5GHz," *14th IEEE International Conference on Electronics, Circuits and System* vol. 4, pp. 111-114, 2007.

[28] T. Rappaport, *Wireless communications: Principles & practice*. Upper Saddle River, New Jersey: Prentice Hall, 2002.

[29] S. R. Saunders and F. R. Bonar, "Explicit multiple building diffraction attenuation function for mobile radio wave propagation.," *Electronics Letters*, vol. 27, pp. 1276-1277, 1991.

[30] S. R. Saunders and F. R. Bonar, "Prediction of mobile radio wave propagation over buildings of irregular heights and spacings," *IEEE Transaction Antennas Propagation*, vol. 42, pp. 137-143, 1994.

[31] O. Yasuhiro and T. Koichi, "Advance LOS path loss model in microwave mobile communications," presented at the 10th International Conference on Antennas and Propagation, Edinburgh, United Kingdom, 1997.

[32] N. N. Neskovic, "Micro-cell electric field strength prediction model based upon artificial neural networks," *International Journal of Electronics and Communication*, vol. 64, pp. 733-738, 2010.

[33] A. Alatishe, O. Adu, A. Atayero, and F. Idachaba, "A performance review of the different path loss models for LTE network planning," *World Congress on Engineering*, vol. I, pp. 1-8, 2014.

[34] N. Aleksandar and N. Natasa, "Micro-cell electric field strength prediction model based upon artificial neural networks " *Internation Journal of Electronics and Communications* vol. 64, pp. 733–738, 2010.

[35] E. Anderson, G. Yee, C. Phillips, D. Sicker, and D. Grunwald, "The impact of directional antenna models on simulation accuracy," *7th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks* pp. 1-7, 2009.

[36] V. S. Abhayawardhana, I. J. Wassell, D. Crosbsy, M. P. Sellars, and M. G. Brown, "Comparison of empirical propagation path loss models for fixed wireless access systems," *IEEE Vehicular Technology Conference*, vol. 1, pp. 73-77, 2005.

[37] J. Sumit, "Outdoor propagation model s- A literature review," *International Journal on Computer Science and Engineering* vol. 4, pp. 281-291, 2012.

[38] Y. L. C. De Jong, D. Camir´e, and D. V. Rogers, "Comparison of radio propagation characteristics at 700 and 2500 MHz pertaining to macro-cellular coverage," *Communications Research Centre* vol. 3, pp. 1-32, 2011.

[39] V. Erceg, L. J. Greenstein, S. Tjandra, S. R. Parkoff, A. Gupta, B. Kulic, *et al.*, "An empirically-based path loss model for wireless channels in sub-urban environments," *Global Telecommunications Conference,* vol. 2, pp. 922-927, 1998.

[40] S. M. Michael and K. Michael, "Comparison of empirical propagation path loss models for mobile communication," *Computer Engineering and Intelligent Systems,* vol. 5, pp. 1-4, 2014.

[41] "ITU-R terrestrial land mobile radiowave propagation in the VHF/UHF bands " 2002.

[42] M. S. Mollel and M. Kisangiri, "Comparison of empirical propagation path loss models for mobile communication," *Computer Engineering and Intelligent Systems* vol. 5, pp. 1-10, 2014.

[43] "E. C. Committee within the European conference of postal and telecommunications administration " The analysis of the co-existence of FWA cells in the 3.4-3.8 GHz band," *Electronic communication committee Report 33,* 2003.

[44] S. Noman, T. S. Muhammad, K. Hasnain, and U. Rizwan, "Comparison of radio propagation models for long term evolution network," *International Journal of Next-Generation Networks,* vol. 3 pp. 27-41, 2011.

[45] Y. A. Alqudah, "On the Performance of Cost 231 Walfisch Ikegami Model in Deployed 3.5 GHz Network," *IEEE,* pp. 524-527, 2013.

[46] V. S. Abhayawardhana, I. Wassell, D. Crosby, M. Sellars, and M. Brown, "Comparison of empirical propagation path loss models for fixed wireless access systems," *IEEE Vehicular Technology Conference,* vol. 1, pp. 73-77, 2005.

[47] T. R. Rao, M. S. V. Bhaskara, M. V. S. N. Prasad, M. Sain, A. Iqbal, and D. R. Lakshmi, "Mobile radio propagation path loss studies at VHF/UHF bands in southern India," *IEEE Transactions on Broadcasting,* vol. 46, pp. 158-164, 2000.

[48] F. Ikegami, T. Takeuchi, and S. Yoshida, "Theoretical prediction of mean field strength for urban mobile radio," *IEEE transaction on Antennas Propagation,* vol. 39, pp. 299-302, 1991.

[49] G. Evans, B. Joslin, L. Vinson, and F. B., "The optimization and application of the W.C.Y. Lee propagation model in the 1900 MHz frequency band," *IEEE Vehicular Technology Conference* vol. 1, pp. 87-91, 1997.

[50] K. J. Chandan and J. Reshu, "Literature survey on various outdoor propagation model for fixed wireless network " *International Journal of Science and Research* pp. 1601-1604, 2012.

[51] J. Isabona and S. Azi, "Optimised Walficsh-Bertoni model for path loss prediction in urban propagation environment," *International Journal of Engineering and Innovative Technology* vol. 2, pp. 14-20, 2012.

[52] J. Sumit, "Outdoor propagation models - A literature review," *International Journal on Computer Science and Engineering*, vol. 4, pp. 281-291, 2012.

[53] J. Isabona and M. Babalola, "Statistical tuning of Walfisch-Bertoni path loss prediction model based on building and street geometry sensitivity parameters in built-up terrains," *American Journal of Physics and Applications*, vol. 1, pp. 10-17, 2013.

[54] S. Haykin, *Neural networks: A comprehensive foundation*, 2nd ed.: Prentice Hall, 1998.

[55] J. S. Seybold, *Introduction to radio frequency Propagation*, 2nd ed. United States of America: John Wiley & Sons, 2005.

[56] J. J. Egli, "Radio propagation above 40MC over irregular terrain," *Proceedings of the IRE*, vol. 45, pp. 1383-1391, 1957.

[57] S. Noman, T. S. Muhammad, K. Hasnain, and U. Rizwan, "Comparison of radio propagation models for long term evolutionnetwork," *International Journal of Next-Generation Networks* vol. 3, pp. 27-41, 2011.

[58] E. Ostlin, H. J. Zepernick, and H. Suzuki, "Macro-cell path loss prediction using artificial neural networks," *IEEE Transactions on Vehicular Technology*, vol. 3, pp. 2735-2747, 2010.

[59] M. Piacentini and F. Rinaldi, "Path loss prediction in urban environment using learning machines and dimensionality reduction techniques," *Springer-Verlag* pp. 1-15, 2010.

[60] P. Sridhar, "Novel artificial neural network path loss propagation models for wireless communications," *Advances in Wireless and Mobile Communications*, vol. 10, pp. 233-237, 2017.

[61] O. Ali Riza, A. Mustafa, K. Mehmet, G. Mehmet, and H. S. Murat, "The prediction of propagation loss of FM radio station using artificial neural network," *Journal of Electromagnetic Analysis and Applications*, vol. 6, pp. 358-365, 2014.

[62] J. Isabona and V. M. Srivastava, "A neural network based model for signal coverage propagation loss prediction in urban radio communication environment," *International Journal of Applied Engineering Research*, vol. 11, pp. 11002-11008, 2016.

[63] R. Rojas, *Neural Networks: A systematic introduction*, 2nd ed.: Springer-Verlag, 1996.

[64] E. D. Turkan, Y. H. Berna, and A. Aysen, "Fuzzy adaptive neural network approach to path loss prediction in urban areas at GSM-900 band," *Turkısh Journal of Electrıcal Engıneerıng & Computer Scıences,* vol. 18, pp. 1077-1094, 2010.

[65] S. P. Sotiroudis, K. Siakavara, and J. N. Sahalos, "A neural network approach to the prediction of the propagation path-loss for mobile communications systems in urban environments," *Proceedings of the progress in Electromagnetics Research Symposium*, pp. 162-166 2007.

[66] R. D. De Veaux and L. H. Ungar, "A brief introduction to neural networks," 2017.

[67] S. Haykin, *Neural networks: A comprehensive foundation*, 2nd ed. New York: Macmillan, 1994.

[68] S. Haykin, *Neural networks: A comprehensive foundation*, 2nd ed. United States of America: Prentice Hall, 2001.

[69] J. Hertz, Krogh A. and R. G. Palmer, *Introduction to the theory of neural computation*, 1st ed. Massachusetts: Addison-Welsey, 1991.

[70] J. A. Anderson, "A simple neural network generating an interactive memory " *Mathematical Biosciences,* vol. 14, pp. 197-220 1972.

[71] K. David, "A brief introduction to neural networks," 2005.

[72] R. D. De Veaux and L. G. Ungar. (2017, 20/04). *A brief introduction to neural networks*. Available: http://www.cis.upenn.edu/~ungar/Datamining/Publications/nnet-intro.pdf.

[73] A. Barto, R. Sutton, and C. Anderson, "Neuron-like adaptive elements that can solve difficult learning control problems," *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 13, pp. 834-846, 1983.

[74] S. Haykin, *Neural networks. A comprehensive foundation*, 2nd ed.: Prentice Hall, 1999.

[75] A. R. Barron, R. L. Barron, and E. J. Wegman, "Statistical learning networks: A unifying view," *Computer Science and Statistics: Proceedings of the 20th Symposium on the Interface* pp. 192-203, 1992.

[76] S. Haykin, *Neural network* 2nd ed. Delhi, India: Pearson Prentice Hall, 2005.

[77] D. E. Rumelhart, E. Hinton, and J. Williams, "Learning internal representation by error propagation," *Parallel Distributed Processing*, vol. 4, pp. 318-362, 1986.

[78] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel distributed processing : Explorations in the microstructure of cognition*, 2nd ed Massachusetts: MIT Press 1986, pp. 318-362.

[79] H. White, "Learning in neural networks: A statistical perspective," *Neural Computation,* vol. 1, pp. 425- 464, 1989.

[80] J. C. Dunn, "Comparison of three backpropagation training algorithms," *Indian Journal of Engineering & Material Sciences,* vol. 12, pp. 434-442, 2005.

[81] R. Robinson, "Neural networks offer an alternative to traditional regression. ," *Geobyte,* pp. 14-19, 1991.

[82] M. W. Gardner and S. R. Dorling, "Artificial neural networks (the multi-layer perceptron): A review of applications in the atmospheric sciences," *Atmospheric Environment* vol. 32, pp. 2627-2636, 1998.

[83] A. B. Ericsson, "LTE an Introduction," *White Paper,* 2009.

[84] V. Erceg, K. V. S. Hari, M. S. Smith, D. S. Baum, K. P. Sheikh, and C. Tappenden, "Channel Model for Fixed Wireless Applications," *IEEE 802.16 Broadband Wireless Access Working Group,* 2011.

[85] X. Clerck, A. Lozano, S. Sesia, C. V. Rensburg, and C. B. Papadias, "A journey towards cellular technology"3GPP LTE and LTE-Advanced"," *Wireless Communication. Network,* vol. 1, 2009.

[86] J. G. Andrews "Efficient and effective cellular technology: " Femtocells: Past, Present and Future," *IEEE journal on selected areas in communications,* vol. 30, 2012.

[87] P. Wararkar and S. S. Dorle, "Improving Handoff Addressing For IEEE 802.15.4 Based Vehicular Adhoc Networks (VANET‟s) By Particle Swarm Optimization (PSO)," *International Journal of Research in Engineering and Applied Sciences,* vol. 3, pp. 94-101, 2015.

[88] D. Mahjabeen, A. Ahmed, and S. Rafique, "Use of LTE for the Interoperability between Different Generations of Wireless Communication," *Int. J. Communications, Network and System Sciences (IJCNS),* vol. 4 pp. 424-429, 2011.

[89] C. Phillips, D. Sicker, and D. Grunwald, "Bounding the practical error of path loss models," *International Journal of Antennas and Propagation,* vol. 1, pp. 1-21, 2012.

[90] N. Faruk, A. Ayeni, and Y. A. Adediran, "On the study of empirical path loss models for accurate prediction of TV signal for secondary users," *Progress In Electromagnetics Research* vol. 49, pp. 155-176, 2013.

[91] Y. Atsushi, O. Koichi, H. Tetsuo, K. Akihito, and F. Masayuki, "Path loss prediction models for intervehicle communication at 60 GHz," *IEEE Transactions on Vehicular Technology* vol. 57, pp. 1-10, 2008.

[92] S. C. Randeep, S. Yuvraj, S. Sandeep, and G. Rakesh, "Performance & evaluation of propagation models for sub-urban areas," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering,* vol. 4 pp. 607-614, 2015.

[93] H. K. Sharma, S. Sahu, and S. Sharma, "Enhanced COST 231 W. I. propagation model in wirelesss network," *International Journal of Computer Applications* vol. 19, pp. 36-42, 2011.

[94] P. Pardeep, K. Parveen, and B. R. Shashi, "Performance evaluation of different path loss models for broadcasting applications," *American Journal of Engineering Research* vol. 03, pp. 335-342, 2014.

[95] O. Shoewu and A. Adedipe, "Investigation of radio waves propagation models in Nigerian rural and sub-urban areas," *American Journal of Scientific and Industrial Research*, pp. 227-232, 2010.

[96] J. Ramkumar and R. Gunasekaran, "A new path loss model for LTE network to address propagation delay," *International Journal of Computer and Communication Engineering*, vol. 2, pp. 413-416, 2013.

[97] F. A. A. Ignacio, A. R. A. Juan, and P. e. F. a. Fernado, "Influence of training set selection in artificial neural network-based propagation path loss predictions," *International Journal of Antennas and Propagation*, pp. 351-487, 2012.

[98] C. A. Deme, "A generalized regression neural network model for path loss prediction at 900 MHZ for Jos city, Nigeria " *American Journal of Engineering Research*, vol. 5, pp. 1-7, 2016.

[99] E. Östlin, H.-J. e. Zepernick, and H. S. Uzuki, "Macro-cell path loss prediction using artificial neural networks," *IEEE Transactions on Vehicular Technology*, vol. 59, pp. 2735-2746, 2010.

[100] E. Amaldi, A. Capone, and F. Malucelli, "Radio planning and coverage optimization of 3G cellular networks," *Wireless Networks*, vol. 14, pp. 435-447, 2008.

[101] E. Ostlin, H. J. Zepernick, and H. Suzuki, "Macro-cell path loss prediction using artificial neural networks," *IEEE Transactions on Vehicular Technology*, vol. 59, pp. 2735-2747, 2010.

[102] K. Joshua, "Neural network prediction of NFL football games," *ECE 539 unpublished*, 2003.

[103] O. U. Anyama and C. P. Igiri, "An application of linear regression & artificial neural network model in the NFL result prediction," *International Journal of Engineering Research & Technology* vol. 4, pp. 457-461, 2015.

[104] C. P. Igiri, O. U. Anyama, and A. I. Silas, "Effect of learning rate on artificial neural network in machine learning," *International Journal of Engineering Research & Technology* vol. 4, pp. 395-363, 2015.

[105] "Cisco white paper. cisco visual networking index: global mobile data traffic forecast update, 2012-2017," available at http://www.cisco.com.

[106] Z. H. Talukder, S. S. Islam, D. Mahjabeen, A. Ahmed, S. Rafique, and M. Rashid, "Cell coverage evaluation for LTE and WiMAX in wireless communication system," *World Applied Sciences Journal* vol. 22, pp. 1486-1491, 2013.

[107] J. Ramkumar and R. Gunasekaran, "A new path loss model for LTE network to address propagation delay," *International Journal of Computer and Communication Engineering* vol. 2, pp. 413-416, 2013.

[108] P. Pardeep, K. Parveen, and B. R. Shashi, "Performance evaluation of different path loss models for broadcasting applications," *American Journal of Engineering Research* vol. 03, pp. 335-342, 2014.

[109] S. Yuvraj, "Comparison of Okumura, Hata and COST-231 models on the basis of path loss and signal strength," *International Journal of Computer Applications* vol. 59, 2012.

[110] N. Zia and I. A. Muhammad, "Pathloss determination using Okumura-Hata model and cubic regression for missing data for Oman " *Proceedings of the International Multi-conference of Engineers and Computer Scientists* vol. 2, pp. 1-4, 17-19 2010.

[111] D. Walter, "Radio frequency path loss & transmission distance calculations," *Technical Memorandum,* p. 1, 2006.

[112] T. Balandier, A. Caminada, V. Lemoine, and F. Alexandre, "170 MHz field strength prediction in urban environments using neural nets," *Proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communication,* vol. 1, pp. 120-124, 1995.

[113] W. C. Y. Lee, *Mobile communication engineering,* 1st ed.: Mc Gram-Hill, 1982.

[114] H. Simon, *Neural networks - A comprehensive foundation*, 2nd ed. Englewood Cliffs Prentice-Hall, 1998.

[115] B. Cheng and D. M. Titterington, "Neural networks: A review from a statistical perspective," *Statistical Science,* vol. 9, pp. 2-54, 1994.

[116] K. Hornik, M. Stinchcombe, and H. White, "Multi-layer feedforward networks are universal approximators," *Neural Networks,* vol. 2, pp. 359-366, 1989.

[117] H. Memarian and S. K. Balasundram, "Comparison between multi-layer perceptron and radial basis function networks for sediment load estimation in a tropical watershed," *Journal of Water Resource and Protection* pp. 870-876, 2012.

[118] T. B. Ludermir, "Hybrid optimization algorithm for the definition of MLP neural network architectures and weights," *IEEE Proceedings of the 5th International Conference on Hybrid Intelligent Systems* pp. 149-154, 2005.

[119] J. Isabona and V. Srivastava, M. , "Radio channel propagation characterization and link reliability estimation in shadowed sub-urban macro-cells," *International Journal on Communications Antenna and Propagation* vol. 3, pp. 7-11, 2017.

[120] J. Isabona and V. M. Srivastava, "Coverage and link quality trends in sub-urban mobile broadband HSPA network environments " *Wireless Personal Communication,* vol. 92, pp. 3955-3968, 2017.

[121] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by backpropagating errors," *Nature,* vol. 323, pp. 533-536, 1986.

[122] P. J. Werbos, *The roots of backpropagation: From ordered derivatives to neural networks and political forecasting*. New York: John Wiley & Sons, 1994.

[123] S.-S. Shai and B.-D. Shai, *Understanding machine learning: From theory to algorithms*, 3rd ed. Cambridge: Cambridge University Press, 2014.

[124] E. Kandel, R., J. Schwartz, H., and T. Jessell, M. , *Principles of neural science*, 4 ed. United State of America: McGraw-Hill, 2000.

[125] D. Graupe, *Principles of artificial neural networks: Advanced series on circuits and systems*, 2nd ed. vol. 6. Singapore World Scientific Publishing, 2007.

[126] A. Griewank and A. Walther, *Evaluating derivatives: Principles and techniques of algorithmic differentiation*, 2nd ed.: Society for Industrial And Applied Mathematics, 2008.

[127] E. Mizutani, S. Dreyfus, and K. Nishio, "On derivation of MLP backpropagation from the Kelley-Bryson optimal-control gradient formula and its application," *Proceedings of the IEEE International Joint Conference on Neural Networks,* vol. 3, pp. 1-6, 2000.

[128] O. Abdel-Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn, and D. Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing,* vol. 22, pp. 1533-1545, 2014.

[129] E. R. Kandel, J. H. Schwartz, and T. M. Jessell, *Principles of Neural science*, 2nd ed.: Appleton & Lange, 2000.

[130] J. A. Snyman, *Practical mathematical optimization: An introduction to basic optimization theory and classical and new gradient-based algorithms*, 2nd ed. New York: Springer, 2005.

[131] B. Sharma and K. Venugopalan, "Comparison of neural network training functions for hematoma classification in brain CT images," *Journal of Computer Engineering*, vol. 16, pp. 31-35, 2014.

[132] M. J. D. Powell, "Restart procedures for the Conjugate Gradient method," *Mathematical Programming*, vol. 2, pp. 241-254, 1977.

[133] Moller, "A scaled conjugate gradient algorithm for fast supervised learning neural networks," *Neural Networks*, vol. 6, pp. 525-533, 1993.

[134] F. LiMin, "Neural networks in computer intelligence," *Ed-McGraw-Hill*, 1994.

[135] Z. Zakaria, N. A. M. Isa, and S. A. Suandi, "A study on neural network training algorithm for multi-face detection in static images," *International Conference on Computer, Electrical, Systems Science and Engineering*, pp. 170-173, 2010.

[136] C. Igel and M. Hüsken, "Empirical evaluation of the improved RPROP learning algorithm," *Neuro-Computing*, vol. 50, pp. 105-123, 2003.

[137] M. Remzi and B. Djavan, "Artificial neural networks for decision making in urologic oncology," *Ann Urol (Paris)*, pp. 110-115, 2013.

[138] R. Haelterman, "Analytical study of the least squares quasi-Newton method for interaction problems," PhD, Ghent University, 2009.

[139] Gill, Murray, and Wright, *Practical optimization*, 1981.

[140] C. Kanzow, N. Yamashita, and M. Fukushima, "Levenberg-Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints," *Journal of Clinical and Analytical Medicine*, vol. 172, pp. 375-97, 2004.

[141] D. Pham and S. Sagiroglu, "Training multi-layer perceptrons for pattern recognition: A comparative study of four training algorithms," *International Journal of Machine Tools and Manufacture*, vol. 41, pp. 419-430, 2001.

[142] MacKay, "Neural Computation," vol. 4, pp. 415-447, 1992.

[143] F. D. Forsee and M. T. Hagan, "Gauss-Newton approximation to Bayesian regularization," *International Joint Conference on Neural Network*, 1997.

[144] MacKay, *Neural Computation*, vol. 4, pp. 415-447, 1992.

[145] Foresee and Hagan, "Gauss-Newton approximation to Bayesian regularization," *Proceedings of the International Joint Conference on Neural Networks*, 1997.

[146] D. J. C. MacKay, "Bayesian interpolation," *Neural Computation,* vol. 4, pp. 415-447, 1992.

[147] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error over the root mean square error in assessing average model performance," *Climate Research,* vol. 30, pp. 79-82, 2005.

[148] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting,* vol. 22, pp. 679-688, 2006.

[149] J. M. Bland and D. G. Altman, "Statistics notes: Measurement error," *British Medical Association,* vol. 312, 1996.

[150] A. L. Edwards, *The Correlation coefficient: An introduction to linear regression and correlation.* San Francisco: W. H. Freeman, 1976.

[151] S. Ghahramani, *Fundamentals of Probability*, 2nd ed. New Jersey: Prentice Hall, 2000.

[152] A. Buda and A. Jarynowski, "Life time of correlations and its applications," *Wydawnictwo Niezależne,* pp. 5-21, 2010.

[153] H. Anysz, A. Zbiciak, and N. Ibadova, "The influence of input data standardization method on prediction accuracy of artificial neural networks," *Procedia Engineering,* vol. 153, pp. 66 - 70, 2016.

[154] T. Sheela, N. Ram, and J. Rameshwar, "Comparative study of backpropagation algorithms in neural network based identification of power system," *International Journal of Computer Science & Information Technology,* vol. 5, 2013.

[155] K. Murat, "Predictive abilities of Bayesian regularization and Levenberg–Marquardt algorithms in artificial neural networks: Acomparative empirical study on social data," *Mathematical and Computational Applications,* vol. 21, 2016.

[156] B. Widrow and M. A. Lehr, "30 years of adaptive neural networks: perceptron, madaline, and backpropagation," *Proceedings of the IEEE,* vol. 78, pp. 1415-1442, 1990.

[157] R. P. Lippman, "An introduction to computing with neura-nets," *IEEE Acoustics, Speech,and Signal Processing Society Magazine,* vol. 4, pp. 4-21, 1987.

[158] S. P. Sotiroudis, K. Siakavara, and J. N. Sahalos, "A neural network approach to the prediction of the propagation path-loss for mobile communications systems in urban environments," *Proceedings of the Progress in Electromagnetics Research Symposium*, pp. 162-166, 2007

[159] M. J. D. Powell, *Radial basis functions for multivariable interpolation: A review in algorithms for approximation* 2nd ed. Oxford: Clarendon Press, 1987.

[160] C. S. Kumar Dash, A. Kumar Behera, S. Dehuri, and S.-B. Cho, "Radial basis function neural networks: A topical state-of-the-art survey," *Open Computer Science,* vol. 6, pp. 33-63, 2016.

[161] N. B. Karayiannis, "Reformulated radial basis neural networks trained by gradient descent," *IEEE Transaction on Neural Networks*, vol. 10, pp. 657-671, 2002.

[162] D. S. Broomhead and D. Lowe, "Radial basis functions, multi-variable functional interpolation and adaptive networks," 1988.

[163] B. M. Wilamowski, "Modified EBP algorithm with instant training of the hidden layer," *Proceedings of Industrial Electronic Conference* pp. 1097-1101, 1997.

[164] J. S. R. Jang, C. T. Sun, and E. Mizutani, *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, 2nd ed. United State of America: Prentice Hall, 1997.

[165] H. B. Celikoglu and H. K. Cigizoglu, "Public transportation trip flow modeling with generalized regression neural networks," *Advance Engineering Software*, vol. 38, pp. 71-9, 2007.

[166] S. Abdul Hannan, R. R. Manza, and R. J. Ramteke, "Generalized regression neural network and radial basis function for heart disease diagnosis," *International Journal of Computer Applications* vol. 7, 2010.

[167] P. D. Wasserman and T. Schwartz, "Neural networks: what are they and why is everybody so interested in them now?," *IEEE Expert*, vol. 3, pp. 10-15, 1988.

[168] J. C. Principe, N. R. Euliano, and W. C. Lefebvre, *Neural adaptive systems: Fundamentals through simulations*. Hoboken: John Wiley & Sons Inc, 2000.

[169] W. Duch and N. Jankowski, "Survey of neural transfer functions," *Neural Computing Surveys* vol. 3, pp. 163-212, 1999.

[170] C. Bishop, *Neural networks for pattern recognition*, 2nd ed. New York: Oxford university Press, 1995.

[171] A. Orriols-Puig, J. Castillas, and E. Bernado-Mansilla, "A comparative study of several genetic-based supervised learning systems," *Computational Intelligence,* vol. 125, pp. 205-230, 2008.

[172] Y. Li, Y. Fu, H. Li, and S. W. Zhang, "The improved training algorithm of backpropagation neural network with self-adaptive learning rate," *International Conference on Computational Intelligence and Natural Computing,* vol. 1, pp. 73-76, 2009

[173] H. H. Chen, M. T. Manry, and H. Chandrasekaran, "A neural network training algorithm utilizing multiple sets of linear equations," *Neuro Computing,* vol. 25, pp. 55-72, 1999.

[174] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks.,* vol. 61, pp. 85-117, 2015.

[175] C. M. Bishop, *Neural networks for pattern recognition*, 2nd ed. Oxford: Clarendon press, 1995.

[176] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," *Nature,* pp. 318-362, 1986.

[177] P. McCullagh and J. A. Nelder, *Generalized linear models*, 2nd ed. London: Chapman & Hall, 1989.

[178] O. Giustolisi and D. Laucelli, "Improving generalization of artificial neural networks in rainfall–runoff modelling," *Journal of Hydrological Sciences,* pp. 439-457, 2005.

[179] H. Okut, X. L. Wu, G. J. M. Rosa, S. Bauck, B. W. Woodward, R. D. Schnabel, *et al.*, "Predicting expected progeny difference for marbling score in Angus cattle using artificial neural networks and Bayesian regression models," *Genetic Selection Evolution,* vol. 45, pp. 1-8, 2013.

[180] D. J. C. MacKay, "A practical Bayesian framework for backpropagation networks," *Neural Computation,* vol. 4, pp. 448-472, 1992.

[181] F. D. Foresee and M. T. Hagan, "Gauss-Newton approximation to Bayesian learning," *IEEE International Conference on Neural Networks,* pp. 1930-1935, 1997.

[182] E. Amaldi, A. Capone, and F. Malucelli, "Radio planning and coverage optimization of 3G cellular networks," *Wireless Networks,* vol. 14, pp. 435-447, 2008.

[183] Neskovic. and N. Neskovic, "Micro-cell electric field strength prediction model based upon artificial neural networks " *International Journal of Electronics and Communication* vol. 64, pp. 733-738, 2010.

[184] E. Ostlin, H.-J. Zepernick, and H. Suzuki, "Macro-cell path loss prediction using artificial neural networks," *IEEE Transactions on Vehicular Technology,* vol. 59, pp. 2735-2747, 2010.

[185] A. Neskovic, N. Neskovic, and D. Paunovic, "Indoor electric field level prediction model based on the artificial neural networks," *IEEE Communication Letters,* vol. 4, pp. 190-192, 2000.

[186] I. Popescu, D. Nikitopoulos, P. Constantinou, and I. Nafornita, "ANN prediction models for outdoor environment," *17th IEEE International Symposium on Personal, Indoor, Mobile Radio Communication,* pp. 1-5, 2006.

[187] H. Peng and S. Zhu, "Handling of incomplete data sets using ICA and SOM in data mining " *Neural Computing & Applications,* vol. 16, pp. 167-172, 2007.

[188] S. H. Wang, "Application of self-organising maps for data mining with incomplete data sets " *Neural computing & applications,* vol. 12 pp. 42-48, 2003

[189] S. Xu and L. Chen, "Adaptive higher order neural networks " *Global Congress on Intelligent Systems,* vol. 3, pp. 26-30., 2009.

[190] S. Sharma, V. Kumar, and R. Kumar, "Spervised online asaptive control of inverted pendulum system using adaline artificail neural networks with varying system parameters and external disturbance," *Intelligent Systems and Applications,* vol. 8, pp. 53-61, 2012.

[191] X. M. Gao, X. Gao, J. M. A. Tanskanen, and S. J. Ovaska, "Power prediction in mobile communication systems using an optimal neural-network structure," *IEEE Transactions on Neural Networks,* vol. 8, pp. 11-14, 1997.

[192] A. B. Tammam, A. Rabie, and K. S. Mustafa, "Neural network approach to model the propagation path loss for great tripoli area at 900, 1800, and 2100 MHz bands," *16th International Conference on Sciences and Techniques of Automatic Control & Computer Engineering* pp. 793-798, 2015.

[193] A. Torre, F. Garcia, I. Moromi, P. Espinoza, and L. Acuna, "Prediction of compression strength of high performance concrete using artificial neural networks," *Journal of Physics,* vol. 3, pp. 1-6, 2015.

[194] R. V. Ramana, B. Krishna, S. R. Kumar, and N. G. Pandey, "Monthly rainfall prediction using wavelet neural networks analysis," *Water Resources Management* vol. 27, pp. 3696-3711, 2013.

[195] C. C. Chou, "A threshold based wavelet denoising method for hydrological data modelling," *Water Resource Management,* pp. 1809-1830, 2011.

[196] J. Adamowski and K. Sun, "Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds," *Journal of Hydrology,* vol. 390, pp. 85-91, 2010.

[197] D. Wu, J. Wang, and Y. Teng, "Prediction of under-groundwater levels using wavelet decompositions and transforms," *Journal of Hydrology Engineering,* vol. 5, pp. 34-39, 2004.

[198] A. Ali, R. Ghazali, and D. M. Mat, "The wavelet multi-layer perception for the prediction of earthquake time series data," *Proceedings of the 13th International Conference on Information Integration and Web-based applications and services,* pp. 138-143, 2011.

[199] C. L. Wu, K. W. Chau, and C. Fan, "Prediction of rainfall time series using modular artificial neural networks coupled with data preprocessing techniques," *Journal of Hydrology,* vol. 389, pp. 146-167, 2010.

[200] B. D. Brabandere, X. J. Tuytelaars, T. , and L. V. Gool, "Dynamic filter networks," *30th Conference on Neural Information Processing Systems* 2016.

[201] V. Jothiprakash and A. S. Kote, "Improving the performance of data-driven techniques through data pre-processing for modelling daily reservoir inflow," *Hydrological Science,* vol. 56, pp. 168-186, 2011.

[202] N. Nithya, Vijayakumar, and P. Beth, "Prediction of missing events in sensor data streams using kalman filters," pp. 1-8.

[203] K. A. Dotche, F. Sekyere, and W. Banuenulmah, "LPC for signal analysis in cellular network coverage," *Journal of Open Access Library,* vol. 3, pp. 1-10, 2016.

[204] R. R. Venkata, B. Krishna, S. R. Kumar, and N. G. Pandey, "Monthly rainfall prediction using wavelet neural network analysis," *Water Resource Management,* vol. 27, pp. 3697-3711, 2013.

[205] V. R. Tripathi, "Image denoising using non-linear filter," *International Journal of Modern Engineering Research* vol. 6, pp. 4543-4546, 2012.

[206] N. R. Kumar and J. U. Kumar, "A spatial mean and median filter for noise removal in digital images," *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering,* vol. 4, pp. 246-253, 2015.

[207] J. W. Tukey, "Non-linear (nonsuperposable) methods for smoothing data," *Proceedings of Congress Record* p. 673, 1974.

[208] R. Lukac, K. N. Plataniotis, and B. Smolka, "Generalized selection weighted vector filters," *Journal on Applied Signal Processing,* vol. 12, pp. 1870-1885, 2004.

[209] W. Ye and Z. Liao, "Generalized correlativity of median filtering operator on signals," *Open Journal of Discrete Mathematics,* vol. 2,, pp. 83-87, 2012.

[210] A. C. Bovik, H. T. S., and D. C. M, "A generalization of median filtering using linear combinations of order statistics," *IEEE Transactions on Acoustics, Speech and Signal Processing,* vol. 31, pp. 1342-1349, 1983.

[211] R. Oten and F. R. J. P., "An efficient method for L-filter design," *IEEE Transactions on Signal Processing,* vol. 51, pp. 193-203, 2003.

[212] D. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal on Applied Mathematics,* vol. 11, pp. 431-441, 1963.

# APPENDIX I

The map of Port-Harcourt, Nigeria