Outdoor User Location over 802.11 Ad-hoc Network

By

MOHD AZRIK BIN ROSLAN

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

DECEMBER 2006

Universiti Teknologi Petronas
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

**OUTDOOR USER LOCATION
OVER
802.11 AD-HOC NETWORK**

by

Mohd Azrik Bin Roslan

A project dissertation submitted to the
Electrical & Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical & Electronics Engineering)

Approved:

Ms. Norashikin Bt. Yahya
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

DECEMBER 2006

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

Mohd Azrik Bin Roslan

ABSTRACT

The objective of this project is to implement outdoor user location which is build on top of IEEE 802.11 technology. In this solution, four laptops are used where three of the laptops will serve as the reference points and the other one will be the target. The signal strengths from each reference laptop to the target are measured to serve as arguments for localization algorithms, which is triangulation. In order to perform this algorithm, the locations of the three reference laptops and their separation distance from the target laptop are needed to be known. The locations of the reference laptops are arbitrarily assigned but the separation distances are estimated based on the signal strength information received by the target laptop. To find the relationship between the *separation distance* and *signal strength received* from each reference laptop, a series of signal strength measurements are conducted. From the collected data, the radio propagation models are determined. A Graphical User Interface (GUI) has been developed to make the process of location determination become less complicated and more user friendly. At the end of this work, the site testing has validated that this algorithm can be used to determine the user location in outdoor environment with considerable accuracy.

ACKNOWLEDGEMENTS

Praise to Allah S.W.T, The Most Gracious, and The Most Merciful for His Guidance and Blessing for giving the strength and chances to undergo the project.

Special acknowledgement and sincere gratitude to Ms. Norashikin Yahya and Ms Nasreen Badruddin, the project supervisors for their assistance on the project, advice, guidance and motivation throughout the semester for the author to achieve what is targeted at the beginning of the semester.

I would also like to express my fullest appreciation to my family especially my beloved mother and father, for their moral support throughout this project.

Special thanks also go to the technologists, Miss Siti Hawa, who deliberately giving ideas and helpful tips for a good quality of Final Year Project output, lending the author equipment for experimenting purposes and preparing the workstation needed to undergo the project.

Last but not least, thanks to everyone involved directly and indirectly towards the successful of the project.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Background of Study

The location-aware services and applications are fast growing in this era of mobile computing and wireless communication. Many research efforts were spent on obtaining location information within wireless network. Back in 90's, the location estimation were based on the IR technology. Then, the Global Positioning System came into place. This system although can perform very well in tracking the object for outdoor situation, is useless for indoor case because buildings will block the GPS Transmission. The recent technology which is based on RF technology using standard IEEE 802.11 wireless LAN has become an advantage for the location aware services. Many techniques as well as the algorithms have been introduced by experts in this area. Majority of them used the Access Point (AP) in their system and some of them try the ad-hoc approach in detecting the user location.

1.2 Problem Statement

Location information has many applications, such as location- sensitive content delivery, cooperative wireless routing and real-time roadmap. In this project, the location of a user (laptop) in a wireless network using an ad-hoc mode will be computed based on the signal strength information from nearby laptops. At least 3 laptops act as reference points need to be connected with the target laptop (the one that the location needs to be determined) The signal strength transmitted by each reference laptop will be measured by the target laptop using software and based on the signal strength information, the location of the target laptop will be computed using triangulation algorithm.

1.3 Objective and Scope of Study

The main objective of this project is to implement user location technique using wireless ad-hoc network for outdoor environment. At the end of this project, it is expected that the location of users for outdoor environment can be predicted based on the signal strength information of the devices within the network (laptops) and the triangulation method will be used for computation.

This project involves the design of RF propagation model for our measuring site. There are some parameters in this model need to be determined before this model can be used in triangulation. For this reason, some experiments have been conducted to collect training data and determine those parameters by the training model. Using the RF propagation model, the next step would be to write a program that will automatically calculate the user location given a set of signal strength.

The last part of this project will involve the performance of this system that is, by measuring the precision achieved when using signal strength to estimate distance.

CHAPTER 2
LITERATURE REVIEW

2.1 Previous Works on RF-based User Location and Tracking System

RF-based user location and tracking system is becoming popular nowadays. Many techniques and methods have been developed and introduced by the researchers. Back to the year 2000, Bahl et al [1] explained the method of using IEEE 802.11b access point signals to detect the location of a user. They have introduced two approaches to cater this problem. The first one is the empirical method. In this method they have built a database of the signal strength of three access points at various locations inside a building, and then used these data together with nearest neighbor's classifier to determine the location of a mobile user. In the second approach, they tried to model the radio signal propagation inside the building by taking into consideration the walls and the objects inside the building. It was found that the first method produced better results compare to the latter one. This can be caused by the complexity involved in modeling the radio propagation model hence the model they produced might not be so accurate.

Siddharta et al [2] also did similar work with the one described before but with some modifications. They built a database of signal strengths at several locations in a building and also adopted the empirical approach. However, they employed more sophisticated classifiers which take into account the distribution of the data. They have studied three methods for classification: (i) nearest neighbor classifier, (ii) back propagation neural network, (iii) histogram matching. They have found out that the second classifier give more accurate results compare to others. They have concluded that the accuracy of location determination depends on how close the two points and how many access points are being used. Using more number of access points surely makes it easy for the location detection system but at the same time will increase the cost of the system. They also concluded that the accuracy and the precision of the system do not change significantly with the increase or decrease of the sampling rate.

In 2004, Youngjune [3] have presented the algorithm for estimating the location of stationary and mobile users based on the heterogeneous indoor RF technologies. They have proposed two location algorithms, (i) Selective Fusion Location Estimation (SELFLOC), (ii) Region of Confidence (ROC). Both of these algorithms can be used in conjunction with classical location algorithms such as triangulation, K-nearest neighbor averaging (KNN) and the smallest M-vertex polygon (SMP). The SELFLOC algorithm infers the user location by selectively fusing location information from multiple wireless technologies and/or multiple classical location algorithms in a theoretically optimal manner. The ROC algorithm attempts to overcome the problem of aliasing in the signal domain, where different physical locations have similar RF characteristics, which is particularly acute when users are mobile. They have empirically validated the proposed algorithms using wireless LAN and Bluetooth technology. Their experimental results show that applying SELFLOC for stationary users when using multiple wireless technologies and multiple classical location algorithms can improve location accuracy significantly, with mean distance errors as low as 1.6 m. For mobile users, they find that using RoC, they can obtain mean errors as low as 3.7 m. Both algorithms can be used in conjunction with a commercial location estimation system and improve its accuracy further.

All the three methods discussed before used the access point (AP) in their system but the next method uses different approach. Song Li et al in [4] choose another mode, named "Ad-Hoc". An Ad-Hoc wireless network does not require an AP to be present. This mean, every wireless station is treated equally and they are free to communicate with each other. They used two approaches in determining the user location. The first one was the triangulation approach and the second one is the radio propagation model. They found out that both of their approaches, although can somehow provide the location information, are very hard to obtain location information of higher accuracy. This is due several factors such as the obstruction, reflection, diffusion, etc that affecting the signal strength significantly. In addition, the propagation models are not accurately taking into consideration the effect of obstructions in the tested area.

2.2 IEEE 802.11

This is the standard for the physical layer of the network. The physical layer defines how the bits and bytes of data are transferred to and from the physical medium of the network (in this case, the electromagnetic spectrum of the wireless LAN). It was released in 1997 and it specifies two raw data rates of 1 and 2 megabits per second (Mbit/s). It uses the Infra Red (IR) signal transmitting in the Industrial Scientific Medical frequency band at 2.4 GHz. The media access method used in this standard is the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). [5]

There are three widely used 802.11 standards in the market nowadays (802.11 a/b/g). The 802.11b and 802.11g systems used the same spectrum (in 2.4 GHz Band) while the 802.11a uses different set of frequencies in 5 GHz band. This mean the 802.11 b and g systems can talk to each other and neither of these systems can connect to an 802.11a system.

This project uses the 802.11b standard. So it is important to know the detail about this standard. The 802.11b was published in 1999. It has a maximum raw data rate of 11 Mbit/s and uses the same CSMA/CA media access method defined in the original standard. Due to the CSMA/CA protocol overhead, in practice the maximum 802.11b speed that an application can achieve is about 5.9 Mbit/s over Transmission Control Protocol (TCP) and 7.1 Mbit/s over User Datagram Protocol (UDP).

802.11b products appeared on the market very quickly, since 802.11b is a direct extension of the DSSS (Direct-sequence spread spectrum) modulation technique defined in the original standard. Technically, the 802.11b standard uses Complementary code keying (CCK) as its modulation technique, which is a variation on CDMA. Hence, chipsets and products were easily upgraded to support the 802.11b enhancements. The dramatic increase in throughput of 802.11b (compared to the original standard) along with substantial price reductions led to the rapid acceptance of 802.11b as the definitive wireless LAN technology.

802.11b is usually used in a point-to-multipoint configuration, wherein an access point communicates via an omni-directional antenna with one or more clients that are located in a coverage area around the access point. Typical indoor range is 30 m at 11 Mbit/s and 90 m at 1 Mbit/s. With high-gain external antennas, the protocol can also be used in fixed point-to-point arrangements, typically at ranges up to eight kilometers (km) although some report success at ranges up to 80–120 km where line of sight can be established. This is usually done in place of costly leased lines or very cumbersome microwave communications equipment. Designers of such installations who wish to remain within the law must however be careful about legal limitations on effective radiated power.

802.11b cards can operate at 11 Mbit/s, but will scale back to 5.5, then 2, then 1 Mbit/s when signal quality becomes an issue. Since the lower data rates use less complex and more redundant methods of encoding the data, they are less susceptible to corruption due to interference and signal attenuation. Extensions have been made to the 802.11b protocol in order to increase speed to 22, 33, and 44 Mbit/s, but the extensions are proprietary and have not been endorsed by the IEEE. Many companies call enhanced versions "802.11b+". These extensions have been largely obviated by the development of 802.11g, which has data rates up to 54 Mbit/s and is backwards-compatible with 802.11b.

CHAPTER 3
METHODOLOGY / PROJECT WORK

3.1 Procedure Identification

Several main procedures have been identified towards accomplishing the project. Block diagram in Fig. 1 summarizes on the tasks that have been performed in completing this project. The details about the steps and procedures are explained in the next sections.

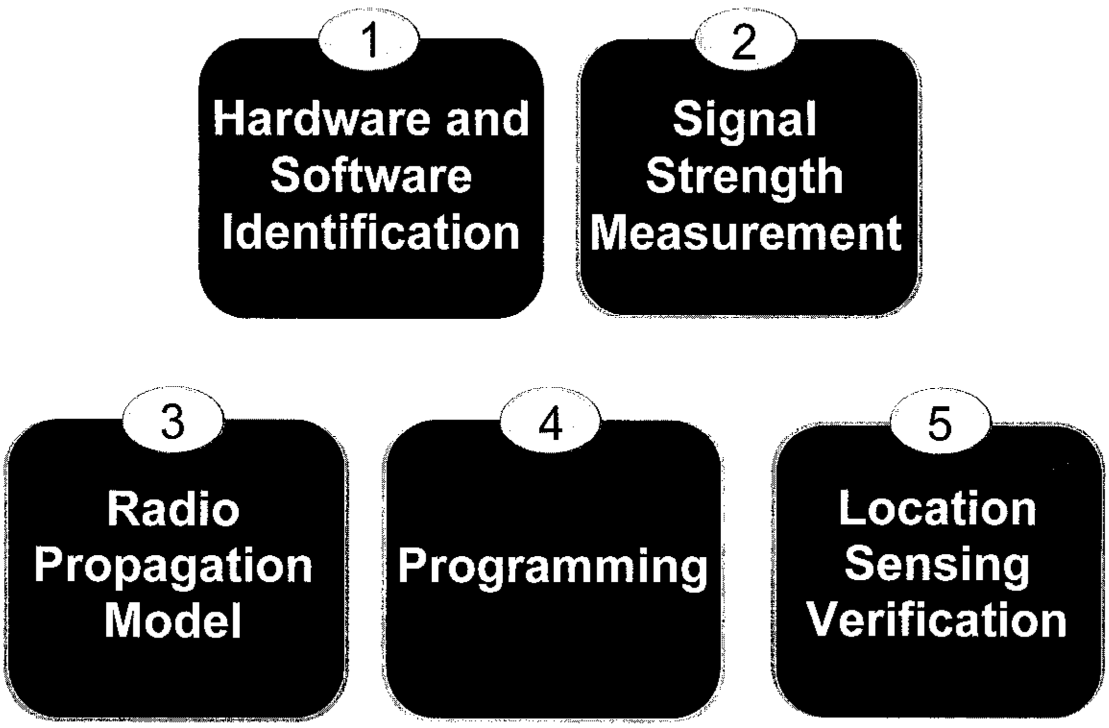


Fig. 1: Procedures Involved in Completing the Project

3.2 Hardware and Software Identification

For this project, all laptops involved in the location sensing are connected in wireless ad-hoc network. The four laptops are fully equipped with the PCMCIA WIFI Adapters; i) *Cisco Air-PCM 352* ii) *LINKSYS Wireless-G WPC54G Notebook Adapter*. Two software; i) *NetStumbler*, ii) *MATLAB* are used in this project.

3.2.1 Hardware

The technical details for both the wireless adapters are listed in the TABLE 1 and TABLE 2.

Cisco Air-PCM 352

TABLE 1

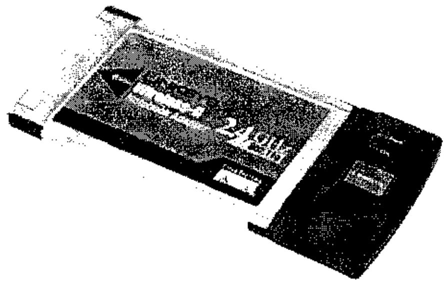
Technical Details for Cisco-PCM 352 [11]

Technical Details
 • **Model**: AIR-PCM352 • **Standard:** 2.4GHz/802.11b wireless standard. • **Hardware Platform**: PC, Mac, Unix • **Networking Feature**: Network adapter • **Form Factor**: Plug-in module • **Network Interface Description**: DSSS Type II PC Card • **Connectivity Technology**: Wireless • **Security:** Up to 128-bit WEP encryption • **Data Link Protocol**: IEEE 802.11b • **Network Data Transfer Rate**: 11 Megabits Per Second • **Maximum Range Indoors**: 350 feet (1Mbps) , 130 feet (11 Mbps) • **Minimum system requirements**: Microsoft DOS, Microsoft Windows 95/98, Linux, Microsoft Windows CE 2.11, Microsoft Windows CE 3.0, Microsoft Windows 2000 / NT4.0, Microsoft Windows Millennium Edition, Apple MacOS 9.x, Apple MacOS X, Microsoft Windows XP • **Width**: 4.4 inches • **Depth**: 2.1 inches • **Height**: 0.1 inches • **Weight**: 1.6 Ounces • **Warranty**: Limited lifetime warranty

LINKSYS Wireless-G WPC54G

TABLE 2

Technical Details for LINKSYS Wireless-G WPC54G

Technical Details (Refer Appendix II)
 • **Model:** Wireless-G WPC54G • **Standards:** IEEE 802.11g, IEEE 802.11b • **Channels:** 11 Channels (USA, Canada) 13 Channels (Europe, Japan) • **LEDs :** Power, Link • **Transmit Power:** 18 dBm • **Protocols:** 802.11b: CCK (11 Mbps), DQPSK (2 Mbps), DBPSK (1 Mbps); 802.11g: OFDM • **Network Protocols:** TCP/IP, IPX/SPX, NetBEUI • **Security Features:** WEP, AES, TKIP, 802.1x • **WEP Key Bits:** 64, 128 Bit • **Dimensions:** 4.53" x 2.13" x 0.30" (115 mm x 54 mm x 7.5 mm) • **Unit Weight:** 1.66 oz. (0.047 kg.) • **Certifications:** FCC, IC-03, CE • **Operating Temp:** 32°F to 131°F (0°C to 55°C) • **Storage Temp:** -13°F to 158°F (-25°C to 70°C) • **Operating Humidity:** 5% to 95%, Non-Condensing • **Storage Humidity:** 5% to 95%, Non-Condensing

3.2.2 Software

This project involves signal strength measurement, triangulation computation and construction of graphical user interface (GUI). For signal strength measurement, the NetStumbler is used due to its considerable performance in measuring the signal strength and also because it is a freeware. For the computation and GUI, the MATLAB is chosen due to its powerful computational engine and availability of GUI design environment. The overviews of these softwares are explained in the next paragraphs.

NetStumbler

NetStumbler is an active scanning network monitoring tool. It finds and monitors networks by actively sending out probe requests on all the WIFI channels supported by the WiFi adapter or card in the PC [6]. Most networks respond to these probe requests, but not all do. Other functions of this software [7] are as follows:

- Find locations with poor coverage in WLAN.
    - NetStumbler can verify that an area is well covered by a good quality signal. It can also be used to see how far the coverage area extends beyond its intended boundary.
- Detect other networks that might be causing interference with our network.
    - A site survey typically includes finding out what existing items (microwave ovens, cordless phones and radio hams) are using the radio frequencies as the wireless LAN. A survey should be done before installation of a new wireless LAN, and then subsequent surveys should be performed after installation.
- Detect unauthorized "rogue" access points in workplace.
    - A corporate network administrator needs assurance that the wired LAN is not being exposed to unauthorized users. This can often happen when users set up their own wireless LANs for convenience. Such wireless LANs often have little or no security, which poses a risk to the entire LAN. The network administrator can use NetStumbler to detect the presence of these "rogue" wireless LANs.
- Help aim directional antennas for long-haul WLAN links.
    - When setting up an antenna, NetStumbler can be used to assist with optimal antenna position and aim.

Fig. 2 below shows the NetStumbler main window while scanning the network. It will display MAC address of the network device, network Service Set Identifier (SSID), the channel that is being used, the speed of the link, the type of the mode being used the signal strength in *dBm* the signal to noise ratio (SNR) and also the magnitude of the noise. Not all the network adapters are supported by this software. For unsupported adapters, the value of SNR, signal strength and noise are not displayed correctly or not displayed at all.

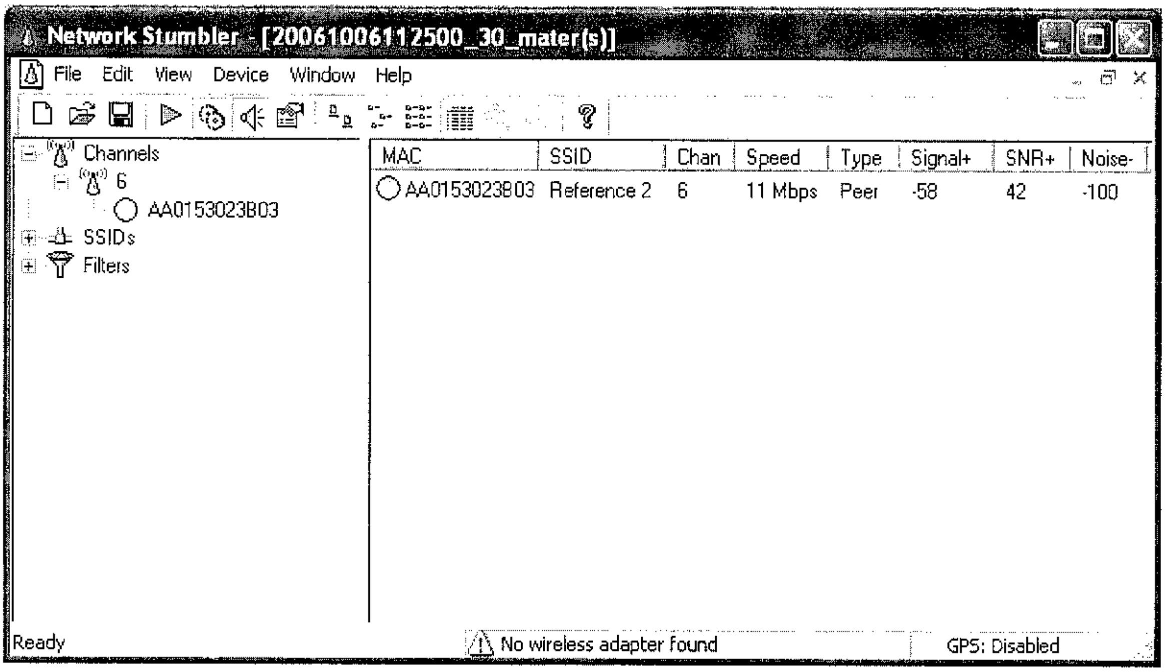


Fig. 2: Scanning Operation in Detail View

NetStumbler also allows the scanning process to be viewed in graphical view as shown in Figure 3. This view can be started by selecting the MAC address at the left panel of the main window. In this mode, the results of the scanning will be displayed in column chart where the y-axis is the signal strength in *dBm* and the x-axis is the time of the measurement.

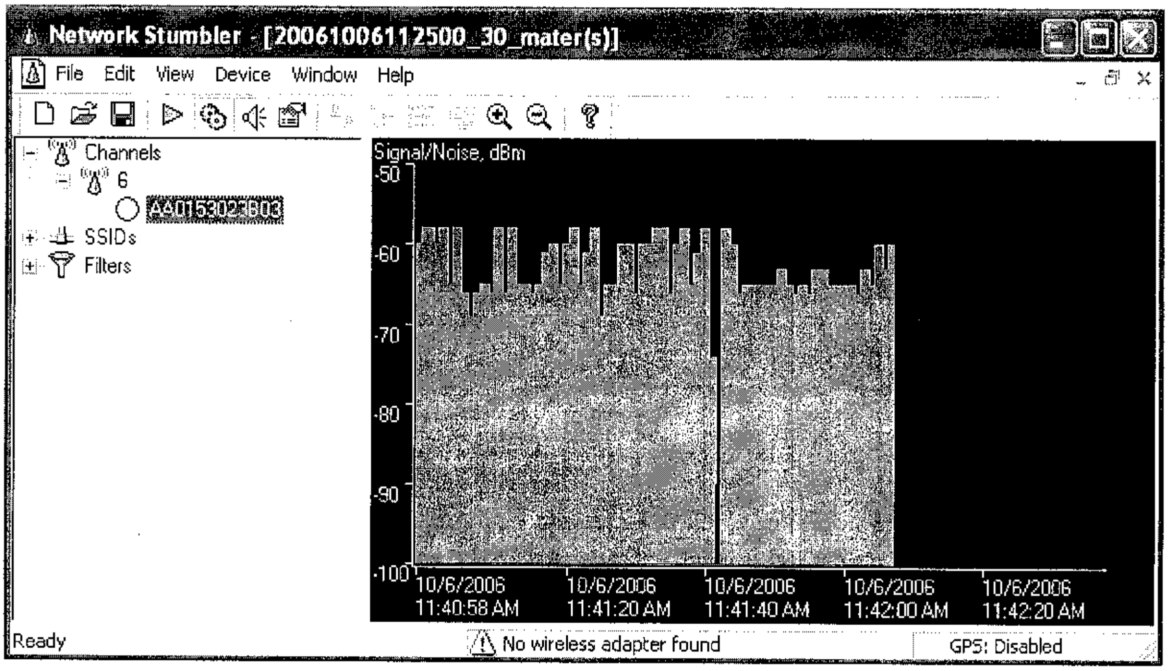


Fig. 3: Scanning Operation in Graphical View

MATLAB

MATLAB® is a high-performance language for technical computing. The name MATLAB stands for matrix laboratory and it was created by MathWorks. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation [10]. Typical uses include:

- Math and computation
- Algorithm development
- Data acquisition
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including graphical user interface building

3.3 Signal Strength Measurement

This project involves several signal strength measurements in order to know the trend of signal strength at different separation distance between the target (receiver) and each reference laptop (transmitter). The measurements are conducted for each of the network adapters. This information is used to derive the radio propagation model that will be used in the programming part of this project.

3.3.1 The Test Site

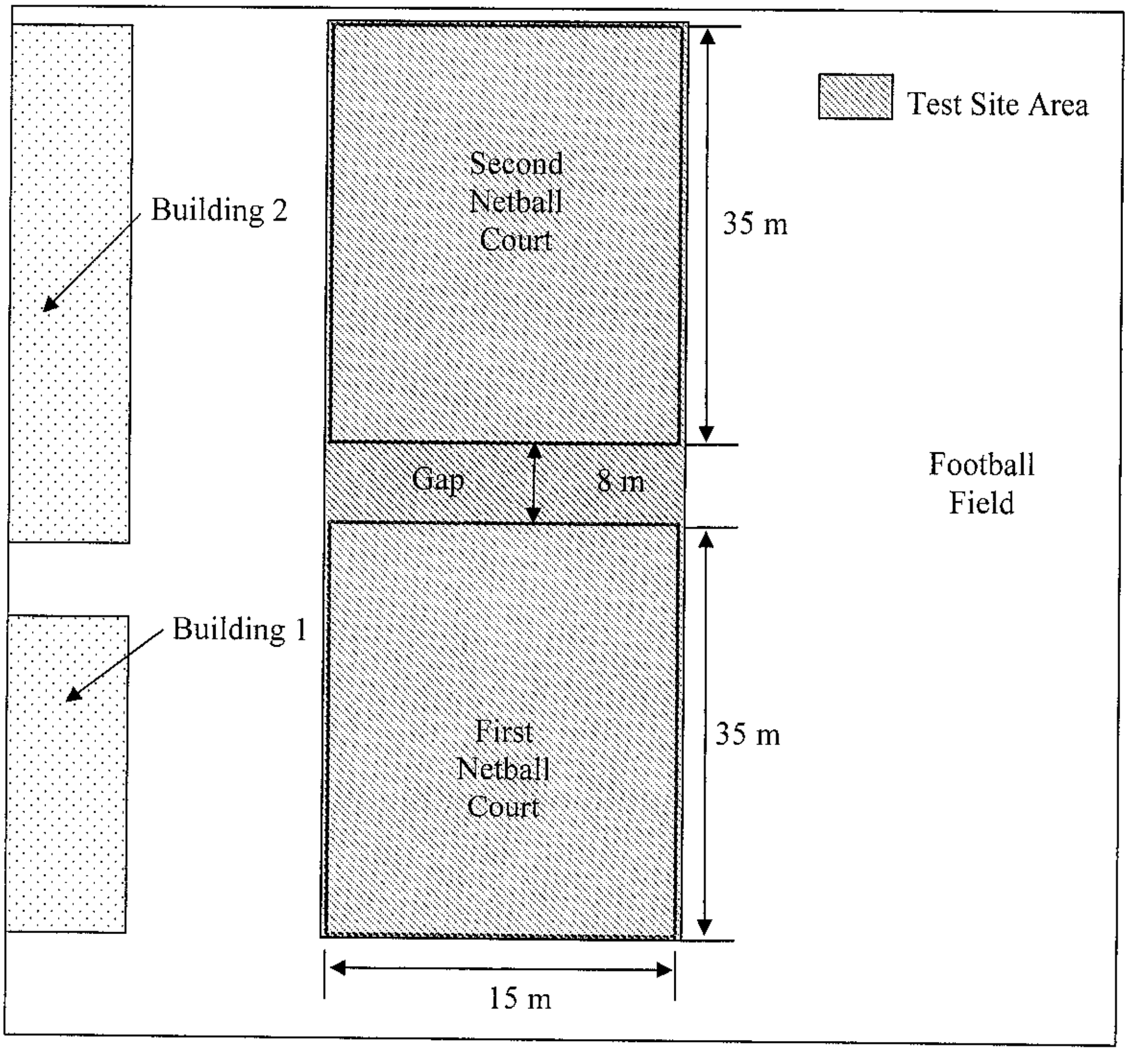


Fig. 4: Test Site Area

The test site is shown in Fig. 4 above. This site was chosen due to several reasons:

- It is located on the open space.
- It is far away from the source of interference (other 802.11 wireless networks)
- The site is large enough for the signal strength measurement to be conducted.

This test site is a combination of two basketball courts. Each court has the dimension of 17 m × 35 m so the combination of them will be 17 m × 70 m. The test site includes the 8 m gap between the first and the second courts so the overall size for the test site is 17 m × 78 m.

3.3.2 Measurement Procedures

This measurement is conducted to study the relationship between the signal strength and separation distance of the transmitter and receiver. This relationship will be used to develop the radio propagation model. In this measurement, only two laptops were used; i) *reference laptop* ii) *target laptop*. Each of the laptops is placed on a chair so that their height from the ground is 1 m. This height is used throughout this project for all the signal strength measurement for standardization. The reference laptop is placed at a fixed location which will initiate the ad-hoc network. On the other hand, the target laptop is used to measure the signal strength from reference laptop using the NetStumbler software.

The initial separation between the two laptops is 1 m. At this point the signal strength transmitted by the reference laptop is measured using the software for about 2 minutes. This long time is needed in the measurement process so that the variations in the received signal strength due to changing in environment which will affect the wireless channel can be captured. Several reading is taken and the average values are used for the calculation. By doing this, the results will be more accurate. The separation distance is increased by moving the target laptop away from reference laptop in the following sequence:

- 2 m
- 3 m
- 4 m
- 5 m
- 7 m
- 10 m
- 15 m
- 20 m
- 25 m
- 30 m
- 32 m
- 42 m
- 45 m
- 50 m
- 55 m
- 60 m
- 65 m
- 70 m
- 75 m

The reason to choose these separation distance distances is because at the near distances (1 m to 10 m) the small changes in distance will significantly affect the received signal strength. However this is not the case for the far distance (1 m to 75 m). For these far distances, the increment of separation distance has to be large enough to affect the received signal strength. Hence the increment of 5 m is used for the signal strength measurement at far distance. However after the measurement at 30 m, separation is increased to 32 and 42. This is due to the 8 m gaps between the two basketball courts. The target laptop cannot be placed within the gap because the height of the target laptop from the ground will not be the same anymore as the area of the gap is lower compare to the basketball court. The collected data are used to plot the graphs and deriving the radio propagation model.

There are 4 PCMCIA wireless adapters being used in this project. One of them which are the LINKSYS card is used by the target laptop and the remaining three are used by the reference laptops. The measurement procedures explained above are performed for all the three PCMCIA cards so that the radio propagation model between the target and each of them can be constructed. The individual signal strength measurement for each PCMCIA card is needed because every wireless adapter card has its own receiving and transmitting characteristics. In other word, the performance of two identical network adapters will not be the same. This is due to the difference in the antenna tuning and the transmit power of the adapters.

3.4 Radio Propagation Model

A Radio Propagation Model, also known as the Radio Wave Propagation Model or the Radio Frequency Propagation Model, is an empirical mathematical formulation for the characterization of radio wave propagation as a function of frequency, distance and other conditions. A single model is developed to predict the behavior of propagation for all similar links under similar constraints. It is created with the goal of formalizing the way radio waves propagated from one place to another. The models typically predict the path loss along a link or the effective coverage area of a transmitter.[8]

Each individual wireless link has to encounter different terrain, path, obstructions, atmospheric conditions and other phenomena, so it is impossible to formulate the exact path loss for all wireless links in a single mathematical equation. As a result, different models exist for different types of radio links under different conditions.

For this project, there are two ways to obtain the model, the first one using *median path loss* and the other using *combination modeling*. These two models are discussed in the next two sections

3.4.1 Median Path Loss Model

The median path loss model [9], as shown in (1) gives signal power relationship between transmitters and receivers placed at a distance d. The signal strength at some reference distance, do, $P(d_o)$ values is depending on the specific hardware and the transmission power. Other environmental factors affecting the wireless channel such as reflection and diffraction are captured in path loss exponent, γ. In other word, it indicates the rate at which the path loss increases with distance. This value may be affected by reflection, diffraction, air temperature, etc.

$$P(d)_{dBm} = P(d_o)_{dBm} - 10\gamma \log(d/d_o) \qquad (1)$$

Where

$P(d)$ = signal power when sender and receiver are separated by a distance d.
$P(d_o)$ = signal strength at some reference distance d_o.
γ = path loss exponent

The value of γ can be obtained by finding the minimum mean square error of the power measurement given by (2).

$$F(\gamma) = \sum_{d=1}^{J} [P(d)_{dBm} - P_T - (P(d_o) - 10\gamma \log_{10}(d))]^2 \qquad (2)$$

$F(\gamma)$ is dependent on separation distance between the target laptop and a reference laptop, d and P_T is the transmit power of the transmitter. In order to obtain the value of γ, equation (2) first need to be differentiated with respect to γ

$$\frac{\partial F(\gamma)}{\partial \gamma} = \frac{\partial}{\partial \gamma}\left[\sum_{d=1}^{J} [P(d)_{dBm} - P_T - (P(d_o) - 10\gamma \log_{10}(d))]^2\right] \qquad (3)$$

Then by setting (3) to zero and solve the equation, the value of γ can be obtained.

3.4.2 Combination Modeling

The combination modeling is a method to obtain the radio propagation model in which two or more equations will be combined in order to represent an accurate signal path loss in a certain environment. For this project, this model can only be obtained after the signal strength measurement data being analyzed. From the graph of "Signal Strength at Different Separation Distance", the best equations that fit the curve best is used and combined to represent the model. Usually the 6^{th} order polynomial equation is used to represent the non linear part of the graph and linear equation will be used to represent the straight line portion of the graph. Both of these equations can be obtained by calculating the least squares fit for the graph and this function is readily available in Microsoft Excel.

3.5 Programming

Programming part is necessary for this project as it involves computation some computation. The main programming tasks involved are to create the Matlab Graphical User Interface (GUI) and to coding for triangulation calculation.

3.5.1 Matlab Graphical User Interface (GUI)

This project involves creating the graphical user interface using Matlab for the triangulation calculation. The coding for the Matlab GUI is shown in Appendix I. The reasons for choosing Matlab to create the GUI are as follows:

- Matlab will be used to do the triangulation calculation, so it is better to use Matlab itself for the GUI.
- Matlab can easily import data from Microsoft Excel. So it is easier to take the data from the signal strength measurement which is saved as Excel files.

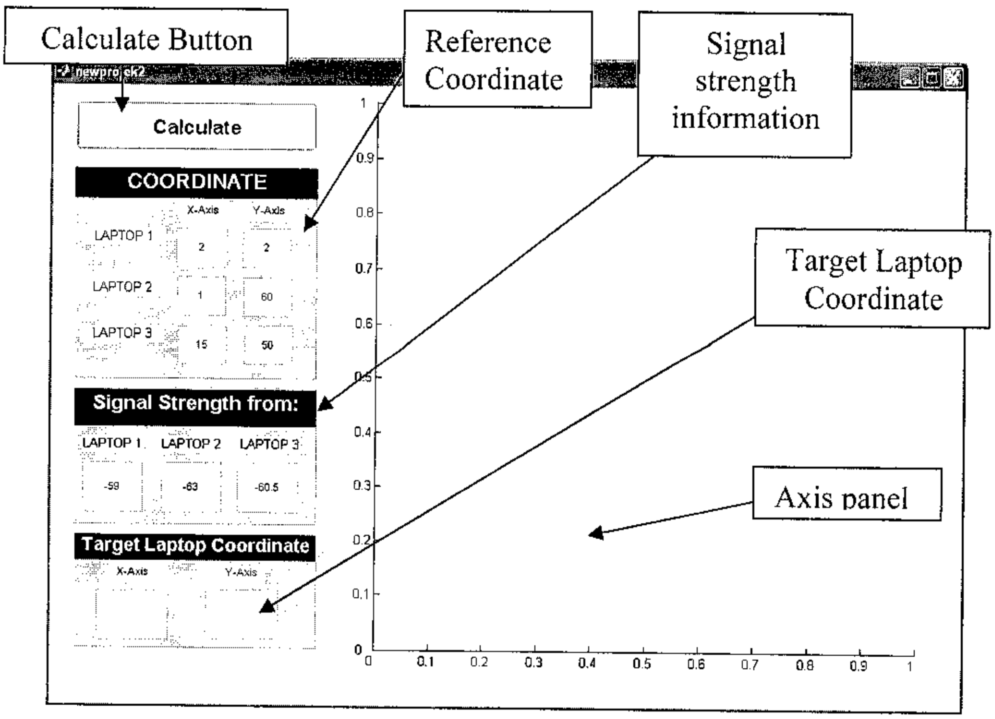


Fig. 5: The Graphical User Interface (GUI) Layout

Fig. 5 shows the layout for the GUI. The explanations of the layout are provided in TABLE 3.

TABLE 3
GUI Layout Explanation

Axis panel	This window will display the coordinate of the three reference laptops and also the target laptop after the triangulation calculation.
Reference Coordinate	The user will key in the coordinate of the each of the three reference laptops.
Signal Strength Information	The signal strength from each reference laptop measured by the target laptop will be keyed in into the corresponding boxes.
Compute Button	To calculate the location of the target laptop.

3.5.2 Triangulation Coding

This code is included in the GUI m-file and its main purpose is to find the coordinate of the target laptop when the following parameters are known:

- Coordinate of the 3 reference laptops
- The distance of each reference laptop from the target laptop

This code will compute the possible coordinate of the target laptop based on the triangulation algorithm. The inputs for this code are the *coordinates* of the three reference laptops and also the *signal strength* from each reference laptop measured by the target laptops. Other than computing the possible target laptop position, this coding also plot three circles each centered at the three reference points. The radii of the circles are determined from the signal strength measurement from the target laptop to the three reference laptop. The plots are shown in Fig. 6 where the 3 reference points are labeled as R1, R2, and R3. From the circles, the coordinate of the target laptop can be estimated by calculating the average of the three intersection points on intersection (A, B, C) of the three circles. However, this coordinate is not really the exact position of the target laptop. The best estimation that can be made is that the target laptop is located within the intersection region of the three circles. The triangulation coding together with some explanation is available in Appendix I starting from line 163 until 363.

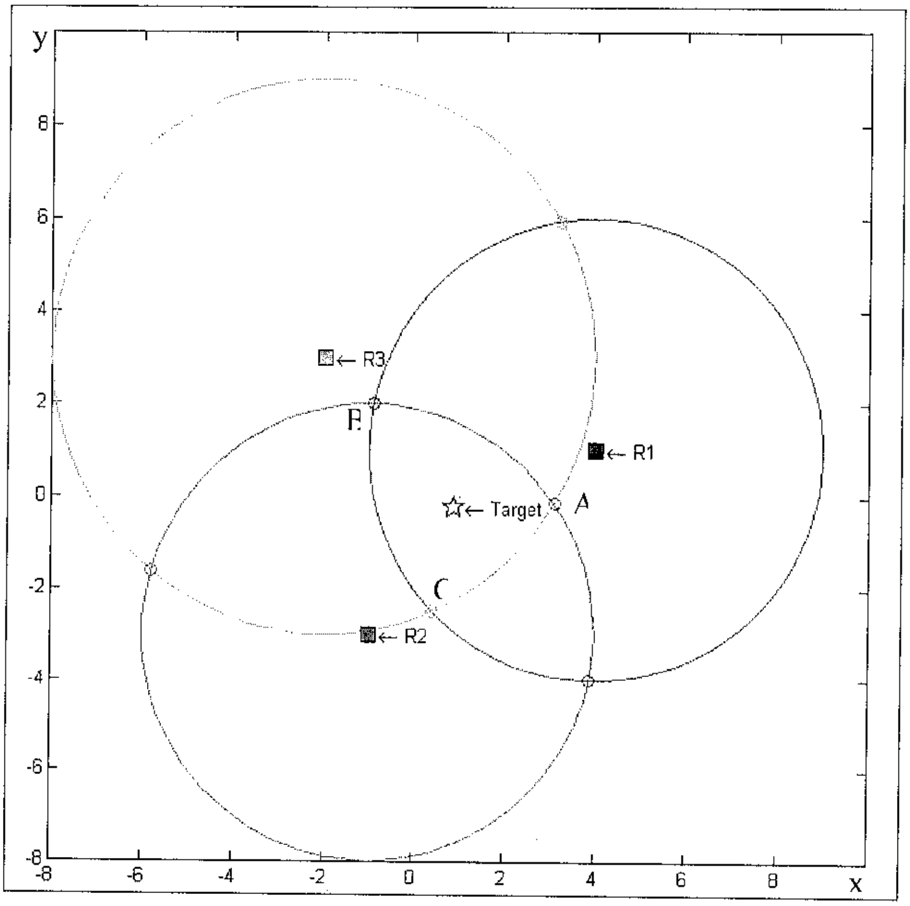
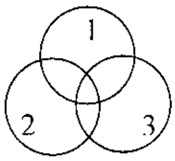
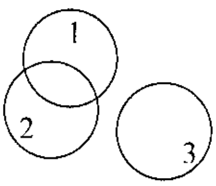
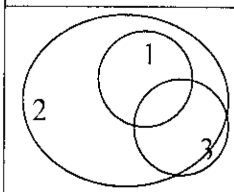
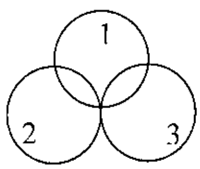
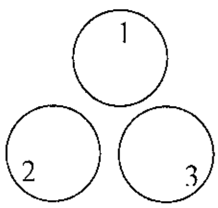
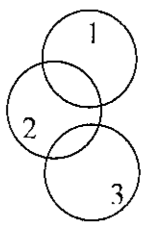


Fig. 6: Plot of 3 Circles on x-y Plane

3.5.3 Triangulation Plot Pattern

Using the triangulation coding, several patterns of the plot can be produced and it depends on the accuracy of the NetStumbler software to measure the signal strength and also the performance of each network adapters to transmit constant power. TABLE 4 below summarizes some common patterns of the triangulation plot and some description about each of them.

TABLE 4
Some Patterns of the Triangulation Plot

Shape	Description	Shape	Description
	Most common pattern that can occur. This shows that the software scan correctly.		Shows that signal strength measurement from R3 is not correct and need to be repeated.
	Shows that signal strength measurement from R2 is not correct and need to be repeated.		The most ideal case where only one solution can be found but this is rarely happen.
	Signal strength measurement is not correct and all the measurements have to be repeated.		Signal strength measurement from R1 and R3 need to be repeated.

3.6 Location Sensing Verification

In this testing, 4 laptops are used to setup an ad-hoc WLAN. The coordinates of the 3 reference laptops R1, R2, and R3 are shown in TABLE 5. The target laptop is placed at arbitrary locations and NetStumbler software is initiated to measure the signal strength received from each reference laptop. Two tests are conducted. For the first test, the target laptop is placed at location (5,5) and for the second test the location of target laptop is set to (0,20). Then all the parameters; i) *the coordinates of the three reference laptops* ii) *signal strength from each reference laptop,* are used to estimate the coordinate of the target laptop.

TABLE 5
Reference Laptop Coordinate for the Final Testing

Reference Laptop	Coordinates
1	(0,0)
2	(15,15)
3	(0,30)

CHAPTER 4
RESULTS AND DISCUSSION

4.1 Measurement between 2 Laptops using Different Network Adapters

Three measurements are conducted for three different links, i) LINKSYS-LINKSYS[R1], ii) LINKSYS – Cisco[R2], iii) LINKSYS – Cisco[R3]. The details on the network adapters used and the transmit power of the transmitter for each of the measurement are shown in Table 6 below.

TABLE 6
Network Adapters Setup for each Measurement

Links	Network Adapter for Target Laptop	Network Adapter for Reference Laptop	
	Model	**Model**	**Transmit Power**
LINKSYS-LINKSYS[R1]	LINKSYS WPC54G (S/N:BDH61F437214)	LINKSYS WPC54G (S/N:BDH61F437206)	18 dBm
LINKSYS–Cisco[R2]	LINKSYS WPC54G (S/N:BDH61F437214)	Cisco AIR-PCM352 (S/N:AMB07220M7E)	20 dBm
LINKSYS–Cisco[R3]	LINKSYS WPC54G (S/N:BDH61F437214)	Cisco AIR-PCM352 (S/N: AMB07220528)	20 dBm

The following are the results for the signal strength measurements and the derivation of the radio propagation model for every link.

4.1.1 LINKSYS – LINKSYS[R1]

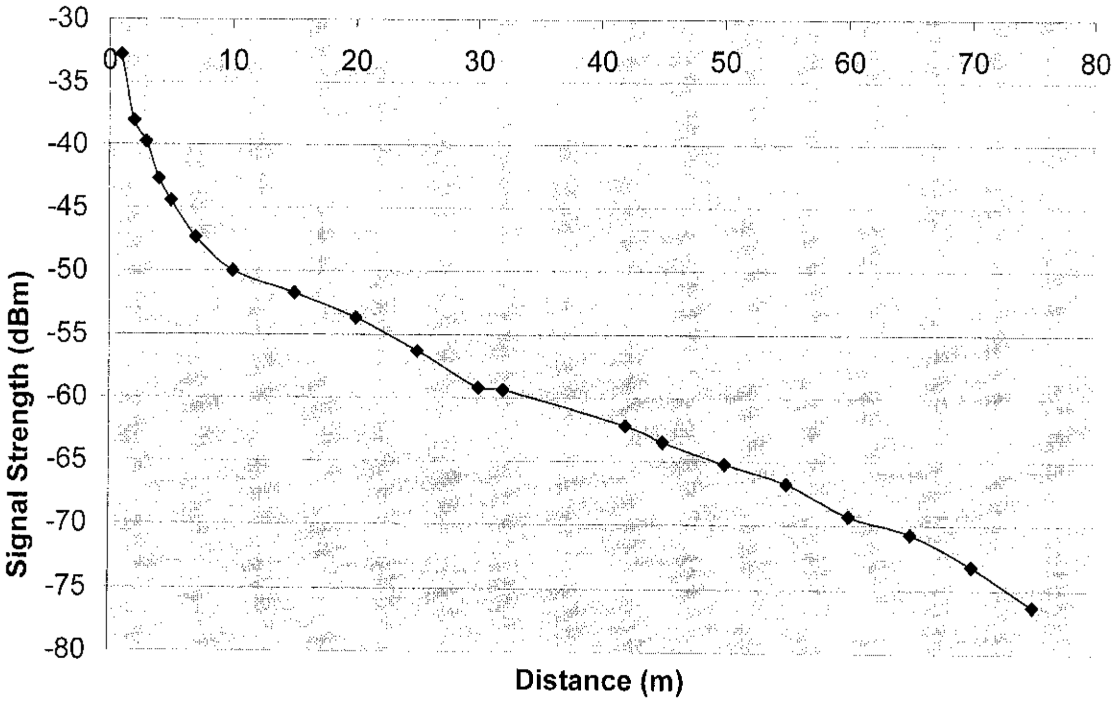


Fig. 7: Signal Strength for Different Separation Distance (LINKSYS-LINKSYS[R1])

Fig. 7 shows the signal strength received by the target laptop at different separation distances for LINKSYS-LINKSYS[R1] link. At near distances (1m to 10 m), signal strength drop rapidly when the separation distance increases. However for the larger separation distance (11m to 75 m), the signal strength dropped linearly with the increase of the separation distance.

The calculation to obtain the radio propagation model for this link will be discussed in the next two sections.

Median Path Loss Model

The path loss exponent, γ for this link is obtained using (1), (2) and (3). The value of γ is found out to be 3.1304

So by using equation (1) with d_o set to 1 m and γ equal 3.1304 the radio propagation model for the LINKSYS-LINKSYS[R1] link becomes

$$P(d)_{dBm} = -32.8018 - 31.304\ \log(d) \qquad (4)$$

Combination Modeling

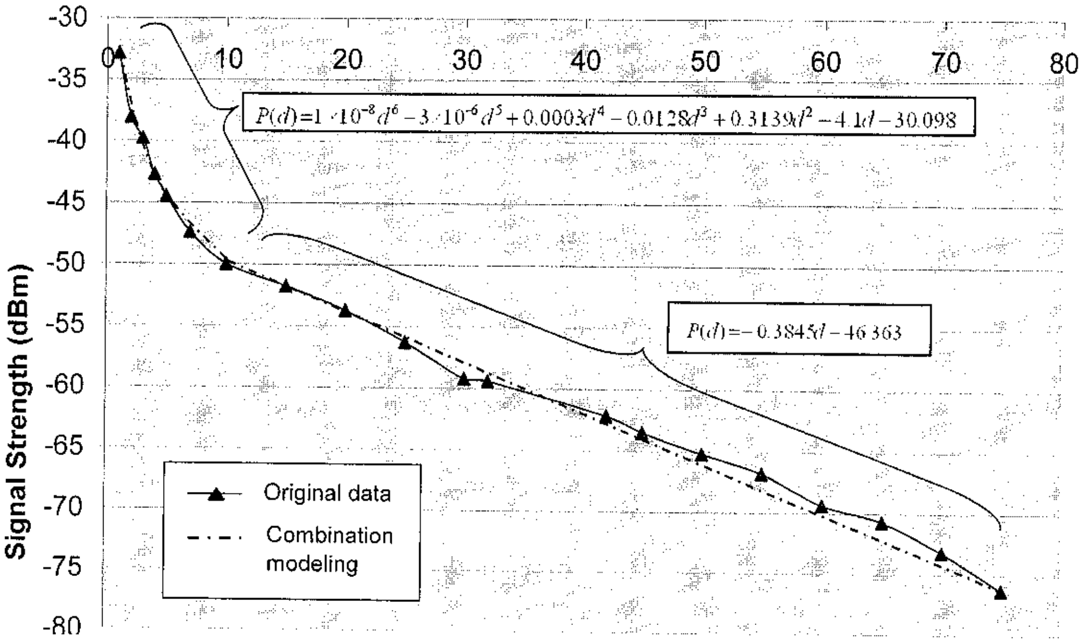


Fig. 8: Combination Modeling (LINKSYS-LINKSYS[R1])

Fig. 8 shows the combination modeling derivation for LINKSYS-LINKSYS[R1] link. From the figure, it is found out that for signal strength, $P(d) \geq$ -49. 98655322, the radio propagation model for this link can be represent by the 6th order polynomial

$$P(d) = 1\times10^{-8}d^6 - 3\times10^{-6}d^5 + 0.0003d^4 - 0.0128d^3 + 0.3139d^2 - 4.1d - 30.098 \quad (5)$$

For signal strength, $P(d) < -49.98655322$, the model can be represented by the following linear equation

$$P(d) = -0.3845d - 46.363 \tag{6}$$

Using the median path loss model, the integral absolute error, which is the sum of the differences between the model and the original data, is 1.3642. For the combination modeling, the integral absolute error is 0.0285. The second model is far superior compare to the first one.

4.1.2 LINKSYS – Cisco [R2]

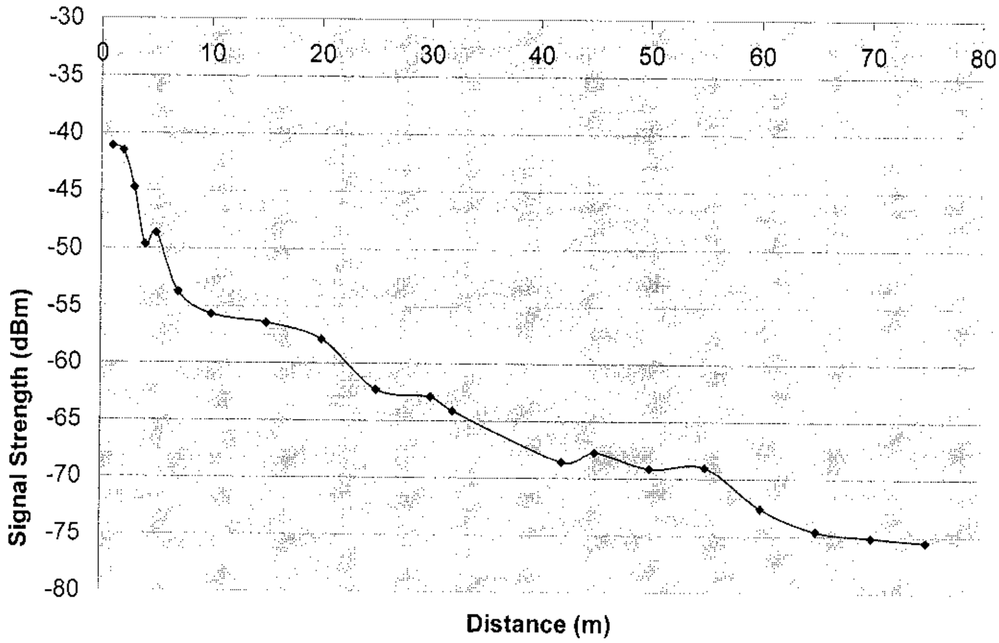


Fig. 9: Signal Strength for Different Separation Distance (LINKSYS-Cisco[R2])

Fig. 9 shows the signal strength received by the target laptop at different separation distances for LINKSYS-Cisco[R2] link. When the separation distance is increased from 4 m to 5 m, the signal strength increases a little bit by 2dBm and not decreases as expected. This is maybe due to reflection of the signal to the ground or near building and this is in some way additively increasing the signal strength received by the target laptop. For the separation distance of 40 m to 55 m, the signal strength is maintained at -68 *dBm*. This is maybe due to the deflection effect caused by 8 m gap between the first and the second courts (refer section 3.3.1).

The calculation to obtain the radio propagation model for this link will be discussed in the next two sections.

i) Median Path Loss Model

The path loss exponent, γ for this link is obtained using (1), (2) and (3). The value of γ is found out to be 2.9683

So by using equation (1) with d_o set to 1 meter and γ equal 2.9683 the radio propagation model for the LINKSYS-Cisco[R2] link becomes

$$P(d)_{dBm} = -41.105691 - 29.683 \log(d) \quad (7)$$

ii) Combination Modeling

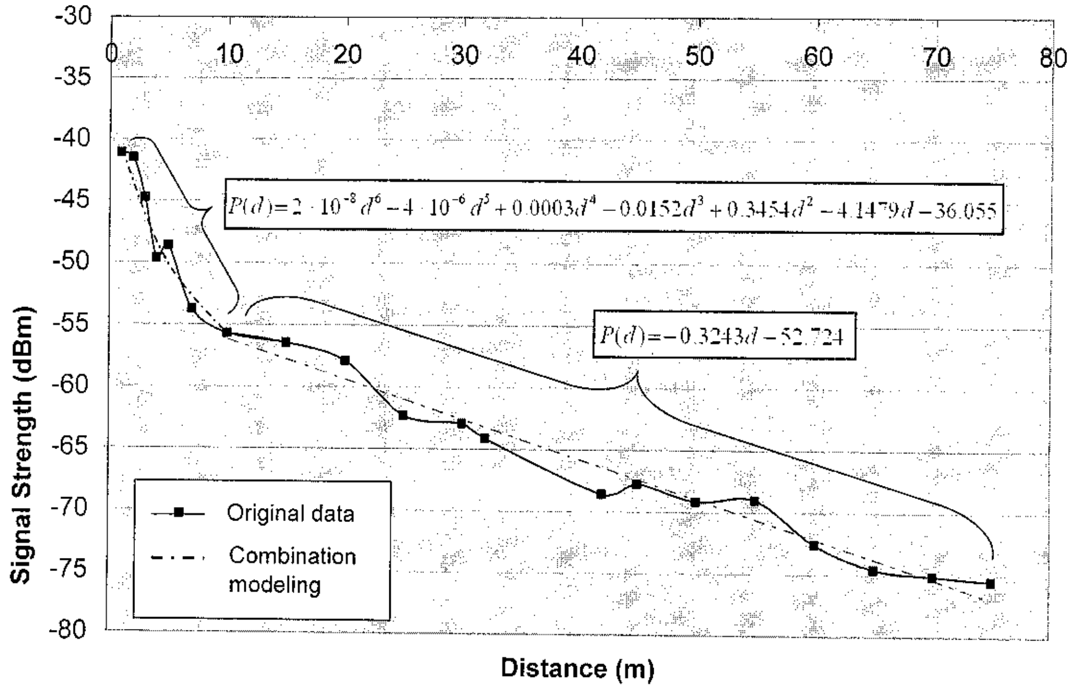


Fig. 10: Combination Modeling (LINKSYS-CISCO[R2])

Fig. 10 shows the combination modeling derivation for LINKSYS-Cisco[R2] link. From the figure, it is found out that for signal strength, $P(d) \geq -55.71869$, the radio propagation model for this link can be represented by the 6th order polynomial

$$P(d) = 2\times10^{-8} d^6 - 4\times10^{-6} d^5 + 0.0003 d^4 - 0.0152 d^3 + 0.3454 d^2 - 4.1479 d - 36.055 \quad (8)$$

For signal strength, $P(d)$ < -55.71869, the model can be represented by the following linear equation

$$P(d) = -0.3243d - 52.724 \tag{9}$$

Using the median path loss model, the integral absolute error, which is the sum of the differences between the model and the original data, is 1.6832. For the combination modeling, the integral absolute error is 0.1069. The second model is far superior compare to the first one.

4.1.3 LINKSYS – Cisco [R3]

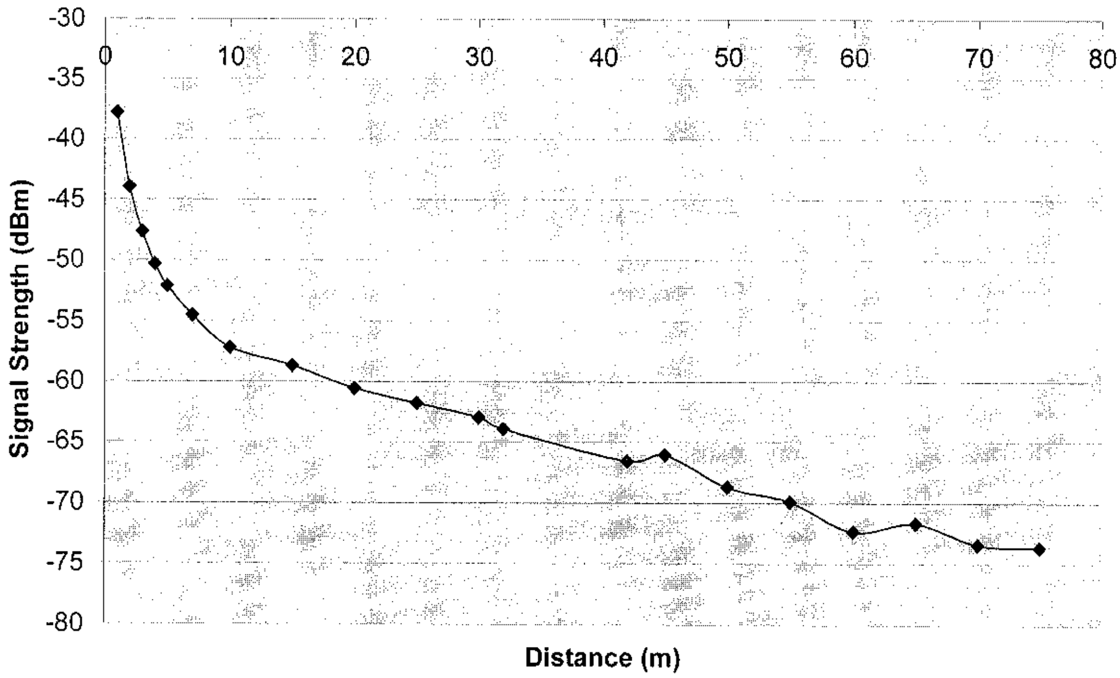


Fig. 11: Signal Strength for Different Separation Distance (LINKSYS-Cisco[R3])

Fig. 11 shows the signal strength received by the target laptop at different separation distances for LINKSYS-Cisco[R3] link. At near distance (1m to 10 m), the signal strength drop significantly and the rate of the decrement is decreased when the separation distance is increased. However for the larger separation distance (11m to 75 m), the signal strength dropped linearly with the increment of the separation distance. The trend for this link is somehow quite similar to the LINKSYS-LINKSYS[R1] link.

The calculation to obtain the radio propagation model for this link will be discussed in the next two sections.

i) Median Path Loss Model

The path loss exponent, γ for this link is obtained using (1), (2) and (3). The value of γ is found out to be 3.1771

So by using equation (1) with d_o set to 1 meter and γ equal 3.1771 the radio propagation model for the LINKSYS-Cisco[R3] link becomes

$$P(d)_{dBm} = -37.79464 - 31.771 \log(d)$$

(10)

ii) Combination Modeling

□

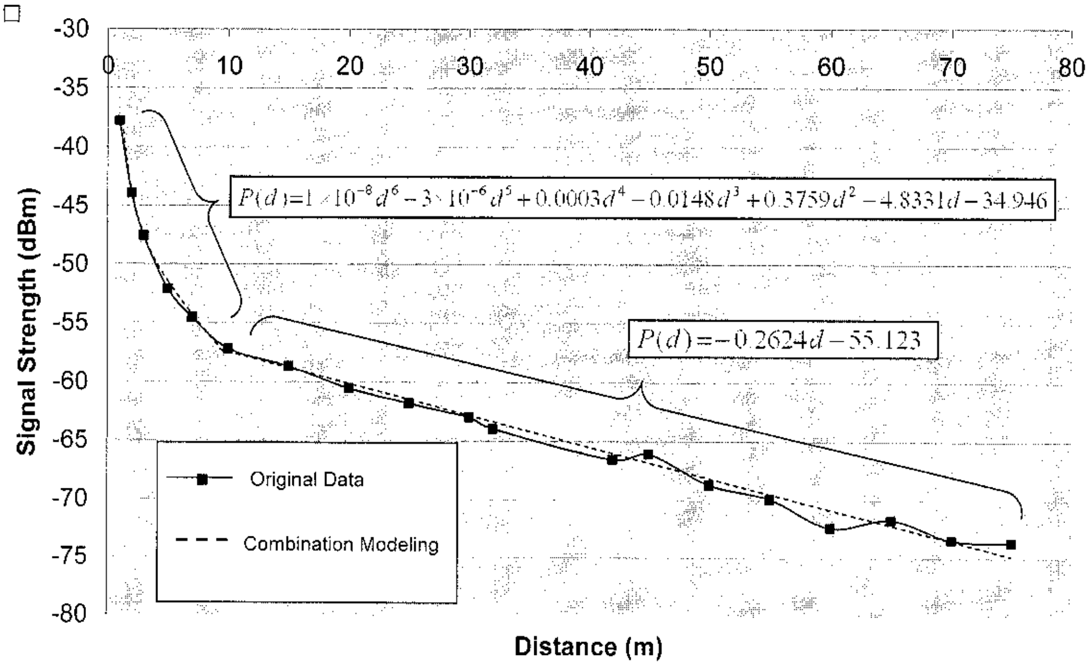


Fig. 12: Combination Modeling (LINKSYS-CISCO[R3])

Fig. 12 shows the combination modeling derivation for LINKSYS-Cisco[R3] link. From the figure, it is found out that for signal strength, $P(d) \geq -57.2035$ the radio propagation model for this link can be represented by the 6th order polynomial

$$P(d) = 1\times10^{-8}d^6 - 3\times10^{-6}d^5 + 0.0003d^4 - 0.0148d^3 + 0.3759d^2 - 4.8331d - 34.946 \quad (11)$$

For signal strength, $P(d)$ < -57.2035 the model can be represented by the following linear equation

$$P(d) = -0.2624d - 55.123 \quad (12)$$

Using the median path loss model, the integral absolute error, which is the sum of the differences between the model and the original data, is 1.8156. For the combination modeling, the integral absolute error is 0.0781. The second model is far superior compare to the first one.

For all the measurements, the combination modeling produced better results in minimizing the integral absolute error. So these models are used for the calculation part in the triangulation coding to determine the distance with a given signal strength information.

4.2 Location Sensing Verification

The coordinates of the three reference laptops, the signal strength from each reference laptop and the computed target laptop coordinate are shown in Fig. 13 below:

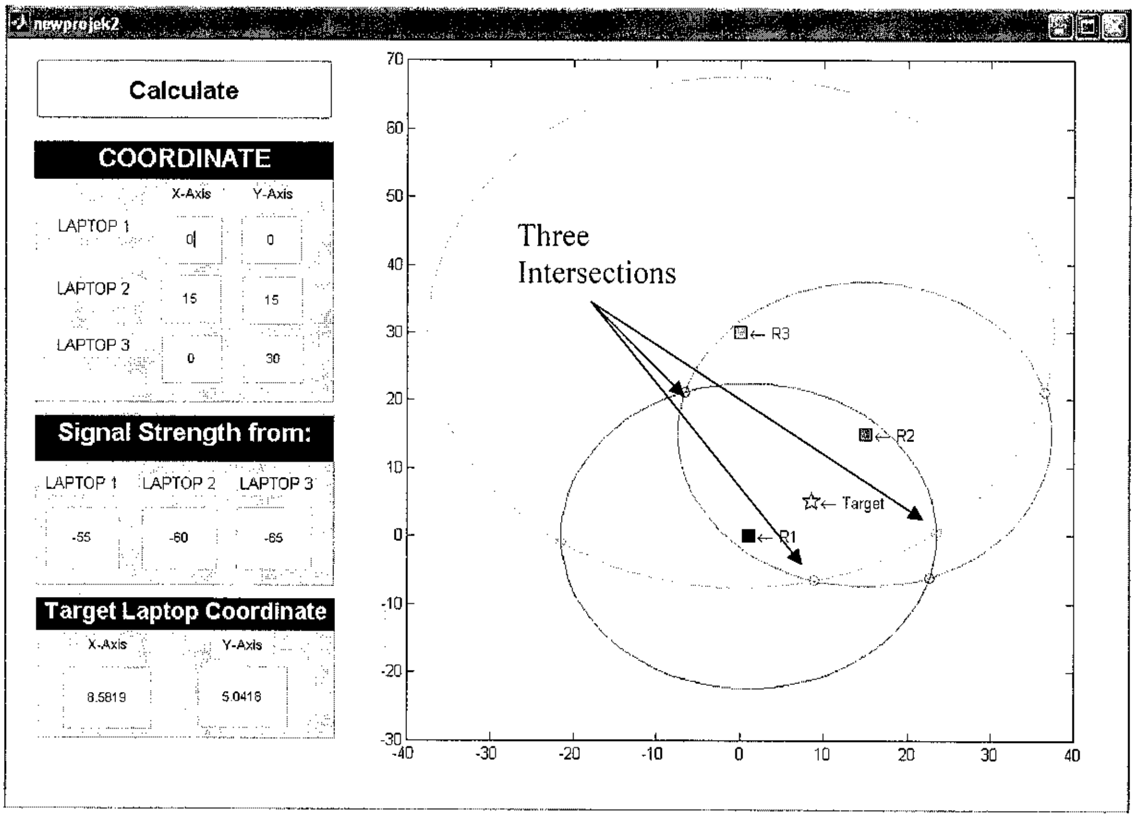


Fig. 13: Results for the First Testing

For the first testing, the target laptop was placed at coordinate (5 , 5) and from the result, the estimated target laptop coordinate is (8.5819 , 5.0418). This is about 3.6 m deviated from the real location. The estimated target location is shown by the 'star' mark in the plot. The location is determined by calculating the average of the three intersection points of the three circles.

For the second testing, the target laptop was placed at coordinate (0, 20) and the estimated target location from the triangulation calculation is (0.32077 , 25.2784). This shows the deviation of approximately 5 m from the real location. The result for the second testing is shown in Fig. 14 below.

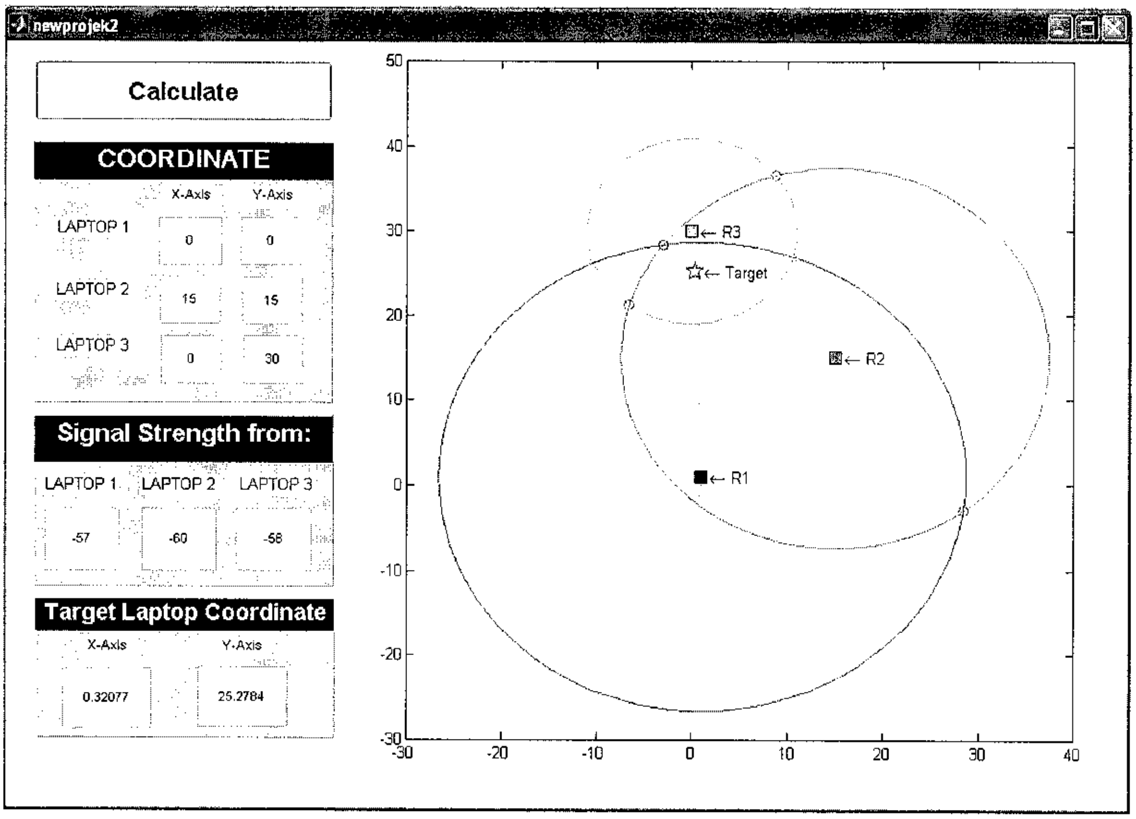


Fig. 14: Results for the Second Testing

CHAPTER 5
CONCLUSION

In this project, the user location technique is implemented over the 802.11 ad-hoc network for outdoor environment. This project proved that the location of user can be estimated provided that we know the *location* and also the *received signal strength* from at least three reference laptops. Several signal strength measurements have been conducted for different network adapter configurations and the radio propagation models are derived for each of the configuration. One program has been developed using Matlab and its function is to estimate the target laptop coordinate. The triangulation algorithm has been embedded in that program. The best radio propagation model for each configuration is chosen to be included in the triangulation algorithm. Several testing during the last portion of this project showed that the margin or error for this technique is about 5 to 10 m. The performance of this technique is totally relying on the accuracy of measuring software (NetStumbler) and also the performance of the wireless adapters.

CHAPTER 6
RECOMMENDATION

There are several improvements can be made so that this project can produce a better results with higher accuracy. All the recommendations are listed TABLE 7 below:

TABLE 7
Recommendation for the Project

Recommendation	**Reason**
• Use high performance network adapters.	• To improve the accuracy of the location estimation.
• Use the same model of network adapters.	• To produce less variation in radio propagation model.
• Find better measuring software and more accurate software. (Linux based software).	• The accurate measuring software can increase the location estimation performance.
• Automate the process of : o Collecting data o Finding the coordinate of the target laptop.	• The real time location estimation can be performed.
• Perform more advance algorithm to calculate the target laptop coordinate.	• This is to increase the accuracy of the location estimation.

REFERENCES

[1] Paramvir Bahl and Venkata N Padmanabhan, "RADAR: An in-building RF Based User Location and Tracking System," (*Proc. of IEEE INFOCOM 2000*) pp. 775-784.

[2] Siddharta Saha, Kamalika Chaudhuri, Dheeraj Sanghi, Pravin Bhagwat, "Location Determination of a Mobile Device Using IEEE 802.11b Access Point," (*Proc. of IEEE Wireless Communications and Networking Conference 2003*)

[3] Youngjune Gwon, Ravi Jain, and Toshiro Kawahara, "Robust Indoor Location Estimation of Stationary and Mobile Users," (*Proc. of IEEE INFOCOM 2004*)

[4] Song Li, Gang Zhou and Lin Liao, "User Location Service over an 802.11 Ad-Hoc Network,"

[5] Details on IEEE 802.11 standard, **http://en.wikipedia.org/wiki/IEEE_802.11** , 3rd March 2006.

[6] NetStumbler Website, **http://www.netstumbler.com/**, 1st March 2006.

[7] Introduction to NetStumbler, **http://www.wi-fiplanet.com/tutorials/article.php/3589131**, 1st March 2006

[8] Wikipedia, the free encyclopedia, "Radio Propagation Model" **http://en.wikipedia.org/wiki/Radio_propagation_model**, 15th July 2006

[9] S. Haykin and M. Moher, "Median Path Loss," in *Modern Wireless Communications*, Pearson Prentice Hall, 2005, pp. 30-31.

[10] Wikipedia, the free encyclopedia, "MATLAB" **http://en.wikipedia.org/wiki/Matlab** , 29th August 2006

[11] Rent/Lease Computers, "Cisco PCM 352 Product Detail" **http://www.erpcrentals.com/products/search/detail.cfm?MFRModel=CIS-AIR-PCM352&CurrentPage=2** 10th October 2006

APPENDICES

Appendix I

Matlab GUI Coding

```
unction varargout = SIAP(varargin)
 SIAP M-file for SIAP.fig

   Graphical User Interface for "Outdoor User Location using 802.11 Ad-Hoc
   Network" project.

 Copyright 2002-2003 The MathWorks, Inc.
 Begin initialization code - DO NOT EDIT
ui_Singleton = 1;
ui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @SIAP_OpeningFcn, ...
                  'gui_OutputFcn',  @SIAP_OutputFcn, ...
                  'gui_LayoutFcn',  @SIAP_LayoutFcn, ...
                  'gui_Callback',   []);
f nargin && ischar(varargin{1})
   gui_State.gui_Callback = str2func(varargin{1});
nd

f nargout
   [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
lse
   gui_mainfcn(gui_State, varargin{:});
nd
 End initialization code - DO NOT EDIT

 --- Executes just before SIAP is made visible.
unction SIAP_OpeningFcn(hObject, eventdata, handles, varargin)

 Choose default command line output for SIAP
andles.output = hObject;

 Update handles structure
uidata(hObject, handles);

 --- Outputs from this function are returned to the command line.
unction varargout = SIAP_OutputFcn(hObject, eventdata, handles)

 Get default command line output from handles structure
arargout{1} = handles.output;

unction R1x_Callback(hObject, eventdata, handles)
 hObject    handle to R1x (see GCBO)

 --- Executes during object creation, after setting all properties.
unction R1x_CreateFcn(hObject, eventdata, handles)
 hObject    handle to R1x (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction R1y_Callback(hObject, eventdata, handles)
 hObject    handle to R1y (see GCBO)
```

```
 --- Executes during object creation, after setting all properties.
unction R1y_CreateFcn(hObject, eventdata, handles)
 hObject    handle to R1y (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction R2x_Callback(hObject, eventdata, handles)
 hObject    handle to R2x (see GCBO)

 --- Executes during object creation, after setting all properties.
unction R2x_CreateFcn(hObject, eventdata, handles)
 hObject    handle to R2x (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction R2y_Callback(hObject, eventdata, handles)
 hObject    handle to R2y (see GCBO)

 --- Executes during object creation, after setting all properties.
unction R2y_CreateFcn(hObject, eventdata, handles)
 hObject    handle to R2y (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction R3x_Callback(hObject, eventdata, handles)
 hObject    handle to R3x (see GCBO)

 --- Executes during object creation, after setting all properties.
unction R3x_CreateFcn(hObject, eventdata, handles)
 hObject    handle to R3x (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction R3y_Callback(hObject, eventdata, handles)
 hObject    handle to R3y (see GCBO)

 --- Executes during object creation, after setting all properties.
unction R3y_CreateFcn(hObject, eventdata, handles)
 hObject    handle to R3y (see GCBO)
```

```
f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction D1_Callback(hObject, eventdata, handles)
 hObject    handle to D1 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction D1_CreateFcn(hObject, eventdata, handles)
 hObject    handle to D1 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction D2_Callback(hObject, eventdata, handles)
 hObject    handle to D2 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction D2_CreateFcn(hObject, eventdata, handles)
 hObject    handle to D2 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction D3_Callback(hObject, eventdata, handles)
 hObject    handle to D3 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction D3_CreateFcn(hObject, eventdata, handles)
 hObject    handle to D3 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

 --- Executes on button press in Calculate.
unction Calculate_Callback(hObject, eventdata, handles)
 hObject    handle to Calculate (see GCBO)

==========================================================================
===========================TRIANGULATION CODE=============================
==========================================================================
 Begin triangulation code - DO NOT EDIT

old off;
```

```
lot(0);
equation of circle.. to find intersection between circle 1 and circle 2
x,y]=solve('(x-R1(1))^2+(y-R1(2))^2=a^2','(x-R2(1))^2+(y-R2(2))^2=b^2');

equation of circle.. to find intersection between circle 3 and circle 2
xx,yy]=solve('(xx-R3(1))^2+(yy-R3(2))^2=c^2','(xx-R2(1))^2+(yy-R2(2))^2=b^2');

equation of circle.. to find intersection between circle 3 and circle 1
xxx,yyy]=solve('(xxx-R3(1))^2+(yyy-R3(2))^2=c^2','(xxx-R1(1))^2+(yyy-R1(2)) ↙
');

Get value of reference point inputted by user
l(1) = str2double(get(handles.R1x,'String'));
l(2) = str2double(get(handles.R1y,'String'));
2(1) = str2double(get(handles.R2x,'String'));
2(2) = str2double(get(handles.R2y,'String'));
3(1) = str2double(get(handles.R3x,'String'));
3(2) = str2double(get(handles.R3y,'String'));

Get value of signal strength inputted by user
) = str2double(get(handles.D1,'String'));
) = str2double(get(handles.D2,'String'));
) = str2double(get(handles.D3,'String'));
=sym('n');

Solving the combination modeling for LINKSYS-LINKSYS link
f a0 >= -49.98655322
   FN1=1E-08*n^6 - 3E-06*n^5 + 0.0003*n^4 - 0.0128*n^3 + 0.3139*n^2 - 4.1*n - ↙
 - a0;
   AA=solve(FN1);
   a=AA(1);
   a=eval(a);
lse
   FN1=-0.3845*n - 46.363 - a0;
   AA=solve(FN1);
   a=eval(AA); %this is the distance between target laptop and Reference 1
nd

Solving the combination modeling for LINKSYS-Cisco[R2] link
f b0 >= -55.7186991869919
   FN2=0.00000002*n^6 - 0.000004*n^5 + 0.0003*n^4 - 0.0152*n^3 + 0.3454*n^2 - ↙
*n - 36.055 - b0;
   BB=solve(FN2);
   b=BB(1);
   b=eval(b);
lse
   FN2=-0.3243*n - 52.724 - b0;
   BB=solve(FN2);
   b=eval(BB); %this is the distance between target laptop and Reference 2
nd

Solving the combination modeling for LINKSYS-Cisco[R3] link
f c0 >= -57.2035714285714
   FN3=1E-08*n^6 - 3E-06*n^5 + 0.0003*n^4 - 0.0148*n^3 + 0.3759*n^2 - 4.8331*n - ↙
 - c0;
   CC=solve(FN3);
```

```
   c=CC(1);
   c=eval(c);
lse
   FN3=-0.2624*n - 55.123 - c0;
   CC=solve(FN3);
   c=eval(CC); %this is the distance between target laptop and Reference 3
nd

Define the size of circle to be plotted
lcircle=rsmak('circle',a,[R1(1),R1(2)]);
?circle=rsmak('circle',b,[R2(1),R2(2)]);
3circle=rsmak('circle',c,[R3(1),R3(2)]);

evaluate the equation of circle to find point of intersection by using the
distance (a b and c) calculated earlier
l = eval (x);
? = eval (xx);
3 = eval (xxx);
l = eval (y);
? = eval (yy);
3 = eval (yyy);



==========================================================================
Plotting the three circles
old on;

nplt(R1circle,'b',0.25), axis square
nplt(R2circle,'r',0.25), axis square
nplt(R3circle,'g',0.25), axis square
lot(R1(1),R1(2),'sr','MarkerEdgeColor','k',...
                'MarkerFaceColor','b',...
                'MarkerSize',10)
ext(R1(1),R1(2),'  \leftarrow R1')
lot(R2(1),R2(2),'sr','MarkerEdgeColor','k',...
                'MarkerFaceColor','r',...
                'MarkerSize',10)
ext(R2(1),R2(2),'  \leftarrow R2')
lot(R3(1),R3(2),'sg','MarkerEdgeColor','k',...
                'MarkerFaceColor','g',...
                'MarkerSize',10)
ext(R3(1),R3(2),'  \leftarrow R3')
==========================================================================
==========================================================================
Plotting the three Reference Points
or i=1:2
  plot (x1(i),y1(i),'ob')
nd

or i=1:2
 plot (x2(i),y2(i),'or')
nd

or i=1:2
   plot (x3(i),y3(i),'og')
```

```
ıd

ɔom on;
==========================================================================
==========================================================================
Finding the x-axis coordinate for the target laptop

ɔr i=1:2
   Gx1(i)=abs(x1(i)- R3(1));
   Gx2(i)=abs(x2(i)- R1(1));
   Gx3(i)=abs(x3(i)- R2(1));
ıd

‹x1=min(Gx1);
‹x2=min(Gx2);
‹x3=min(Gx3);

ɔr i=1:2
   if Gxx1==Gx1(i)
      T1(1)=x1(i)
   end
ıd
ɔr i=1:2
   if Gxx2==Gx2(i)
      T2(1)=x2(i)
   end
ıd
ɔr i=1:2
   if Gxx3==Gx3(i)
      T3(1)=x3(i)
   end
ıd
==========================================================================
==========================================================================
Finding the y-axis coordinate for the target laptop

ɔr i=1:2
   Gy1(i)=abs(y1(i)- R3(2));
   Gy2(i)=abs(y2(i)- R1(2));
   Gy3(i)=abs(y3(i)- R2(2));
ıd

yy1=min(Gy1);
yy2=min(Gy2);
yy3=min(Gy3);

ɔr i=1:2
   if Gyy1==Gy1(i)
      T1(2)=y1(i)
   end
ıd
ɔr i=1:2
   if Gyy2==Gy2(i)
      T2(2)=y2(i)
   end
ıd
```

```
or i=1:2
   if Gyy3==Gy3(i)
      T3(2)=y3(i)
   end
nd

==========================================================================
Finding the coordinate of the three nearest points to the target laptop
The target coordinate
Target = (T1+T2+T3)/3;

or i=1:2
   Target(i)=(T1(i)+T2(i)+T3(i))/3;
nd

plot (Target(1),Target(2),'*b')
lot (Target(1),Target(2),'pb','MarkerEdgeColor','k',...
              'MarkerFaceColor','y',...
              'MarkerSize',12)
ext(Target(1),Target(2),'  \leftarrow Target')

Display the estimated target laptop coordinate on the GUI
et(handles.Tx,'String',...
   num2str(Target(1)));

et(handles.Ty,'String',...
   num2str(Target(2)));

==========================================================================
======================END OF TRIANGULATION CODE=========================
==========================================================================

unction Tx_Callback(hObject, eventdata, handles)
 hObject    handle to Tx (see GCBO)

 --- Executes during object creation, after setting all properties.
unction Tx_CreateFcn(hObject, eventdata, handles)
 hObject    handle to Tx (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction Ty_Callback(hObject, eventdata, handles)
 hObject    handle to Ty (see GCBO)

 --- Executes during object creation, after setting all properties.
unction Ty_CreateFcn(hObject, eventdata, handles)
 hObject    handle to Ty (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
```

```
nd

unction edit22_Callback(hObject, eventdata, handles)
 hObject    handle to edit22 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit22_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit22 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit23_Callback(hObject, eventdata, handles)
 hObject    handle to edit23 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit23_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit23 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit24_Callback(hObject, eventdata, handles)
 hObject    handle to edit24 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit24_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit24 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit25_Callback(hObject, eventdata, handles)
 hObject    handle to edit25 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit25_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit25 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit26_Callback(hObject, eventdata, handles)
 hObject    handle to edit26 (see GCBO)
```

```
 --- Executes during object creation, after setting all properties.
unction edit26_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit26 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit27_Callback(hObject, eventdata, handles)
 hObject    handle to edit27 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit27_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit27 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit28_Callback(hObject, eventdata, handles)
 hObject    handle to edit28 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit28_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit28 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit29_Callback(hObject, eventdata, handles)
 hObject    handle to edit29 (see GCBO)

unction edit29_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit29 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit30_Callback(hObject, eventdata, handles)
 hObject    handle to edit30 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit30_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit30 (see GCBO)
```

```
f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit31_Callback(hObject, eventdata, handles)
 hObject    handle to edit31 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit31_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit31 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

unction edit32_Callback(hObject, eventdata, handles)
 hObject    handle to edit32 (see GCBO)

 --- Executes during object creation, after setting all properties.
unction edit32_CreateFcn(hObject, eventdata, handles)
 hObject    handle to edit32 (see GCBO)

f ispc
   set(hObject,'BackgroundColor','white');
lse
   set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
nd

 --- Executes on button press in pushbutton2.
unction pushbutton2_Callback(hObject, eventdata, handles)
 hObject    handle to pushbutton2 (see GCBO)

 --- Creates and returns a handle to the GUI figure.
unction h1 = SIAP_LayoutFcn(policy)
 policy - create a new figure or use a singleton. 'new' or 'reuse'.

ersistent hsingleton;
f strcmpi(policy, 'reuse') & ishandle(hsingleton)
   h1 = hsingleton;
   return;
nd

ppdata = [];
ppdata.GUIDEOptions = struct(...
   'active_h', [], ...
   'taginfo', struct(...
   'figure', 2, ...
   'edit', 33, ...
   'pushbutton', 3, ...
   'uipanel', 9, ...
   'axes', 3, ...
   'text', 34), ...
```

```
    'override', 0, ...
    'release', 13, ...
    'resize', 'none', ...
    'accessibility', 'callback', ...
    'mfile', 1, ...
    'callbacks', 1, ...
    'singleton', 1, ...
    'syscolorfig', 1, ...
    'blocking', 0, ...
    'lastSavedFile', 'C:\MATLAB7\work\SIAP.m');
ppdata.lastValidTag = 'figure1';
ppdata.GUIDELayoutEditor = [];

1 = figure(...
Units','characters',...
Color',[0.925490196078431 0.913725490196078 0.847058823529412],...
Colormap',[0 0 0.5625;0 0 0.625;0 0 0.6875;0 0 0.75;0 0 0.8125;0 0 0.875;0 0 ↙
;0 0 1;0 0.0625 1;0 0.125 1;0 0.1875 1;0 0.25 1;0 0.3125 1;0 0.375 1;0 0.4375 1;0 ↙
0 0.5625 1;0 0.625 1;0 0.6875 1;0 0.75 1;0 0.8125 1;0 0.875 1;0 0.9375 1;0 1 1;↙
 1 1;0.125 1 0.9375;0.1875 1 0.875;0.25 1 0.8125;0.3125 1 0.75;0.375 1 0.6875;↙
 1 0.625;0.5 1 0.5625;0.5625 1 0.5;0.625 1 0.4375;0.6875 1 0.375;0.75 1 0.3125;↙
 1 0.25;0.875 1 0.1875;0.9375 1 0.125;1 1 0.0625;1 1 0;1 0.9375 0;1 0.875 0;1↙
 0;1 0.75 0;1 0.6875 0;1 0.625 0;1 0.5625 0;1 0.5 0;1 0.4375 0;1 0.375 0;1 0.3125↙
25 0;1 0.1875 0;1 0.125 0;1 0.0625 0;1 0 0;0.9375 0 0;0.875 0 0;0.8125 0 0;0.75 0↙
75 0 0;0.625 0 0;0.5625 0 0],...
IntegerHandle','off',...
InvertHardcopy',get(0,'defaultfigureInvertHardcopy'),...
MenuBar','none',...
Name','newprojek2',...
NumberTitle','off',...
PaperPosition',get(0,'defaultfigurePaperPosition'),...
Position',[103.8 12.4615384615385 190 49],...
Renderer',get(0,'defaultfigureRenderer'),...
RendererMode','manual',...
Resize','off',...
HandleVisibility','callback',...
Tag','figure1',...
UserData',[],...
Behavior',get(0,'defaultfigureBehavior'),...
Visible','on',...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'uipanel1';

2 = uipanel(...
Parent',h1,...
Units','characters',...
BorderType','beveledout',...
BackgroundColor',[0.752941176470588 0.752941176470588 0.752941176470588],...
Position',[4.6 25.8461538461538 50.6 16.6923076923077],...
Tag','uipanel1',...
UserData',[],...
Behavior',get(0,'defaultuipanelBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );
```

```
ppdata = [];
ppdata.lastValidTag = 'R1x';

3 = uicontrol(...
Parent',h2,...
Units','characters',...
BackgroundColor',[1 1 1],...
Callback','SIAP(''R1x_Callback'',gcbo,[],guidata(gcbo))',...
CData',[],...
Position',[21.2 8.84615384615385 10.6 3.07692307692308],...
String','2',...
Style','edit',...
CreateFcn', {@local_CreateFcn, 'SIAP(''R1x_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
a} ,...
Tag','R1x',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'));

ppdata = [];
ppdata.lastValidTag = 'R1y';

4 = uicontrol(...
Parent',h2,...
Units','characters',...
BackgroundColor',[1 1 1],...
Callback','SIAP(''R1y_Callback'',gcbo,[],guidata(gcbo))',...
CData',[],...
Position',[35 8.84615384615385 10.2 3.07692307692308],...
String','2',...
Style','edit',...
CreateFcn', {@local_CreateFcn, 'SIAP(''R1y_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
a} ,...
Tag','R1y',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'));

ppdata = [];
ppdata.lastValidTag = 'R2x';

5 = uicontrol(...
Parent',h2,...
Units','characters',...
BackgroundColor',[1 1 1],...
Callback','SIAP(''R2x_Callback'',gcbo,[],guidata(gcbo))',...
CData',[],...
Position',[21.4 5.07692307692308 10.2 3.07692307692308],...
String','1',...
Style','edit',...
CreateFcn', {@local_CreateFcn, 'SIAP(''R2x_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
a} ,...
Tag','R2x',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'));

ppdata = [];
ppdata.lastValidTag = 'R2y';
```

```
6 = uicontrol(...
Parent',h2,...
Units','characters',...
BackgroundColor',[1 1 1],...
Callback','SIAP(''R2y_Callback'',gcbo,[],guidata(gcbo))',...
CData',[],...
Position',[35.2 5 10.2 3.07692307692308],...
String','60',...
Style','edit',...
CreateFcn', {@local_CreateFcn, 'SIAP(''R2y_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
a} ,...
Tag','R2y',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'));

ppdata = [];
ppdata.lastValidTag = 'R3x';

7 = uicontrol(...
Parent',h2,...
Units','characters',...
BackgroundColor',[1 1 1],...
Callback','SIAP(''R3x_Callback'',gcbo,[],guidata(gcbo))',...
CData',[],...
Position',[21.6 1.23076923076923 10.2 3.07692307692308],...
String','15',...
Style','edit',...
CreateFcn', {@local_CreateFcn, 'SIAP(''R3x_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
a} ,...
Tag','R3x',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'));

ppdata = [];
ppdata.lastValidTag = 'R3y';

8 = uicontrol(...
Parent',h2,...
Units','characters',...
BackgroundColor',[1 1 1],...
Callback','SIAP(''R3y_Callback'',gcbo,[],guidata(gcbo))',...
CData',[],...
Position',[35.4 1.23076923076923 10.2 3.07692307692308],...
String','50',...
Style','edit',...
CreateFcn', {@local_CreateFcn, 'SIAP(''R3y_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
a} ,...
Tag','R3y',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'));

ppdata = [];
ppdata.lastValidTag = 'text1';

9 = uicontrol(...
```

```
Parent',h2,...
Units','characters',...
BackgroundColor',[0 0 0],...
CData',[],...
FontName','T',...
FontSize',16,...
FontWeight','bold',...
ForegroundColor',[1 1 1],...
Position',[0 14.3076923076923 50.6 2.38461538461538],...
String','COORDINATE',...
Style','text',...
Tag','text1',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'text2';

10 = uicontrol(...
Parent',h2,...
Units','characters',...
CData',[],...
FontSize',10,...
Position',[1.2 10.5384615384615 17.8 1.38461538461538],...
String','LAPTOP 1',...
Style','text',...
Tag','text2',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'text6';

11 = uicontrol(...
Parent',h2,...
Units','characters',...
CData',[],...
FontSize',10,...
Position',[1 6.53846153846154 17.8 1.38461538461538],...
String','LAPTOP 2',...
Style','text',...
Tag','text6',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'text7';

12 = uicontrol(...
Parent',h2,...
Units','characters',...
CData',[],...
FontSize',10,...
```

```
Position',[1 2.92307692307692 17.8 1.38461538461538],...
String','LAPTOP 3',...
Style','text',...
Tag','text7',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'text8';

13 = uicontrol(...
Parent',h2,...
Units','characters',...
CData',[],...
Position',[21.4 12.4615384615385 10.4 1.38461538461538],...
String','X-Axis',...
Style','text',...
Tag','text8',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'text9';

14 = uicontrol(...
Parent',h2,...
Units','characters',...
CData',[],...
Position',[35 12.4615384615385 10.4 1.38461538461538],...
String','Y-Axis',...
Style','text',...
Tag','text9',...
UserData',[],...
Behavior',get(0,'defaultuicontrolBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'axes1';

15 = axes(...
Parent',h1,...
Units','characters',...
Position',[65.4 4.38461538461539 117.6 43.3846153846154],...
ALim',get(0,'defaultaxesALim'),...
ALimMode','manual',...
CameraPosition',[0.5 0.5 9.16025403784439],...
CameraPositionMode','manual',...
CameraTarget',[0.5 0.5 0.5],...
CameraTargetMode','manual',...
CameraUpVector',[0 1 0],...
CameraUpVectorMode','manual',...
CameraViewAngle',6.60861036031192,...
CameraViewAngleMode','manual',...
CLim',get(0,'defaultaxesCLim'),...
```

```
CLimMode','manual',...
Color',get(0,'defaultaxesColor'),...
ColorOrder',get(0,'defaultaxesColorOrder'),...
DataAspectRatio',get(0,'defaultaxesDataAspectRatio'),...
DataAspectRatioMode','manual',...
LooseInset',[24.7 5.39 18.05 3.675],...
PlotBoxAspectRatio',get(0,'defaultaxesPlotBoxAspectRatio'),...
PlotBoxAspectRatioMode','manual',...
TickDir',get(0,'defaultaxesTickDir'),...
TickDirMode','manual',...
XColor',get(0,'defaultaxesXColor'),...
XLim',get(0,'defaultaxesXLim'),...
XLimMode','manual',...
XTick',[0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1],...
XTickLabel',{  '0  '; '0.1'; '0.2'; '0.3'; '0.4'; '0.5'; '0.6'; '0.7'; '0.8';↙
 '1  ' },...
XTickLabelMode','manual',...
XTickMode','manual',...
YColor',get(0,'defaultaxesYColor'),...
YLim',get(0,'defaultaxesYLim'),...
YLimMode','manual',...
YTick',[0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1],...
YTickLabel',{  '0  '; '0.1'; '0.2'; '0.3'; '0.4'; '0.5'; '0.6'; '0.7'; '0.8';↙
 '1  ' },...
YTickLabelMode','manual',...
YTickMode','manual',...
ZColor',get(0,'defaultaxesZColor'),...
ZLim',get(0,'defaultaxesZLim'),...
ZLimMode','manual',...
ZTick',[0 0.5 1],...
ZTickLabel','',...
ZTickLabelMode','manual',...
ZTickMode','manual',...
Tag','axes1',...
UserData',[],...
Behavior',get(0,'defaultaxesBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

16 = get(h15,'title');

et(h16,...
Parent',h15,...
Color',[0 0 0],...
HorizontalAlignment','center',...
Position',[0.499113475177305 1.01329787234043 1.00005459937205],...
VerticalAlignment','bottom',...
HandleVisibility','off',...
Behavior',struct());

17 = get(h15,'xlabel');

et(h17,...
Parent',h15,...
Color',[0 0 0],...
HorizontalAlignment','center',...
Position',[0.499113475177305 -0.0416666666666667 1.00005459937205],...
```

```
VerticalAlignment','cap',...
HandleVisibility','off',...
Behavior',struct());

18 = get(h15,'ylabel');

et(h18,...
Parent',h15,...
Color',[0 0 0],...
HorizontalAlignment','center',...
Position',[-0.0505319148936169 0.499113475177305 1.00005459937205],...
Rotation',90,...
VerticalAlignment','bottom',...
HandleVisibility','off',...
Behavior',struct());

19 = get(h15,'zlabel');

et(h19,...
Parent',h15,...
Color',[0 0 0],...
HorizontalAlignment','right',...
Position',[-0.601950354609929 1.02570921985816 1.00005459937205],...
HandleVisibility','off',...
Behavior',struct(),...
Visible','off');

ppdata = [];
ppdata.lastValidTag = 'uipanel2';

20 = uipanel(...
Parent',h1,...
Units','characters',...
BorderType','beveledout',...
BackgroundColor',[0.752941176470588 0.752941176470588 0.752941176470588],...
Position',[4.6 14.1538461538462 50.8 11.0769230769231],...
Tag','uipanel2',...
UserData',[],...
Behavior',get(0,'defaultuipanelBehavior'),...
CreateFcn', {@local_CreateFcn, '', appdata} );

ppdata = [];
ppdata.lastValidTag = 'D1';

21 = uicontrol(...
Parent',h20,...
Units','characters',...
BackgroundColor',[1 1 1],...
Callback','SIAP(''D1_Callback'',gcbo,[],guidata(gcbo))',...
CData',[],...
Position',[2 1 12.6 4],...
String','-59',...
Style','edit',...
CreateFcn', {@local_CreateFcn, 'SIAP(''D1_CreateFcn'',gcbo,[],guidata(gcbo))',↙
a} ,...
Tag','D1',...
```

```
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'));

appdata = [];
appdata.lastValidTag = 'D2';

h22 = uicontrol(...
'Parent',h20,...
'Units','characters',...
'BackgroundColor',[1 1 1],...
'Callback','SIAP(''D2_Callback'',gcbo,[],guidata(gcbo))',...
'CData',[],...
'Position',[18.4 1 12.6 4],...
'String','-63',...
'Style','edit',...
'CreateFcn', {@local_CreateFcn, 'SIAP(''D2_CreateFcn'',gcbo,[],guidata(gcbo))',↙
ta} ,...
'Tag','D2',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'));

appdata = [];
appdata.lastValidTag = 'D3';

h23 = uicontrol(...
'Parent',h20,...
'Units','characters',...
'BackgroundColor',[1 1 1],...
'Callback','SIAP(''D3_Callback'',gcbo,[],guidata(gcbo))',...
'CData',[],...
'Position',[34.2 1 12.8 4],...
'String','-60.5',...
'Style','edit',...
'CreateFcn', {@local_CreateFcn, 'SIAP(''D3_CreateFcn'',gcbo,[],guidata(gcbo))',↙
ta} ,...
'Tag','D3',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'));

appdata = [];
appdata.lastValidTag = 'text10';

h24 = uicontrol(...
'Parent',h20,...
'Units','characters',...
'BackgroundColor',[0 0 0],...
'CData',[],...
'FontSize',16,...
'FontWeight','bold',...
'ForegroundColor',[1 1 1],...
'Position',[0.2 7.92307692307692 50.4 2.92307692307692],...
'String','Signal Strength from:',...
'Style','text',...
'Tag','text10',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
```

```
'CreateFcn', {@local_CreateFcn, '', appdata} );

appdata = [];
appdata.lastValidTag = 'text11';

h25 = uicontrol(...
'Parent',h20,...
'Units','characters',...
'CData',[],...
'FontSize',10,...
'Position',[1.6 5.76923076923077 13 1.38461538461538],...
'String','LAPTOP 1',...
'Style','text',...
'Tag','text11',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );

appdata = [];
appdata.lastValidTag = 'text12';

h26 = uicontrol(...
'Parent',h20,...
'Units','characters',...
'CData',[],...
'FontSize',10,...
'Position',[18 5.76923076923077 13 1.38461538461538],...
'String','LAPTOP 2',...
'Style','text',...
'Tag','text12',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );

appdata = [];
appdata.lastValidTag = 'text13';

h27 = uicontrol(...
'Parent',h20,...
'Units','characters',...
'CData',[],...
'FontSize',10,...
'Position',[34.4 5.76923076923077 13 1.38461538461538],...
'String','LAPTOP 3',...
'Style','text',...
'Tag','text13',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );

appdata = [];
appdata.lastValidTag = 'uipanel4';

h28 = uipanel(...
'Parent',h1,...
'Units','characters',...
```

```
'BorderType','beveledout',...
'BackgroundColor',[0.752941176470588 0.752941176470588 0.752941176470588],...
'Position',[5 4.38461538461539 50.6 9.07692307692308],...
'Tag','uipanel4',...
'UserData',[],...
'Behavior',get(0,'defaultuipanelBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );

appdata = [];
appdata.lastValidTag = 'text18';

h29 = uicontrol(...
'Parent',h28,...
'Units','characters',...
'BackgroundColor',[0 0 0],...
'CData',[],...
'FontSize',14,...
'FontWeight','bold',...
'ForegroundColor',[1 1 1],...
'Position',[0 6.92307692307692 50.4 2.07692307692308],...
'String','Target Laptop Coordinate',...
'Style','text',...
'Tag','text18',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );

appdata = [];
appdata.lastValidTag = 'text19';

h30 = uicontrol(...
'Parent',h28,...
'Units','characters',...
'CData',[],...
'Position',[7 5.07692307692307 10.4 1.38461538461538],...
'String','X-Axis',...
'Style','text',...
'Tag','text19',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );

appdata = [];
appdata.lastValidTag = 'text20';

h31 = uicontrol(...
'Parent',h28,...
'Units','characters',...
'CData',[],...
'Position',[29.6 5.07692307692307 10.4 1.38461538461538],...
'String','Y-Axis',...
'Style','text',...
'Tag','text20',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );
```

```
appdata = [];
appdata.lastValidTag = 'Tx';

h32 = uicontrol(...
'Parent',h28,...
'Units','characters',...
'BackgroundColor',[1 1 1],...
'Callback','SIAP(''Tx_Callback'',gcbo,[],guidata(gcbo))',...
'CData',[],...
'Position',[4.6 0.615384615384615 15 3.92307692307692],...
'Style','edit',...
'CreateFcn', {@local_CreateFcn, 'SIAP(''Tx_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
ta} ,...
'Tag','Tx',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'));

appdata = [];
appdata.lastValidTag = 'Ty';

h33 = uicontrol(...
'Parent',h28,...
'Units','characters',...
'BackgroundColor',[1 1 1],...
'Callback','SIAP(''Ty_Callback'',gcbo,[],guidata(gcbo))',...
'CData',[],...
'Position',[27.4 0.692307692307692 15.2 3.92307692307692],...
'Style','edit',...
'CreateFcn', {@local_CreateFcn, 'SIAP(''Ty_CreateFcn'',gcbo,[],guidata(gcbo))', ↙
ta} ,...
'Tag','Ty',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'));

appdata = [];
appdata.lastValidTag = 'Calculate';

h34 = uicontrol(...
'Parent',h1,...
'Units','characters',...
'Callback','SIAP(''Calculate_Callback'',gcbo,[],guidata(gcbo))',...
'CData',[],...
'FontSize',16,...
'FontWeight','bold',...
'Position',[4.8 43.9230769230769 50.2 3.84615384615385],...
'String','Calculate',...
'Tag','Calculate',...
'UserData',[],...
'Behavior',get(0,'defaultuicontrolBehavior'),...
'CreateFcn', {@local_CreateFcn, '', appdata} );


hsingleton = h1;
```

```
% --- Set application data first then calling the CreateFcn.
function local_CreateFcn(hObject, eventdata, createfcn, appdata)

if ~isempty(appdata)
   names = fieldnames(appdata);
   for i=1:length(names)
       name = char(names(i));
       setappdata(hObject, name, getfield(appdata,name));
   end
end

if ~isempty(createfcn)
   eval(createfcn);
end


% --- Handles default GUIDE GUI creation and callback dispatch
function varargout = gui_mainfcn(gui_State, varargin)


%   GUI_MAINFCN provides these command line APIs for dealing with GUIs
%
%      SIAP, by itself, creates a new SIAP or raises the existing
%      singleton*.
%
%      H = SIAP returns the handle to a new SIAP or the handle to
%      the existing singleton*.
%
%      SIAP('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in SIAP.M with the given input arguments.
%
%      SIAP('Property','Value',...) creates a new SIAP or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before untitled_OpeningFunction gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to untitled_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".

%   Copyright 1984-2004 The MathWorks, Inc.
%   $Revision: 1.4.6.8 $ $Date: 2004/04/15 00:06:57 $

gui_StateFields =  {'gui_Name'
                    'gui_Singleton'
                    'gui_OpeningFcn'
                    'gui_OutputFcn'
                    'gui_LayoutFcn'
                    'gui_Callback'};
gui_Mfile = '';
for i=1:length(gui_StateFields)
    if ~isfield(gui_State, gui_StateFields{i})
        error('Could not find field %s in the gui_State struct in GUI M-file %s', ↙
tateFields{i}, gui_Mfile);
    elseif isequal(gui_StateFields{i}, 'gui_Name')
        gui_Mfile = [gui_State.(gui_StateFields{i}), '.m'];
```

```
    end
end

numargin = length(varargin);

if numargin == 0
    % SIAP
    % create the GUI
    gui_Create = 1;
elseif isequal(ishandle(varargin{1}), 1) && ispc && iscom(varargin{1}) && isequal↙
rgin{1},gcbo)
    % SIAP(ACTIVEX,...)
    vin{1} = gui_State.gui_Name;
    vin{2} = [get(varargin{1}.Peer, 'Tag'), '_', varargin{end}];
    vin{3} = varargin{1};
    vin{4} = varargin{end-1};
    vin{5} = guidata(varargin{1}.Peer);
    feval(vin{:});
    return;
elseif ischar(varargin{1}) && numargin>1 && isequal(ishandle(varargin{2}), 1)
    % SIAP('CALLBACK',hObject,eventData,handles,...)
    gui_Create = 0;
else
    % SIAP(...)
    % create the GUI and hand varargin to the openingfcn
    gui_Create = 1;
end

if gui_Create == 0
    varargin{1} = gui_State.gui_Callback;
    if nargout
        [varargout{1:nargout}] = feval(varargin{:});
    else
        feval(varargin{:});
    end
else
    if gui_State.gui_Singleton
        gui_SingletonOpt = 'reuse';
    else
        gui_SingletonOpt = 'new';
    end

    % Open fig file with stored settings.  Note: This executes all component
    % specific CreateFunctions with an empty HANDLES structure.

    % Do feval on layout code in m-file if it exists
    if ~isempty(gui_State.gui_LayoutFcn)
        gui_hFigure = feval(gui_State.gui_LayoutFcn, gui_SingletonOpt);
        % openfig (called by local_openfig below) does this for guis without
        % the LayoutFcn. Be sure to do it here so guis show up on screen.
        movegui(gui_hFigure,'onscreen')
    else
        gui_hFigure = local_openfig(gui_State.gui_Name, gui_SingletonOpt);
        % If the figure has InGUIInitialization it was not completely created
        % on the last pass.  Delete this handle and try again.
        if isappdata(gui_hFigure, 'InGUIInitialization')
```

```
        delete(gui_hFigure);
        gui_hFigure = local_openfig(gui_State.gui_Name, gui_SingletonOpt);
    end
end

% Set flag to indicate starting GUI initialization
setappdata(gui_hFigure,'InGUIInitialization',1);

% Fetch GUIDE Application options
gui_Options = getappdata(gui_hFigure,'GUIDEOptions');

if ~isappdata(gui_hFigure,'GUIOnScreen')
    % Adjust background color
    if gui_Options.syscolorfig
        set(gui_hFigure,'Color', get(0,'DefaultUicontrolBackgroundColor'));
    end

    % Generate HANDLES structure and store with GUIDATA
    guidata(gui_hFigure, guihandles(gui_hFigure));
end

% If user specified 'Visible','off' in p/v pairs, don't make the figure
% visible.
gui_MakeVisible = 1;
for ind=1:2:length(varargin)
    if length(varargin) == ind
        break;
    end
    len1 = min(length('visible'),length(varargin{ind}));
    len2 = min(length('off'),length(varargin{ind+1}));
    if ischar(varargin{ind}) && ischar(varargin{ind+1}) && ...
            strncmpi(varargin{ind},'visible',len1) && len2 > 1
        if strncmpi(varargin{ind+1},'off',len2)
            gui_MakeVisible = 0;
        elseif strncmpi(varargin{ind+1},'on',len2)
            gui_MakeVisible = 1;
        end
    end
end

% Check for figure param value pairs
for index=1:2:length(varargin)
    if length(varargin) == index
        break;
    end
    try set(gui_hFigure, varargin{index}, varargin{index+1}), catch break, end
end

% If handle visibility is set to 'callback', turn it on until finished
% with OpeningFcn
gui_HandleVisibility = get(gui_hFigure,'HandleVisibility');
if strcmp(gui_HandleVisibility, 'callback')
    set(gui_hFigure,'HandleVisibility', 'on');
end

feval(gui_State.gui_OpeningFcn, gui_hFigure, [], guidata(gui_hFigure), ↙
```

```
gin{:});

    if ishandle(gui_hFigure)
        % Update handle visibility
        set(gui_hFigure,'HandleVisibility', gui_HandleVisibility);

        % Make figure visible
        if gui_MakeVisible
            set(gui_hFigure, 'Visible', 'on')
            if gui_Options.singleton
                setappdata(gui_hFigure,'GUIOnScreen', 1);
            end
        end

        % Done with GUI initialization
        rmappdata(gui_hFigure,'InGUIInitialization');
    end

    % If handle visibility is set to 'callback', turn it on until finished with
    % OutputFcn
    if ishandle(gui_hFigure)
        gui_HandleVisibility = get(gui_hFigure,'HandleVisibility');
        if strcmp(gui_HandleVisibility, 'callback')
            set(gui_hFigure,'HandleVisibility', 'on');
        end
        gui_Handles = guidata(gui_hFigure);
    else
        gui_Handles = [];
    end

    if nargout
        [varargout{1:nargout}] = feval(gui_State.gui_OutputFcn, gui_hFigure, [],↙
andles);
    else
        feval(gui_State.gui_OutputFcn, gui_hFigure, [], gui_Handles);
    end

    if ishandle(gui_hFigure)
        set(gui_hFigure,'HandleVisibility', gui_HandleVisibility);
    end
end

function gui_hFigure = local_openfig(name, singleton)

try
    gui_hFigure = openfig(name, singleton, 'auto');
catch
    % OPENFIG did not accept 3rd input argument until R13,
    % toggle default figure visible to prevent the figure
    % from showing up too soon.
    gui_OldDefaultVisible = get(0,'defaultFigureVisible');
    set(0,'defaultFigureVisible','off');
    gui_hFigure = openfig(name, singleton);
    set(0,'defaultFigureVisible',gui_OldDefaultVisible);
end
```

Appendix II

WPC54G Product Data

INKSYS®

ivision of Cisco Systems, Inc.

np-start your wireless network! This Wireless-G Network for Notebooks provides you with a Wireless-G oadband Router (the heart of your network), and a reless-G Notebook Adapter to get your PC nnected, quick and easy.

e Wireless Router lets you securely share your DSL or ible Modem Internet connection with all the mputers in the house, both wireless and wired. You in connect four computers directly to the Router by ernet cables, and daisy-chain out to more hubs and itches to create as big a network as you need.

Or, use the Router's built-in Wireless Access Point to connect your PC to the network without stringing any wires. That's where the Wireless-G Notebook Adapter comes in. Just plug it into your notebook through a PC Card slot. Wireless networking doesn't get any easier.

Of course, you can connect more computers to the Router and your high-speed Internet link -- just provide a wired or wireless network adapter for each one. And once your computers are connected to the Router, they can communicate with each other too, sharing resources like printers and all kinds of files -- music, digital pictures, and documents.

To protect your data and privacy, all wireless communications can be protected by industrial-strength 256-bit WPA encryption, while the Router helps keep Internet intruders out of your computers. It's all easier than you think -- the included Setup Wizards walk you through configuring the Router and Adapter, step by step.

With the Wireless-G Network Kit for Notebooks, you're ready to start sharing printers, files, and your Internet connection, the easy way.

- mplete solution cludes an Internet-sharing Wireless-G Router and Nireless-G Adapter for ur notebook
- Router shares a high-speed Cable or DSL Internet connection and other resources with both wired and wireless PCs
- Wireless communications at up to 54Mbps data rate -- secured by 128-bit encryption
- Included Setup Wizards take you through configuration, step by step

No. **WKPC54G**

CISCO SYSTEMS

Network Kit for Notebooks

es

nplies with 802.11g
802.11b (2.4GHz)
ıdards

ırpassed Wireless
urity with Wi-Fi Pro-
ed Access™ (WPA)

AN Ports Support
)-Crossover (MDI/MDI-
No Need for Cross-
' Cables

oit CardBus Interface

anced security fea-
s including WEP, AES,
and 802.1x

npatible with Win-
's 98SE, Millennium,
) and XP

Specifications for the Router

Model Number	WRK54G
Standards	IEEE 802.3, IEEE 802.3u, IEEE 802.11g, IEEE 802.11b
Channels	11 Channels (US, Canada) 13 Channels (Europe) 14 Channels (Japan)
Ports/Buttons	Internet: One 10/100 RJ-45 Port LAN: Four 10/100 RJ-45 Switched Ports One Power Port One Reset Button
Cabling Type	UTP CAT 5
LEDs	Power, DMZ, WLAN, Ethernet (1, 2, 3, 4), Internet
RF Power Output	16.5 dBm
UPnP able/cert	able
Security Features	NAT Firewall
Wireless Security	Wi-Fi Protected Access™ (WPA), WEP, Wireless MAC Filtering

Environmental for the Router

Dimensions W x H x D	7.32" x 1.89" x 6.06" 186 mm x 48 mm x 154 mm
Unit Weight	12 oz. (0.34 kg)
Power	External, 12V DC, 1.0A
Certifications	FCC, IC-03, CE, Wi-Fi (802.11b, 802.11g), WPA
Operating Temperature	32°F to 104°F (0°C to 40°C)
Storage Temperature	-4°F to 158°F (-20°C to 70°C)
Operating Humidity	10% to 85% Non-Condensing
Storage Humidity	5% to 90% Non-Condensing
Warranty	3-yearLimited

Specifications for the Notebook Adapter

Model Number	WPC54G v4
Standards	802.11g, 802.11b
LEDs	Power, Link
Transmit Power	15 dBm
Security features	WEP, AES, TKIP, 802.1x
WEP key bits	64-bit and 128-bit

Environmental for the Notebook Adapter

Dimensions W x H x D	4.53" x 2.13" x 0.30" 115 mm x 54 mm x 7.5 mm
Unit Weight	1.7 oz. (0.05 kg)
Power	3.3V Bus powered
Certifications	FCC, IC-03, CE, Wi-Fi (802.11b), WHQL:2000 and XP
Operating Temperature	32°F to 131°F (0°C to 55°C)
Storage Temperature	-13°F to 158°F (-25°C to 70°C)
Operating Humidity	5% to 95% Non-Condensing
Storage Humidity	5% to 95% Non-Condensing
Warranty	Three Years

n of Cisco Systems, Inc.
iller Avenue
A 92612 USA

sales@linksys.com
support@linksys.com

http://www.linksys.com

roducts are available in more than 50 supported by 12 Linksys Regional Offices it the world. For a complete list of local ales and Technical Support contacts, visit wide Web Site at www.linksys.com.

Minimum Requirements

- 200 MHz or Faster Processor
- 64 MB of RAM
- Internet Explorer 6.0 or Netscape Navigator 6 or Higher for Web-based configuration
- CD-ROM Drive
- Windows 98SE, Me, 2000, or XP
- Network Adapter

Package Contents

- Wireless-G Broadband Router
- Wireless-G Notebook Adapter
- Two Setup CD-ROM with Symantec Internet Security
- Two User Guide on CD-ROM
- Power Adapter
- Ethernet Network Cable
- Quick Installation Guide

KL

Model No. WKPC54G