Journal of the Nigerian Association of Mathematical Physics
***Volume28, No. 1,* (*November, 2014*), pp 415 – 420**

Graphical User Interface (GUI) for Calculating Path Loss for Wireless Metropolitan Area Network (Benin City) Using Java Programing Language

[1]Osunkwor E. O., [2]Olafenwa E. O., [3]Azi S.O.

Physics Department, Faculty of Physical science, University of Benin.

Abstract

GSM technology has noticeably improved the quality of life in the community and creating an attractive business climate. This technology provides speedy, inexpensive and convenient means of communication, yet it is plagued with constant consumer dissatisfaction due to the quality of service offered by the network providers. This work is based on an area that affects the Quality of Service (QoS) in the mobile Telecommunication Network world over, with emphasis on Benin City; Path loss. Path loss has a strong impact on the quality of the links, and hence, it needs to be accurately estimated for efficient design and operation of wireless networks. A Graphical User Interface (GUI) using JAVA programming language was developed for simulation of results and for prediction purposes, based on COST 231-Hata (modified Okumura-Hata model). The GUI was then tested using data obtained from the test drive carried out and the resulting graphs were then compared with graphs, which used the classical method of computing the Mean Path Loss in COST 231 – Hata Model, found in literature.

Keywords: Graphical User Interface (GUI), JAVA, Base station, Signals, Path Loss, COST 231 **–** Hata Model.

1.0 Introduction

When designing any system, engineers and system architects usually want to know how well the various components of the system will perform. In a microprocessor, logic or other digital system these questions are usually answered in terms of clock speeds, instructions per second or data throughput. With wireless communication the most common performance questions involve range [1].

Unlike digital systems, trying to quantify the range performance of radio frequency (RF) communication systems can be difficult due to the large role the environment has on radio frequency signals. Buildings, trees, obstructions and lack of antenna height can all contribute to a decrease in signal strength at the receiving end. In order to estimate the communication distance (transmission range) for a system, four factors must be considered:

- Transmit Power- The power that is broadcast by the transmitter. This is usually measured in watts or milliwatts.
- Receive Sensitivity- A measure of the of the minimum signal strength that a receiver can discern.
- Antenna gain- The amount of signal gain provided by the antennas.
- Path Loss- The signal decrease that occurs as the radio waves travel through the air or through obstacles.

Radio propagation path lossmodel is an important tool that characterizes the quality of service (QoS) provided by mobile telecommunication companies. It determines effective radio coverage, as well as network optimization. The path loss models also predict to a high level of accuracy the true signal strength reliability of the network and the quality of coverage [2,3]. With appropriate propagation path loss model, the coverage area of mobile communication system, the signal-to-noise ratio as well as the carrier-to-interference ratio can be determined easily [4,5]**.**

Furthermore, with the worldwide proliferation and subsequent congestion of wireless Metropolitan Area Networks (WMANs), careful network planning and propagation modelling have become essential to current deployments of Base stations.In order to meet this demand, the network calculations of the desired path loss model must be carried out such that it is both accurate, exhaustive, less time consuming and with minimal errors.

Hence, the goal of this work is to design a Graphical User Interface (GUI) for calculating path loss of a Wireless Metropolitan Area Network (Benin City) for optimal performance using JAVA programming language that is both accurate and less time consuming.

Our objective here is to design a Graphical User Interface (GUI) for calculating Path Loss in Benin City, using the COST 231-Hata Model. To design GUI that will reduce the rigors in calculating path loss. Test run the GUI with data gotten from Drive Test. Compare the result with literature(s). To provide data that could serve as a reference point for further work

Corresponding author: Osunkwor E. O,E-mail: offormata.osunkwor@uniben.edu, Tel.: +2347060470621

1.1 Graphical User Interface (GUI)

A graphical user interface (GUI) is a graphical display in one or more windows containing controls, called components that enable a user to perform interactive tasks. The user of the GUI does not have to create a script or type commands at the command line to accomplish the tasks. Unlike coding programs to accomplish tasks, the user of a GUI need not understand the details of how the tasks are performed.

GUI components can include menus, toolbars, push buttons, radio buttons, list boxes, and sliders—just to name a few. GUIs tools can also perform any type of computation, read and write data files, communicate with other GUIs, and display data as tables or as plots [6].

1.1.1 Graphical Systems: Advantages and Disadvantages

Graphical systems burst upon the office with great promise. The simplified interface they presented was thought to reduce the memory requirements imposed on the user, make more effective use of one's information-processing capabilities, and dramatically reduce system learning requirements. Experience indicates that for many people they have done all these things [7].

The success of graphical systems has been attributed to a host of factors. The following have been commonly referenced in literature and endorsed by their advocates as advantages of these systems; Symbols are usually recognized faster than text, GUI support Faster learning, less time consuming and high problem solving efficiency, Easier remembering of operational components, Less anxiety concerning use, Relatively More attractive for even the most skeptical user, Low typing requirements.

Nonetheless, the body of positive research, hypotheses, and comment concerning graphical systems is being challenged by some studies, findings, and opinions that indicate that graphical representation and interaction may not necessarily always be better. Indeed, in some cases, it may be poorer than pure textual or alphanumeric displays. Trying to force all system components into a graphical format may be doing a disservice to the user. Some also feel that, as graphical systems are becoming increasingly sophisticated and continue to expand, interfaces have become increasingly more complex, sometimes arcane, and even bizarre. Some of the disadvantages put forth are these; Inefficient for touch(keyboard) typists and not always the fastest style of interaction like in an automated bank teller machine. GUI were inferior to textual instructions.

1.1.2 Characteristics Of The Graphical User Interface

A graphical system possesses a set of defining concepts. These defined steps encompasses its characteristics and they are listed below;

- Sophisticated Visual Presentation
- Pick-and-Click Interaction
- Restricted Set of Interface Options
- Visualization
- Object Orientation

1.2 Java Overview

Java programming language was originally developed by Sun Microsystems which was initiated by James Gosling and released in 1995 as core component of Sun Microsystems' Java platform (Java 1.0 [J2SE]). As of December 2008, the latest release of the Java Standard Edition is 6 (J2SE). With the advancement of Java and its widespread popularity, multiple configurations were built to suite various types of platforms. Ex: J2EE for Enterprise Applications, J2ME for Mobile Applications. Sun Microsystems has renamed the new J2 versions as Java SE, Java EE and Java ME, respectively. Java is guaranteed to be Write **Once, Run Anywhere**. Some advantages of Java are: object oriented, platform independent, simple,secure, architectural-neutrality, portable, robust, multithreaded, easily interpreted, high performance, distributed and dynamic.

1.3 History of Java

James Gosling initiated the Java language project in June 1991 for use in one of his many set-top box projects. The language, initially called Oak after an oak tree that stood outside Gosling's office, also went by the name Green and ended up later being renamed as Java, from a list of random words. Sun released the first public implementation as Java 1.0 in 1995. It promised Write Once, Run Anywhere (WORA), providing no-cost run-times on popular platforms. On 13 November 2006, Sun released much of Java as free and open source software under the terms of the GNU General Public License (GPL). On 8 May 2007, Sun finished the process, making all of Java's core code free and open-source, aside from a small portion of code to which Sun did not hold the copyright.

In order to carry out Java programming will have certain requirement and they are; a Pentium 200-MHz computer with a minimum of 64 MB of RAM (128 MB of RAM recommended). You also will need the following softwares: Linux 7.1 or Windows 95/98/2000/XP/7/8 operating system, Java JDK 5, Microsoft Notepad or any other text editor.

1.3.1 Java Basic Syntax

When we consider a Java program, it can be defined as a collection of objects that communicate via invoking each other's methods. Some important terms in java programming are given below;

- Object - Objects have states and behaviors. Example: A dog has states- gcolor, name, breed as well as behaviors -wagging, barking, eating. An object is an instance of a class.
- Class - A class can be defined as a template/blue print that describes the behaviors/states that object of its type support.
- Methods - A method is basically a behavior. A class can contain many methods. It is in methods where the logics are written, data is manipulated and all the actions are executed.
- Instance Variables - Each object has its unique set of instance variables. An object's state is created by the values assigned to these instance variables[8].

1.3.2 Sample Java Program:

Let us look at a simple code that would print the words *Hello World*.

```
public class MyFirstJavaProgram{
/* This is my first java program.
 * This will print 'Hello World' as the output
 */
public static void main(String[]args){
System.out.println("Hello World");// prints Hello World
}
}
```

Let's look at how to save the file, compile and run the program. Please follow the steps given below:

- Open notepad and add the code as above.
- Save the file as: SampleJavaProgram.java.
- Open a command prompt window and go o the directory where you saved the class. Assume it's C:\.
- Type ' javac SampleProgram.java ' and press enter to compile your code. If there are no errors in your code, the command prompt will take you to the next line(Assumption : The path variable is set).
- Now, type ' java SampleProgram ' to run your program.
- You will be able to see ' Hello World ' printed on the window.

```
C :> javac SampleJavaProgram.java
C :> java SampleJavaProgram
HelloWorld
```

2.0 Methodology

2.1 GUI Design

The GUI used in calculating the mean Path Loss, using the COST 231-Hata Model was designed using the Java Programing Language and is as shown in Figure 4.1.

The GUI was designed using Eclipse JAVA Compiler. The GUI takes in height of the Base station in metre (h_b), height of the mobile station in metre (h_m), frequency of transmission in MHz (f), and distance between the base station and mobile station in m (r) as inputs. It simulates the constants used in calculating the Mean Path Loss (F, B, and E) and the predicted Mean Path Loss (L_{dB}) simultaneously as outputs. The GUI is designed such that it can take either single variable or several variables of the distance between the antenna and the mobile station, producing either single or multiple values for the calculated Mean Path Loss.

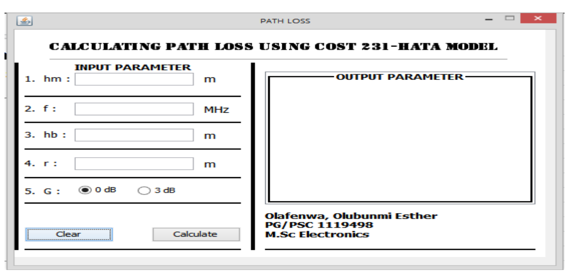

Figure 1: GUI for Calculating Mean Path Loss (L_{dB}), using the COST 231 – Hata Model

The GUI is then tested using data gathered from the drive test carried out and the result gotten is then compared with existing results from journals.

3.0 Data Collection Method

To test run the GUI, real-time data from field measurement during a drive test carried out on 29th March, 2014 is used. Data were collected at varying distances on live radio from two different Telecommunication Network Providers which we shall refer to as: Base Station 1, with transmitting frequency of 2100MHz and Base Station 2 with transmitting frequency of 1800MHz.

Drive test tools used for collecting data included a laptop equipped with drive test Ericsson software, Sony Ericsson Test Equipment Mobile Software (TEMS) phone for idle and dedicated mode and a Global Position System (GPS). The test was carried out at two locations in Benin City: Ekiadolor, with co-ordinate (6.49230^{o}N 5.55860^{o}E, Elevation 150m) and Agbor Park road, with co-ordinate (6.34107^{o}N 5.670211^{o}E, Elevation 109m)

The Ericsson TEMS phone, mobile GPS receiver and the Dongle probe were coupled to a laptop placed in a car. The laptop was powered on in order to launch TEMS investigation software. All the equipment were connected and detected on TEMS interface. The car was driven around through a predefined route in the direction of the Active Sector (AS) of the directional antenna away from the site until it got to the coverage border. The car was driven at an average speed of 30km/h.

The received signal power is measured using Ericsson handset and transferred to the TEMS log file in the laptop. The GPS receiver gave the location and distance from the BTS and the received power level reading and was recorded on the laptop.

Figures 2-3 shows a sample of a log files obtained for the field measured data during the drive test.

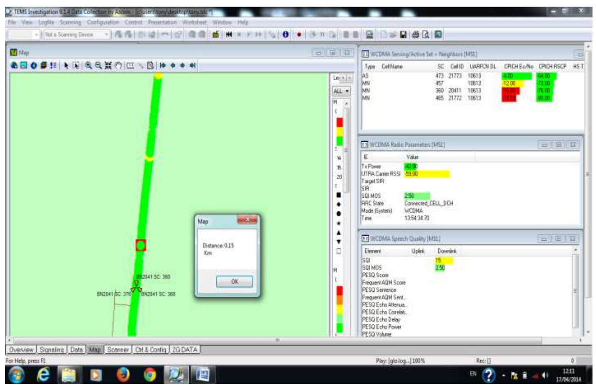

Figure 2: Log file showing the received signal level distribution for Base Station 1

Table 1: Summary of the field measured data obtained from Base Station 1

Distance between base station and mobile (m)	Signal Coverage(dB)	Signal Quality(dB)
50	-67.0	-5.0
100	-61.0	-4.5
200	-70.0	-7.0
300	-75.0	-5.0
400	-80.0	-6.0
500	-80.0	-5.0
600	-87.0	-8.0
700	-81.0	-6.0
800	-77.0	-4.5
900	-75.0	-4.5
1000	-87.0	-7.0

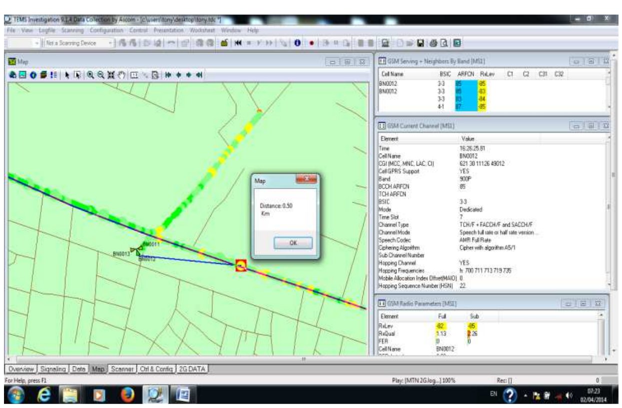

Figure 3: Log file showing the received signal level distribution for Base Station 2

Table 2: Summary of the field measured data obtained from Base Station 2

Distance between base station and mobile (m)	Signal Coverage (dB)	Signal Quality (dB)
100	-79.0	-1.13
200	-70.0	-7.0
300	-76.0	-0.14
400	-79.0	9.05
500	-85.0	2.26
600	-89.0	4.53
700	-77.0	0.14
750	-89.0	0.28
800	-82	0.57
900	-82.0	0.14
1000	-85.0	0.57

3.1 Result and Discussion

After Simulation to get the Mean Path Loss (L_{dB}) using designed GUI. The results are as follows:

3.1.1 Result For Base Station 1

Input Parameter:
f_c (MHz) Carrier Frequency = 2100MHz
h_m (m) Mobile Station antenna height from ground = 1.23m
h_b (m) Base Station antenna height from ground = 38m
and distance between mobile station and the base station, r (m)

Table 3: Output Parameter for Base Station 1

Distance (m)	Mean Path Loss (dB)
50	95.886
100	106.287
200	116.688
300	122.773
400	127.090
500	130.438
600	133.174
700	135.487
800	137.491
900	139.258
1000	140.839

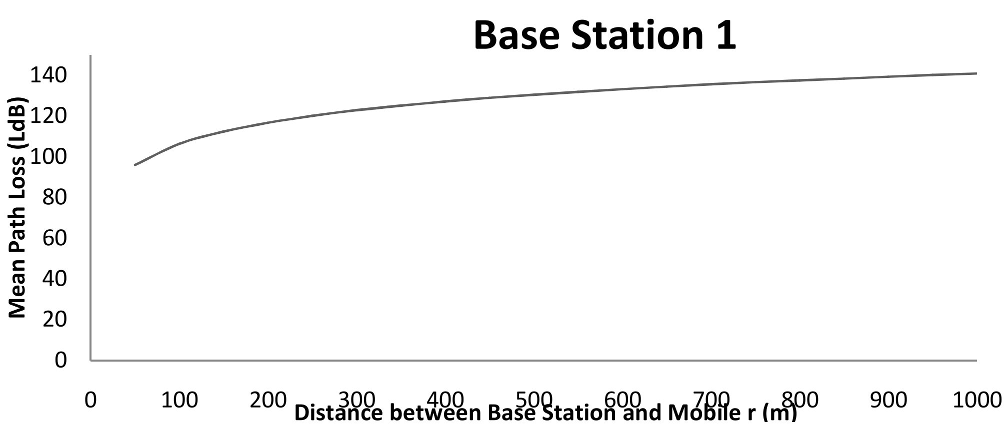

Figure 4: Graph of Mean Path Loss (L_{dB}) against Distance between base station and Mobile station (r) for Base station 1.

3.1.2 Result for Base Station 2

Input Parameter:
f_c (MHz) Carrier Frequency = 1800MHz
h_m (m) Mobile Station antenna height from ground = 1.23m
h_b (m) Base Station antenna height from ground = 30m
and r (m)

Table 4: Output Parameter for Base Station 2

Distance (m)	Mean Path Loss (dB)
100	104.750
200	115.354
300	121.556
400	125.957
500	129.371
600	132.160
700	134.518
800	136.561
900	138.363
1000	139.975

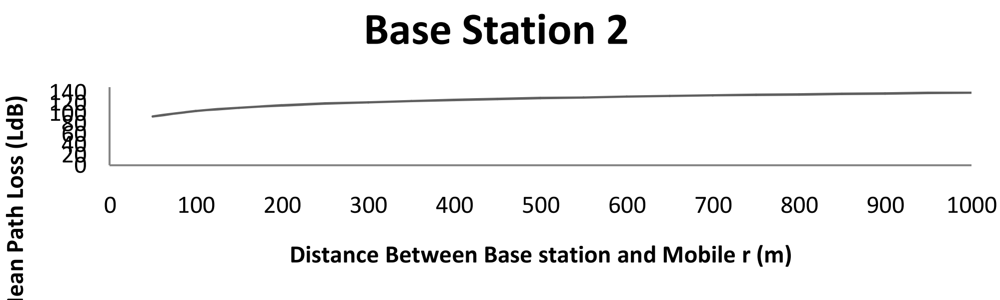

Figure 5: Graph of Mean Path Loss (L_{dB}) against Distance between base station and Mobile station (r) for Base station 2.

4.0 Conclusion

The Graphs obtained through calculating the path loss using the GUI (Figures 4 and 5) were found to compare favourably with the graphs obtained by Kumar and Malik[8], Ogbulezie et al[9], and Akinwole & Biebuma [10]. All of whom predicted the mean Path Loss using the classical method for calculating for COST 231- Hata Model.

From this, we can draw the conclusion that the designed GUI is an easier and less cumbersome method for calculating Mean Path Loss.

Also, from the analysis of the data obtained, we observed that increase in distance inevitably leads to greater signal attenuation which will eventually lead to dropped calls due to signal loss, or delay handover. This in turn will lead to poor quality of service from the service providers and dissatisfied customer.

5.0 References

[1] Digi (2012), Application Note (XST-AN005a-Indoor), www.digi.com.Email:rf-experts@digi.com as at 15/03/2014

[2] Anderson R.H. (2006). Coverage Prediction for Digital Mobile Systems. PrismBusiness media Inc.., Overland Park, KS. USA.

[3] Bachir B. & Saunder, S., (2004). Accurate coverage prediction and optimization – for digital broadcasting. EBU Technical Review, CellularDesign Services-Spectrum Planning. Pp 84-89.

[4] Lee, W.C.Y., (1993). Mobile Communication Design Fundamentals. JohnWiley and Sons Inc.

[5] Tiong, S.K., Z. Akma, L.H. Kar and P.T. Perng, (2004) Propagation loss modelscharacterization for GSM 900MHz At Kuala Lumpur and Putrajaya, Collegeof Engineering, University of Tenaga, Malaysia (UNITEN)

[6] Matlab (2013), Creating Graphical User Interfaces, MA: MathWorks,

[7] Galitz O. William (2002). Essential Guide to User Interface Design: AnIntroduction to GUI Design, Principles and Techniques. NY: John Wiley andSons.

[8] Eck D. J. (2011) Introduction to programming using Java, Department ofMathematics and Computer Science, Hobart and William Smith Colleges,Geneva, NY 14456. Online book(http://math.hws.edu/javanotes/.) as at 24/04/2014

[9] Kumar, Mukesh, Kumar Vijay, & Malik Suchika, (2012). Performance and Analysis of Propagation Models for Predicting RSS for Efficient Handoff. International Journal of Advanced Scientific and Technical Research. Issue 2, Vol. 1, pp 61-70

[10] Ogbulezie, J.C. ,Onuu, M.U. , Bassey, D.E & Etienam-Umoh, S. (2013). Site specific measurements and propagation models for GSM in three cities in Northern Nigeria. American Journal of Scientific and Industrial Research. ISSN: 2153-649X, doi:10.5251/ajsir.2013.4.2. pp238-245

[11] Akinwole B.O.H & Biebuma J.J. (2013). Comparative Analysis of Empirical Path Loss Model for Cellular Transmission in Rivers State, American Journal of Engineering Research (AJER), Vol-02, Issue-08, pp-24-31.