AREA OF OPERATION FOR A RADIO-FREQUENCY IDENTIFICATION (RFID) TAG IN THE FAR-FIELD

by

Charles Edward Greene

Bachelor of Science in Electrical Engineering, University of Pittsburgh, 2001

Master of Science in Electrical Engineering, University of Pittsburgh, 2002

Submitted to the Graduate Faculty of

the School of Engineering in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering

University of Pittsburgh

2006

UNIVERSITY OF PITTSBURGH

SCHOOL OF ENGINEERING

This dissertation was presented

by

Charles Edward Greene

It was defended on

February 13, 2006

and approved by

Michael R. Lovell, Associate Professor, Mechanical Engineering

John Boston, Professor, Electrical and Computer Engineering

Ronald Hoelzeman, Associate Professor, Electrical and Computer Engineering

James T. Cain, Professor, Electrical and Computer Engineering

Marlin H. Mickle, Nickolas A. DeCecco Professor, Electrical and Computer Engineering
Dissertation Director

AREA OF OPERATION FOR A RADIO-FREQUENCY IDENTIFICATION (RFID) TAG IN THE FAR-FIELD

Charles Edward Greene, Ph.D.

University of Pittsburgh, 2006

In Radio Frequency Identification (RFID) applications, it is beneficial to know where in a three-dimensional space an RFID tag will operate with respect to the interrogating transmitter. It becomes a very complex problem containing numerous variables including transmitted power, antenna gains, orientation, etc. One well-known equation used to approximate the power that a tag can receive from an interrogating transmitter is the Friis Equation. However, the commonly used form of the Friis Equation contains assumptions that limit the validity to a single point, orientation, and polarization in space, which is usually the most favorable. These simplifications eliminate the reflection coefficients and polarization terms, and the gains lose their angular dependences. This dissertation will provide a mathematical model that describes the operation of a tag in the far-field from a more realistic perspective in a three-dimensional space. The complete form of the Friis equation will be used as the basic formulation to model the amount of power a tag can receive for any orientation at a given point in space. The dissertation will also include mathematical analyses of how the location of the data base station affects the performance of the system by applying the physics embodied in the complete Friis equation to the return transmission link from the tag to the data base station. The complete mathematical expression will be used to evaluate the performance of an RFID tag by depicting the three-dimensional area of operation. The functioning volume will be solved using the developed scaling factor method and will give an accurate portrayal of where a tag can be successfully read as a specified percentage of reads when all orientations and polarizations are examined.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 INTRODUCTION

1.1 STATEMENT OF THE PROBLEM

The operation of an RFID tag is generally described using the Radar Equation shown in (1.1) [1].

$$P_R = P_T \frac{\sigma G_T G_R}{4\pi} \left[\frac{\lambda}{4\pi R_1 R_2} \right]^2 \tag{1.1}$$

where

P_R – received data power
P_T – interrogator transmit power
G_R – maximum receiver gain
G_T – maximum transmitter gain
σ – radar cross section (RCS) of the tag
R_1 – distance between the transmitter and tag
R_2 – distance between the tag and receiver
λ – wavelength

It is, however, generally more advantageous to apply the Friis Equation shown in (1.2) to the RF link between the interrogating transmitter and the tag in order to determine the maximum read range of the system [1]. Then, the Radar Equation can be used to determine the minimum sensitivity required by the data base station in order to receive the data when the tag is located at its maximum read range.

$$P_R = P_T \frac{G_T G_R \lambda^2}{(4\pi r)^2} \tag{1.2}$$

An example of this analysis was given in a presentation by Matrics (the current RFID arm of Symbol Corporation) [2]. In the example, the maximum read range was calculated using the Friis Equation to be 9.45 meters using an Effective Isotropic Radiated Power (EIRP is equal to $P_T G_T$) of the FCC limit of 4 Watts (W) [3], a tag gain of a dipole or 1.64, a frequency of 915MHz, and a required tag power of 50μW. The Radar Equation was then used to calculate the minimum sensitivity required by the data base station using a radar cross section (RCS) of 10cm^2 and a receiver gain of 6dB. The resulting receiver sensitivity was presented as –69.6dBm. These calculations can be seen in Appendix A.

The problem with this analysis is that the results are only valid if the tag is polarization matched (see Section 3.1.1) with the interrogating transmitter and data base station and the antennas are aligned for maximum radiation and reception, i.e. the tag is in its optimum location and orientation. This realization means that the results calculated in the example are only valid for a single point, orientation, and polarization in space. This, however, is not acceptable in certain situations due to the fact that the tag has a very small probability of being located in the optimum position. One practical example of this is an RFID system in a retail setting where a shopping cart may be full of numerous items all with essentially random orientations and polarizations. It is the objective of the current research to examine all possible positions and orientations by limiting the number of assumptions made.

1.2 RESEARCH OVERVIEW

The current research provides a mathematical analysis of an RFID system, which minimizes the number of assumptions used to accurately describe the operation of an RFID tag in a three-dimensional space. Specifically, the simplified Friis Equation shown in (1.2) can be expanded to include terms that do not limit the results to the optimum position. The complete Friis Equation to be used herein is shown in (1.3) [1].

$$P_R = P_T \frac{G_T(\theta_T,\phi_T) G_R(\theta_R,\phi_R)\lambda^2}{(4\pi r)^2}(1-|\Gamma_T|^2)(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_T \cdot \hat{\mathbf{p}}_R|^2 \qquad (1.3)$$

where

P_R – received power
P_T – transmit power
$G_R(\theta_R,\phi_R)$ – angular dependent receiver gain
$G_T(\theta_T,\phi_T)$ – angular dependent transmitter gain
Γ_T – transmitter reflection coefficent
Γ_R – receiver reflection coefficent
$\hat{\mathbf{p}}_T$ – transmitter polarization vector
$\hat{\mathbf{p}}_R$ – receiver polarization vector
r – distance between the transmitter and receiver
λ – wavelength

Likewise, the Radar Equation can also be expanded to include non-limiting terms, however, the equation shown in (1.4) is generally used to describe scattering from a target rather than an antenna. Therefore, the equation will be modified in the subsequent sections to reflect this condition.

$$P_R = P_T \frac{\sigma(\theta_{TAG},\phi_{TAG})G_T(\theta_T,\phi_T)G_R(\theta_R,\phi_R)}{4\pi}\left[\frac{\lambda}{4\pi R_1 R_2}\right]^2 (1-|\Gamma_T|^2)(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_S \bullet \hat{\mathbf{p}}_R|^2 \qquad (1.4)$$

where

P_R – received power
P_T – transmit power
$G_R(\theta_R,\phi_R)$ – angular dependent receiver gain
$G_T(\theta_T,\phi_T)$ – angular dependent transmitter gain
Γ_T – transmitter reflection coefficent
Γ_R – receiver reflection coefficent
$\hat{\mathbf{p}}_S$ – scattered wave polarization vector
$\hat{\mathbf{p}}_R$ – receiver polarization vector
$\sigma(\theta_{TAG},\phi_{TAG})$ – target angular dependent radar cross section (RCS)
R_1 – distance between the transmitter and target
R_2 – distance between the target and receiver
λ – wavelength

As can be seen in the Friis and Radar Equations, the gain terms have an angular dependence, which means a three-dimensional (3D) antenna gain pattern is needed for each antenna. For most antenna systems, the 3D gains can be obtained from measurements in an anechoic camber. For the purpose of the current research, an analytic expression will be derived for the 3D gain of two common RFID antennas, the half-wave dipole and the half-wave square patch, in order to present a thorough mathematical representation of the RFID system. The gains will be modeled in the far-field region, which is valid for the UHF RFID tags that are the focus of the current research.

As previously stated, equation (1.4) is generally used to describe the scattering from a target rather than an antenna. It therefore becomes necessary to identify a relationship between the three-dimensional gain and the three-dimensional Radar Cross Section (RCS) of an antenna. It is also necessary to include in the relationship the effects on the RCS caused by the backscatter modulation, which is used to transmit data from the tag. Backscatter modulation is described in detail in Section 2.3.2.

It is not only necessary to define a mathematical expression in order to solve the RFID system, but also, it become necessary to define a method of solving the problem to describe the area of operation in three dimensions. It is obvious that the problem can be solved for every point in space, but the amount of resulting data would be overwhelming and would present an extreme challenge in terms of processing capabilities and graphical representation. As an example, for a tag at a given point in space there are 64,800 gain values for a one-degree resolution (180° x 360°). This means the tag can be positioned in one of 64,800 possible positions with respect to the interrogating transmitter. The problem is complicated further when the effects of polarization are included. At a given point in space with a given gain value (1 of 64,800), there are 360 possible polarization positions for a one-degree resolution. This number can be reduced to 90 possible polarizations by the realization that all four quadrants will produce the same results. Therefore, at a given point in space there are 5,832,000 possible gain and polarization positions. If the problem is then solved for one cubic meter in front of the interrogating transmitter in 10cm steps, there are 1000 points with each having 5,832,000 possible gain and polarization positions. The result is 5,832,000,000 positions for one cubic meter in space. If it is assumed that the results are stored in 32-bit (4-byte) floating point numbers, the amount of memory needed would be 23.328 Giga-bytes (GB) which is not practical on most computer systems. Given this result, the current research will provide a method for solving the RFID system that is manageable for a laptop computer yet useful for evaluating the performance of the overall system.

In summary, there are two primary goals of the research as described in Section 1.3.

1.3 RESEARCH OBJECTIVES

The research reported in this dissertation provides the three-dimensional area of operation for an ARS and a backscatter RFID tag. To complete this research, two primary goals were established and are given as follows.

1. Formulate a mathematical model based on the Friis equation to describe the operation of an ARS and a backscatter RFID tag accounting for all currently known physical factors affecting the performance of the RFID system.
    a. Derive the three-dimensional antenna gains for two widely use RFID antennas, the half-wave dipole and the half-wave square patch, to demonstrate the functionality of the mathematical model.
    b. Establish a relationship between the three-dimensional gain of an antenna and the three-dimensional radar cross-section including the effects of the terminating load and polarization of the antenna.
2. Develop a method for solving the mathematical model to allow the determination of the percentage of reads possible for a given tag location in a three-dimensional space with respect to the interrogating transmitter and receiving base station when all possible orientations and polarizations are examined.

2.0 BACKGROUND

2.1 WHAT IS RFID?

Radio-Frequency Identification (RFID) is a method of communicating between an interrogating transmitter, a tag, and a receiving base station by the use of electromagnetic waves. RFID can be broken into several categories including passive and active, with passive being the focus of the current research. An active tag includes an on-board power supply used to energize the tag, which acts as a miniature radio receiver and transmitter. A typical passive system includes an interrogating transmitter that produces an electromagnetic field that is used to supply power and data to the tag. The tag captures the electromagnetic energy and converts some of the energy into usable Direct Current (DC) power. The tag generally monitors the amplitude of the incoming energy for the purpose of extracting data included by the interrogating transmitter through amplitude modulation although other modulation schemes are possible. The tag then transmits identification data back to a receiving base station, which logs the successful identification of the tag. Figure 1 shows a simple block diagram of an RFID system. Other facets of RFID such as collision resolution when multiple tags respond are outside the scope of this research.

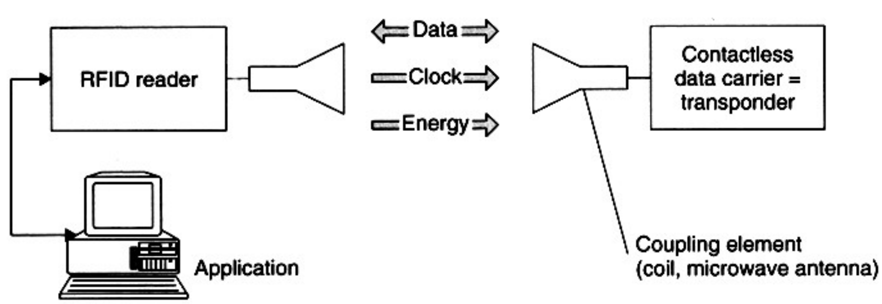

Figure 1 - Block Diagram of an RFID System [4]

2.2 HISTORY OF RFID

The origin of RFID can be traced back to a paper by Harry Stockman entitled "Communications by Means of Reflected Power" published in October 1948 [5]. In this publication, Stockman makes the statement that "Evidently, considerable research and development work has to be done before the remaining basic problems in reflected-power communication are solved, and before the field of useful applications is explored." From the 1960's to the 1980's considerable research and development was performed with significant advances being made by the Los Alamos Scientific Laboratory, Northwestern University, and the Microwave Institute Foundation in Sweden [6]. One important advance was presented by Koelle et. al. in the paper "Short-range Radio-telemetry for Electronic Identification using Modulated Backscatter" in 1975 [6]. Considerable advances in microelectronics, embedded software, and Radio-Frequency (RF) integrated circuits during the 1990's and 2000's have enable significant growth in the RFID field [7]. However, the statement made by Stockman in 1948 still applies to RFID today. There are fundamental problems that need to be addressed in order for RFID to completely replace barcodes. A few of these problems include effects caused by metal, attenuation by water or moisture inherent to the tagged product, and effects caused by the non-optimal orientation of the tag with respect to the interrogating transmitter. The latter will be explored in this dissertation with the effects made apparent through mathematics and graphical representations.

2.3 TYPES OF RFID

There are many types of RFID tags. The simplest type is classified as a 1-bit transponder due to the fact that it has only two states: 1 or 0 [4]. This type of tag is used to identify the presence or absence of a tag in applications such as anti-theft devices at the exits of stores. This type of tag can be implemented by simply placing a diode across the connections of a dipole antenna. The nonlinear nature of the diode will cause the second harmonic to be transmitted from the tag when interrogated. The existence of the second harmonic at the reader indicates the presence of the tag.

From this simple tag, more complicated tags have been developed that use different methods of transmitting data and also contain much more memory. These types of tags have expanded the uses of RFID to include many different applications and thus it becomes important to understand their operation. Three types of tags will be examined in the following sections to give a better understanding on how they obtain their power and communicate.

2.3.1 Inductive Coupling

Passive tags designed to use inductive coupling rely on a transformer effect to transfer power and data. An inductively coupled tag is a near-field device where the near-field is commonly accepted to be distances less than $\frac{\lambda}{2\pi}$. The reader and tag each contain an inductor in the form of a coil that is resonated with a parallel capacitance. The coils can be viewed as individual antennas or a loosely coupled transformer. Power is transferred to the tag by energizing the reader coil, which produces a time varying electromagnetic field. Because the wavelength of the frequency used in inductive coupling systems is typically much larger than the distance between the reader and tag (i.e. the tag is in the near-field), the electromagnetic field can be treated as a simple time varying magnetic field [4]. Part of the magnetic field or flux created by the reader is coupled to the tag's coil, which is resonant at the desired frequency. The coupled magnetic field induces a time varying voltage across the coil. The voltage is then rectified and used to provide power to the tag's electronic circuits, which are used to communicate with the reader.

Inductively coupled RFID tags communicate by modulating the load impedance on the tag's antenna. This modulation produces voltage changes at the reader's antenna (essentially the primary of a transformer) that can be deciphered into usable data. The load modulation effects on the reader can be seen by examining a transformer circuit. The load on the secondary side of the transformer can be reflected to the primary side using an impedance transformation based on the turns-ratio. Therefore, changes in the secondary side load will be reflected to the primary side, which in turn changes the voltage and current on the primary side. This technique can be used in evaluating inductively coupled tags although the impedance transfer is based on a loosely coupled transformer.

2.3.2 Backscatter

Passive backscatter RFID tags, like inductive tags, rely on the reader to supply operational power, however, the power is obtained in a different way. Backscatter tags are designed to operate in the far-field region rather than the near-field region used for inductive tags. Generally, backscatter tags operate at frequencies greater than 30MHz [4] meaning the wavelengths are smaller than those used in inductive systems. This allows the antenna to be on the order of a wavelength especially for frequencies in the UHF band such as 868MHz in Europe and 915MHz in the U.S. Designing the antenna with dimensions comparable to the wavelength allows the antenna to have a much higher efficiency than antennas in inductive systems, which leads to an increase in the power available to the tag. The increase in available power allows the tags to operate at greater distances from the reader. This fact makes UHF backscatter tags an attractive solution for a number of RFID applications requiring greater read distances.

Backscatter tags get their name from how they communicate with the reader. The implementation in the tag is similar to that of inductive coupling tags, however, the resulting effect is different because the tag is generally located in the far-field. A backscatter tag communicates by modulating the Radar Cross Section (RCS) of the antenna. As is known from radar theory, objects on the order of a wavelength of the interrogating wave will reflect or backscatter part of the incident wave. The resulting backscattered wave can be sensed to determine if an object is present. The same technique used for conventional radar can be used to sense data from an RFID tag. This is done by varying the load across the antenna terminals. The effect is to change or modulate the RCS of the antenna, which in turn varies the amount of power backscattered to the reader. The changes in backscattered energy can be sensed by the reader and interpreted as data.

2.3.3 Active Remote Sensing (ARS)

Active Remote Sensing (ARS) is a term used at the University of Pittsburgh to describe a form of RFID that separates the powering and communicating parts of the system on separate frequencies. As an example, ARS systems at the University of Pittsburgh have been designed that establish an operational power channel at 915MHz while tag communication to and from the reader is performed at 2.45GHz. The resulting tag has numerous benefits including reduced noise effects caused by the interrogator, a much simpler design, and optimized power and communication channels. The drawback to this approach is that the amount of power transmitted from the tag is less than that available in backscatter systems for a given reader power level. However, the stated benefits make this method an attractive solution for numerous RFID applications.

2.4 ANTENNA BASICS

In order to understand the operation of RFID tags, it becomes important to understand the basic principles of antenna theory and how to implement them. Several important antenna concepts will be discussed in detail in the subsequent sections.

2.4.1 Radiation Pattern

The radiation pattern of an antenna is defined as "a mathematical function or graphical representation of the radiation properties of the antenna as a function of space coordinates. In most cases, the radiation pattern is determined in the far-field region and is represented as a function of the directional coordinates" [8]. In other words, the antenna pattern shows where the antenna will radiate energy and the relative magnitude.

2.4.2 Gain

The gain of an antenna is defined as "the ratio of the intensity, in a given direction, to the radiation intensity that would be obtained if the power accepted by the antenna were radiated isotropically" [8]. In other words, the gain is a measure of how an antenna directs or focuses its input power with respect to a hypothetical isotropic radiator.

The gain of an antenna is measured using a reference antenna with a known gain value. Either the reference antenna or test antenna is rotated within a plane while the other antenna has a fixed position. The power received can then be used to calculate the gain of the test antenna for various angles within the plane.

The gain of an antenna is usually given as the maximum value, e.g. 2.15dBi for a dipole, and the entire gain pattern is plotted in the two major planes, horizontal and vertical or azimuth and elevation. From these two planes, antennas can be easily compared to evaluate the gains at various angles within the planes. Antenna gains are rarely given in three-dimensions due to the increase in the number of measurement points and the fact that most antenna systems are designed in order to utilize the maximum gain value in a favorable direction.

2.4.3 Effective Area

The effective area of an antenna is defined as "the ratio of the available power at the terminals of a receiving antenna to the power flux density of a plane wave incident on the antenna from that direction, the wave being polarization matched to the antenna" [8]. Simply put, the effective area is the power delivered to the load divided by the power density per square meter of the incoming wave. This equation can be seen in (2.1).

$$A_e = \frac{P_R}{S} \tag{2.1}$$

where

A_e is the effective area of the antenna (m)
P_R is the power received at the load (W)
S is the power density $\left(\frac{W}{m^2}\right)$

Using the Friis Equation derived in Section 2.4.6 and shown in (2.2), a relationship between the gain and effective area can be established.

$$P_R = P_T \frac{G_T(\theta_T,\phi_T)G_R(\theta_R,\phi_R)\lambda^2}{(4\pi r)^2}(1-\left|\Gamma_T\right|^2)(1-\left|\Gamma_R\right|^2)\left|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_R\right|^2 \tag{2.2}$$

By noting that $\frac{P_T G_T(\theta_T,\phi_T)}{4\pi r^2}(1-\left|\Gamma_T\right|^2)$ is equal to the power density, S, equation (2.2) can be written as

$$P_R = \frac{S\,G_R(\theta_R,\phi_R)\lambda^2}{4\pi}(1-\left|\Gamma_R\right|^2)\left|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_R\right|^2 \tag{2.3}$$

Rearranging (2.1)

$$P_R = S\,A_e \tag{2.4}$$

Comparing (2.3) and (2.4) yields the following gain and effective area relationship.

$$A_e = \frac{G_R(\theta_R,\phi_R)\lambda^2}{4\pi}(1-\left|\Gamma_R\right|^2)\left|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_R\right|^2 \tag{2.5}$$

where

A_e – effective area
$G_R(\theta_R,\phi_R)$ – angular dependent receiver gain
Γ_R – receiver reflection coefficent
$\hat{\mathbf{p}}_T$ – incident wave polarization vector
$\hat{\mathbf{p}}_R$ – receiver polarization vector
λ – wavelength

2.4.4 Reciprocity

The reciprocity theorem applied to an antenna states that an antenna located in a linear and isotropic medium will have the same properties whether in transmit or receive mode. Reciprocity can also be applied to an antenna system. If two antennas are separated by a linear and isotropic medium and the first antenna is used to transmit to the second antenna, the power received will be the same if the second antenna is used to transmit to the first. The reciprocity theorem allows antenna properties such as gain to be measured in either transmit or receive mode, which can be important depending on the size of the test antenna. It may be difficult to rotate the test antenna with respect the reference antenna in order to measure the principle planes of radiation.

2.4.5 Far-Field Boundary

As has been stated, the current research focuses on RFID tags operating in the far-field. It, therefore, becomes important to determine a boundary between the near-field and the far-field. The common method is to determine where the parallel ray approximation begins to breakdown [9]. Basically, it must be determined where the wave exiting the antenna can no longer be approximated as plane waves impinging on the receiving antenna. Typically, a phase error of $\frac{\pi}{8}$ is considered acceptable thereby yielding the common near-field/far-field boundary given in the following equation.

$$r > \frac{2D^2}{\lambda} \tag{2.6}$$

where

r – distance from the antenna
D – maximum dimension of the transmitting or receiveing antenna
λ – wavelength

This condition is sufficient for antennas operating in the VHF band and above, however, for lower frequencies the far-field boundary may need to be defined at distances greater than those found using (2.6) [9].

2.4.6 Derivation of the Friis Equation

The Friis Equation was developed at Bell Telephone Laboratory by Harold T. Friis and was published in 1946 [10]. The equation is used to predict the amount of power received at a distance, r, from a transmitting antenna given detailed information about the system and antennas. To derive the Friis Equation, one must first realize that the power density transmitted from an antenna can be written as the following.

$$S = \frac{P_T G_T(\theta_T,\phi_T)}{4\pi r^2}(1-|\Gamma_T|^2) \tag{2.7}$$

where

S – transmitted power density $\left(\frac{W}{m^2}\right)$
P_T – transmitted power
$G_T(\theta_T,\phi_T)$ – angular dependent transmitter gain
Γ_T – transmitter reflection coefficent
r – distance from the transmitter
λ – wavelength

As was shown in Section 2.4.3, the power received by an antenna can be expressed by the following equation.

$$P_R = S\,A_e \tag{2.8}$$

Inserting (2.5) into (2.8) yields

$$P_R = S\frac{G_R(\theta_R,\phi_R)\lambda^2}{4\pi}(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_R|^2 \tag{2.9}$$

Substituting (2.7) in (2.9) gives the complete Friis Equation.

$$P_R = P_T\frac{G_T(\theta_T,\phi_T)G_R(\theta_R,\phi_R)\lambda^2}{(4\pi r)^2}(1-|\Gamma_T|^2)(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_R|^2 \tag{2.10}$$

where

P_R – received power
P_T – transmit power
$G_R(\theta_R,\phi_R)$ – angular dependent receiver gain
$G_T(\theta_T,\phi_T)$ – angular dependent transmitter gain
Γ_T – transmitter reflection coefficent
Γ_R – receiver reflection coefficent
$\hat{\mathbf{p}}_T$ – transmitter polarization vector
$\hat{\mathbf{p}}_R$ – receiver polarization vector
r – distance between the transmitter and receiver
λ – wavelength

2.4.7 Radar Equation

The Radar Equation is used to predict the amount of power reflected from an object to a receiver when the object is interrogated by a transmitting antenna. The derivation of the Radar Equation is similar to the Friis Equation except for the fact that the receiving object is no longer an antenna but rather a target. Because the target is not an antenna, it will not have an associated gain term. Therefore, it becomes necessary to define a new term to describe how the object receives power. This term is the Radar Cross Section (RCS). The RCS of an object is defined as "the area intercepting that amount of power which, when scattered isotropically, produces at the receiver a density which is equal to that scattered by the actual target" [1].

Using the definition of RCS, consider a target that initially captures energy and then reradiates it isotropically. The power received by the target can be described by the following equation [1].

$$P_C = \sigma\, S = \sigma\, \frac{P_T G_T(\theta_T,\phi_T)}{4\pi\, R_1^2}(1-|\Gamma_T|^2) \tag{2.11}$$

The power is then reradiated by the target to produce a power density, S_S.

$$S_S = \frac{P_C}{4\pi\, R_2^2} = \sigma\, \frac{P_T G_T(\theta_T,\phi_T)}{\left(4\pi\, R_1 R_2\right)^2}(1-|\Gamma_T|^2) \tag{2.12}$$

The scatter power is then captured by a receiver with an associated effective area. The resulting power is given by (2.13) and (2.14), which represent forms of the complete Radar Equation.

$$P_R = S_S A_e = \sigma\, \frac{P_T G_T(\theta_T,\phi_T)}{\left(4\pi\, R_1 R_2\right)^2}(1-|\Gamma_T|^2)\frac{G_R(\theta_R,\phi_R)\lambda^2}{4\pi}(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_S \bullet \hat{\mathbf{p}}_R|^2 \tag{2.13}$$

$$P_R = \sigma\, \frac{P_T G_T(\theta_T,\phi_T) G_R(\theta_R,\phi_R)}{4\pi}\left(\frac{\lambda}{4\pi\, R_1 R_2}\right)^2 (1-|\Gamma_T|^2)(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_S \bullet \hat{\mathbf{p}}_R|^2 \tag{2.14}$$

3.0 RESEARCH RESULTS

3.1 RADAR AND FRIIS EQUATIONS

As previously stated, the common Radar and Friis Equations shown in (1.1) and (1.2), respectively, make assumptions that limit their validity to a single point on a line in space that connects the transmitting and receiving antennas along the axes of maximum gain. The two equations are also limited to a polarization-matched system. Figures 2 and 3 show where equations (1.1) and (1.2), respectively, are valid in a three-dimensional space for a polarization-matched system.

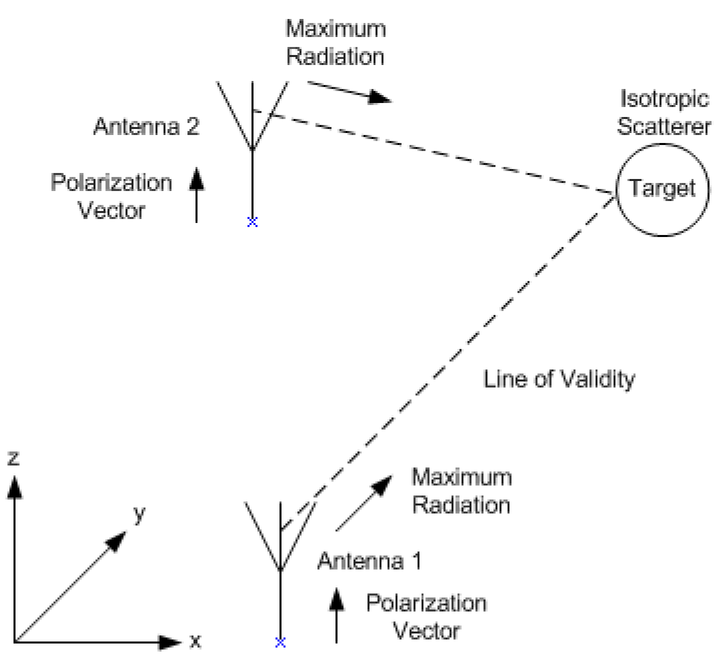

Figure 2 - Graphical Representation of where the Radar Equation in (1.1) is Valid

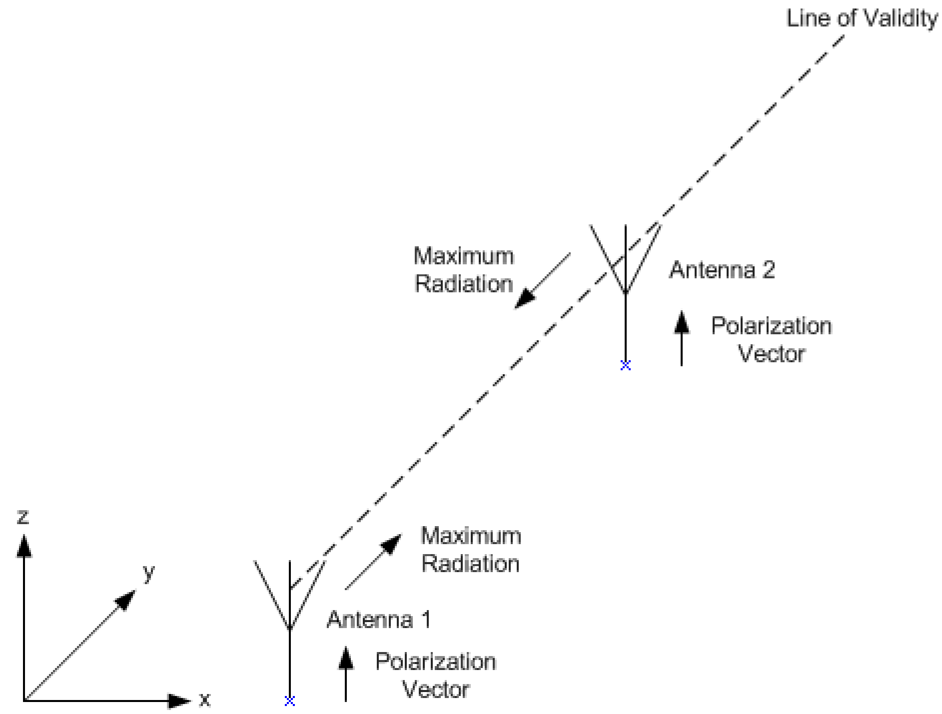

Figure 3 - Graphical Representation of where the Friis Equation in (1.2) is Valid

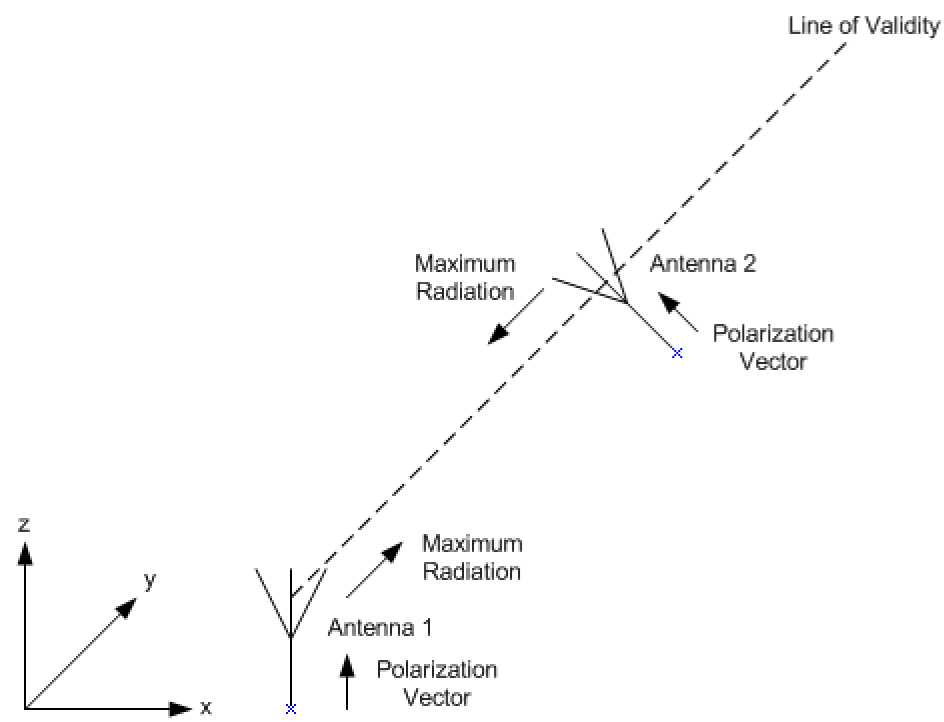

Figure 4 - Example of a Tag Position not Valid for Equation (1.2)

As Figures 2 and 3 show, the validity of the two equations is very limiting when examining the operation of an RFID system. An example of this is shown in Figure 4. If the tag is merely rotated in its polarization plane equations (1.1) and (1.2) can no longer be used.

To combat the issues just described, additional terms must be added to (1.1) and (1.2) to accurately describe any tag position in space. These equations are well documented and are given below [1]. Equation (3.1) is the three-dimensional version of the Radar Equation. Equation (3.2) is the three-dimensional version of the Friis Equation. The equation terms were briefly described in Section 1.2 and additional detail will be supplied in subsequent sections.

$$P_R = P_T \frac{\sigma(\theta_{TAG}, \phi_{TAG}) G_T(\theta_T, \phi_T) G_R(\theta_R, \phi_R)}{4\pi} \left[\frac{\lambda}{4\pi R_1 R_2} \right]^2 (1 - |\Gamma_T|^2)(1 - |\Gamma_R|^2) |\hat{\mathbf{p}}_S \bullet \hat{\mathbf{p}}_R|^2 \quad (3.1)$$

$$P_R = P_T \frac{G_T(\theta_T, \phi_T) G_R(\theta_R, \phi_R) \lambda^2}{(4\pi r)^2} (1 - |\Gamma_T|^2)(1 - |\Gamma_R|^2) |\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_R|^2 \quad (3.2)$$

3.1.1 Polarization Loss Factor (PLF) Term

As can be seen by examining (3.1) and (3.2), a polarization loss factor (PLF), $|\hat{\mathbf{p}}_1 \bullet \hat{\mathbf{p}}_2|^2$, has been added to account for rotation in the antenna's polarization plane. The polarization plane is a plane in which the gain remains constant while the antenna is rotated. As an example, two dipoles may be broadside to one another. However, they may both be vertically polarized as shown in Figure 3 or one may be horizontally polarized while the other is vertically polarized as shown in Figure 5. The gains in both cases for both antennas will be 1.64 for ideal half-wave dipoles even though the link in the latter case will have no power throughput due to the polarization mismatch. The addition of this term accounts for polarization variations.

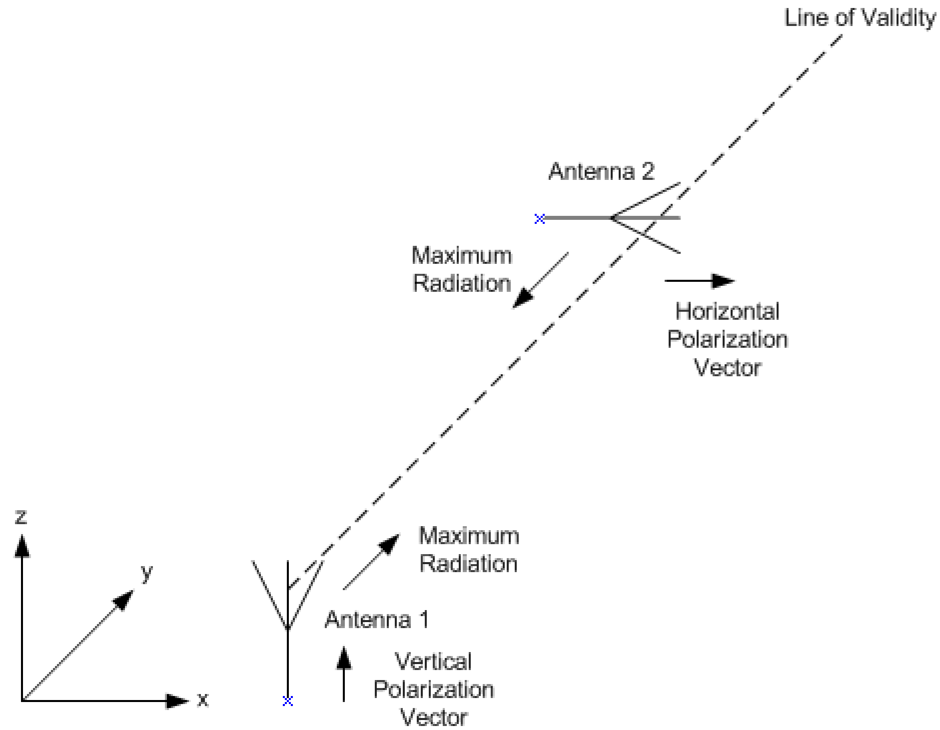

Figure 5 - Two antennas oriented broadside to each other in opposite polarizations

The two unit vectors, $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$, represent the polarization of the antennas with reference to an arbitrary polarization, which is typically that of the interrogating transmitter [1]. The polarization term is implemented using the following equality where ψ_1 and ψ_2 are the polarizations of each antenna. A plot of (3.3) can be seen in Figure 6. From the figure, it can be seen that the PLF is a symmetric function around zero degrees with a repetitive nature every 180 degrees.

$$\left|\hat{\mathbf{p}}_1 \bullet \hat{\mathbf{p}}_2\right|^2 = \cos^2(\psi_1 - \psi_2) \tag{3.3}$$

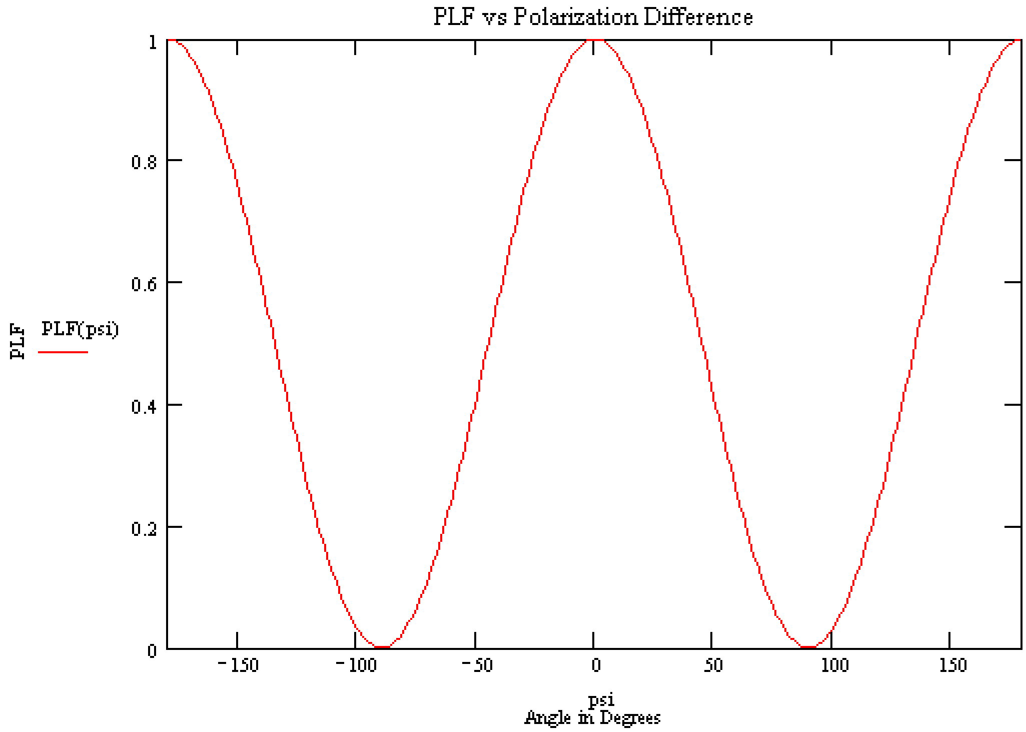

Figure 6 - Plot of the Polarization Loss Factor (PLF)

An example of rotation in the polarization plane is shown in Figure 7 where the transmitting antenna is vertically polarized and the receiving antenna is a dipole. As can be seen from the figure, if the antennas are polarization matched, the PLF is one and the term drops out of the equation. However, if the antennas have polarizations that are oriented ninety degrees apart, the PLF becomes zero and the power at the receiver is reduced to zero.

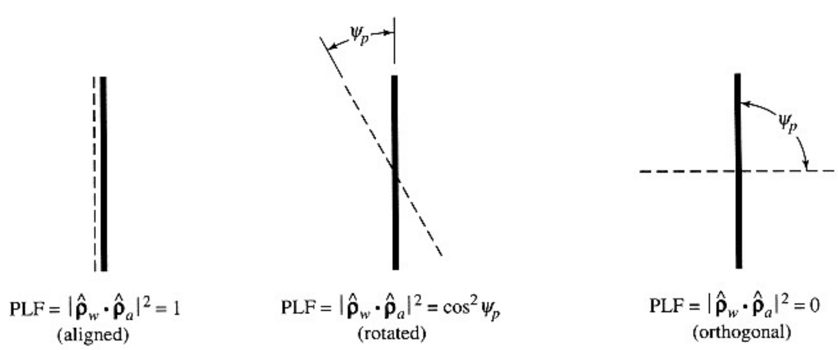

Figure 7 - Polarization Effects on Antenna Performance [1]

To verify the validity of (3.3) and Figure 7, the PLF was measured using calibrated equipment in a non-anechoic chamber environment. An RF signal generator was set to an output power of 10dBm at 915MHz and was connected to an A.H. Systems Model FCC-4 precision half-wave dipole. An additional and identical precision half-wave dipole was placed at approximately one meter from the transmitting antenna. The receiving antenna was mounted to a stepper motor with 7.2-degree rotation in the polarization plane and was connected to an RF power meter. The results can be seen in Figure 8 where the measured power in the optimum position has been normalized to one. The data can be seen in Appendix B.

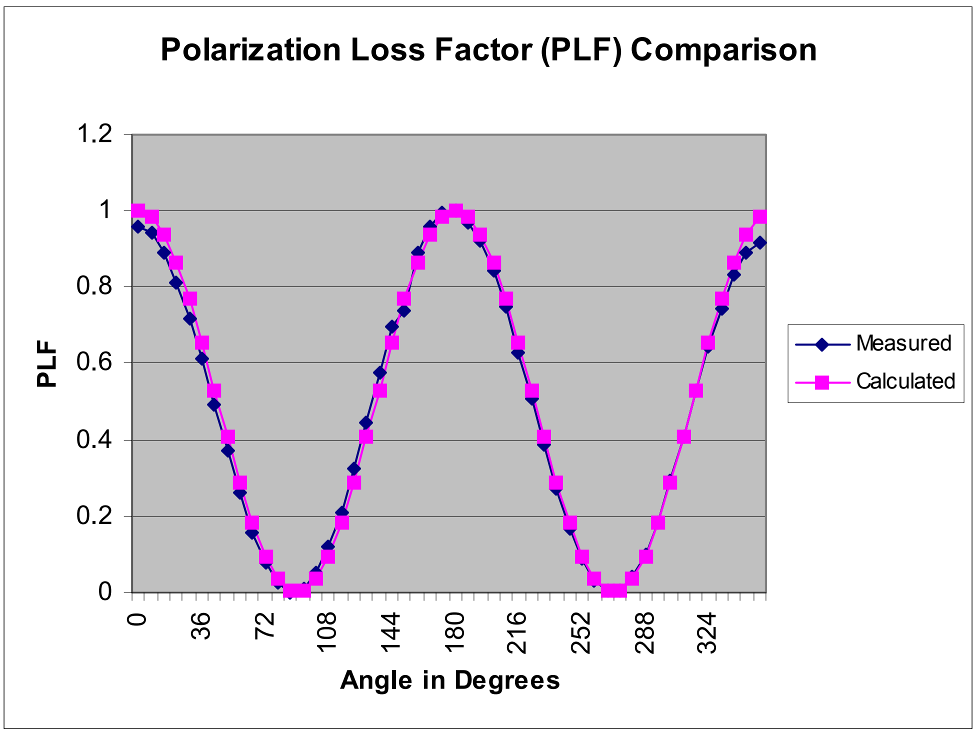

Figure 8 - Comparison of Calculated and Simulated PLF

As can be noted from Figure 8, the PLF is very accurately described by the cosine-squared function given in (3.3). There are very minor differences at 0 and 360 degrees that can be attributed to a number of factors including the non-anechoic environment and minor misalignments in the transmitting and receiving antennas.

It should be noted that the analysis above has focused on linear polarization although many RFID systems use circularly polarized antennas specifically on the interrogating transmitter and data receiver to decrease the PLF effects [11]-[12]. A circularly polarized interrogating transmitter will supply an equal amount of power to a linearly polarized tag antenna in all polarizations. The effect of this type of system is to eliminate the PLF term. However, there is a 3dB loss in gain due to the fact that the linearly polarized tag antenna can only capture a single component of the dual component circularly polarized wave. As an example, a circularly polarized wave traveling in the direction of the y-axis $(\theta = 90^{\circ}, \phi = 90^{\circ})$ can be represented by the following polarization vector.

$$\hat{\mathbf{p}}_1 = \frac{\hat{a}_\theta + j\hat{a}_\phi}{\sqrt{2}} = \frac{-\hat{a}_z - j\hat{a}_x}{\sqrt{2}} \tag{3.4}$$

If the wave described by (3.4) impinges on a linear polarized antenna with a polarization vector in the z-direction as shown in (3.5), the resulting PLF will be 0.5 or a 3dB loss as shown in (3.6).

$$\hat{\mathbf{p}}_2 = \hat{a}_z \tag{3.5}$$

$$\left|\hat{\mathbf{p}}_1 \bullet \hat{\mathbf{p}}_2\right|^2 = \left|\left(\frac{-\hat{a}_z - j\hat{a}_x}{\sqrt{2}}\right)\bullet\hat{a}_z\right|^2 = \left|\frac{-1}{\sqrt{2}}\right|^2 = 0.5 \tag{3.6}$$

The preceding analysis makes the assumption that the polarization is constant over the entire pattern although it is very difficult to design an antenna that maintains a constant polarization state [1]. This assumption is of little consequence because most of the RFID system analysis will be performed within the main lobe of the interrogating antenna.

3.1.2 Reflection Coefficient Terms

As can be seen by examining (3.1) and (3.2), two reflection coefficient terms, $(1-\left|\Gamma_T\right|^2)$ and $(1-\left|\Gamma_R\right|^2)$, have been added to account for impedance mismatch between the antenna and circuitry in the transmitter and receiver, respectively. For the transmitting case, any mismatch between the transmitter and the antenna will cause power to be reflected back to the transmitter rather than being radiated. In the receiving case, any mismatch between the antenna and tag will cause power captured by the antenna to reflect back rather than it being captured for power or data. Therefore, the reflection in both cases is considered a source of loss.

The reflection coefficient, Γ, can be calculated using the following equation.

$$\Gamma = \frac{Z_L - Z_O}{Z_L + Z_O} \tag{3.7}$$

where

$Z_L -$ load impedance
$Z_O -$ characteristic impedance

From Equation (3.7), it can be seen that the value of the reflection coefficient can take on values from minus one to one with zero representing an impedance-matched load or $Z_L = Z_O$. An open circuit load, $Z_L = \infty$, will give a gamma of one while a short circuit load, $Z_L = 0$, yields a reflection coefficient of minus one.

Because the reflection coefficient is rarely given as a specification or measurement, it becomes important to rewrite (3.7) in terms of a more practical term such as the Voltage Standing Wave Ratio (VSWR). The reflection coefficient in terms of VSWR is given in (3.8).

$$|\Gamma| = \frac{VSWR - 1}{VSWR + 1} \tag{3.8}$$

As was previously stated, the value of gamma can range from minus one to one meaning the VSWR has a range of one to infinity with one being a perfect impedance match.

3.1.3 Modeling the Three-Dimensional Gain Pattern

In order to implement equations (3.1) and (3.2), the three dimensional gains of the interrogating transmitter, the tag, and the receiving base station must be obtained. Generally, gain is given as the maximum value. As an example, it is common knowledge among RF engineers that the gain of a half-wave dipole is 1.64 or 2.15dB. This gain figure is true in the H-plane of the antenna, $(\theta = 90^{\circ} \text{ or broadside})$, however, at other values of theta the gain is less. The three-dimensional gains can be obtained in numerous ways with the most common being simulation and measurements in an anechoic chamber. However, in order to demonstrate a mathematical model of an RFID system, the three-dimensional gains of a half-wave dipole and a half-wave square patch will be mathematically analyzed in the subsequent sections. This is obviously not an option for most antenna types due to the complexity of the antenna structure. Its sole purpose is to provide a basis for the demonstration of the model developed as a part of this research. If actual dipole and patch antennas are used in the RFID system, simulated or measured data will provide results reflecting any variations in manufacturing or assembly.

3.1.4 Three-Dimensional RCS

In order to successfully calculate the minimum sensitivity required by the receiving base station for a backscatter RFID tag, the three-dimensional RCS must be determined in order to use the Radar Equation shown in (3.1). This data, unlike the three-dimensional antenna gains, cannot be easily simulated or measured due to the fact that the tag data is created by modulating the RCS of the tag. The difficulty comes from the fact that the RCS does not linearly change for two switched loads (the first for a data zero and the second for a data one) as the incident angle is varied. This phenomenon will be explored in more detail in Section 3.3 with the goal of relating the three-dimensional gain, which is easily measured, to the three-dimensional RCS.

3.2 ANTENNA PATTERN

3.2.1 Green's Function

To understand the radiation from a dipole, Green's function must be solved using Maxwell's equations in k-space [13].

$$-jk \times \vec{H} = j\omega\varepsilon \vec{E} + \vec{J}_E \tag{3.9}$$

$$\vec{k} \times \vec{E} = \omega\mu \vec{H} \tag{3.10}$$

Crossing each side of (3.10) by k yields

$$\vec{k} \times \vec{k} \times \vec{E} = \omega\mu \vec{k} \times \vec{H} \tag{3.11}$$

Rearranging (3.9) and substituting into (3.11), yields

$$\vec{k} \times \vec{k} \times \vec{E} = \omega\mu(-\omega\varepsilon \vec{E} + j\vec{J}_E) \tag{3.12}$$

Knowing that [14]

$$\nabla \times \nabla \times \vec{A} = \nabla\nabla \cdot \vec{A} - \nabla^2 \vec{A} \tag{3.13}$$

equation (3.12) simplifies to

$$\vec{k}\vec{k} \cdot \vec{E} - k^2 \vec{E} = \omega\mu(-\omega\varepsilon \vec{E} + j\vec{J}_E) \tag{3.14}$$

Rearranging terms

$$k^2 \vec{E} - \omega^2 \mu\varepsilon \vec{E} - \vec{k}\vec{k} \cdot \vec{E} = -j\omega\mu \vec{J}_E \tag{3.15}$$

and defining

$$k_o^2 \equiv \omega^2 \mu \varepsilon \tag{3.16}$$

equation (3.15) simplifies to

$$(k^2 - k_o^2)\vec{E} - \vec{k}\vec{k} \cdot \vec{E} = -j\omega\mu \vec{J}_E \tag{3.17}$$

Further simplification yields

$$\left[(k^2 - k_o^2)\overline{\overline{I}} - \vec{k}\vec{k}\right] \cdot \vec{E} = -j\omega\mu \vec{J}_E \tag{3.18}$$

Defining

$$\overline{\overline{W}}(\vec{k},\omega) \equiv \left[(k^2 - k_o^2)\overline{\overline{I}} - \vec{k}\vec{k}\right] \tag{3.19}$$

equation (3.18) reduces to

$$\overline{\overline{W}}(\vec{k},\omega) \cdot \vec{E} = -j\omega\mu \vec{J}_E \tag{3.20}$$

To solve for the electric field, an inverse of the W matrix must exist. Assuming a valid inverse, the electric field can be written as

$$\vec{E}(\vec{k},\omega) = -j\omega\mu \left[\overline{\overline{W}}(\vec{k},\omega)\right]^{-1} \cdot \vec{J}_E(\vec{k},\omega) \tag{3.21}$$

The equation in (3.21) can be examined from a systems point of view by realizing that the electric field (the output), $\vec{E}(\vec{k},\omega)$, is given by the current density (the input), $\vec{J}_E(\vec{k},\omega)$, dotted with a system response of $-j\omega\mu\left[\overline{\overline{W}}(\vec{k},\omega)\right]^{-1}$. The impulse response of this system is known as Green's Function. As with system theory, the impulse response is obtained by introducing an impulse as the input to the system. In this case, the input impulse is in four dimensions; x, y, z, and time. Therefore,

$$\vec{J}_E(\vec{r},t) = \hat{a}\delta(\vec{r})\delta(t) \tag{3.22}$$

where

$$\delta(\vec{r}) = \delta(x)\delta(y)\delta(z) \tag{3.23}$$

Transforming the current density into the frequency and k-domain yields

$$\vec{J}_E(\vec{k},\omega) = \hat{a}\int_{-\infty}^{\infty} e^{j\vec{k}\cdot\vec{r}}d^3r\int_{-\infty}^{\infty} e^{j\vec{k}\cdot\vec{r}}d^3r \tag{3.24}$$

Simplifying the integrals leads to

$$\vec{J}_E(\vec{k},\omega) = \hat{a} \tag{3.25}$$

The impulse response electric field therefore becomes

$$\vec{E}(\vec{k},\omega) = -j\omega\mu\left[\overline{\overline{W}}(\vec{k},\omega)\right]^{-1}\cdot\hat{a} \tag{3.26}$$

As has been shown by [13]

$$\left[\overline{\overline{W}}(\vec{k},\omega)\right]^{-1} = \left[(k^2-k_o^2)\overline{\overline{I}} - \vec{k}\vec{k}\right]^{-1} = \frac{k_o^2\overline{\overline{I}} - \vec{k}\vec{k}}{k_o^2(k^2-k_o^2)} \tag{3.27}$$

where $\overline{\overline{I}}$ is the identity dyad and is represented by [15]

$$\overline{\overline{I}} = [\hat{x}\hat{x} + \hat{y}\hat{y} + \hat{z}\hat{z}] \quad \text{(cartesian coordinates)}$$
$$\overline{\overline{I}} = [\hat{r}\hat{r} + \hat{\theta}\hat{\theta} + \hat{\phi}\hat{\phi}] \quad \text{(spherical coordinates)} \tag{3.28}$$

Equation (3.27) is termed the Dyadic Green's Function and is represented by $\overline{\overline{G}}(\vec{k},\omega)$.

$$\overline{\overline{G}}(\vec{k},\omega) = \frac{k_o^2\overline{\overline{I}} - \vec{k}\vec{k}}{k_o^2(k^2 - k_o^2)} \tag{3.29}$$

Writing the system in terms of its impulse response

$$\vec{E}(\vec{k},\omega) = -j\omega\mu\overline{\overline{G}}(\vec{k},\omega)\cdot\vec{J}_E(\vec{k},\omega) \tag{3.30}$$

Equation (3.30) is given in the $\vec{k},\omega$ space while the result is desired in the $\vec{r},\omega$ domain. As with other linear transformations, multiplication in one domain is represented by convolution in the other. The general form of convolution is shown by the following equations [16].

$$Y(\omega) = X(\omega)H(\omega) \tag{3.31}$$

$$y(t) = x(t) * h(t) = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)\,d\tau \tag{3.32}$$

In the case of the electric field, the conversion is between spaces and does not involve time. Therefore, t is analogous to $\vec{r}$ and the impulse response, $H(\omega)$, is analogous to the Dyadic Green's Function, $\overline{\overline{G}}(\vec{k},\omega)$. Also, the dummy variable for integration will be represented by $\vec{r}'$ rather than τ. These modifications produce the following equations.

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\overline{\overline{G}}(\vec{r},\omega) * \vec{J}_E(\vec{r},\omega) \tag{3.33}$$

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\int_V \overline{\overline{G}}(\vec{r}-\vec{r}',\omega)\cdot\vec{J}_E(\vec{r},\omega)d^3r' \tag{3.34}$$

The next step is to convert the Dyadic Green's Function from k-space to the r-space. The transformation relationship is shown in the following equation.

$$F(\vec{r}) = \frac{1}{(2\pi)^3} \int F(\vec{k})\, e^{-j\vec{k}\cdot\vec{r}}\, d^3k \tag{3.35}$$

Using the relationship

$$\overline{\overline{G}}(\vec{r}-\vec{r}',\omega) = \frac{1}{(2\pi)^3} \int \overline{\overline{G}}(\vec{k},\omega)\, e^{-j\vec{k}\cdot(\vec{r}-\vec{r}')}\, d^3k \tag{3.36}$$

where

$$\overline{\overline{G}}(\vec{k},\omega) = \frac{k_o^2 \overline{\overline{I}} - \vec{k}\vec{k}}{k_o^2 (k^2 - k_o^2)} \tag{3.37}$$

Therefore,

$$\overline{\overline{G}}(\vec{r}-\vec{r}',\omega) = \frac{1}{(2\pi)^3} \int \frac{k_o^2 \overline{\overline{I}} - \vec{k}\vec{k}}{k_o^2 (k^2 - k_o^2)}\, e^{-j\vec{k}\cdot(\vec{r}-\vec{r}')}\, d^3k \tag{3.38}$$

However, the transformation used previously can be applied in reverse to simplify equation (3.38). The transformation was given as

$$\nabla \rightarrow -j\vec{k} \tag{3.39}$$

This transformation was from r-space to k-space. The reverse transformation can be written as

$$\vec{k} \rightarrow j\nabla \tag{3.40}$$

Making this substitution into (3.38) yields,

$$\overline{\overline{G}}(\vec{r}-\vec{r}',\omega)=\frac{1}{(2\pi)^3}\int\frac{k_o^2\overline{\overline{I}}+\nabla\nabla}{k_o^2(k^2-k_o^2)}e^{-j\vec{k}\cdot(\vec{r}-\vec{r}')}d^3k \tag{3.41}$$

Removing constant terms from the integral gives

$$\overline{\overline{G}}(\vec{r}-\vec{r}',\omega)=\left(\overline{\overline{I}}+\frac{\nabla\nabla}{k_o^2}\right)\frac{1}{(2\pi)^3}\int\frac{e^{-j\vec{k}\cdot(\vec{r}-\vec{r}')}}{k^2-k_o^2}d^3k \tag{3.42}$$

$$\overline{\overline{G}}(\vec{r}-\vec{r}',\omega)=\left(\overline{\overline{I}}+\frac{\nabla\nabla}{k_o^2}\right)g(\vec{r}-\vec{r}',\omega) \tag{3.43}$$

The right-hand part of equation (3.42) is termed the Scalar Green's Function and is shown in (3.44).

$$g(\vec{r}-\vec{r}',\omega)=\frac{1}{(2\pi)^3}\int\frac{e^{-j\vec{k}\cdot(\vec{r}-\vec{r}')}}{k^2-k_o^2}d^3k \tag{3.44}$$

The integral in (3.44) must be evaluated. For simplicity $\vec{r}-\vec{r}'$ will be replaced by $\vec{R}$. The equation then reduces to

$$g(\vec{R},\omega)=\frac{1}{(2\pi)^3}\int\frac{e^{-j\vec{k}\cdot\vec{R}}}{k^2-k_o^2}d^3k \tag{3.45}$$

Expanding the volume integral using $d^3k=k^2\sin\theta\, d\theta\, d\phi\, dk$ yields

$$g(\vec{R},\omega)=\frac{1}{(2\pi)^3}\int_0^{2\pi}d\phi\int_0^{\pi}\sin\theta\, d\theta\int_0^{\infty}k^2\frac{e^{-j\vec{k}\cdot\vec{R}}}{k^2-k_o^2}dk \tag{3.46}$$

The phi integral produces a factor of 2π that reduces the power in the denominator by one. The remaining two integrals are confronted by using a change of variable [13]. This is shown by the following equations.

$$\begin{aligned} \beta &= -\cos\theta \\ d\beta &= \sin\theta \, d\theta \\ \vec{k}\cdot\vec{R} &= kR\cos\theta = -kR\beta \end{aligned} \tag{3.47}$$

Using these simplifications and adjusting the limits to reflect the limits of β

$$g(\vec{R},\omega) = \frac{1}{(2\pi)^2}\int_{-1}^{1} d\beta \int_{0}^{\infty} k^2 \frac{e^{jkR\beta}}{k^2 - k_o^2} dk \tag{3.48}$$

Rearranging terms yields

$$g(\vec{R},\omega) = \frac{1}{(2\pi)^2}\int_{0}^{\infty} \frac{k^2}{k^2 - k_o^2} dk \int_{-1}^{1} e^{jkR\beta} d\beta \tag{3.49}$$

The β integral can be easily evaluated and provides the following result.

$$\begin{aligned} g(\vec{R},\omega) &= \frac{1}{(2\pi)^2}\int_{0}^{\infty} \frac{k^2}{k^2 - k_o^2} dk \left[\frac{e^{jkR\beta}}{jkR}\right]_{-1}^{1} \\ g(\vec{R},\omega) &= \frac{-j}{R(2\pi)^2}\int_{0}^{\infty} \frac{k}{k^2 - k_o^2}\left[e^{jkR} - e^{-jkR}\right] dk \end{aligned} \tag{3.50}$$

Expanding the result into two integrals provides

$$g(\vec{R},\omega) = \frac{-j}{R(2\pi)^2}\left[\int_{0}^{\infty} \frac{ke^{jkR}}{k^2 - k_o^2} dk - \int_{0}^{\infty} \frac{ke^{-jkR}}{k^2 - k_o^2} dk\right] \tag{3.51}$$

The two integrals can be combined by changing the limits on the first integral and changing the evaluation from negative infinity to zero by replacing k by a negative k [13]. These steps are shown in the following equations.

$$g(\vec{R},\omega) = \frac{-j}{R(2\pi)^2}\left[-\int_{\infty}^{0} \frac{ke^{jkR}}{k^2 - k_o^2} dk - \int_{0}^{\infty} \frac{ke^{-jkR}}{k^2 - k_o^2} dk \right] \tag{3.52}$$

$$g(\vec{R},\omega) = \frac{-j}{R(2\pi)^2}\left[-\int_{-\infty}^{0} \frac{(-k)e^{j(-k)R}}{(-k)^2 - k_o^2} d(-k) - \int_{0}^{\infty} \frac{ke^{-jkR}}{k^2 - k_o^2} dk \right] \tag{3.53}$$

Simplifying the signs yields

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2}\left[\int_{-\infty}^{0} \frac{ke^{-jkR}}{k^2 - k_o^2} dk + \int_{0}^{\infty} \frac{ke^{-jkR}}{k^2 - k_o^2} dk \right] \tag{3.54}$$

The integrals can now be combined.

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2}\left[\int_{-\infty}^{\infty} \frac{ke^{-jkR}}{k^2 - k_o^2} dk \right] \tag{3.55}$$

The integral must be evaluated for k from negative infinity to positive infinity; however, expanding the denominator yields singularities at $k = \pm k_o$.

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2}\left[\int_{-\infty}^{\infty} \frac{ke^{-jkR}}{(k - k_o)(k + k_o)} dk \right] \tag{3.56}$$

To combat this issue, the poles must be shifted off of the k-axis by some factor alpha. This allows the integration to be performed although it introduces an unknown variable. Alpha will be eliminated later by taking the limit of the equation as it approaches zero [13]. Using this technique, (3.56) becomes

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2} \lim_{\alpha \to 0}\left[\int_{-\infty}^{\infty} \frac{ke^{-jkR}}{(k - k_o + j\alpha)(k + k_o - j\alpha)} dk \right] \tag{3.57}$$

The solution to this integral is not trivial and requires a technique from complex analysis called the Residue Theorem. The Residue Theorem states that an integral in the complex plane evaluated around a counterclockwise closed-loop can be determined by multiplying $2\pi j$ by the sum of the residues of the poles enclosed by the contour. The theorem for first order poles is shown by the following equation where n is the number of enclosed poles [13].

$$\oint_C f(x)dx = 2\pi j \sum_n \lim_{x \to x_n} (x - x_n) f(x) \tag{3.58}$$

This theorem only works for closed loop integrals, however, (3.57) is not a closed loop. The solution to this problem is to close the integral by adding an arc from positive to negative infinity which leaves two choices, the upper or lower plane. The plane must be chosen so that the contribution to the integral from the additional path length is small. This can be accomplished by making the magnitude of the exponential term go to zero along the additional path [13].

$$e^{-jkR} \to 0 \tag{3.59}$$

where k is a complex number, yielding

$$e^{-j(\mathrm{Re}\{k\}+\mathrm{Im}\{k\})R} = e^{-j(k_R + jk_I)R} = e^{k_I R} e^{-jk_R R} \tag{3.60}$$

The extra path length introduces the k_I term, which must be negative for the exponential to vanish. Therefore, the correct choice is to close the integral in the negative half of the complex plane. Closing the loop from infinity to negative infinity through an arc in the negative half of the plane represents a clockwise loop rather than the counterclockwise contour required by the Residue theorem. This issue is resolved by adding a negative sign to the result.

Because the contour is in the bottom half of the complex plane, the pole enclosed will be $k = k_o - j\alpha$. Using the Residue Theorem,

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2} \lim_{\alpha \to 0}(-2\pi j) \lim_{k \to k_o - j\alpha} (k - k_o + j\alpha) \frac{ke^{-jkR}}{(k - k_o + j\alpha)(k + k_o - j\alpha)} \tag{3.61}$$

Simplifying (3.61),

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2} \lim_{\alpha \to 0}(-2\pi j) \lim_{k \to k_o - j\alpha} \frac{ke^{-jkR}}{(k + k_o - j\alpha)} \tag{3.62}$$

Taking the right most limit, yields

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2} \lim_{\alpha \to 0}(-2\pi j) \frac{(k_o - j\alpha)e^{-j(k_o - j\alpha)R}}{2(k_o - j\alpha)} \tag{3.63}$$

$$g(\vec{R},\omega) = \frac{j}{R(2\pi)^2} \lim_{\alpha \to 0}(-2\pi j) \frac{e^{-j(k_o - j\alpha)R}}{2} \tag{3.64}$$

The final step is to apply the limit to eliminate the presence of alpha and simplify the expression. The final result is given in (3.65).

$$g(\vec{R},\omega) = \frac{e^{-jk_o R}}{4\pi R} \tag{3.65}$$

Now that the Scalar Green's Function has been derived, it can be inserted back into the expression for the electric field. As previously shown in (3.34) and (3.43), and recalling that $\vec{R} = \vec{r} - \vec{r}'$, the expressions can be written as

$$\vec{E}(\vec{r},\omega) = -j\omega\mu \int_V \overline{\overline{G}}(\vec{r} - \vec{r}',\omega) \cdot \vec{J}_E(\vec{r},\omega) d^3 r' \tag{3.66}$$

$$\overline{\overline{G}}(\vec{r} - \vec{r}',\omega) = \left(\overline{\overline{I}} + \frac{\nabla\nabla}{k_o^2} \right) g(\vec{r} - \vec{r}',\omega) \tag{3.67}$$

$$g(\vec{r} - \vec{r}',\omega) = \frac{e^{-jk_o|\vec{r} - \vec{r}'|}}{4\pi\,|\vec{r} - \vec{r}'|} \tag{3.68}$$

Combining the three expressions yields

$$\vec{E}(\vec{r},\omega) = -j\omega\mu \int_V \left(\overline{\overline{I}} + \frac{\nabla\nabla}{k_o^2} \right) \frac{e^{-jk_o|\vec{r}-\vec{r}'|}}{4\pi\,|\vec{r}-\vec{r}'|} \cdot \vec{J}_E(\vec{r},\omega)\, d^3r' \tag{3.69}$$

3.2.2 Far-field Green's Function

As was previously shown, the electric field from a current density can be described by the expression in (3.69). This expression produces a result that contains electrostatic, inductive, and radiation terms. More specifically, the result is valid for the near and far-fields. However, in most applications only the far-field is of interest. This allows the equation to be simplified by introducing the condition that the point of interest, P, is located within the far-field and therefore does not receive any contribution from the electrostatic $\left(\frac{1}{r^3}\right)$ and inductive $\left(\frac{1}{r^2}\right)$ terms.

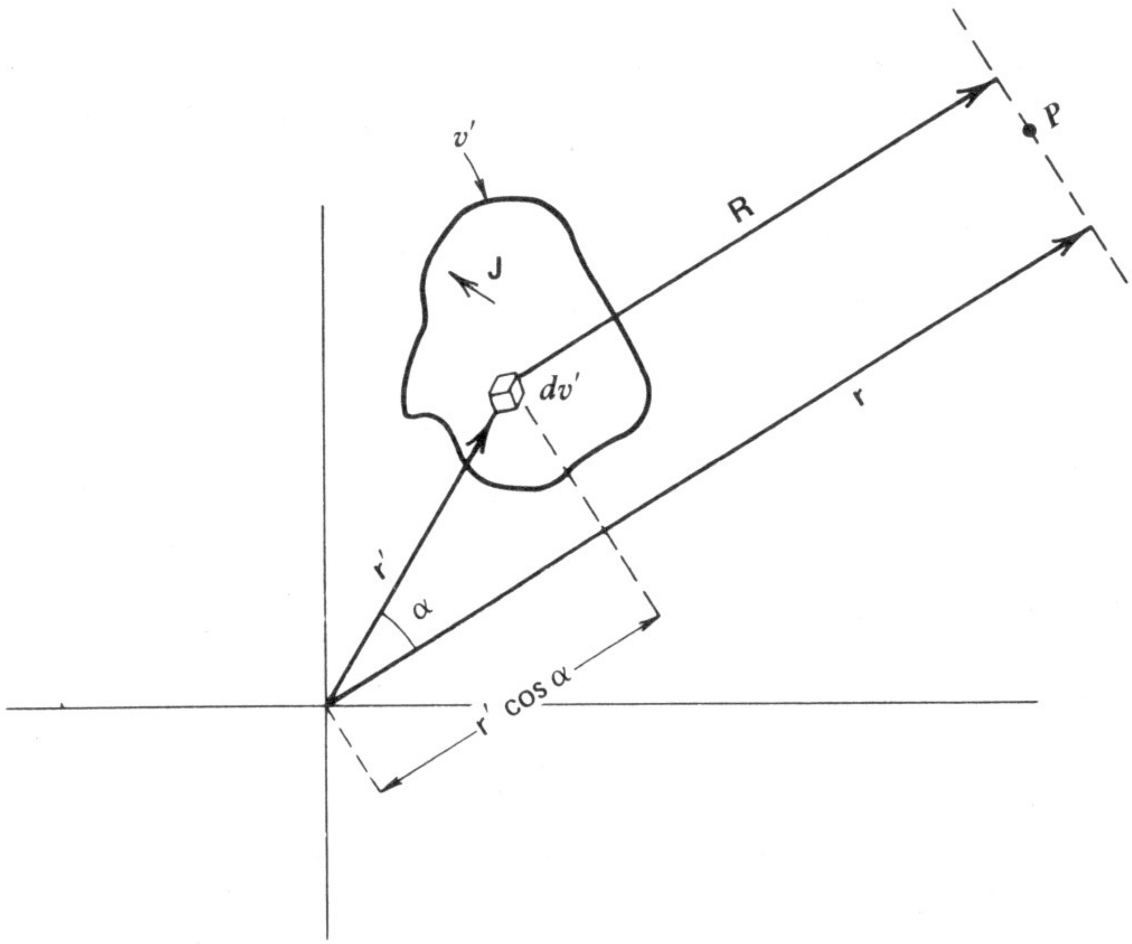

Figure 9 - Antenna with an Observation Point [9]

The expression $|\vec{r}-\vec{r}'|$ can be better understood by examining Figure 9. It is easy to see that when the observation point is located significantly far from the origin and from the current density, J, that $R=|\vec{r}-\vec{r}'|$ can be approximated as r, which is the distance from the origin. When examining (3.69), this approximation is only valid for the $|\vec{r}-\vec{r}'|$ term in the denominator. The exponential term introduces a phase shift and, therefore, $\vec{r}$ and $\vec{r}'$ may be approximately equal but the minor difference may be comparable to 2π thus introducing a significant phase shift. For this reason, the $|\vec{r}-\vec{r}'|$ term in the exponential must be approximated as $\vec{r}-\hat{r}\cdot\vec{r}'$. These simplifications are listed below.

distance

$$\left|\vec{r}-\vec{r}'\right| \approx r \tag{3.70}$$

angle

$$\left|\vec{r}-\vec{r}'\right| \approx r-\hat{r}\cdot\vec{r}' \tag{3.71}$$

Using these simplifications, (3.69) becomes

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\int_V \left(\bar{\bar{I}}+\frac{\nabla\nabla}{k_o^2}\right)\frac{e^{-jk_o(r-\hat{r}\cdot\vec{r}')}}{4\pi r}\cdot\vec{J}_E(\vec{r},\omega)d^3r' \tag{3.72}$$

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\left(\bar{\bar{I}}+\frac{\nabla\nabla}{k_o^2}\right)\cdot\frac{e^{-jk_or}}{4\pi r}\int_V e^{jk_o\hat{r}\cdot\vec{r}'}\vec{J}_E(\vec{r},\omega)d^3r' \tag{3.73}$$

However, in the far-field $\hat{r}$ is in the $\hat{k}$ direction meaning the waves are propagating radially away from the antenna. This distinction allows the $k_o\hat{r}$ term to be simplified to $k_o\hat{k}$ or simply $\vec{k}$. Equation (3.73) then reduces to

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\left(\bar{\bar{I}}+\frac{\nabla\nabla}{k_o^2}\right)\cdot\frac{e^{-jk_or}}{4\pi r}\int_V e^{j\vec{k}\cdot\vec{r}'}\vec{J}_E(\vec{r},\omega)d^3r' \tag{3.74}$$

The right-hand side of (3.74) is termed the Radiation vector, $\vec{R}$ and is shown in (3.75).

$$\vec{R} = \frac{e^{-jk_or}}{4\pi r}\int_V e^{j\vec{k}\cdot\vec{r}'}\vec{J}_E(\vec{r},\omega)d^3r' \tag{3.75}$$

Rewriting (3.74) yields

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\left(\bar{\bar{I}}+\frac{\nabla\nabla}{k_o^2}\right)\cdot\vec{R} \tag{3.76}$$

The equation in (3.76) can be simplified further by applying the identity dyad and the del operators to the radiation vector as shown in (3.77).

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\left(\bar{\bar{I}}\cdot\vec{R} + \frac{\nabla\nabla}{k_o^2}\cdot\vec{R}\right) \tag{3.77}$$

Applying the identity dyad yields

$$\begin{aligned}\vec{E}(\vec{r},\omega) &= -j\omega\mu\left(\hat{r}\hat{r}\cdot\hat{r}R + \hat{\theta}\hat{\theta}\cdot\hat{\theta}R + \hat{\phi}\hat{\phi}\cdot\hat{\phi}R + \frac{\nabla\nabla}{k_o^2}\cdot\vec{R}\right)\\ \vec{E}(\vec{r},\omega) &= -j\omega\mu\left(\hat{r}R_r + \hat{\theta}R_\theta + \hat{\phi}\vec{R}_\phi + \frac{\nabla\nabla}{k_o^2}\cdot\vec{R}\right)\end{aligned} \tag{3.78}$$

Equation (3.78) can be simplified further by applying the del operators to the radiation vector. The del operators will only operate on the exponential term because the integral is a function of $\vec{r}'$. The resulting vector will be in the $\hat{r}$ direction due to the lack of θ and ϕ dependence. This is shown in (3.79) [14].

$$\nabla A = \frac{\delta A}{\delta r}\hat{r} + \frac{1}{r}\frac{\delta A}{\delta\theta}\hat{\theta} + \frac{1}{r\sin\theta}\frac{\delta A}{\delta\phi}\hat{\phi} \tag{3.79}$$

First, the right most del operator is applied to the exponential term using the product rule.

$$\nabla\frac{e^{-jk_o r}}{r} = \left[\frac{1}{r}\frac{\delta}{\delta r}\left(e^{-jk_o r}\right) + e^{-jk_o r}\frac{\delta}{\delta r}\left(\frac{1}{r}\right)\right]\hat{r} \tag{3.80}$$

$$\nabla\frac{e^{-jk_o r}}{r} = \left[\frac{-jk_o}{r}e^{-jk_o r} + \frac{1}{r^2}e^{-jk_o r}\right]\hat{r} \tag{3.81}$$

However, in the far-field (r is large) the $\left(\frac{1}{r^2}\right)$ terms can be neglected. Equation (3.81) then reduces to

$$\nabla \frac{e^{-jk_o r}}{r} = \frac{-jk_o}{r} e^{-jk_o r} \hat{r} \tag{3.82}$$

Applying the second del operator produces the same result and the combined term is given as

$$\nabla\nabla \frac{e^{-jk_o r}}{r} = \nabla \left[\frac{-jk_o}{r} e^{-jk_o r} \hat{r} \right] = -jk_o \nabla \left[\frac{e^{-jk_o r}}{r} \right] \tag{3.83}$$

$$\nabla\nabla \frac{e^{-jk_o r}}{r} = -jk_o \left[\frac{-jk_o}{r} e^{-jk_o r} \hat{r} \right] \hat{r} = -k_o^2 \frac{e^{-jk_o r}}{r} \hat{r}\hat{r} \tag{3.84}$$

Applying the result in (3.84) to (3.78) gives

$$\vec{E}(\vec{r},\omega) = -j\omega\mu \left(\hat{r}R_r + \hat{\theta} R_\theta + \hat{\phi} R_\phi + \frac{1}{k_o^2} \left(-k_o^2 \frac{e^{-jk_o r}}{r} \hat{r}\hat{r} \right) \cdot \int_V e^{j\vec{k}\cdot\vec{r}'} \vec{J}_E(\vec{r},\omega) d^3 r' \right) \tag{3.85}$$

Rearranging terms and simplifying

$$\vec{E}(\vec{r},\omega) = -j\omega\mu \left(\hat{r}R_r + \hat{\theta} R_\theta + \hat{\phi} R_\phi - \left(\hat{r}\hat{r} \right) \cdot \frac{e^{-jk_o r}}{r} \int_V e^{j\vec{k}\cdot\vec{r}'} \vec{J}_E(\vec{r},\omega) d^3 r' \right) \tag{3.86}$$

The right-hand side of (3.86) is the radiation vector, therefore

$$\begin{aligned} \vec{E}(\vec{r},\omega) &= -j\omega\mu \left(\hat{r}R_r + \hat{\theta} R_\theta + \hat{\phi} R_\phi - \left(\hat{r}\hat{r} \right) \cdot \vec{R} \right) \\ \vec{E}(\vec{r},\omega) &= -j\omega\mu \left(\hat{r}R_r + \hat{\theta} R_\theta + \hat{\phi} R_\phi - \hat{r}\vec{R}_r \right) \end{aligned} \tag{3.87}$$

The final result is given by

$$\vec{E}(\vec{r},\omega) = -j\omega\mu \left(\hat{\theta} R_\theta + \hat{\phi} R_\phi \right) \tag{3.88}$$

The result in (3.88) as expected for the far field is independent of the $\hat{r}$ direction and is only dependent on phi and theta. The result in (3.88) can be used to calculate the electric field from an antenna with a known current density by solving the radiation vector, (3.89), and inserting the appropriate components into (3.88).

$$\vec{R} = \frac{e^{-jk_o r}}{4\pi r} \int_V e^{j\vec{k}\cdot\vec{r}'} \vec{J}_E(\vec{r},\omega) d^3 r' \tag{3.89}$$

3.2.3 Dipole Equation

As was shown previously, the far-field radiation from an antenna with a known current density, $J_E(\vec{r}',\omega)$, can be calculated by the using the following equations.

$$\vec{R} = \frac{e^{-jk_o r}}{4\pi r} \int_V e^{j\vec{k}\cdot\vec{r}'} \vec{J}_E(\vec{r},\omega) d^3 r' \tag{3.90}$$

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\left(\hat{\theta} R_\theta + \hat{\phi} R_\phi\right) \tag{3.91}$$

For a half-wave dipole antenna, the current density is well defined and can be accurately described in a sinusoidal nature [1], [10]. This is depicted in Figure 10.

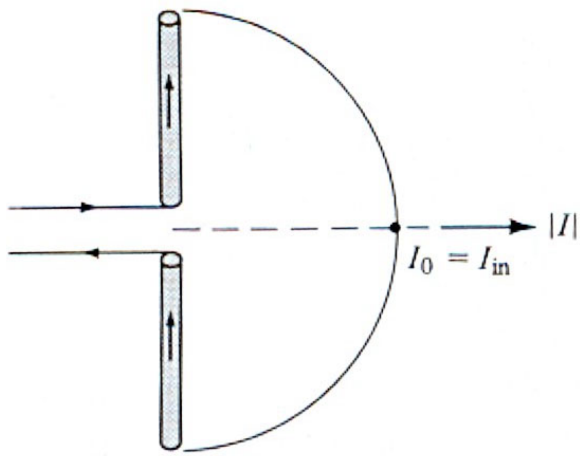

Figure 10 - Current Density of a Half-wave Dipole [10]

If the center of the dipole is assumed to be at the origin, the current density can be written as

$$\vec{J}_E = \begin{cases} \hat{z} I_o \sin\left[k_o\left(\frac{L}{2} - z\right)\right] \partial(x)\partial(y) & z > 0 \\ \hat{z} I_o \sin\left[k_o\left(\frac{L}{2} + z\right)\right] \partial(x)\partial(y) & z < 0 \end{cases} \tag{3.92}$$

where

$$k_o = \omega\sqrt{\mu\varepsilon} = \frac{\omega}{c} = \frac{2\pi f}{c} = \frac{2\pi}{\lambda} \tag{3.93}$$

Inserting (3.93) into (3.92) for the $z > 0$ case

$$\begin{aligned} \vec{J}_E &= \hat{z} I_o \sin\left[\frac{2\pi}{\lambda}\left(\frac{\lambda/2}{2} - z\right)\right] \partial(x)\partial(y) \quad z > 0 \\ \vec{J}_E &= \hat{z} I_o \sin\left[\pi\left(\frac{1}{2} - z\right)\right] \partial(x)\partial(y) \quad z > 0 \end{aligned} \tag{3.94}$$

Examining the result obtained in (3.94) shows that (3.92) accurately describes the current in a half-wave dipole as was depicted in Figure 10. When $z = \frac{1}{2}$ the sine term is zero and therefore the current density is also zero. When $z = 0$ the sine term is one and the current density is at the maximum value of I_o.

Now that the current density has been determined, the radiation vector, (3.90), must be evaluated. Inserting (3.92) into (3.90) yields

$$\vec{R} = \hat{z}\frac{e^{-jk_o r}}{4\pi r}\left[\int_0^{L/2} I_o \sin\left[k_o\left(\frac{L}{2} - z\right)\right] e^{jk_o z\cos\theta} dz + \int_{-L/2}^{0} I_o \sin\left[k_o\left(\frac{L}{2} + z\right)\right] e^{jk_o z\cos\theta} dz\right] \tag{3.95}$$

In (3.95) the delta function reduces the $e^{j\vec{k}\cdot\vec{r}'}$ term to $e^{jk_o z\cos\theta}$ and also minimizes the volume integral to a line integral along the z-axis because the current density is only dependent on z.

Using the formula [17]

$$\int e^{ax}\sin(bx + c)dx = \frac{e^{ax}}{a^2 + b^2}[a\sin(bx + c) - b\cos(bx + c)] \tag{3.96}$$

where

$$a = jk_o \cos x \qquad b = \pm k_o \qquad c = k_o \frac{L}{2} \tag{3.97}$$

equation (3.95) becomes

$$\vec{R} = \hat{z}\frac{e^{-jk_o r}}{4\pi r} I_o \left[\frac{e^{jk_o z\cos\theta}}{(jk_o\cos\theta)^2 + (-k_o)^2}\left[jk_o \cos\theta \sin\left(-k_o z + \frac{k_o L}{2}\right) + k_o \cos\left(-k_o z + \frac{k_o L}{2}\right)\right]\Bigg|_0^{L/2} + \frac{e^{jk_o z\cos\theta}}{(jk_o\cos\theta)^2 + (k_o)^2}\left[jk_o \cos\theta \sin\left(k_o z + \frac{k_o L}{2}\right) - k_o \cos\left(k_o z + \frac{k_o L}{2}\right)\right]\Bigg|_{-L/2}^{0} \right] \quad (3.98)$$

Simplifying (3.98), yields

$$\vec{R} = \hat{z}\frac{e^{-jk_o r}}{4\pi r}\frac{I_o}{k_o^2(1-\cos^2\theta)}\left[\left[\cancel{-jk_o\cos\theta\sin\left(\frac{k_o L}{2}\right)} - k_o\cos\left(\frac{k_o L}{2}\right) + k_o e^{j\frac{k_o L}{2}\cos\theta}\right] + \left[\cancel{jk_o\cos\theta\sin\left(\frac{k_o L}{2}\right)} - k_o\cos\left(\frac{k_o L}{2}\right) + k_o e^{-j\frac{k_o L}{2}\cos\theta}\right]\right] \quad (3.99)$$

$$\vec{R} = \hat{z}\frac{e^{-jk_o r}}{4\pi r}\frac{I_o}{k_o \sin^2\theta}\left[-2\cos\left(\frac{k_o L}{2}\right) + e^{j\frac{k_o L}{2}\cos\theta} + e^{-j\frac{k_o L}{2}\cos\theta}\right] \quad (3.100)$$

Using the property that

$$\cos x = \frac{e^{jx} + e^{-jx}}{2} \quad (3.101)$$

equation (3.100) becomes

$$\vec{R} = \hat{z}\frac{e^{-jk_o r}}{4\pi r}\frac{I_o}{k_o \sin^2\theta}\left[-2\cos\left(\frac{k_o L}{2}\right) + 2\cos\left(\frac{k_o L}{2}\cos\theta\right)\right] \quad (3.102)$$

$$\vec{R} = \hat{z}\frac{e^{-jk_o r}}{2\pi r}\frac{I_o}{k_o \sin^2\theta}\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right] \quad (3.103)$$

The result in (3.103) must be placed in spherical coordinates by using the transformation in (3.104) [14].

$$\hat{z} = \hat{r}\cos\theta - \hat{\theta}\sin\theta \quad (3.104)$$

Using (3.91) and noting the fact that in the far-field there is no $\hat{r}$ dependence, the electric field can be written as

$$\vec{E}(\vec{r},\omega) = -j\omega\mu\left(\hat{\theta}R_\theta + \hat{\phi}R_\phi\right)$$
$$\vec{E}(\vec{r},\omega) = -\hat{\theta}\, j\omega\mu R_\theta \tag{3.105}$$

where

$$\vec{R}_\theta = -\frac{e^{-jk_o r}}{2\pi r}\frac{I_o}{k_o \sin\theta}\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right] \tag{3.106}$$

Combining (3.105) and (3.106) yields

$$\vec{E}(\vec{r},\omega) = \hat{\theta}\, j\omega\mu\frac{e^{-jk_o r}}{2\pi r}\frac{I_o}{k_o \sin\theta}\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right] \tag{3.107}$$

Simplifying (3.107) with the following expressions,

$$k_o = \omega\sqrt{\mu\varepsilon}$$
$$\frac{\mu}{\sqrt{\mu}} = \sqrt{\mu} \tag{3.108}$$
$$\eta = \sqrt{\frac{\mu}{\varepsilon}}$$

where η is the characteristic impedance of the medium or 377 ohms for free space.

The final expression for the electric field radiated from a dipole can then be written as

$$\vec{E}(\vec{r},\omega) = \hat{\theta}\, j\eta\frac{e^{-jk_o r}}{2\pi r}\frac{I_o}{\sin\theta}\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right] \tag{3.109}$$

This expression is valid for any length of dipole where the current can be accurately described by (3.92).

Now that an expression for the electric field from a dipole has been calculated, the directivity can be derived. The directivity of an antenna is defined as the Poynting vector, $S(\theta,\phi)$, divided by the average radiated power. The average radiated power can be calculated by integrating the Poynting vector over a sphere and dividing the result by the surface area of $4\pi r^2$. The directivity can then be written as

$$D(\theta,\phi) = \frac{4\pi r^2 S(\theta,\phi)}{\int_S S(\theta,\phi) dS} \tag{3.110}$$

The Poynting vector in the far-field can be obtain using the following expression.

$$S(\theta,\phi) = \frac{\left|\vec{E}\right|^2}{2\eta}\hat{r} \tag{3.111}$$

Substituting in for the electric field magnitude yields

$$S(\theta,\phi) = \frac{\left(\eta \frac{1}{2\pi r}\frac{I_o}{\sin\theta}\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right]\right)^2}{2\eta}\hat{r} \tag{3.112}$$

$$S(\theta,\phi) = \frac{\eta}{8}\left(\frac{I_o}{\pi r}\right)^2 \left[\frac{\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)}{\sin\theta}\right]^2 \hat{r} \tag{3.113}$$

Integrating the Poynting vector over a sphere produces

$$\int_S S(\theta,\phi) = \int_0^{\pi}\int_0^{2\pi} \frac{\eta}{8}\left(\frac{I_o}{\pi r}\right)^2 \left[\frac{\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)}{\sin\theta}\right]^2 dS \tag{3.114}$$

where [14]

$$dS = r^2 \sin\theta \; d\theta \; d\phi \; \hat{r} \tag{3.115}$$

Therefore, the integral can be written as

$$\int_S S(\theta,\phi) = \int_0^{\pi}\int_0^{2\pi} \frac{\eta}{8}\left(\frac{I_o}{\pi r}\right)^2 \left[\frac{\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)}{\sin\theta}\right]^2 r^2 \sin\theta \; d\theta \; d\phi \; \hat{r} \tag{3.116}$$

$$\int_S S(\theta,\phi) = \frac{\eta}{8}\left(\frac{I_o}{\pi r}\right)^2 \int_0^{2\pi} d\phi \int_0^{\pi} \frac{\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right]^2}{\sin\theta} r^2 \; d\theta \; \hat{r} \tag{3.117}$$

$$\int_S S(\theta,\phi) = \frac{\eta}{4\pi} I_o^{\,2} \int_0^{\pi} \frac{\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right]^2}{\sin\theta} d\theta \; \hat{r} \tag{3.118}$$

Substituting (3.113) and (3.118) into (3.110) yields

$$D(\theta,\phi) = \frac{4\pi r^2 \frac{\eta}{8}\left(\frac{I_o}{\pi r}\right)^2 \left[\frac{\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)}{\sin\theta}\right]^2}{\frac{\eta}{4\pi} I_o^{\,2} \int_0^{\pi} \frac{\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right]^2}{\sin\theta} d\theta} \tag{3.119}$$

Simplifying (3.119)

$$D(\theta,\phi) = \frac{2\left[\frac{\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)}{\sin\theta}\right]^2}{\int_0^\pi \frac{\left[\cos\left(\frac{k_o L}{2}\cos\theta\right) - \cos\left(\frac{k_o L}{2}\right)\right]^2}{\sin\theta} d\theta} \tag{3.120}$$

Equation (3.120) is the directivity of a dipole antenna with a current density given by (3.92). The case of interest is the half-wave dipole, which makes $L = \frac{\lambda}{2}$. The equation then becomes

$$D(\theta,\phi) = \frac{2\left[\frac{\cos\left(\frac{k_o \lambda}{4}\cos\theta\right) - \cos\left(\frac{k_o \lambda}{4}\right)}{\sin\theta}\right]^2}{\int_0^\pi \frac{\left[\cos\left(\frac{k_o \lambda}{4}\cos\theta\right) - \cos\left(\frac{k_o \lambda}{4}\right)\right]^2}{\sin\theta} d\theta} \tag{3.121}$$

where

$$k_o = \frac{2\pi}{\lambda} \tag{3.122}$$

Substituting (3.122) into (3.121) yields

$$D(\theta,\phi) = \frac{2\left[\dfrac{\cos\left(\dfrac{\pi}{2}\cos\theta\right) - \cancel{\cos\left(\dfrac{\pi}{2}\right)}}{\sin\theta}\right]^2}{\displaystyle\int_0^{\pi} \frac{\left[\cos\left(\dfrac{\pi}{2}\cos\theta\right) - \cancel{\cos\left(\dfrac{\pi}{2}\right)}\right]^2}{\sin\theta}\, d\theta} \tag{3.123}$$

$$D(\theta,\phi) = \frac{2\left[\dfrac{\cos\left(\dfrac{\pi}{2}\cos\theta\right)}{\sin\theta}\right]^2}{\displaystyle\int_0^{\pi} \frac{\left[\cos\left(\dfrac{\pi}{2}\cos\theta\right)\right]^2}{\sin\theta}\, d\theta} \tag{3.124}$$

To obtain an expression for the three-dimensional directivity of the half-wave dipole the denominator in (3.124) must be evaluated. The integral can be evaluated numerically using a mathematical program such as Matlab by using adaptive Simpson quadrature. Using the following expression, the integral can be evaluated from 0 to pi.

```
integral_result = quad(' ((cos(3.14159/2*cos(x))).^2)./sin(x) ', 0, 3.14159)
```
(3.125)

The result is given as 1.2188 and is shown in (3.126). Simplifying the expression yields a familiar number of 1.64, which is the maximum directivity for a half-wave dipole [1], [9], [10].

$$D(\theta,\phi) = \frac{2\left[\dfrac{\cos\left(\dfrac{\pi}{2}\cos\theta\right)}{\sin\theta}\right]^2}{1.2188} \tag{3.126}$$

$$D(\theta,\phi) = 1.641\left[\frac{\cos\left(\frac{\pi}{2}\cos\theta\right)}{\sin\theta}\right]^2 \tag{3.127}$$

The three-dimensional directivity for a half-wave dipole antenna is given by equation (3.127). This expression describes how the antenna focuses its radiated power. The parameter of interest is the gain, which describes how the antenna focuses the power obtained at its input terminals. The difference between the two parameters is the antenna efficiency. This is depicted in (3.128) where e is the antenna efficiency.

$$G(\theta,\phi) = e\,D(\theta,\phi) \tag{3.128}$$

The antenna efficiency for a half-wave dipole is approximately one hundred percent and can be realized by noting that the input resistance is approximately equal to the radiation resistance (i.e. there is no loss resistance) [1]. This observation allows (3.127) to be rewritten as

$$G(\theta,\phi) = 1.641\left[\frac{\cos\left(\frac{\pi}{2}\cos\theta\right)}{\sin\theta}\right]^2 \tag{3.129}$$

As can be seen, the gain equation is only a function of theta and has no phi dependence. This result suggests an omnidirectional pattern, which accurately describes the radiation from a half-wave dipole. The maximum radiation described by (3.129) is broadside (90 degrees) to the axis of the antenna and has a maximum value of 1.64, which is the accepted gain for a half-wave dipole. Equation (3.129) is plotted in Figures 11 and 12.

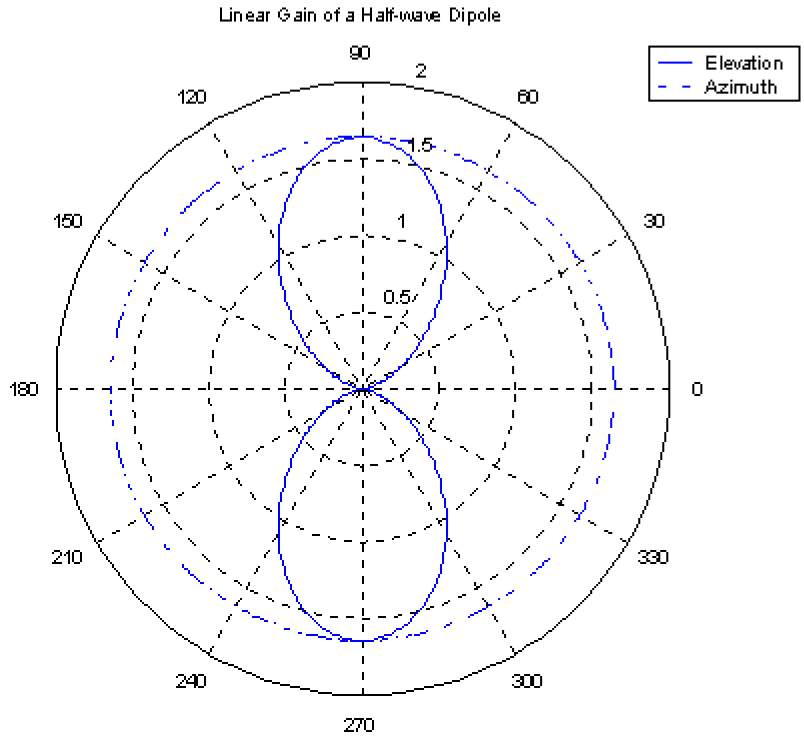

Figure 11 - Two-dimensional view of the gain for a half-wave dipole antenna as described by Equation (3.129)

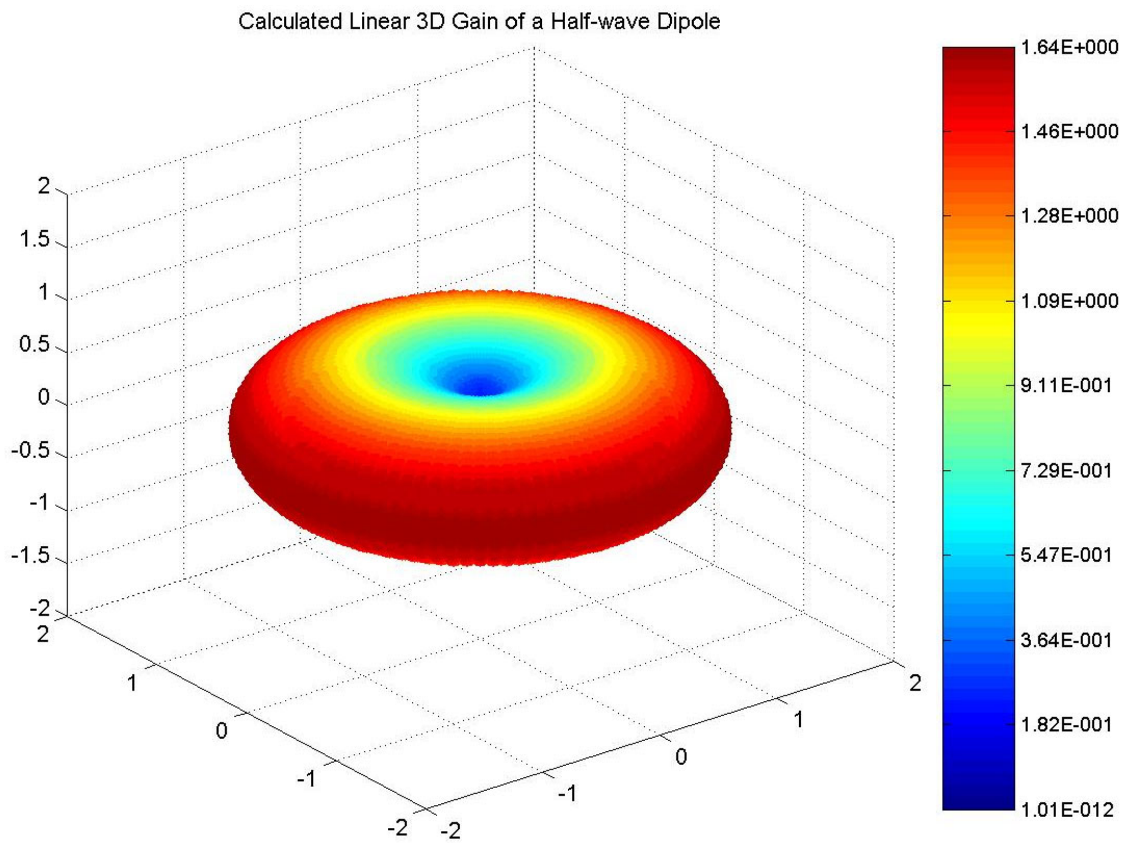

Figure 12 - Three-dimensional view of the gain for a half-wave dipole antenna as described by Equation (3.129)

To further validate the expression for a half-wave dipole antenna given in (3.129), the gain was simulated using SuperNEC. The results are shown in Figures 13-16.

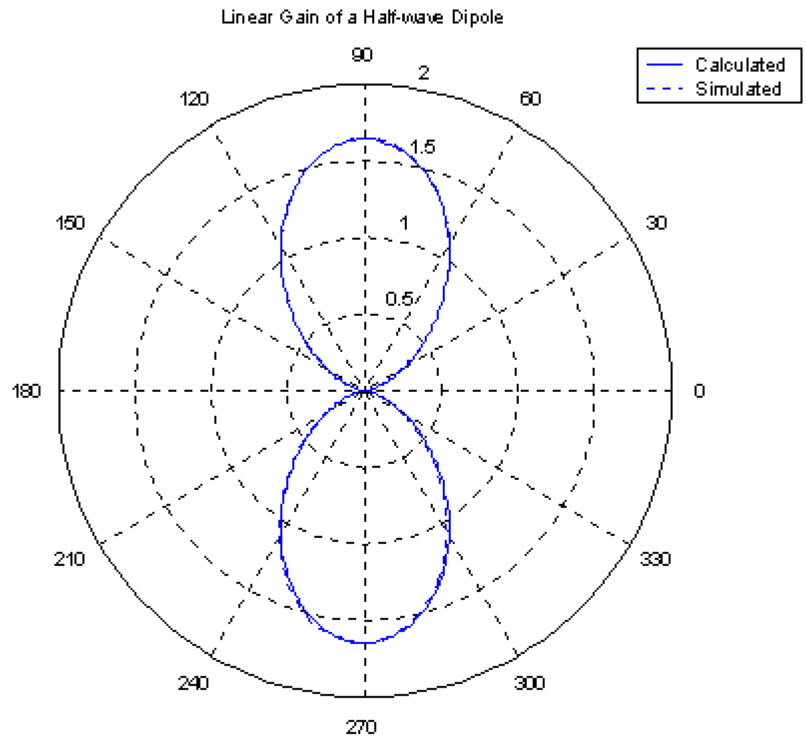

Figure 13 - Calculated and Simulated two-dimensional linear gain for a half-wave dipole antenna (Elevation)

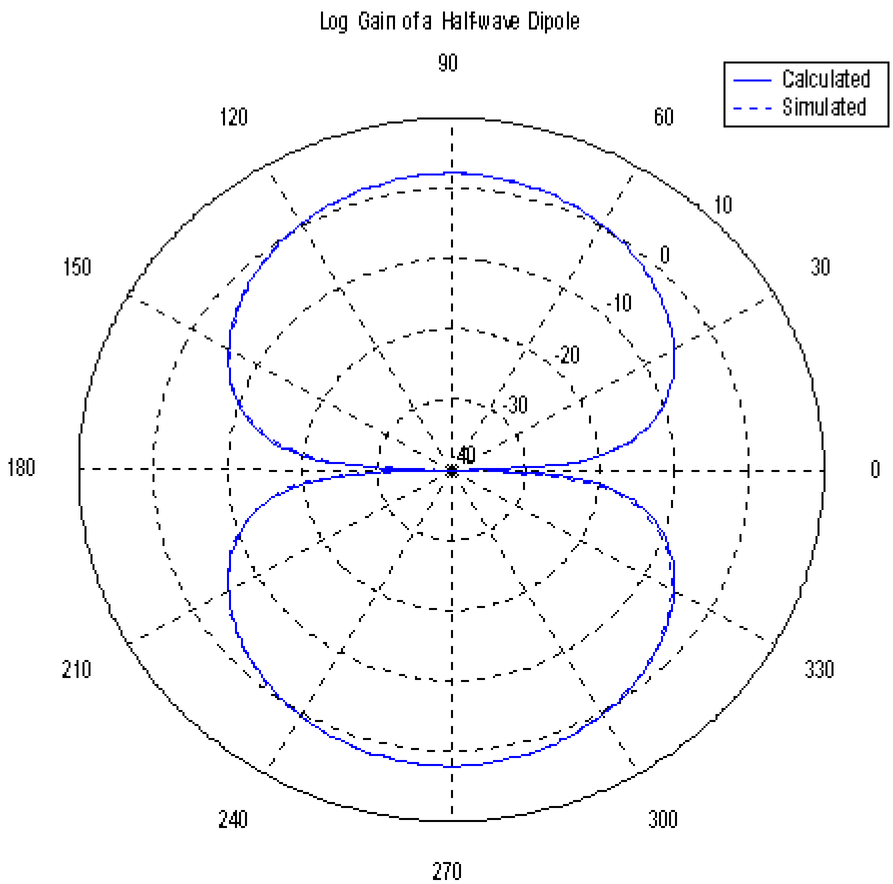

Figure 14 - Calculated and Simulated two-dimensional logarithmic gain for a half-wave dipole antenna (Elevation)

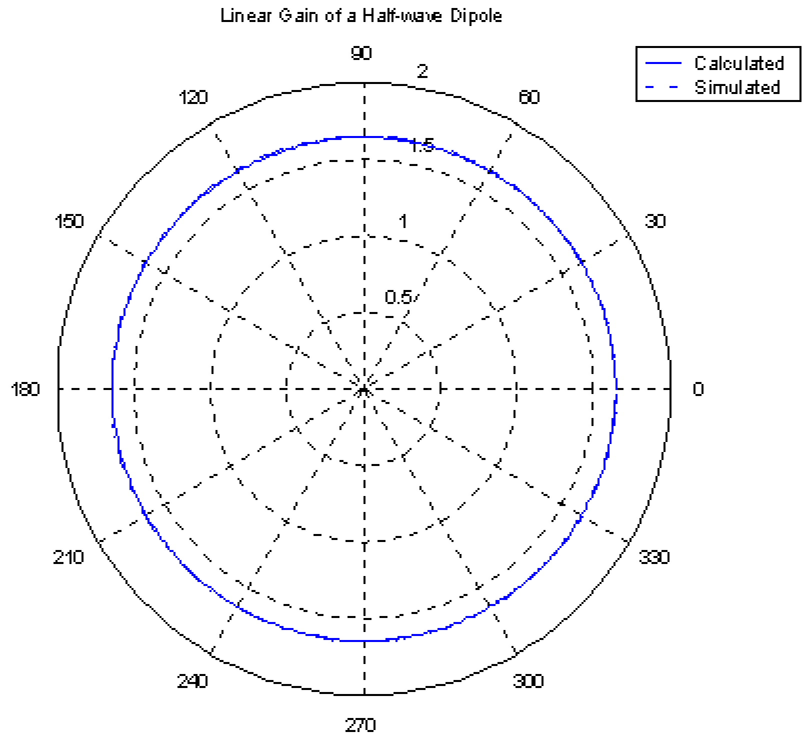

Figure 15 - Calculated and Simulated linear gain for a half-wave dipole antenna (Azimuth)

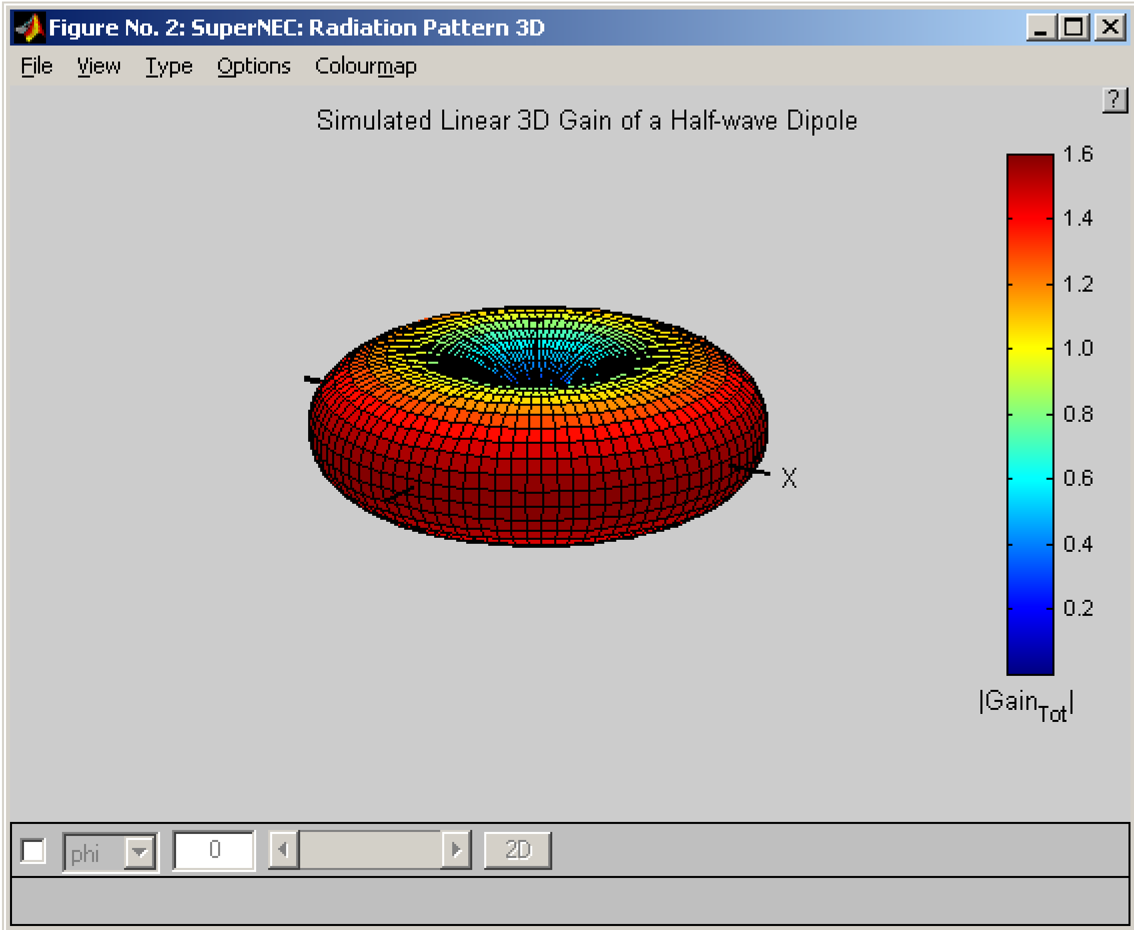

Figure 16 - Simulated three-dimensional linear gain for a half-wave dipole antenna

As can be seen from the preceding figures, the calculated and simulated results are nearly identical which suggests the result in (3.129) is an accurate representation for a half-wave dipole antenna. To further validate (3.129), the gain of a dipole antenna was measured in a non-anechoic chamber environment using two A.H. Systems Model FCC-4 precision half-wave dipoles as the transmitting and receiving antennas. The selected frequency was 915MHz due to its common use in RFID systems. The two dipoles were separated by one meter to insure a far-field gain pattern while helping to minimize the influence of possible reflections. Objects in the room were placed at least ten feet from the setup and the antennas were mounted five feet above the floor. The receiving antenna was mounted on an acrylic test fixture and was rotated using a stepper motor. The power received at each angle was record using a calibrated power meter. The measured data can be seen in Appendix C, and the results are plotted in Figure 17.

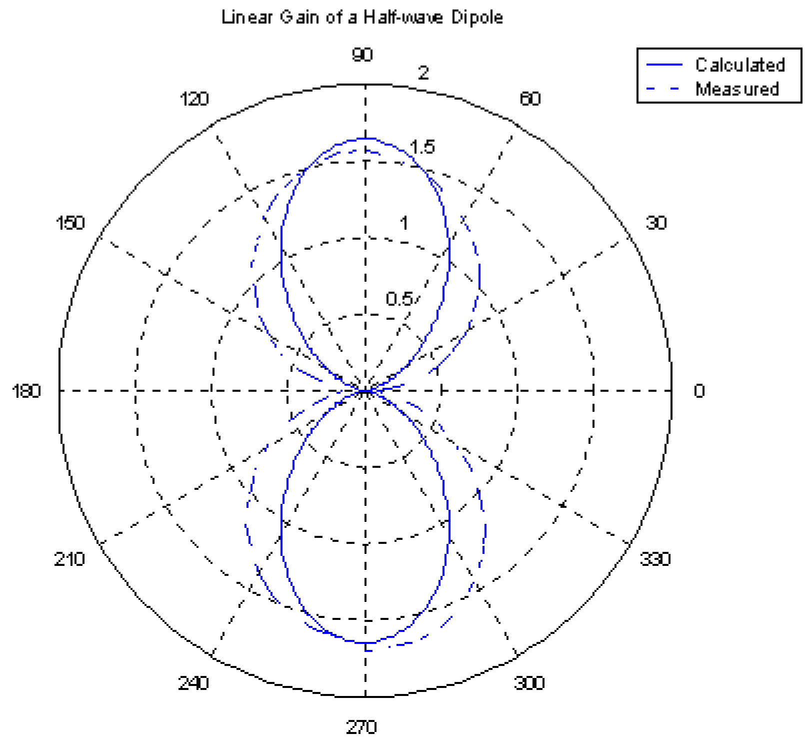

Figure 17 - Calculated and measured gain for a half-wave dipole antenna (Elevation)

The measured gain was similar to the calculated data. The broadside gain readings were very close to the accepted value of 1.64. The beamwidth of the pattern was, however, wider than the calculated data. The minor differences in the calculated and measured patterns can be attributed to the fact that the calculated result does not take into account the finite gap at the feed location and the thickness of the wire used for the dipole sections. Also, the radiation caused by the ideal sinusoidal current produces a field, which disturbs the current distribution [18]. These differences slightly modify the current distribution in the antenna that was given in (3.92). Additionally, the gain pattern was measured at one meter to minimize error cause by reflections due to the non-availability of an anechoic chamber. This distance, although technically in the far-field, may not be sufficient to approximate a plane wave from the transmitting dipole. Therefore, the pattern may not be completely independent of the distance from the transmitting antenna, which could account for some of the deviation seen.

3.2.4 Patch Equation

Calculating the gain for a patch antenna is far more complicated than a simple half-wave dipole antenna. A patch antenna is simply a metal patch placed above a ground plane fed from the side or through the ground plane. An example can be seen in Figure 18.

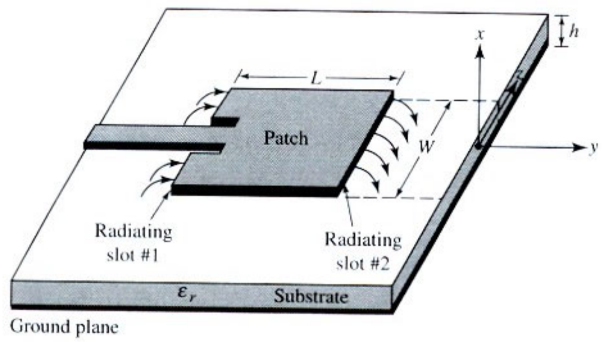

Figure 18 - Example of a patch antenna [1]

It has been shown by [1] that the radiation from a patch antenna can be modeled as two radiating slots at opposite ends of the patch separated by the antenna length, L, and having dimensions of the antenna width, W, by the height, h. Using this model for $h \ll \lambda$, the electric field can be written as

$$E_\phi = j\frac{2hE_o e^{-jk_o r}}{\pi r}\left\{\sin\theta\,\frac{\sin\left(\frac{k_o W}{2}\cos\theta\right)}{\cos\theta}\right\}\cos\left(\frac{k_o L_e}{2}\sin\theta\,\sin\phi\right) \tag{3.130}$$

where E_o is the magnitude of the electric field in the radiating slot and L_e is the effective length of the patch antenna.

Using the far-field definition, the Poynting vector of a patch antenna can be written as

$$S(\theta,\phi)=\frac{|\vec{E}|^2}{2\eta}\hat{r} \tag{3.131}$$

$$S(\theta,\phi)=\frac{2}{\eta}\left(\frac{hE_o}{\pi r}\right)^2\left\{\sin\theta\,\frac{\sin\left(\frac{k_oW}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{k_oL_e}{2}\sin\theta\,\sin\phi\right)\right\}^2\hat{r} \tag{3.132}$$

For the case of a square patch antenna where $W=L_e=\frac{\lambda}{2}$, Equation (3.132) reduces to

$$S(\theta,\phi)=\frac{2}{\eta}\left(\frac{hE_o}{\pi r}\right)^2\left\{\sin\theta\,\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2\hat{r} \tag{3.133}$$

Using the definition of directivity presented previously

$$D(\theta,\phi)=\frac{4\pi r^2 S(\theta,\phi)}{\int_S S(\theta,\phi)dS} \tag{3.134}$$

$$D(\theta,\phi)=\frac{\frac{8\pi r^2}{\eta}\left(\frac{hE_o}{\pi r}\right)^2\left\{\sin\theta\,\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2}{\int_0^\pi\int_0^\pi\frac{2}{\eta}\left(\frac{hE_o}{\pi r}\right)^2\left\{\sin\theta\,\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2 r^2\sin\theta\;d\phi\,d\theta} \tag{3.135}$$

$$D(\theta,\phi)=\frac{4\pi\left\{\sin\theta\,\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2}{\int_0^\pi\int_0^\pi\left\{\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2\sin^3\theta\;d\phi\,d\theta} \tag{3.136}$$

The expression for the directivity can be simplified by solving the integral in the denominator using a mathematical program such as Matlab. The integral can be evaluated using the following expressions.

```
func='((sin(pi/2.*cos(x))./cos(x).*cos(pi/2.*sin(x).*sin(y))).^2).*(sin(x)).^3';
dblquad(func,0,pi,0,pi)
```
(3.137)

The integral is evaluated from zero to pi for both theta and phi because the patch antenna ideally only radiates in one hemisphere. The result of the integration is 3.6067 and can be seen in the following equation.

$$D(\theta,\phi)=\frac{4\pi\left\{\sin\theta\,\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2}{3.6067} \tag{3.138}$$

$$D(\theta,\phi)=3.484\left\{\sin\theta\,\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2 \tag{3.139}$$

Equation (3.139) is the directivity for a square half-wave patch antenna. As stated previously, the parameter of interest is the gain of the antenna, which is described by

$$G(\theta, \phi) = e\,D(\theta, \phi) \tag{3.140}$$

where e is the efficiency of the antenna. Unlike the half-wave dipole case, the efficiency of a patch antenna is not one hundred percent. Therefore, the gain will be less than the calculated directivity. The antenna loss will depend on several variables including the dielectric used between the patch and the ground plane. According to [19] the efficiency can approach ninety percent when an air dielectric is used and the structure is fed properly. Therefore, e will be set to 0.9 in (3.140). The result is shown below.

$$G(\theta, \phi) = 3.136 \left\{ \sin\theta \, \frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta} \cos\left(\frac{\pi}{2}\sin\theta \, \sin\phi\right) \right\}^2 \tag{3.141}$$

To test the validity of (3.141), calculated data was compared with simulated data for a fabricated air dielectric 915MHz square patch antenna [20]. The dimensions can be seen in Figure 19.

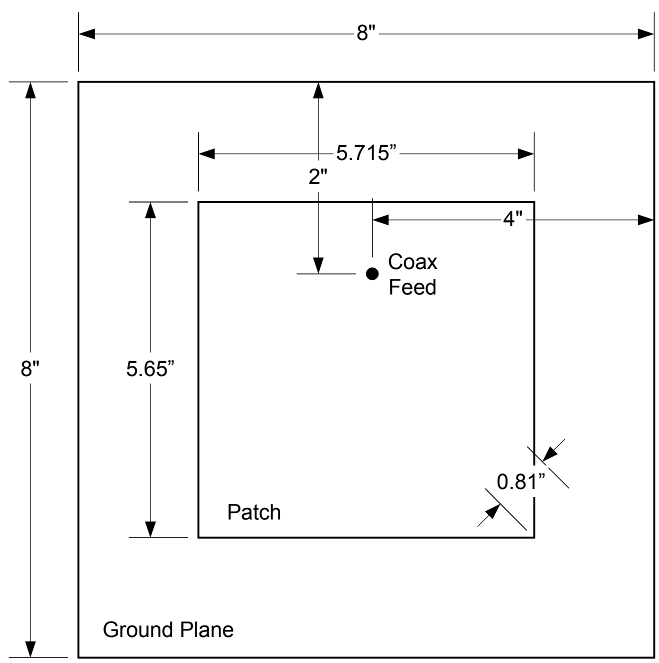

Figure 19 - Dimensions of the Fabricated Patch Antenna used for Simulations

As can be seen from the figure, the dimensions of the antenna are slightly less than a half wavelength as was the case for the dipole in order to create a resonant structure. The two principle planes of the calculated and simulated gain patterns are compared in Figures 20 and 21.

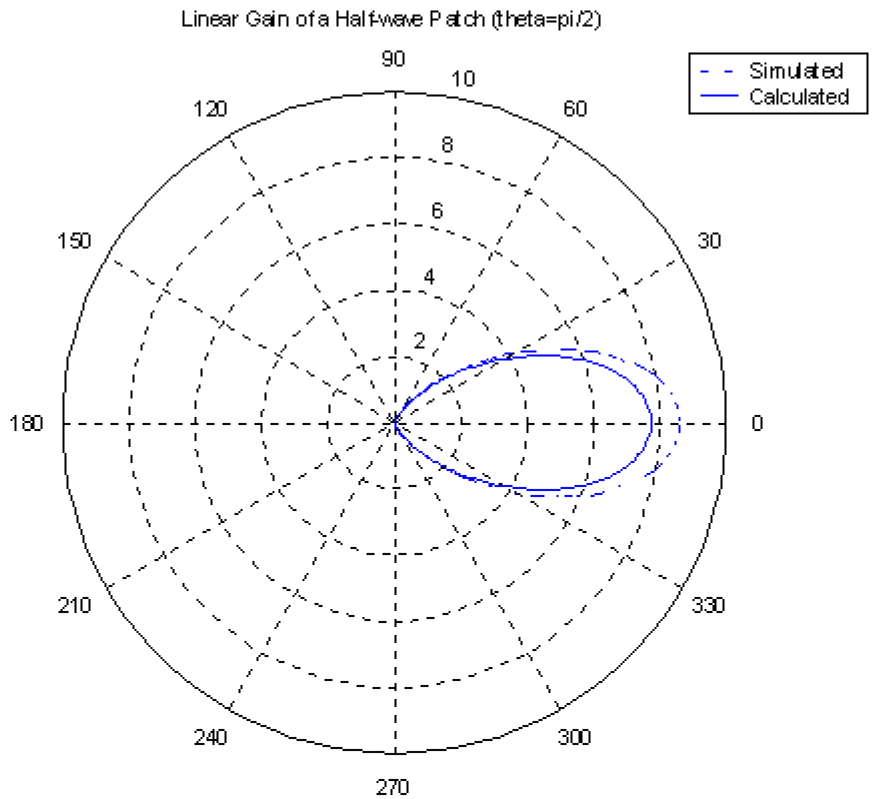

Figure 20 - Calculated and Simulated Data for a Patch Antenna $\left(\theta = \frac{\pi}{2}\right)$

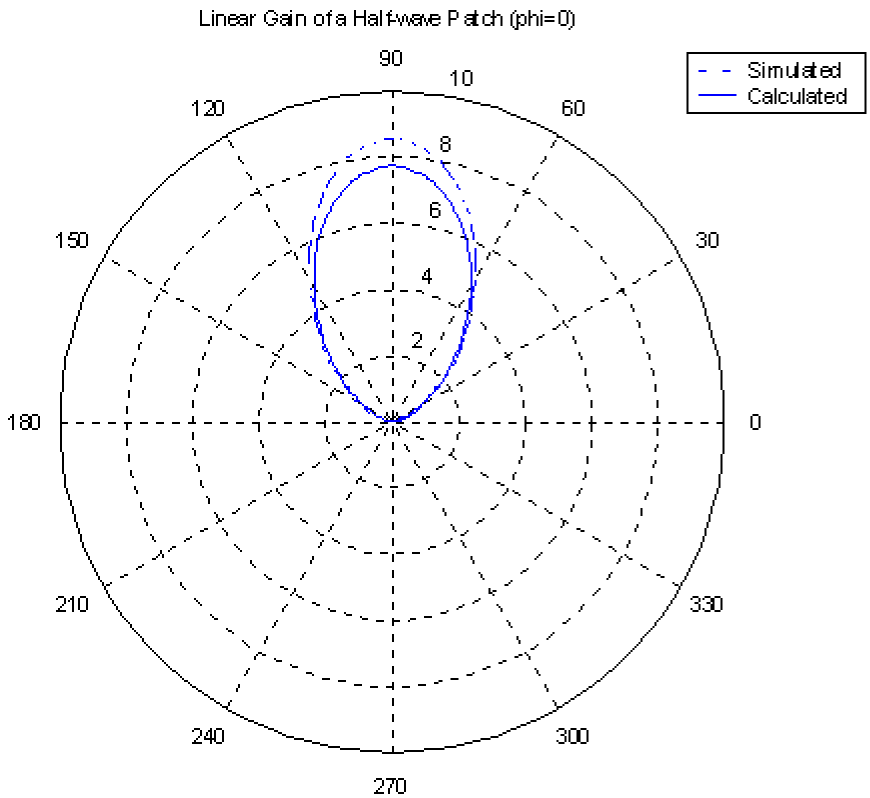

Figure 21 - Calculated and Simulated Data for a Patch Antenna $(\phi = 0)$

As can be seen by examining Figures 20 and 21, the calculated gain pattern is slightly less than the simulated gain. Therefore, the efficiency of the calculated patch antenna gain in (3.140) was adjusted to one hundred percent. The results are shown in Figures 22 and 23.

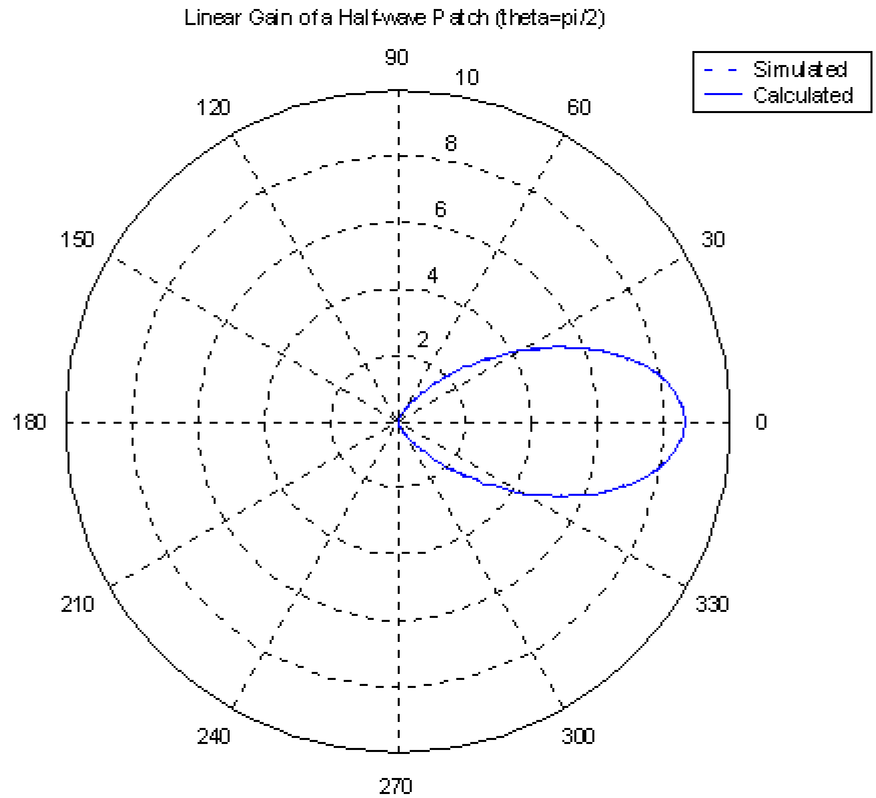

Figure 22 - Calculated and Simulated Data for a Patch Antenna with 100% Efficiency $\left(\theta = \frac{\pi}{2}\right)$

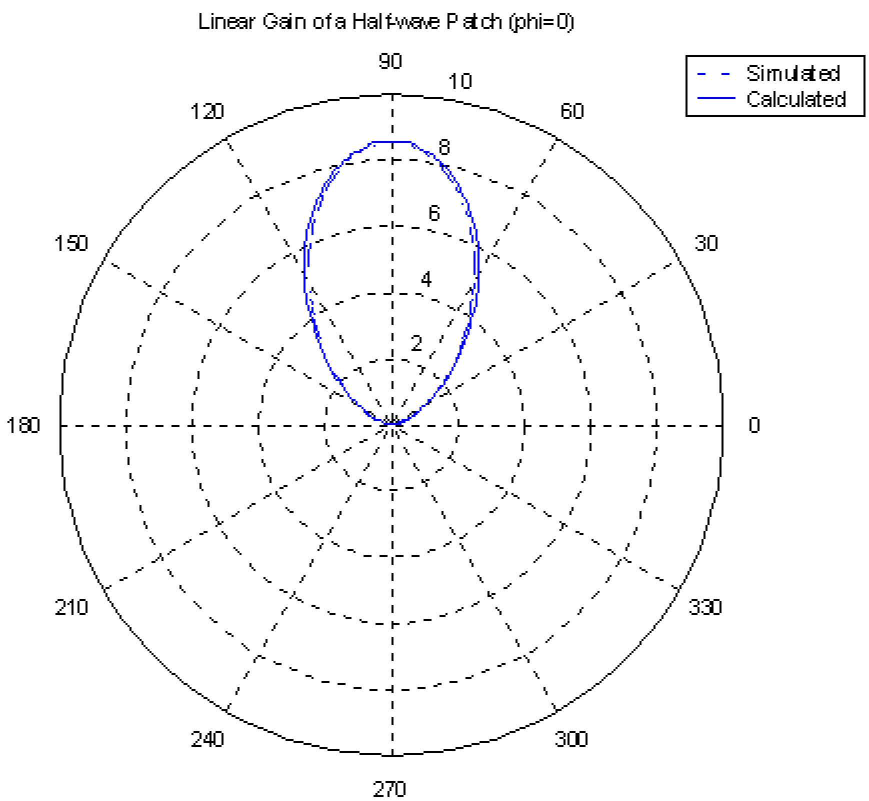

Figure 23 - Calculated and Simulated Data for a Patch Antenna with 100% Efficiency $(\phi = 0)$

As can be seen by Figures 22 and 23, the calculated and simulated data are nearly identical, which suggests that the efficiency of the simulated antenna is very near one hundred percent. To help illustrate the minor difference in the gain patterns, the resulting patterns were plotted on a logarithmic scale as shown in Figures 24 and 25. As these figures show, there are minor differences at the orthogonal angles to the main beam due to the finite size of the ground plane used for the simulations.

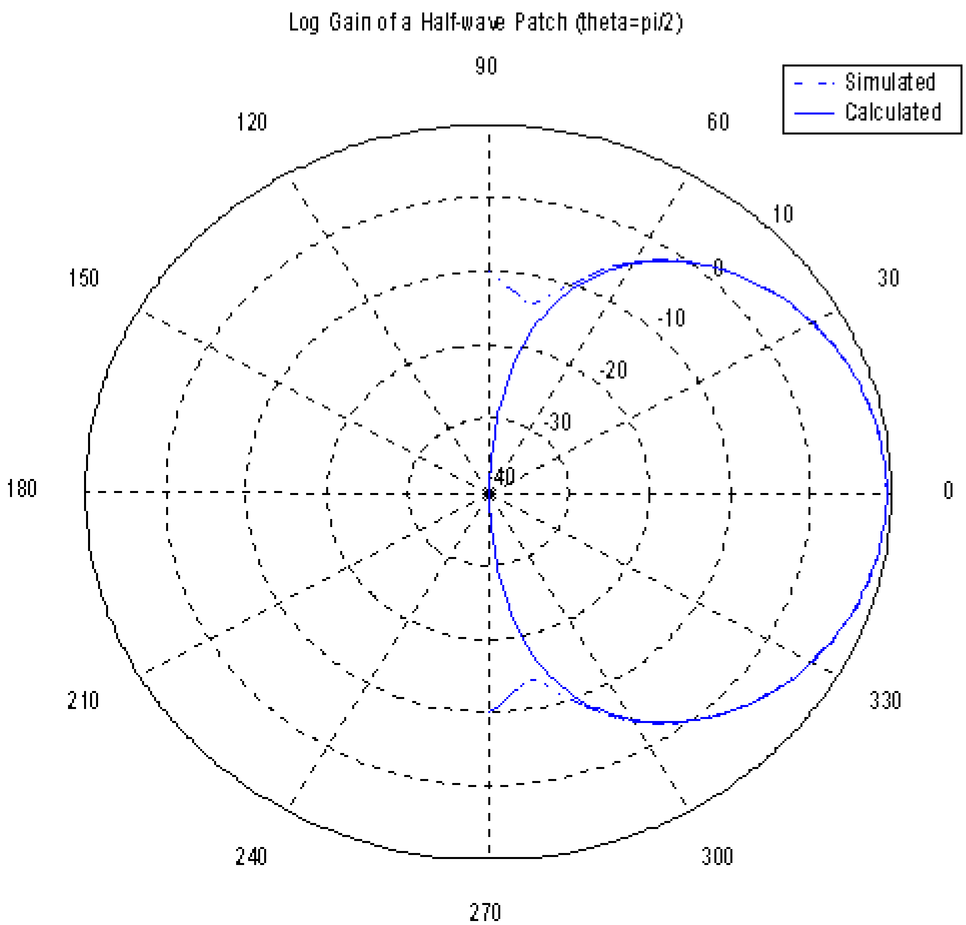

Figure 24 - Calculated and Simulated Data in dB for a Patch Antenna with 100% Efficiency $\left(\theta = \frac{\pi}{2}\right)$

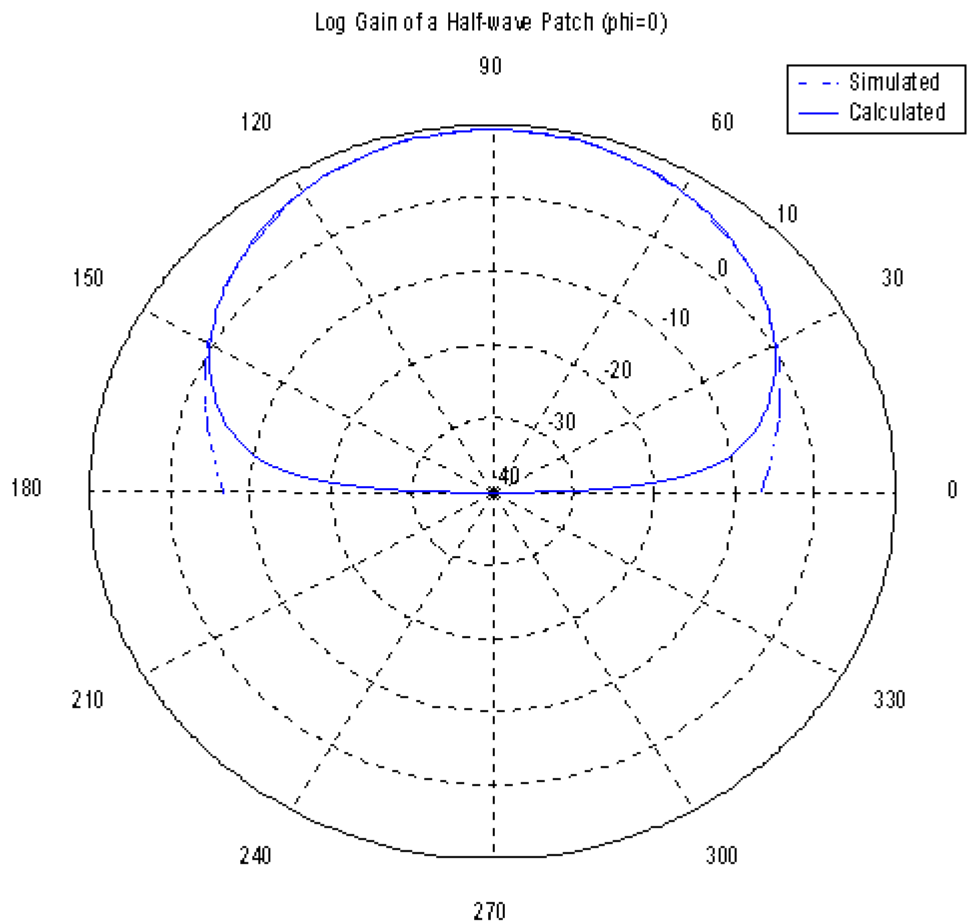

Figure 25 - Calculated and Simulated Data in dB for a Patch Antenna with 100% Efficiency $(\phi = 0)$

3.2.5 Obtaining the Three-Dimensional Gain Pattern

The equations developed in the preceding sections can be used to give an estimate of how an RFID tag will function. However, in most cases it is beneficial to know the exact radiation pattern of the antenna to obtain a more accurate solution. The equations developed do not take into account the effects of the antenna feed or the dielectric used to support the antenna structure. Therefore, it becomes necessary to measure or simulate the three-dimensional gain of the antenna to obtain an accurate solution to the RFID system. Generally, it is easier to simulate the pattern of the antenna rather then measure the three-dimensional gain due to the large number of data points and time required to make the measurements. The number of data points, N, can be calculated for a specific resolution by using the following equation where R is the number of degrees between data points. The equation is derived by the realization that a plane of rotation has 360 degrees and rotating that plane by 180 degrees captures all possible points. As can be seen by (3.142) and Figure 26, the amount of data points grows exponentially as the resolution becomes finer. An important question arises from this analysis. What is the coarsest resolution that can accurately model the RFID system with acceptable error? This question will be addressed in Section 5.1.

$$N = \left(\frac{360}{R} \times \frac{180}{R} \right)$$
$$N = \frac{64{,}800}{R^2} \tag{3.142}$$

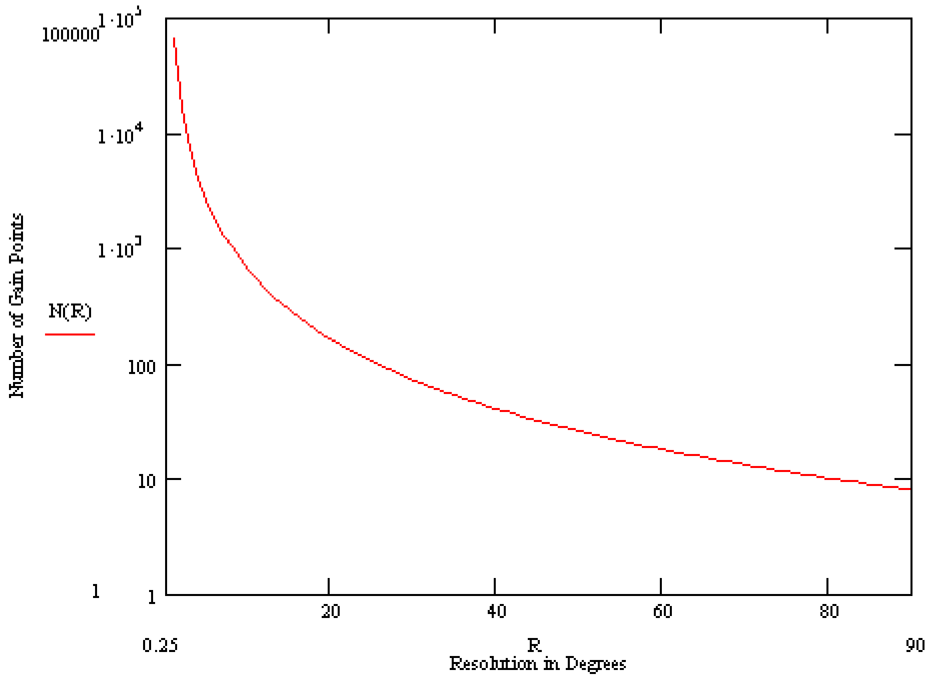

Figure 26 - Number of Gain Points for Different Resolutions

3.3 RADAR CROSS SECTION/GAIN RELATIONSHIP

In order to accurately model a backscatter RFID system, the three-dimensional RCS of the tag antenna must be determined. As previously stated, this parameter is difficult to simulate or measure, and therefore it becomes important to specify the RCS in terms of a known or easily obtainable parameter such as the antenna gain. Several different methods relating RCS to gain have been identified in the literature. Each method will be presented and then compared to determine which method most accurately describes a three-dimensional RFID system.

The first method, referred to as method one in the subsequent text, is presented in [4] and relates the RCS to the gain of the antenna using the impedances of the load and of the antenna. This is illustrated in the following equation.

$$\sigma = \frac{\lambda^2 R_r^2 G_{TAG}^2}{\pi\left[\left(R_r + R_V + R_T\right)^2 + \left(X_A + X_T\right)^2\right]} \tag{3.143}$$

where

σ – radar cross section (RCS) of the tag
G_{TAG} – gain of the transponder antenna
R_r – antenna radiation resistance
R_V – antenna loss resistance
R_T – load resistance
X_A – antenna reactance
X_T – load reactance
λ – wavelength

The second method, referred to as method two in the subsequent text, is presented in [21] and describes a change in the RCS relative to a change in the load impedance. This method is unique in the fact that the RCS is a complex value to account for any phase shift in the reflected power. The equation can be seen below.

$$\Delta\sigma = \frac{\lambda^2 G_{TAG}^2}{4\pi}\frac{\Delta Z}{R_r} \tag{3.144}$$

where

$\Delta\sigma$ – change in tag RCS
G_{TAG} – gain of the transponder antenna
R_r – antenna radiation resistance
ΔZ – change in load impedance
λ – wavelength

The third method, referred to as method three in the subsequent text, is presented in [1], [22], and [23] and relates the RCS of the antenna to its gain using a constant and the reflection coefficient. This can be seen in the following equation.

$$\sigma(Z_T,\theta,\phi)=\frac{\lambda^2}{4\pi}G_t^2(\theta,\phi)\left|C+\Gamma^*\right|^2 \tag{3.145}$$

where

$\sigma(Z_T,\theta,\phi)$ – three-dimensional tag RCS
$G_t(\theta,\phi)$ – three-dimensional gain of the transponder antenna
C – complex parameter independent of the load
Γ^* – modified current reflection coefficient
λ – wavelength

and

$$\Gamma^*=\frac{Z_A^*-Z_T}{Z_A+Z_T} \tag{3.146}$$

where Z_A and Z_T are the impedance of the antenna and tag, respectively.

According to [23] Equation (3.145) can be used to predict the RCS for different incident angles, which is important for predicting the operation of an RFID system. Additionally, [1] and [22] define the complex parameter, C, as the "structural mode" or "residual mode" antenna scattering. Both references approximate a value of $C \simeq 1$ for a half-wave dipole interrogated by an in-band frequency. This approximation simplifies (3.145) to the following.

$$\sigma(Z_T,\theta,\phi) = \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi)\left|1+\Gamma^*\right|^2 \tag{3.147}$$

Equations (3.143), (3.144), and (3.147) can also be simplified by the realization that a resonant dipole has a radiation resistance of about 65 ohms. This value is slightly less than the accepted value of 73 ohms due to the fact that the impedance of a half-wave dipole is 73+j42.5 and therefore, it is generally shortened by a few percent to enable the dipole to resonate without matching. The result is a slight reduction in the radiation resistance; however, the reactive term is reduced to zero [10]. Additionally, if a matched antenna is assumed to be lossless which is a good approximation for a half-wave dipole [9] and the RCS is modulated through purely resistive changes, the equations for the three methods simplify to the following.

$$\begin{aligned} \sigma &= \frac{\lambda^2 R_r^2 G_{TAG}^2}{\pi\left[\left(R_r+R_V+R_T\right)^2+\left(X_A+X_T\right)^2\right]} \\ \sigma &= \frac{\lambda^2 R_r^2 G_{TAG}^2}{\pi\left[\left(R_r+R_T\right)^2\right]} \end{aligned} \tag{3.148}$$

$$\Delta\sigma = \frac{\lambda^2 G_{TAG}^2}{4\pi}\frac{\Delta R}{R_r} \tag{3.149}$$

$$
\begin{aligned}
\sigma(Z_T,\theta,\phi) &= \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi) \left| 1 + \frac{Z_A^* - Z_T}{Z_A + Z_T} \right|^2 \\
\sigma(Z_T,\theta,\phi) &= \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi) \left| 1 + \frac{R_r - R_T}{R_r + R_T} \right|^2 \\
\sigma(Z_T,\theta,\phi) &= \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi) \left| \frac{R_r + R_T}{R_r + R_T} + \frac{R_r - R_T}{R_r + R_T} \right|^2 \\
\sigma(Z_T,\theta,\phi) &= \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi) \left| \frac{2R_r}{R_r + R_T} \right|^2 \\
\sigma(Z_T,\theta,\phi) &= \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi) \left| \frac{2R_r}{R_r + R_T} \right|^2 \\
\sigma(Z_T,\theta,\phi) &= \frac{\lambda^2}{\pi} \frac{G_t^2(\theta,\phi) R_r^2}{\left| R_r + R_T \right|^2}
\end{aligned}
\tag{3.150}
$$

As can be seen by examining the preceding equations, the first and third methods, (3.148) and (3.150), have reduced to the same form. A detailed analysis of the literature also revealed that the second method is an approximation derived from the first method by assuming that the values of ΔR and ΔX (change in load impedance to cause modulation) are small compared to R and X which may not be the case for all RFID systems especially those employing high percentage modulation. It should also be noted that the second method measures a change in RCS, which is not the parameter of interest. The required parameter is the minimum RCS of the tag which will be used to calculate the minimum amount of power scattered from the tag to the receiving base station in order to calculate the minimum sensitivity of the receiver. For these reasons, method three will be used to calculate the RCS of the tag. A plot of (3.147) for a shorted $(\Gamma = 1)$ and matched load $(\Gamma = 0)$ can be seen in Figure 27. As can be seen by examining (3.147), a dipole terminated with an open $(\Gamma = -1)$ produces a response of zero RCS and has been described by Kahn as an invisible antenna [24]. Hansen, however, makes the point that the open circuited dipole still scatters due to scattering from the two wire sections although the resulting RCS is very small [25]. This can be seen by examining the results reported in [1], which are shown in Figure 28.

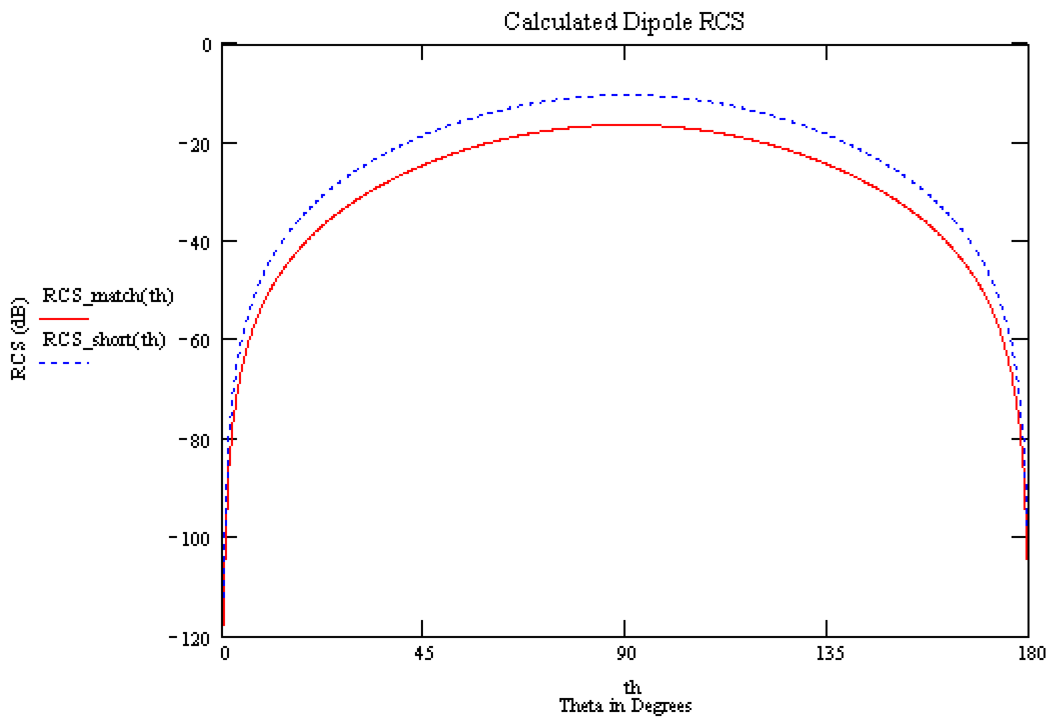

Figure 27 - Plot of Calculated RCS for a Dipole with Various Loads

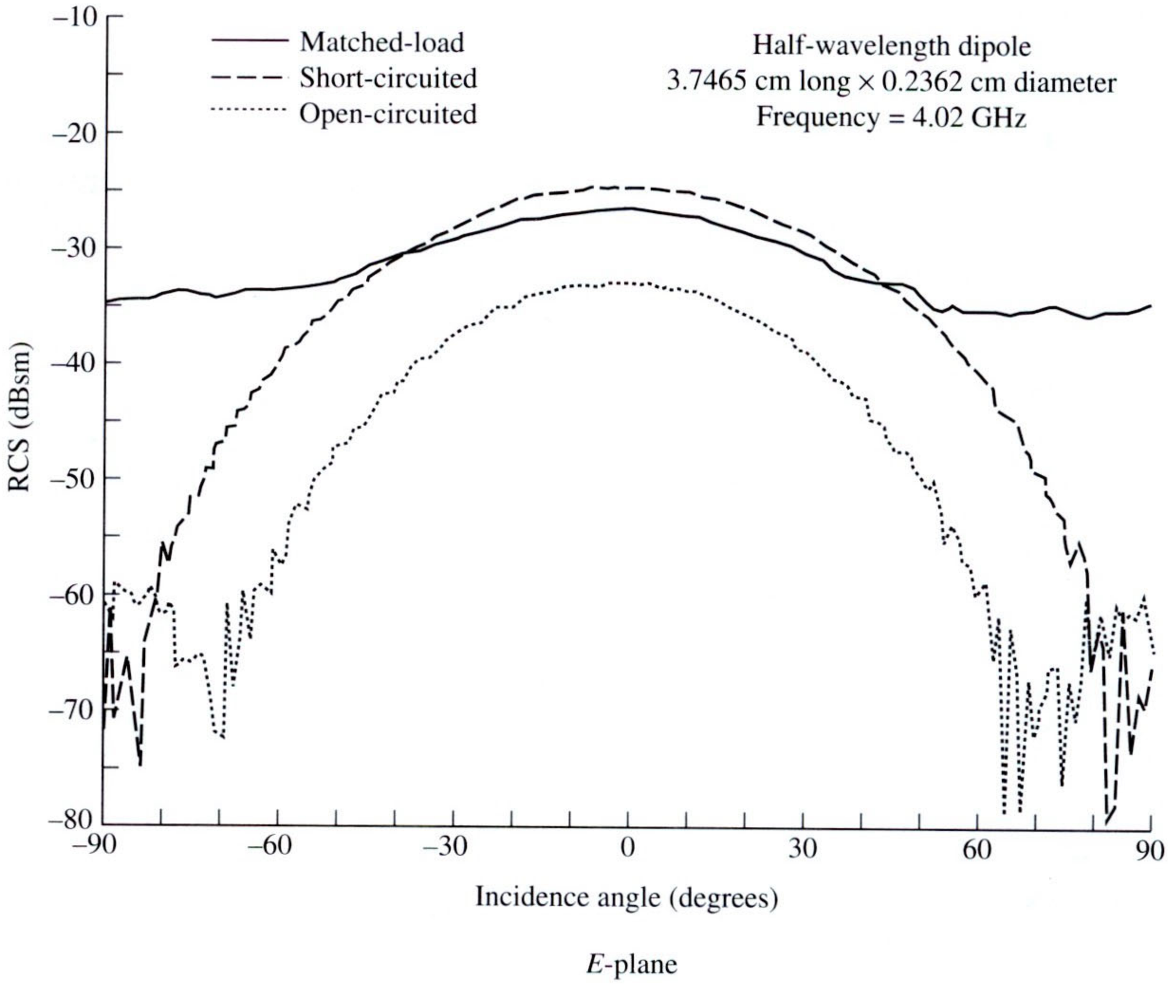

Figure 28 – Measured RCS Data for a Half-wave Dipole with Various Loads [1]

As can be seen by examining Figures 27 and 28, the calculated and measured data show similar results for values of theta (Incidence angle) from –30 to 30 degrees. However, an interesting phenomenon occurs at around ±45 degrees. The RCS for the shorted and matched case is the same. This point is examined in [21] which states that "In this case the difference in $\Delta\sigma$ or delta RCS is zero. Thus the modulation might not cause any change in scattered power. But judging from Eq 2 the phase in the scattered wave has changed. The base station can easily detect this change." Therefore, the reader can still detect the data backscattered by the tag. The major issue with the crossing point between the shorted and matched cases is that a minimum RCS must be established in order to calculate the minimum sensitivity of the base station. As Figure 28 shows, the minimum RCS for theta from –45 to 45 degrees is given by the matched case while other values of theta give a minimum RCS for the shorted case. As was previously stated, Figure 27 shows that the calculated and measured data have similar results for values of theta from –30 to 30 degrees while other angles show that the RCS follows the shape of the shorted case but has a lower value. This means that the calculated RCS will always be smaller for the matched case. Therefore, the minimum sensitivity can be set using the matched load case, and the RCS of an RFID tag using a half-wave dipole can be approximated by the following equation.

$$\sigma(\theta,\phi) = \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi) \tag{3.151}$$

Equation (3.151) can also be expressed in terms of the antenna effective area as is shown in (3.152).

$$\sigma(\theta,\phi) = A_e G_t(\theta,\phi) \tag{3.152}$$

The results presented in (3.151) and (3.152) have been derived from equations presented in the literature. However, none of the literature focused on the effects on the RCS by a mismatch in polarization. Therefore, it is proposed that an additional term be added to (3.151) and (3.152) to describe the PLF. As described in [21], the scattered power from the tag is directly proportional to the square of the current in the antenna. Likewise, the power received by the load is directly proportional to the square of the current in the antenna. The PLF in the Friis equation reduces the amount of power received by the load, which suggests that the current in the antenna is reduced. This result leads to the conclusion that the PLF not only reduces the power at the load but also the power scattered by the antenna. This realization leads to an equation for RCS that accounts for any orientation and polarization of an antenna. The result is shown in (3.153).

$$\sigma(Z_T,\theta,\phi,\hat{\mathbf{p}}_I,\hat{\mathbf{p}}_A) = \frac{\lambda^2}{4\pi} G_t^2(\theta,\phi) \left|C + \Gamma^*\right|^2 \left|\hat{\mathbf{p}}_I \bullet \hat{\mathbf{p}}_A\right|^2 \tag{3.153}$$

where $\hat{\mathbf{p}}_I$ and $\hat{\mathbf{p}}_A$ are the polarizations of the incident wave and antenna, respectively and the value of C is antenna dependent.

Applying this result to (3.151) and (3.152) leads to the following two equations for modeling an RFID system using a half-wave dipole tag antenna.

$$\sigma(Z_T,\theta,\phi,\hat{\mathbf{p}}_I,\hat{\mathbf{p}}_A)=\frac{\lambda^2}{4\pi}G_t^2(\theta,\phi)\left|\hat{\mathbf{p}}_I\bullet\hat{\mathbf{p}}_A\right|^2 \tag{3.154}$$

$$\sigma(Z_T,\theta,\phi,\hat{\mathbf{p}}_I,\hat{\mathbf{p}}_A)=A_eG_t(\theta,\phi)\left|\hat{\mathbf{p}}_I\bullet\hat{\mathbf{p}}_A\right|^2 \tag{3.155}$$

3.4 COMPLETE MATHEMATICAL MODEL

As has been previously stated, in order to predict the area of operation for an RFID system it is necessary to determine where the tag can receive enough power for operation and where the tag can successfully transmit its data back to the base station. Two mathematical models will be presented for two different types of RFID systems. The first is the ARS method described in Section 2.3.3 and the second is the traditional backscatter method discussed in Section 2.3.2.

3.4.1 ARS

The ARS method uses a separate channel for the powering and communicating links. The powering link can be described using the Friis Equation presented in (1.3) and repeated below. For clarity, the subscripts will be changed to T, R, and TAG for the interrogating transmitter, data receiver, and tag, respectively.

$$P_{TAG}=P_T\frac{G_T(\theta_T,\phi_T)G_{TAG-RX}(\theta_{TAG},\phi_{TAG})\lambda_T^2}{(4\pi r_1)^2}(1-\left|\Gamma_T\right|^2)(1-\left|\Gamma_{TAG-RX}\right|^2)\left|\hat{\mathbf{p}}_T\bullet\hat{\mathbf{p}}_{TAG-RX}\right|^2 \tag{3.156}$$

where $G_{TAG-RX}(\theta_{TAG},\phi_{TAG})$ is the three-dimensional gain of the tag's power receiving antenna.

The communicating link can also be described using the Friis Equation presented in (1.3). The resulting equation is given in (3.157).

$$P_R = \alpha P_{TAG} \frac{G_R(\theta_R,\phi_R) G_{TAG-TX}(\theta_{TAG},\phi_{TAG}) \lambda_R^2}{(4\pi r_2)^2} (1-|\Gamma_R|^2)(1-|\Gamma_{TAG-TX}|^2) |\hat{\mathbf{p}}_R \bullet \hat{\mathbf{p}}_{TAG-TX}|^2 \quad (3.157)$$

where $G_{TAG-TX}(\theta_{TAG},\phi_{TAG})$ is the three-dimensional gain of the tag's data transmitting antenna and α is a variable describing the portion of the power captured by the tag that is retransmitted as data. The value of α can range from zero to one and will depend on the power required for the logic and RF circuitry.

The complete model used to predict the operation of an ARS RFID system can be seen in the following equation.

$$\begin{aligned} P_R = \alpha P_T & \frac{G_R(\theta_R,\phi_R) G_{TAG-TX}(\theta_{TAG},\phi_{TAG}) G_T(\theta_T,\phi_T) G_{TAG-RX}(\theta_{TAG},\phi_{TAG}) \lambda_R^2 \lambda_T^2}{(4\pi)^4 (r_1 r_2)^2} \\ & \times (1-|\Gamma_T|^2)(1-|\Gamma_R|^2)(1-|\Gamma_{TAG-TX}|^2)(1-|\Gamma_{TAG-RX}|^2) |\hat{\mathbf{p}}_R \bullet \hat{\mathbf{p}}_{TAG-TX}|^2 |\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_{TAG-RX}|^2 \end{aligned} \quad (3.158)$$

It should be noted that (3.158) can be simplified for a particular RFID system, however, for completeness the full expression is presented.

3.4.2 Backscatter

The backscatter method of RFID must be separated into two parts. The first is used to calculate whether the tag can receive enough power to operate while the second is used to determine the minimum sensitivity required by the base station. The powering link, as with the ARS method, can be described by the Friis Equation shown below where no distinction is necessary for whether the gain of the tag is for the transmitting or receiving case because the same antenna is used for both. It should be noted that the reflection coefficient of the tag is a function of time due to its modulation in order to produce the backscattered signal. It therefore becomes necessary to solve (3.159) for either the average value of gamma if sufficient power can be stored in the tag for operation during the mismatched periods or for the maximum value of gamma to ensure the tag can properly operate.

$$P_{TAG} = P_T \frac{G_T(\theta_T,\phi_T)G_{TAG}(\theta_{TAG},\phi_{TAG})\lambda_T^2}{(4\pi r_1)^2}(1-|\Gamma_T|^2)(1-|\Gamma_{TAG}(t)|^2)|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_{TAG}|^2 \tag{3.159}$$

The communication link can be described using the Radar Equation given in (1.4) which is repeated below.

$$P_R = P_T \frac{\sigma(\theta_{TAG},\phi_{TAG})G_T(\theta_T,\phi_T)G_R(\theta_R,\phi_R)}{4\pi}\left[\frac{\lambda}{4\pi r_1 r_2}\right]^2(1-|\Gamma_T|^2)(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_{TAG} \bullet \hat{\mathbf{p}}_R|^2 \tag{3.160}$$

It was shown in Section 3.3 that the RCS of an antenna interrogated by an in-band frequency can be described by the following equation.

$$\sigma(Z_T,\theta,\phi,\hat{\mathbf{p}}_I,\hat{\mathbf{p}}_A) = \frac{\lambda^2}{4\pi}G_T^2(\theta,\phi)\left|C+\Gamma^*\right|^2|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_{TAG}|^2 \tag{3.161}$$

Combining (3.160) and (3.161) yields a complete expression that describes the operation of a backscatter RFID tag. This is shown in the following equation.

$$\begin{aligned} P_R = P_T &\frac{G_{TAG}^2(\theta,\phi)G_T(\theta_T,\phi_T)G_R(\theta_R,\phi_R)\lambda^4}{(4\pi)^4(r_1 r_2)^2} \\ &\times\left|C+\Gamma^*\right|^2(1-|\Gamma_T|^2)(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_{TAG}|^2|\hat{\mathbf{p}}_{TAG} \bullet \hat{\mathbf{p}}_R|^2 \end{aligned} \tag{3.162}$$

4.0 SOLVING THE MATHEMATICAL MODEL

As was shown in Section 1.2, solving the mathematical models presented in the previous sections can quickly become a massive problem not only from a processing time point of view but also from a pure memory standpoint. The simple example given required 23.3GB of data store to solve a mere 1000 points. As can be seen, the solution can easily spiral out of control as more and more data points are required. The solution to this dilemma is to develop a method to solve the problem that answers the proposed question of where an RFID tag will operate given the ability to be in any position. To do this, the question must be explored in a little more detail. What data are really required? It's obvious from the current research that the typical figure of merit, the maximum read range, is of no use. Therefore, it must be determined what performance criterion must be used. One important parameter presented in the literature is the read accuracy [26]-[27]. Companies exploring RFID are concerned with the read accuracy of the tags because a missed read means lost income for the product and reduced profits. There has been work done by numerous companies in predicting the read accuracy of RFID tags in various orientations [28]-[29]. However, there has not been a comprehensive analysis of an RFID tag with the ability to take on any position, orientation, and polarization. It is, therefore, the objective of this research to give data relative to the read accuracy of an RFID tag when it is free to take on any position, orientation, and polarization.

There is still the problem of having a large amount of data that needs to be calculated. The solution is to define a specific scenario. As an example, an RFID tag in a shopping cart was explored. The tag can take on any position and orientation and moves along a line in front of the reader. This realization simplifies the problem by only requiring data be calculated on the line of movement. The massive data problem still exists, even though it has been greatly simplified, due to the large number of points on the line.

To simplify the problem further, a technique has been developed to limit the intense calculations to a single point. The technique was developed using the realization that the tag's gain and polarization are the only variables in the Friis Equation that are affected by the tag's position at a single point in space. Therefore, these two variables can be combined into a three-dimensional matrix, χ, while the remainder of the Friis Equation can be viewed as an RFID system dependent constant, κ, and a distance variable, r. These parameters are shown in the following equations. A pictorial representation of χ can be seen in Figure 29.

$$\chi(\theta,\phi,\psi) = G_{TAG}(\theta,\phi)\left|\hat{\mathbf{p}}_T \bullet \hat{\mathbf{p}}_R\right|^2 = G_{TAG}(\theta,\phi)\cos^2(\psi) \tag{4.1}$$

$$\kappa = \frac{P_T G_T(\theta_P,\phi_P)\lambda^2}{(4\pi)^2}(1-\left|\Gamma_T\right|^2)(1-\left|\Gamma_R\right|^2) \tag{4.2}$$

where

θ_P is the theta value of the specified point
ϕ_P is the phi value of the specified point

Rewriting the Friis Equation yields

$$P_R = \frac{\kappa\,\chi}{r^2} \tag{4.3}$$

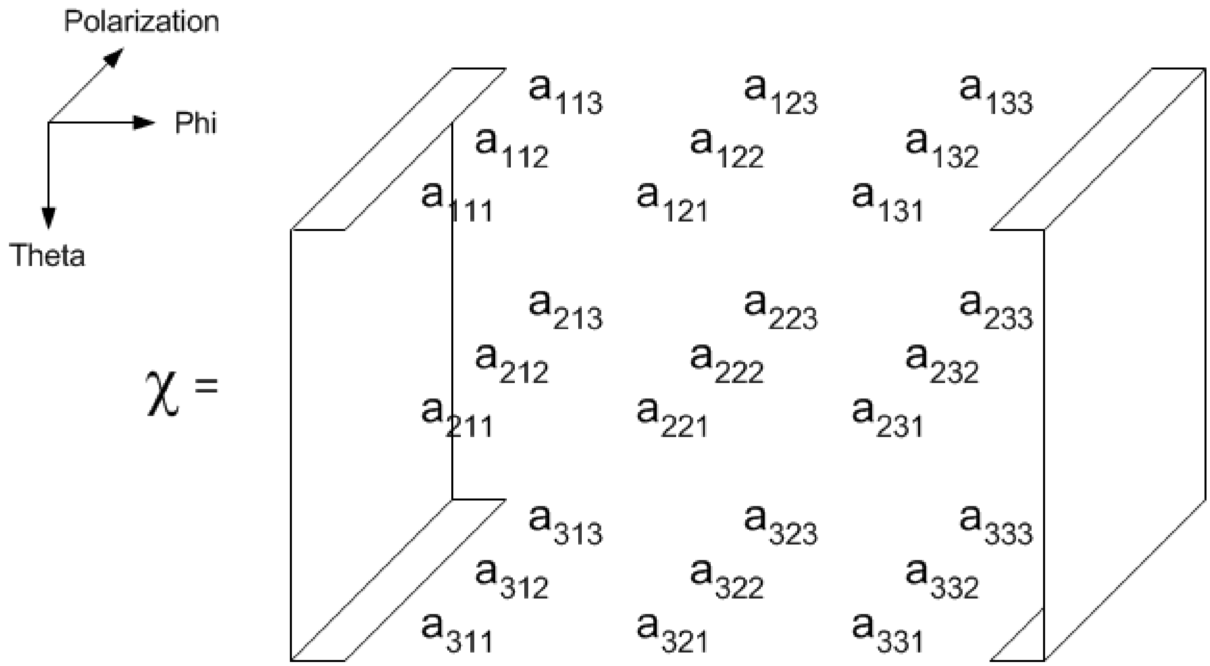

Figure 29 - Pictorial representation of the chi matrix

The rows of the χ matrix represent different values of theta, the columns are the values of phi, and the depth represents a change in psi or polarization. The three-dimensional matrix is a discretized representation of the gain and polarization data. This is generally the case for gain measurements and simulations, which are done in discrete steps of theta and phi. However, the polarization function is continuous in nature, but can easily be adapted to discrete angles. As was shown in Section 3.1.1, the polarization data is the same for all four quadrants and, thus only ninety degrees must be considered. Therefore, the gain/polarization matrix will have dimensions of $180 \times 360 \times 90$ for a one-degree resolution. The polarization data represented by the depth of the matrix will range from the unmodified three-dimensional gain to a matrix with all zeros in steps determined by the cosine-squared function or PLF previously described.

Using the gain/polarization matrix, it becomes possible to solve all gain and polarization combinations at a single point. Because of its widespread use, the optimum point will be chosen. Due to the fact that the tag is at the optimum position, the transmitter gain is at a maximum and (4.2) can be simplified. The result is shown in the following equation.

$$\kappa = \frac{P_T G_{T-MAX} \lambda^2}{(4\pi)^2}(1-|\Gamma_T|^2)(1-|\Gamma_R|^2) \tag{4.4}$$

The next step is to define values of the transmitter gain, transmitted power, wavelength, and minimum required tag power. These values are arbitrary and it will be shown that they will divide out in the end. Next, the distance is varied and the power received by the tag for all gains and polarizations is calculated using (4.3). The result is a three-dimensional matrix containing the power received by the tag in all the possible gains and polarizations. This matrix can easily be evaluated to see what percentage of the points is above the arbitrary minimum power level. The resulting percentage is the read accuracy for any orientation and polarization at the optimum point with respect to the interrogating transmitter. Dividing the maximum power received (optimum tag orientation and polarization) by the minimum power threshold leads to an interesting observation. This ratio, shown in (4.5), provides a scaling factor, S_f, which enables easy calculations for other points in space. A flow chart of the scaling factor method can be seen in Figure 30.

$$S_f = \frac{P_{R-MAX}}{P_{R-THRESHOLD}} \tag{4.5}$$

To clarify, if the minimum power level for tag operation is 50μW and 90% of the possible orientations and polarizations are above this level, the power received at the tag when in the optimum orientation and polarization, say 5mW, can be used to calculate a scaling factor of $\frac{5mW}{50\mu W}$ or 100. Using this realization, it can be seen that if the tag in this example receives 100 times the minimum operating power level at its optimum position, the read accuracy at that point will be 90% when examining all possible orientations and polarizations of the RFID tag.

Different read accuracies can be obtained by varying the distance in the arbitrary system. This will change the maximum power level while the threshold power level remains constant, thus changing the scaling factor. The system is arbitrary due to the ratio taken at the end of the process. The transmitter gain, transmitter power, wavelength, and reflection coefficients simply divide out as shown in (4.6). Therefore, the scaling factor is independent of these values and can be determined using only the gain and polarization data of the tag antenna.

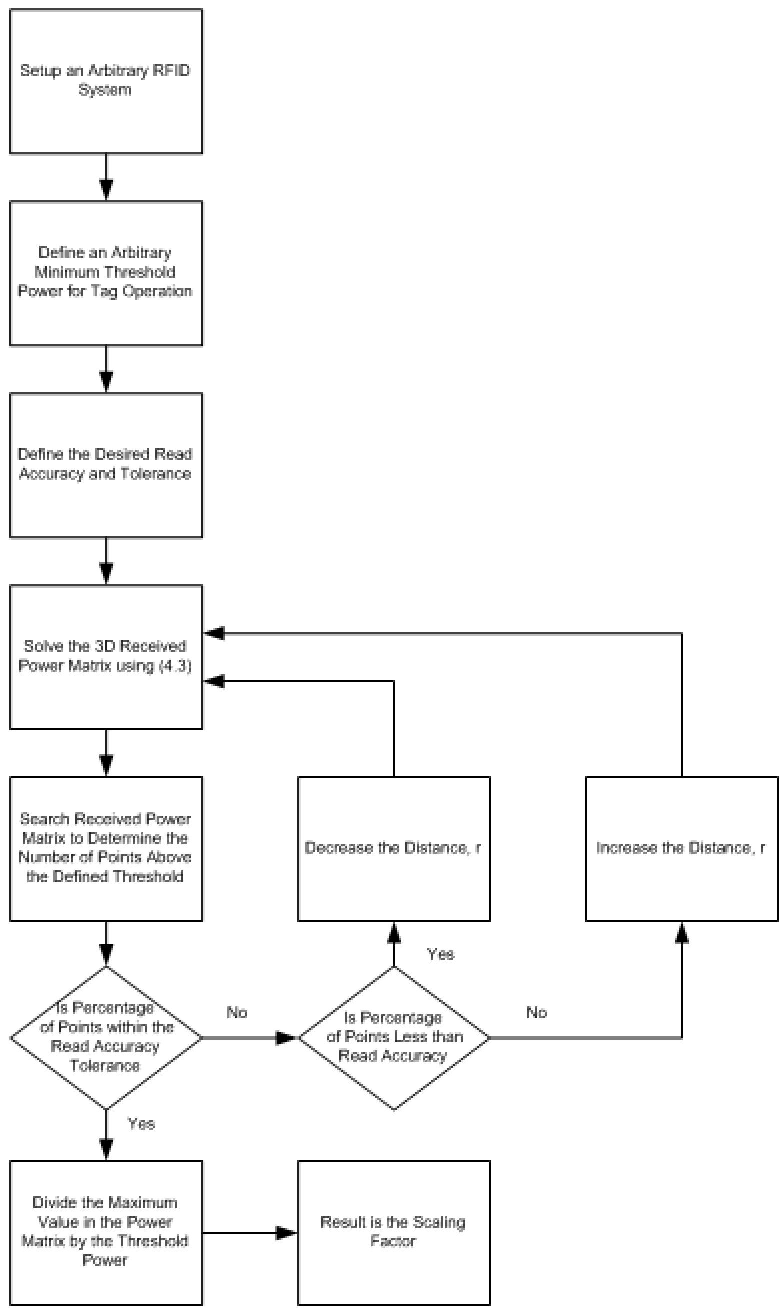

Figure 30 - Flow Chart of the Scaling Factor Method

$$S_f = \frac{P_{R-MAX}}{P_{R-THRESHOLD}} = \frac{\frac{P_T G_T \lambda^2}{(4\pi r)^2}(1-|\Gamma_T|^2)(1-|\Gamma_R|^2) G_{TAG-MAX}}{\frac{P_T G_T \lambda^2}{(4\pi r)^2}(1-|\Gamma_T|^2)(1-|\Gamma_R|^2) G_{TAG}(\theta_P,\phi_P)\cos^2(\psi)} \tag{4.6}$$

$G_{TAG}(\theta_P,\phi_P)$ and $\cos^2(\psi)$ are the values of the gain and PLF of the tag antenna that provide the arbitrary power level to the tag and will vary in order to hold the threshold power level constant.

The minimum power level is also arbitrary for a given read accuracy. As an example, if it is desired to have a read accuracy of 90% and the minimum power level is reduced to 25μW in the previous example, the scaling factor will double to 200 but the read accuracy will be higher than 90%. The distance must then be increased which will in turn reduce the power received in the optimum position and the scaling factor will return to the value calculate above of 100.

Take the simple two-dimensional example of a half-wave dipole antenna gain as shown in Table 1. If this gain data is used in conjunction with the following system parameters, the power received by the RFID tag is as shown in Table 2.

$$\begin{aligned} P_T &= 1 \text{ Watt} \\ G_T &= 4 \\ \lambda &= 0.328 \text{ meters} \\ r &= 1 \text{ meter} \\ \Gamma_{T,R} &= 0 \end{aligned} \tag{4.7}$$

Table 1 - 2D gain data for a half-wave dipole antenna

Theta/Phi	0	20	40	60	80	100	120	140	160	180	200	220	240	260	280	300	320	340	360
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03
20	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13
30	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29
40	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
50	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79
60	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09
70	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37
80	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57
90	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64	1.64
100	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57	1.57
110	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37	1.37
120	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09	1.09
130	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79
140	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51
150	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29	0.29
160	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13
170	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03
180	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 2 - Power received by the RFID tag in microwatts

Theta/Phi	0	20	40	60	80	100	120	140	160	180	200	220	240	260	280	300	320	340	360
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84
20	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
30	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780
40	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396
50	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157
60	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979
70	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736
80	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274
90	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469
100	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274
110	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736
120	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979
130	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157
140	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396
150	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780
160	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
170	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84
180	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

If the arbitrary power threshold is set at 85 microwatts, approximately seventy-nine percent of the orientations will receive enough power for the tag to operate, and the resulting power-scaling factor will be $\frac{4469\mu W}{85\mu W}$ or 53. The powered positions are shown in Table 3 where the non-powered positions have been colored black.

Table 3 - Powered and non-powered positions for the RFID system

Theta/Phi	0	20	40	60	80	100	120	140	160	180	200	220	240	260	280	300	320	340	360
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84
20	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
30	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780
40	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396
50	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157
60	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979
70	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736
80	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274
90	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469
100	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274
110	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736
120	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979
130	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157
140	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396
150	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780
160	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
170	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84
180	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

To demonstrate why the power threshold is arbitrary, consider a new threshold of 343 microwatts while specifying a read accuracy of seventy-nine percent, as was the previous case. The new threshold level means the data in Table 3 is no longer valid. Table 4 shows the new data for the parameters given in (4.7) and a threshold of 343 microwatts.

As can be seen by examining Table 4, the RFID system using the parameters in (4.7) yields a read accuracy of only sixty-eight percent, which is below the required value of seventy-nine. Therefore, the distance, r, must be decreased in order to increase the read accuracy. If the distance is adjusted from one meter to 0.497 meters, the power received by the tag is increased, which in turn, increases the read accuracy to seventy-nine percent. This is shown in Table 5.

Table 4 - Powered and non-powered positions for the RFID system with a larger power threshold

Theta/Phi	0	20	40	60	80	100	120	140	160	180	200	220	240	260	280	300	320	340	360
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84
20	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
30	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780
40	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396
50	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157
60	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979
70	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736
80	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274
90	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469	4469
100	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274	4274
110	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736	3736
120	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979	2979
130	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157	2157
140	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396	1396
150	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780	780
160	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
170	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84	84
180	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 5 - Powered and non-powered positions for the RFID system with a larger power threshold at a distance of 0.497 meters

Theta/Phi	0	20	40	60	80	100	120	140	160	180	200	220	240	260	280	300	320	340	360
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
20	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384
30	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158
40	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653
50	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730	8730
60	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062
70	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124
80	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302
90	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093	18093
100	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302	17302
110	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124	15124
120	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062	12062
130	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731	8731
140	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653	5653
150	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158	3158
160	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384	1384
170	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342	342
180	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

As Table 5 shows, the power-scaling factor for the increased threshold level is $\frac{18093\mu W}{343\mu W}$ or 53, which is the same as previously calculated. Therefore, the threshold level is an arbitrary value for a given read accuracy.

The technique developed through this research does have a large amount of overhead calculations. The user must specify the desired read accuracy and a computer must be used to search for the corresponding scaling factor. However, once this is done, a constant power contour around the interrogating transmitter can be plotted very quickly for a tag optimally aligned. The contour is at a power level of the scaling factor times the minimum power level needed by the tag. It should be noted that this constant power contour also represents a constant read accuracy contour. In other words, tags within the contour area are read at accuracies higher than the contour value and tags outside read at lower accuracies. The contour for a given scaling factor (corresponds to a read accuracy) can be calculated using the following equation.

$$r(\theta,\phi) = \frac{\lambda}{4\pi}\sqrt{\frac{P_T G_T(\theta_T,\phi_T) G_{TAG}(\max)}{S_f P_{\min}}(1-|\Gamma_T|^2)(1-|\Gamma_{TAG}|^2)} \qquad (4.8)$$

where

$G_T(\theta_T,\phi_T)$ is the 3D gain of the interrogating transmitter
$P_{\min}$ is the minimum power level required by the tag
S_f is the scaling factor for the desired read accuracy

The analysis given above has focused on providing operational power to the tag. As has been stated previously, the tag must be able to receive power and communicate to the base station. Therefore, it becomes important to develop a method for determining whether the tag can transmit its data back to the base station. The technique developed through this research can also be applied to the communicating link by the use of reciprocity. The receiving base station can be considered the transmitter and the tag can be defined as the receiver. This realization allows the scaling factor from the powering link to be used in order to reduce calculations and processing time. The reciprocal link transmits from the base station to the optimally aligned tag. The resulting power received at the tag is the scaling factor times the minimum sensitivity. Therefore, the minimum sensitivity required at the base station is the power received in the reciprocal link divided by the scaling factor. This is shown in (4.9).

$$S_{dBm}(\theta_T,\phi_T) = 10\log\left[\frac{1000P_{TAG}G_T(\theta_T,\phi_T)G_{TAG}(\max)\lambda^2}{S_f\left(4\pi r\right)^2}(1-\left|\Gamma_T\right|^2)(1-\left|\Gamma_{TAG}\right|^2)\right] \tag{4.9}$$

It should be noted that the analysis above is valid when the interrogating transmitter and data receiving base station are collocated. If the data base station is positioned away from the interrogator, functional points will be lost due to the fact that points that can communicate may not have received enough operational power and vice versa. The loss of data points leads to a read accuracy that is slightly less than the specified value. This deviation can be minimized by locating the base station outside the nulls of the tag antenna when it is optimally aligned with the interrogator. Additionally, if a symmetric tag antenna pattern is used, the base station can be located on the broadside axis without loss of data points.

5.0 DESIGN OF RFID EVALUATION TOOLS

5.1 SCALING FACTOR TOOL

One of the major focuses of the current research was to determine a method for solving an RFID system that did not require supercomputing capabilities. The proposed solution was to define a scaling factor that can be used to accurately model an RFID tag's three-dimensional operation by only examining its optimum positions and orientations. As was stated in the previous section, the scaling factor method requires a computer to search a very large three-dimensional matrix. In order to give a better understanding of how the method works and is implemented, an evaluation tool was developed using Matlab. The program is used to calculate the scaling factor for a user defined gain file for various read accuracies. A screen shot of the GUI can be seen in the following figure. The Matlab code is presented in Appendix D.

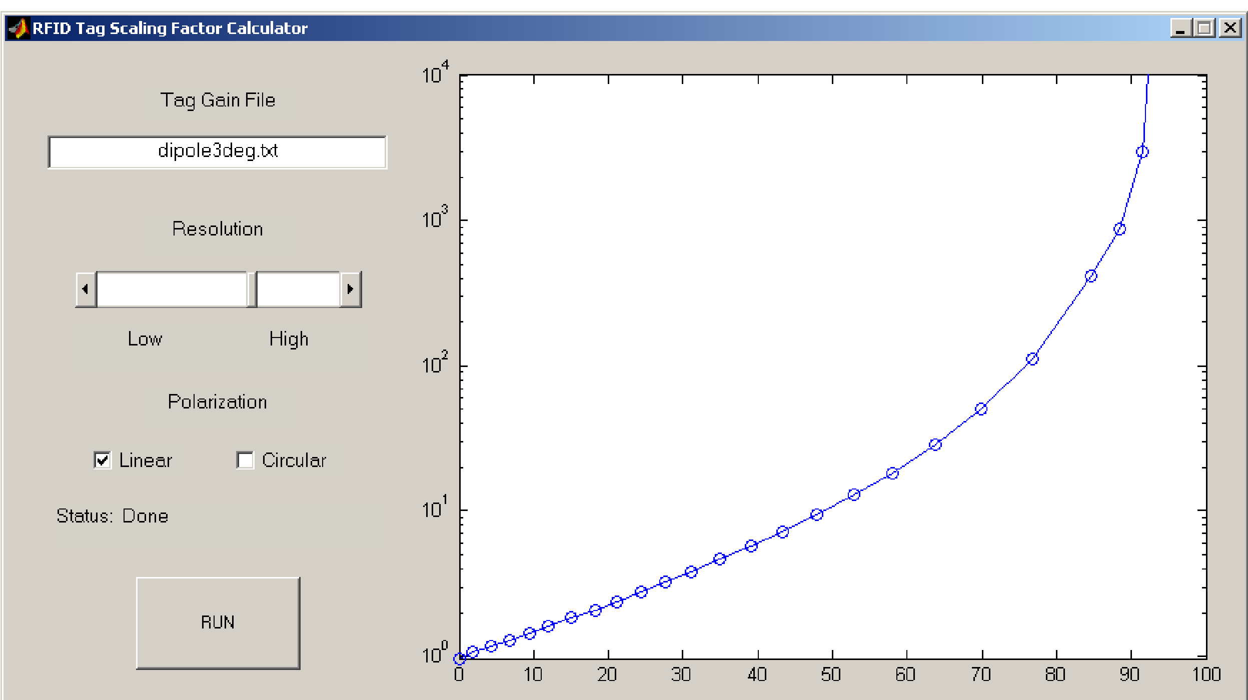

Figure 31 - Scaling Factor Evaluation Tool

As can be seen from Figure 31, the evaluation tool plots the scaling factor for various accuracies. The program also outputs the data points to the main Matlab window to allow user manipulation. An example can be seen in Table 6 where the "Actual Accuracy" and "Scaling Factor" columns have been provided by the evaluation tool. The example is based on the parameters and calculation shown in Appendix A. As can be seen in Table 6, the lowest read accuracy, 0.5%, corresponds to a maximum read range of 9.38 meters. If the limit of the maximum read range is taken as the read accuracy approaches zero, the maximum read range would approach the calculated value of 9.45 meters shown by the calculations in Appendix A. However, for higher read accuracies the maximum read range is reduced. This can be seen in Figure 32. As an example, a read accuracy of 80% reduces the maximum read range to 0.73 meters. The maximum read range for a given accuracy can be calculated using the following equation.

$$r_{\max} = \frac{\lambda}{4\pi}\sqrt{\frac{P_T G_T(\max) G_{TAG}(\max)}{S_f P_{\min}}} \tag{5.1}$$

where S_f is the scaling factor.

It should be noted that the maximum distance has also been calculated in terms of lambda to insure the calculations are being performed in the far-field which was discussed in Section 2.4.5. The far-field boundary for a half-wave antenna given by (2.6) is a half wavelength. Therefore, the read accuracy of 92.5 in Table 7 is actually in the near-field and the corresponding scaling factor will be lower than the calculated one. This is due to the fact that the antenna will also be receiving reactive and near-field radiated power.

Table 6 - Scaling Factor Data for a Half-wave Dipole for Various Accuracies

Target Accuracy (%)	Actual Accuracy (%)	Scaling Factor	Maximum Distance (m) (min 50uW 915MHz)	Maximum Distance (lambda) (min 50uW 915MHz)
0.5	0.48	1.02	9.38	28.62
1	0.95	1.03	9.31	28.39
5	5.02	1.21	8.60	26.23
10	10.05	1.48	7.78	23.71
15	14.97	1.82	7.01	21.39
20	20.04	2.27	6.28	19.15
25	24.96	2.86	5.59	17.06
30	30.04	3.60	4.99	15.21
35	34.96	4.65	4.39	13.38
40	39.98	6.01	3.86	11.77
45	45.06	7.92	3.36	10.25
50	49.97	10.82	2.88	8.77
55	55.00	15.23	2.42	7.39
60	59.97	22.58	1.99	6.07
65	64.99	33.04	1.65	5.02
70	70.02	50.81	1.33	4.05
75	75.04	91.95	0.99	3.01
80	80.01	169.50	0.73	2.22
82.5	82.55	270.58	0.58	1.75
85	85.03	457.28	0.44	1.35
87.5	87.47	696.97	0.36	1.09
90	92.54	1484.56	0.25	0.75
92.5	90.01	7938.93	0.11	0.32

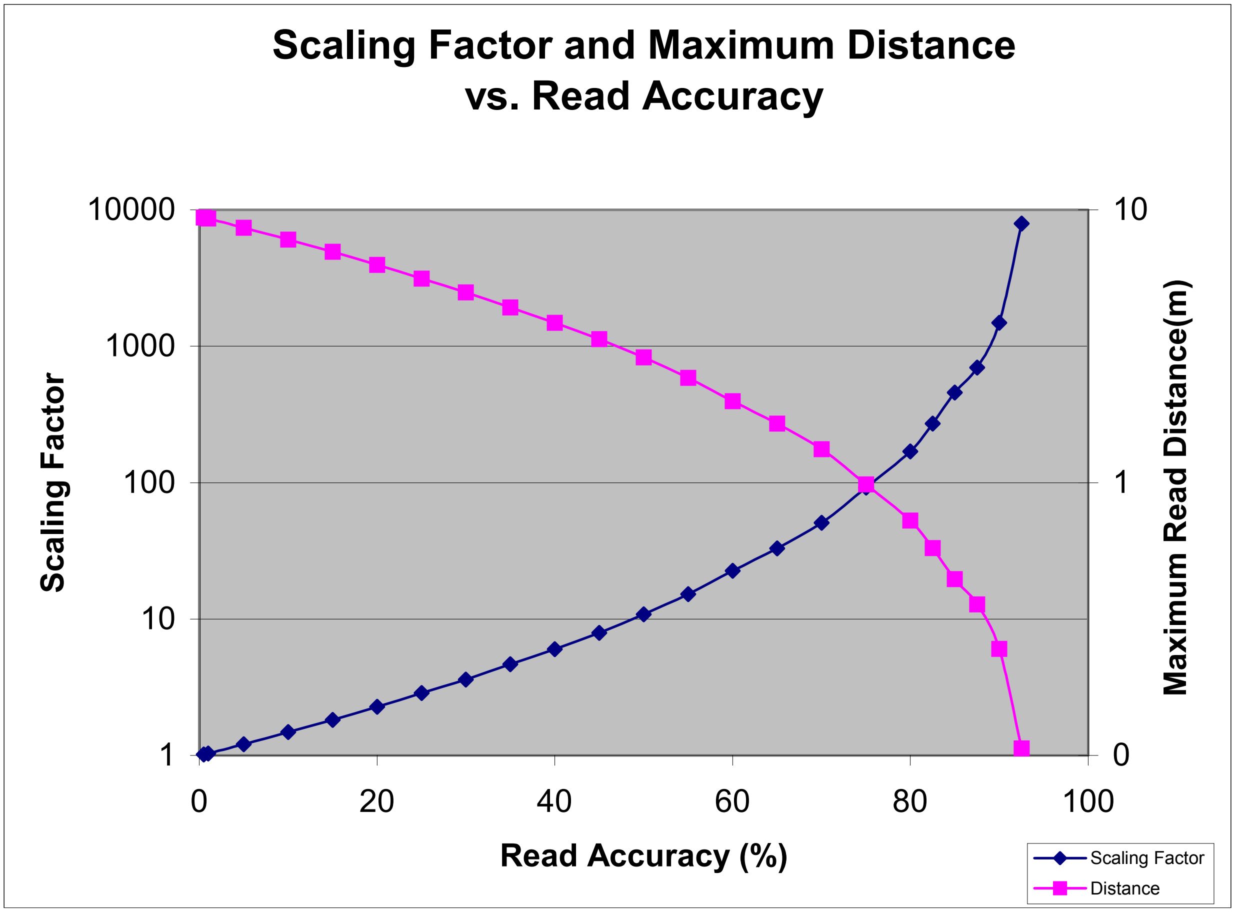

Figure 32 - Scaling Factor and Maximum Read Range Versus Read Accuracy for a Half-wave Dipole

One important question raised in Section 3.2.5, was related to the resolution of the gain file. The higher the resolution, the larger the matrix, and therefore, the calculations become slower. To help answer this question, the equation derived for a half-wave dipole in Section 3.2.3 was used to generate three-dimensional gain data for various resolutions. The resulting scaling factors for these gain resolutions can be seen in Figure 33.

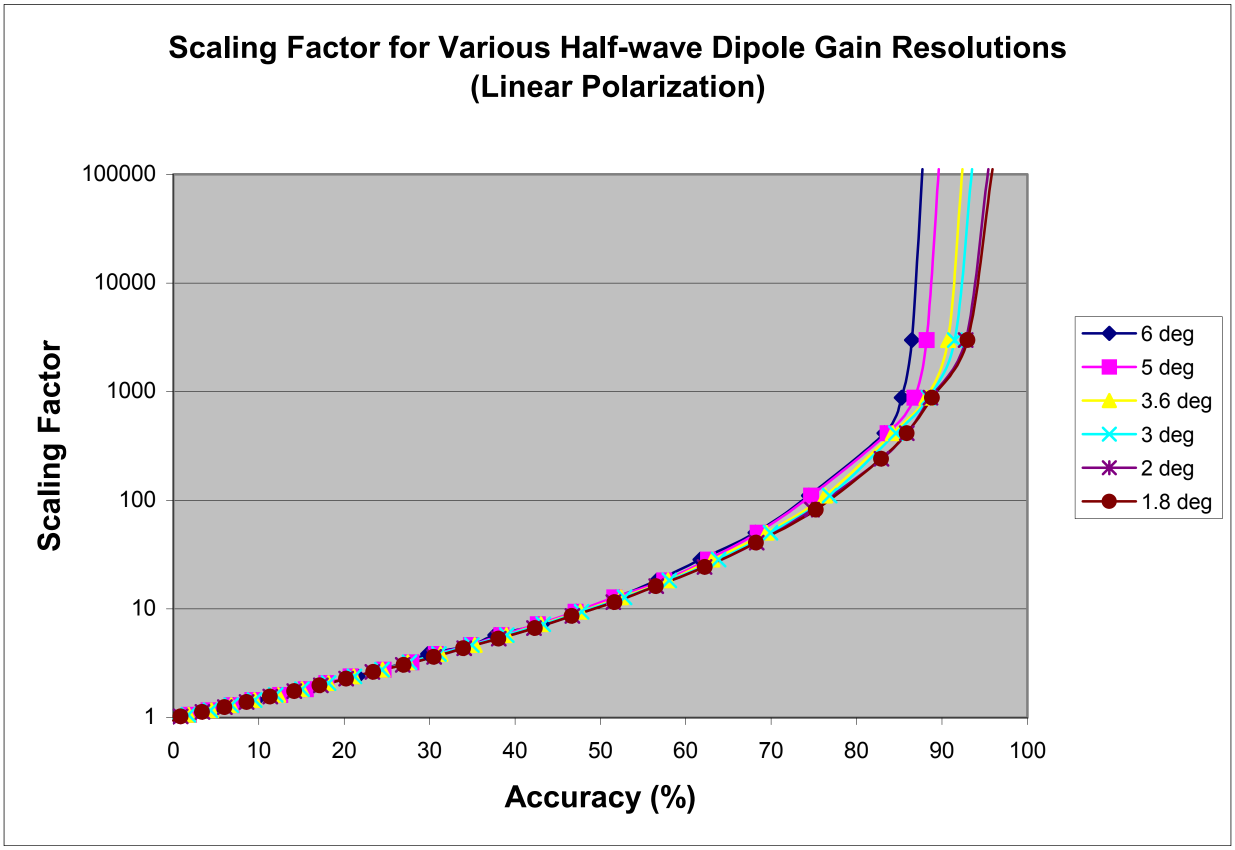

Figure 33 - Scaling Factor for Various Half-wave Dipole Gain Resolutions

As Figure 33 shows, the scaling factors are very similar at the lower read accuracies. However, at high accuracies, the data begins to spread. The graph shows that resolutions above three degrees show a high degree of error although those at and below three degrees are fairly similar. Therefore, the results would suggest that for a half-wave dipole antenna the gain file should be at a resolution of three degrees or less.

Another feature included in the scaling factor evaluation tool due to its popularity in RFID systems is circular polarization on the transmitted power. Circular polarization eliminates the polarization dependence on the tag, which results in a gain/polarization matrix that has constant gain throughout the depth of the matrix. This essentially means the polarization data or the third dimension can be removed from the matrix meaning the computer need only search a two-dimensional matrix. This will significantly increase the speed of the program for circularly polarized systems. To illustrate the effects of circular polarization, the graph of various half-wave dipole gain resolutions shown in Figure 33 was repeated. As can be seen in Figure 34, the scaling factor is rather jagged and has a stair-step fashion. This is caused by the limited number of different valued data points due to the lack of dependence on polarization. Examining the figure in detail, it can be seen that as the resolution is increased, the data becomes smoother due to the increase in data values.

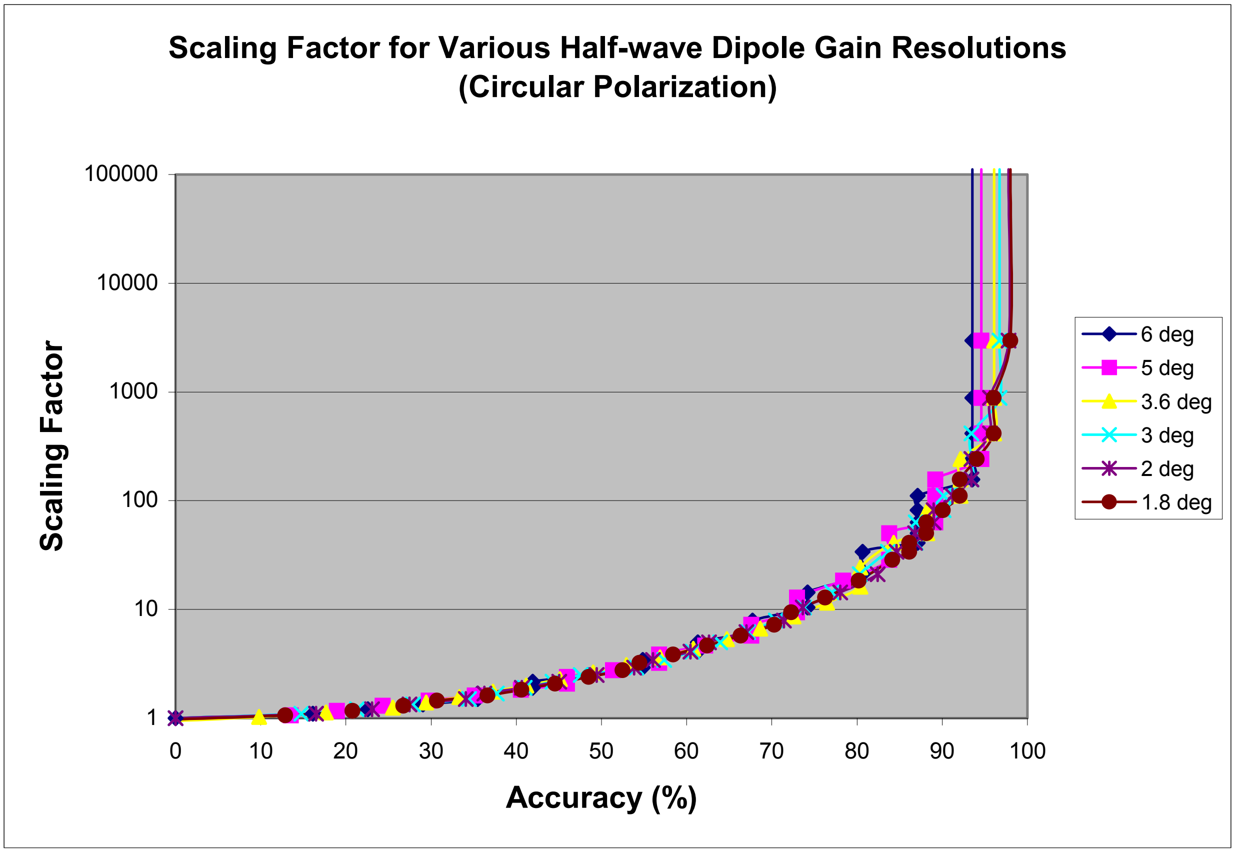

Figure 34 - Scaling Factor for Various Half-wave Dipole Gain Resolutions with Circular Polarization

As with the linear polarization case, the output data can be manipulated in order to calculate the maximum read range for a specific accuracy. The data can be seen in Table 7 and Figure 35.

Table 7 - Scaling Factor Data for a Half-wave Dipole for Various Accuracies with Circular Polarization

Target Accuracy (%)	Actual Accuracy (%)	Scaling Factor	Maximum Distance (m) (min 50uW 915MHz)	Maximum Distance (lambda) (min 50uW 915MHz)
0.5	0.423	1.006	9.430752242	28.76379434
1	0.952	1.016	9.384226294	28.6218902
5	5.077	1.099	9.022906971	27.51986626
10	10.048	1.21	8.599092898	26.22723334
15	15.019	1.333	8.192760348	24.98791906
20	20.095	1.476	7.785780346	23.74663006
25	25.066	1.644	7.377248611	22.50060826
30	29.931	1.819	7.013405933	21.39088809
35	34.955	2.019	6.656978686	20.30378499
40	40.032	2.307	6.22761174	18.99421581
45	45.003	2.716	5.739584355	17.50573228
50	49.974	3.329	5.184279747	15.81205323
55	55.05	4.055	4.697317292	14.32681774
60	59.915	5.175	4.158054227	12.68206539
65	64.992	7.109	3.547652464	10.82034002
70	70.069	9.481	3.071979071	9.369536168
75	75.04	14.199	2.51024694	7.656253167
80	79.905	23.62	1.946280266	5.936154812
82.5	82.549	32.345	1.663189552	5.072728134
85	84.982	42.247	1.455282774	4.438612459
87.5	87.5	66.617	1.158918219	3.534700569
90	90.058	124.046	0.849285958	2.590322172
92.5	92.491	197.923	0.672352762	2.050675924
95	94.923	663.786	0.367139643	1.119775912

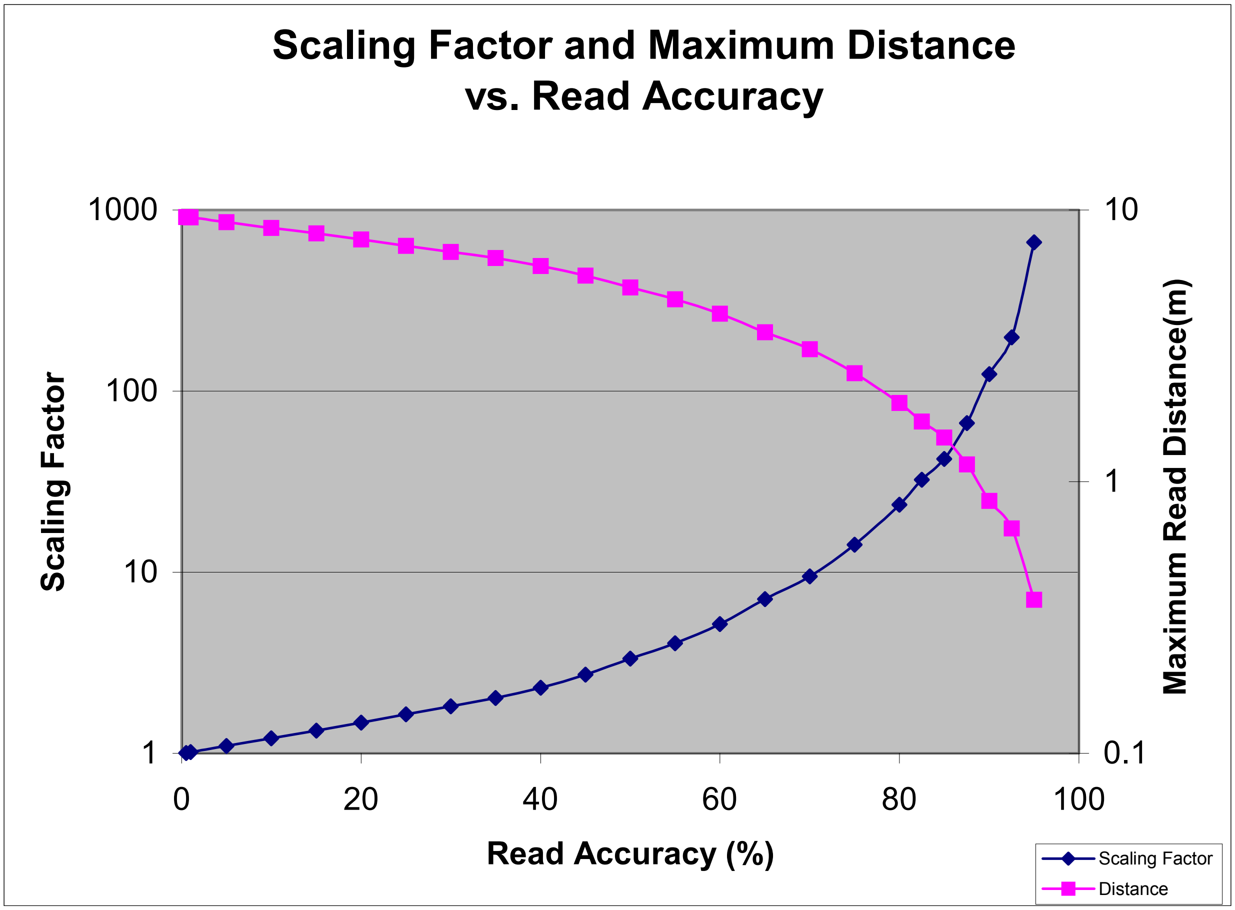

Figure 35 - Scaling Factor and Maximum Read Range Versus Read Accuracy for a Half-wave Dipole using Circular Polarization

The elimination of the polarization dependence will significantly decrease the value of the scaling factor leading to an increase in the read range for a given accuracy. A comparison of linear and circular polarization scaling factors can be seen in the following figure. Additionally, the maximum read ranges are compared in Figure 37.

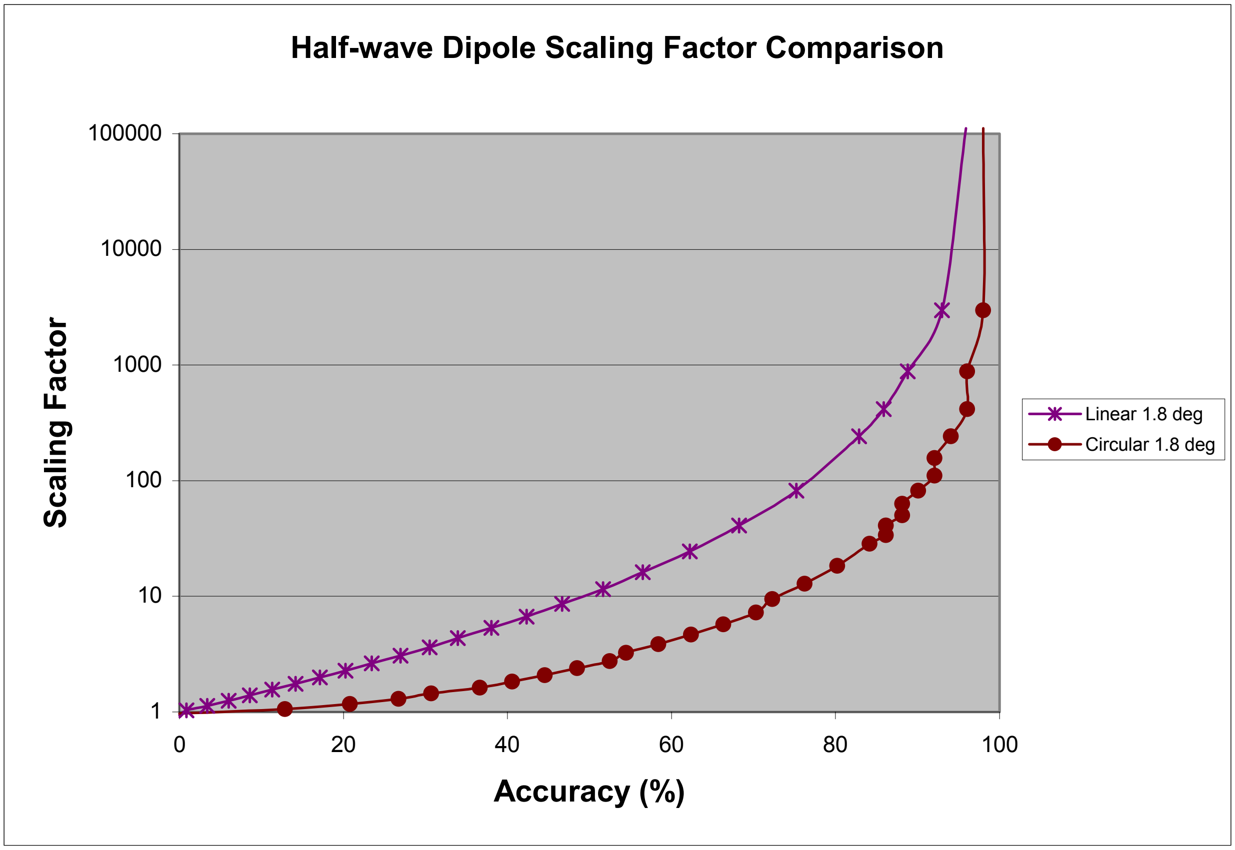

Figure 36 - Comparison of Linear and Circular Polarization Scaling Factors

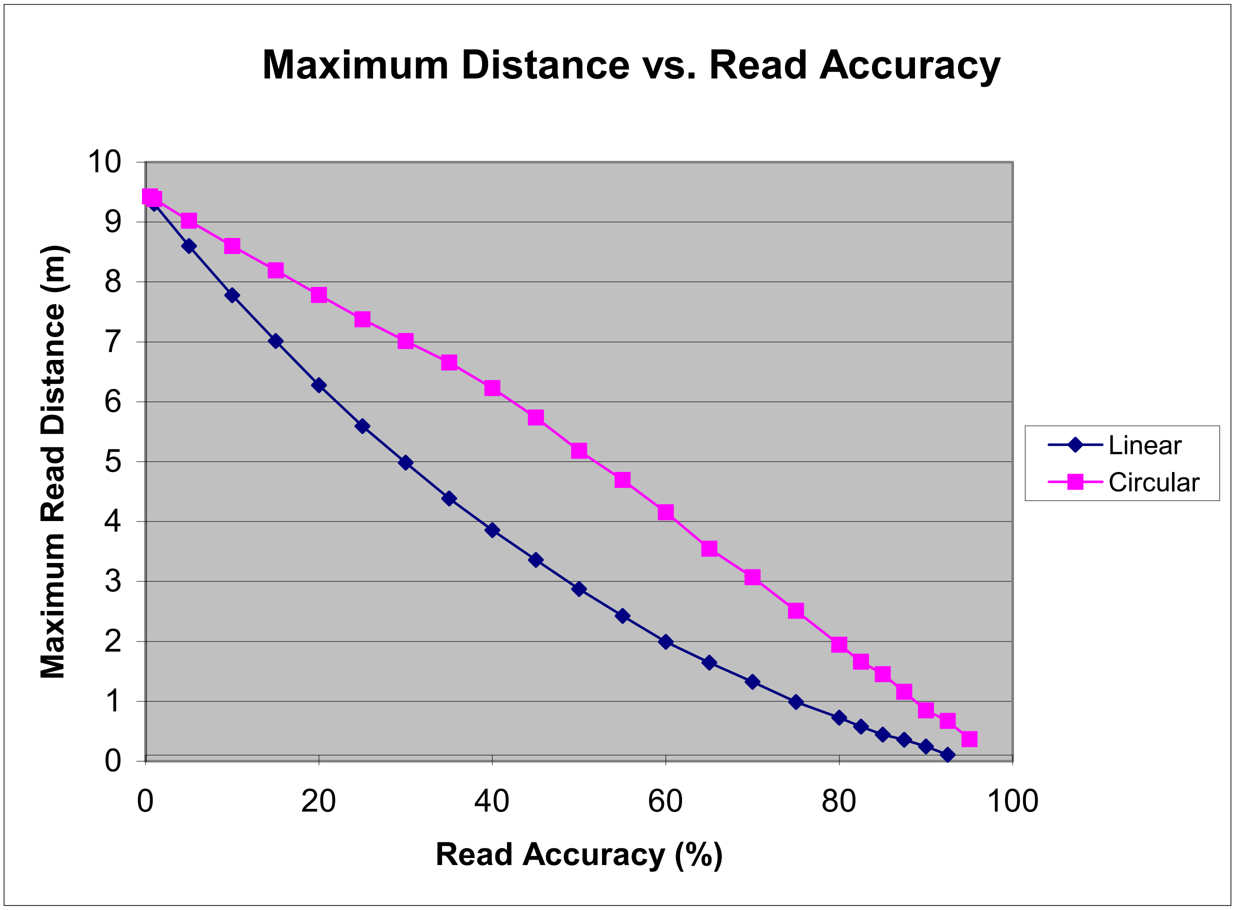

Figure 37 - Comparison of Linear and Circular Polarization Read Ranges

The scaling factors presented so far have been purely theoretical and have been limited by the fact that nulls in the gain pattern go completely to zero. The actual measured gain will have some finite value, which will allow those values to scale unlike a zero value. It should be noted that the theoretical results are limited by the number of zeros within the matrix. Measured results, however, are only limited by the amount of power that can be transmitted in order to obtain a required scaling factor. Figure 38 shows a comparison of the scaling factors for theoretical and measured gains of a half-wave dipole antenna. As can be seen, the scaling factors of the measured dipole are much less than those obtained from the theoretical results.

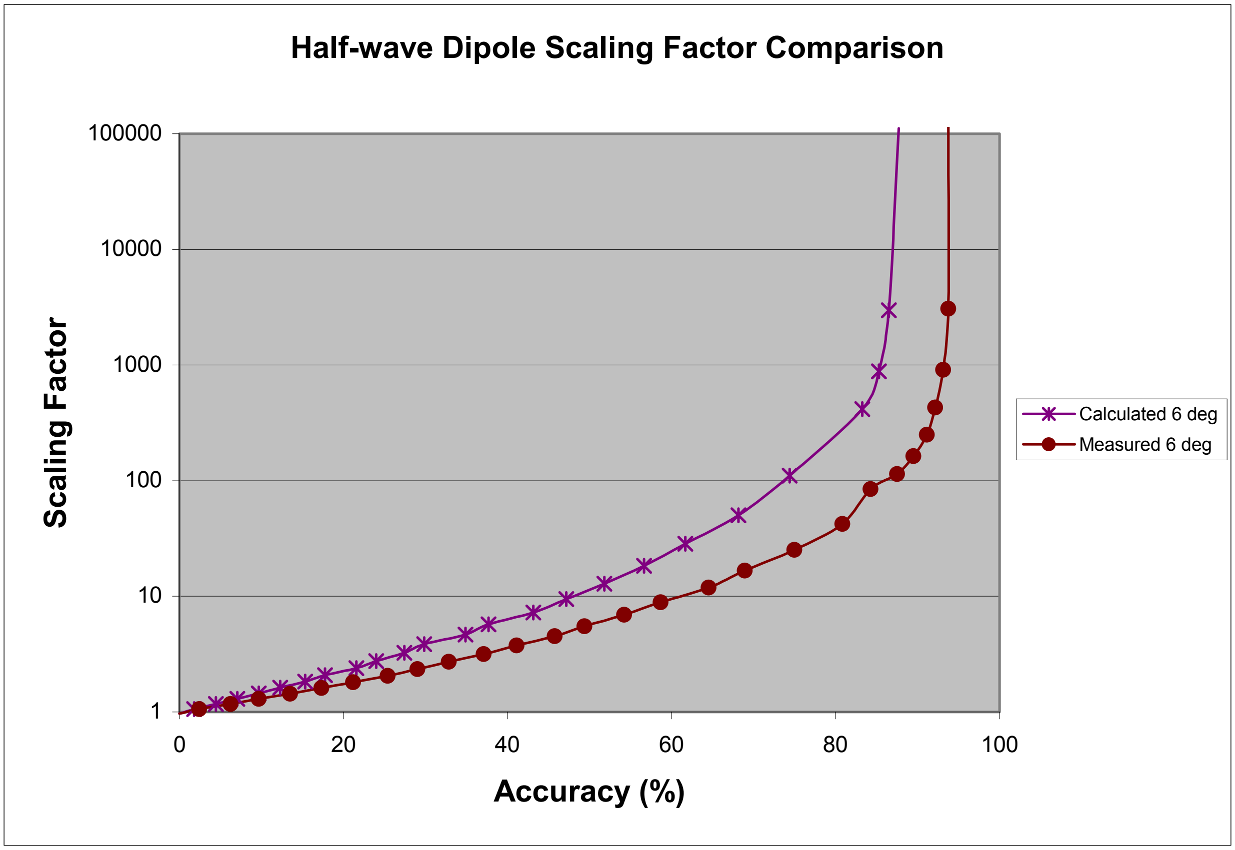

Figure 38 - Comparison of the Scaling Factors for Calculated and Measured Half-wave Dipole Gains

As the scaling factor evaluation tool has shown, the use of circular polarization on the incident wave significantly reduces the scaling factor, which extends the maximum read range at a specific accuracy. Additionally, the use of measured gain data further reduces the value of the scaling factor and increases the maximum read range due to the lack of zeros in the gain/polarization matrix. Therefore, the RFID system with the best read accuracy at a given distance will use circular polarization transmitted to an RFID tag employing an omnidirectional antenna with linear polarization.

5.2 ARS RFID SYSTEM TOOL

The development of the scaling factor method allows the implementation of an evaluation tool for the entire RFID system that can be executed on a standard computer. This was not previously an option due to the large memory and processing capabilities required for calculating and storing the system data.

The first RFID system that was implemented was the ARS method described in Section 2.3.3. The mathematical model for an ARS RFID system was developed in Section 3.4.1 and the equation is given in (3.158) and is repeated below for convenience.

$$P_R = \alpha P_T \frac{G_R(\theta_R,\phi_R)G_{TAG-TX}(\theta_{TAG},\phi_{TAG})G_T(\theta_T,\phi_T)G_{TAG-RX}(\theta_{TAG},\phi_{TAG})\lambda_R^2\lambda_T^2}{(4\pi)^4(r_1r_2)^2}$$
$$\times(1-|\Gamma_T|^2)(1-|\Gamma_R|^2)(1-|\Gamma_{TAG-TX}|^2)(1-|\Gamma_{TAG-RX}|^2)|\hat{\mathbf{p}}_R\bullet\hat{\mathbf{p}}_{TAG-TX}|^2|\hat{\mathbf{p}}_T\bullet\hat{\mathbf{p}}_{TAG-RX}|^2 \quad (5.2)$$

This mathematical model was implemented in Matlab in conjunction with the scaling factor method described in the Section 4. A screen shot of the ARS Evaluation Tool GUI can be seen in the following figure. The Matlab Code is given in Appendix E.

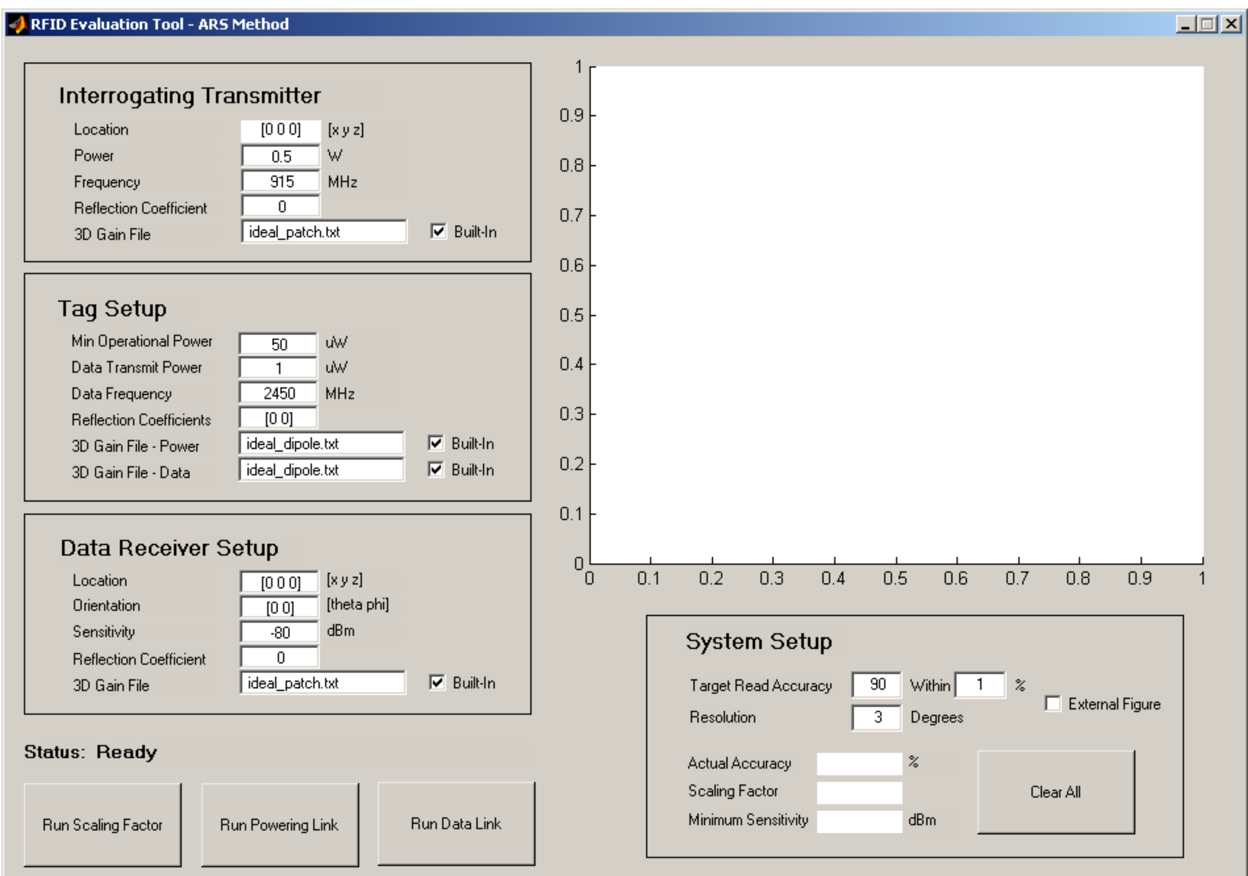

Figure 39 - Screen Shot of the ARS RFID Evaluation Tool

The evaluation tool requires detailed input from the user in order to accurately model the RFID system. The evaluation tool also outputs data that can be used to graphically and numerically evaluate the system performance. All of these parameters are outlined below for each of the four input sections.

Interrogating Transmitter

- **Location** - is a fixed value for the location of the interrogating transmitter. [0,0,0] is used as the reference location for the entire system.
- **Power** - is the power level of the interrogating transmitter. This is not the effective radiated power but the power delivered to the antenna.
- **Frequency** - is the fundamental frequency of the interrogating transmitter.
- **Reflection Coefficient** - is the reflection coefficient describing the impedance mismatch between the interrogating transmitter and the antenna.
- **3D Gain File** - is the three-dimensional gain file for the interrogating transmitter's antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave patch antenna by clicking the checkbox. The default is the built-in gain file. The resolution of a user defined gain file must match the resolution give in the System Setup box to avoid an error message.

Tag Setup

- **Min Operational Power** - is the minimum power required for the tag to operate its logic and transmitting circuits.
- **Data Transmit Power** - is the amount of power transmitted from the ARS RFID tag as data.
- **Data Frequency** - is the fundamental frequency used by the communication link.
- **Reflection Coefficients** - are the reflection coefficients describing the impedance mismatch between the tag's power antenna and rectification circuitry and the mismatch between the data transmitting circuit and the data antenna. The first entry in the vector is the powering mismatch while the second is the communicating mismatch.

- **3D Gain File - Power** - is the three-dimensional gain file for the tag's power antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave dipole antenna by clicking the checkbox. The default is the built-in gain file. The resolution of a user defined gain file must match the resolution given in the System Setup box to avoid an error message.
- **3D Gain File - Data** - is the three-dimensional gain file for the tag's data antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave dipole antenna by clicking the checkbox. The default is the built-in gain file. The resolution of a user defined gain file must match the resolution given in the System Setup box to avoid an error message.

Data Receiver Setup

- **Location** - is the location of the data receiving base station with respect to the interrogating transmitter location of [0,0,0].
- **Orientation** - is the orientation in degrees of the base station antenna with respect to the interrogating transmitters antenna. Theta is the rotation of the z-axis in the x-z plane. Phi is the rotation of the x-axis in the x-y plane. The user must use caution in specifying the orientation of the data receiver to insure that the polarization of the data receiver's antenna is the same as the interrogating transmitter's antenna to avoid erroneous results.
- **Sensitivity** - is the sensitivity of the data receiver used to determine whether the tag can transmit sufficient power to communicate.
- **Reflection Coefficient** - is the reflection coefficient describing the impedance mismatch between the data receiver and the antenna.
- **3D Gain File** - is the three-dimensional gain file for data receiver's antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave patch antenna by clicking the checkbox. The default is the built-in gain file. The resolution of a user defined gain file must match the resolution given in the System Setup box to avoid an error message.

System Setup

- **Target Read Accuracy** - is the read accuracy for which the user wishes to plot a constant read accuracy contour. The user can also specify a range used by the program when searching for the target accuracy.
- **Resolution** - is the resolution used by the program when calculating and plotting the data. Finer resolutions increase the number of data points exponentially and therefore increase the processing time. The resolution specified here must match the resolution of any user defined gain files to avoid an error message.
- **External Figure** - is used to plot the data in an external window for user manipulation.
- **Actual Accuracy** - is an output from the program specifying the actual accuracy, within the user-defined range, of the contour that will be plotted.
- **Scaling Factor** - is an output from the program specifying the power-scaling factor that is required to plot the constant accuracy contour.
- **Minimum Sensitivity** - is the minimum sensitivity needed by the data receiver in order to read all tag positions and orientations at the specified target read accuracy.

Once the user has entered the system parameters, the evaluation tool is ready for operation. The first step is to click the "Run Scaling Factor" button that enables the built-in scaling factor tool that was described in the previous section. The program will search for the required Target Read Accuracy and will display the Actual Accuracy and Scaling Factor when completed. Additionally, the interrogating transmitter's antenna will be added to the plot with the X- and Z-axes for reference. This is shown in the following figure.

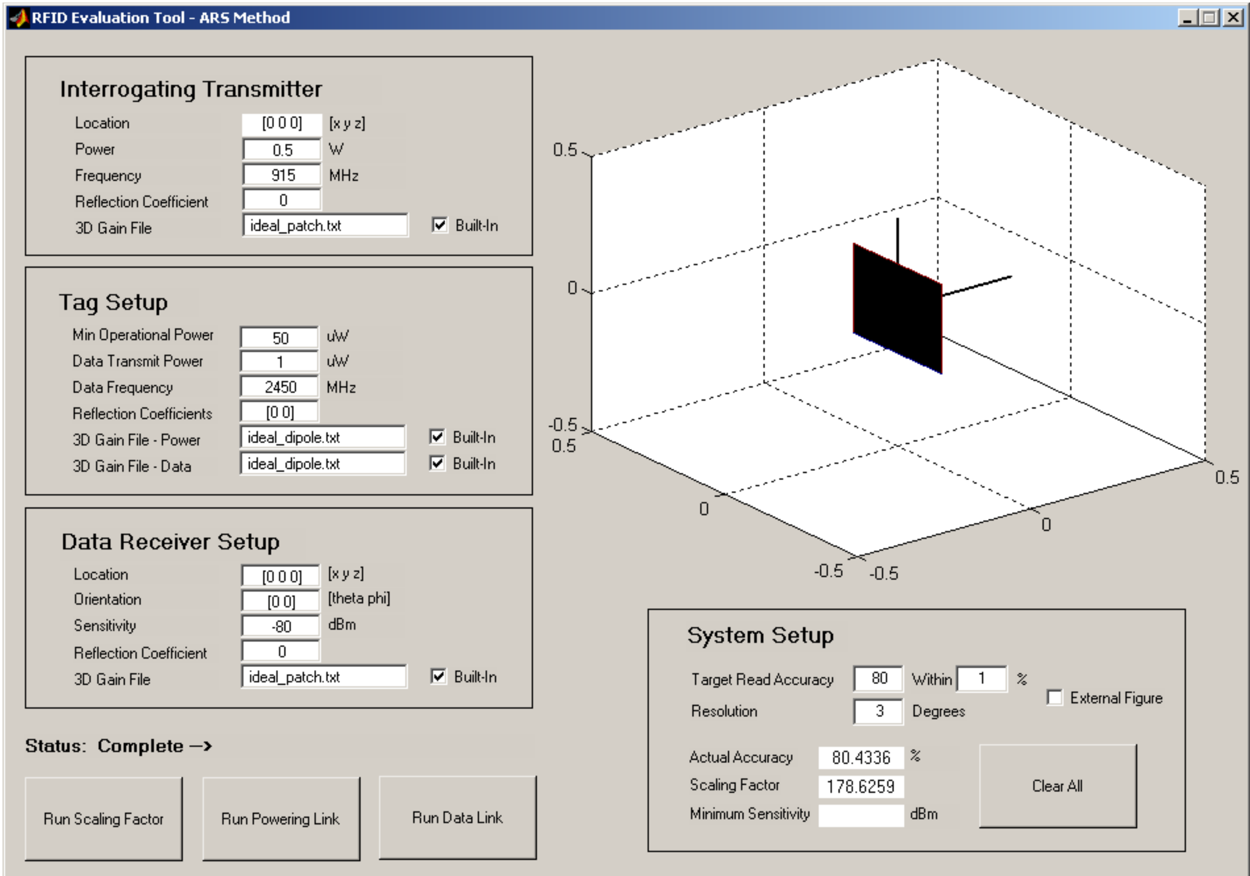

Figure 40 - Screen Shot of the ARS RFID Evaluation Tool after Running the Scaling Factor

As can be seen from Figure 40, the actual accuracy is slightly more than 80% but is within the 1% requirement. The calculated scaling factor is approximately 179 times the minimum operational power. In this case, the minimum operational power was specified as 50µW meaning the tag must receive 179*50µW or 8.95mW of power when in its optimum position in order to read 80.43% of the orientations and polarizations. Using this information, the program can very quickly plot the constant read accuracy contour by simply calculating the contour where the interrogating transmitter can supply an optimally positioned tag 8.95mW of power. This step is done by clicking the "Run Powering Link" button. The output is shown in the following figure.

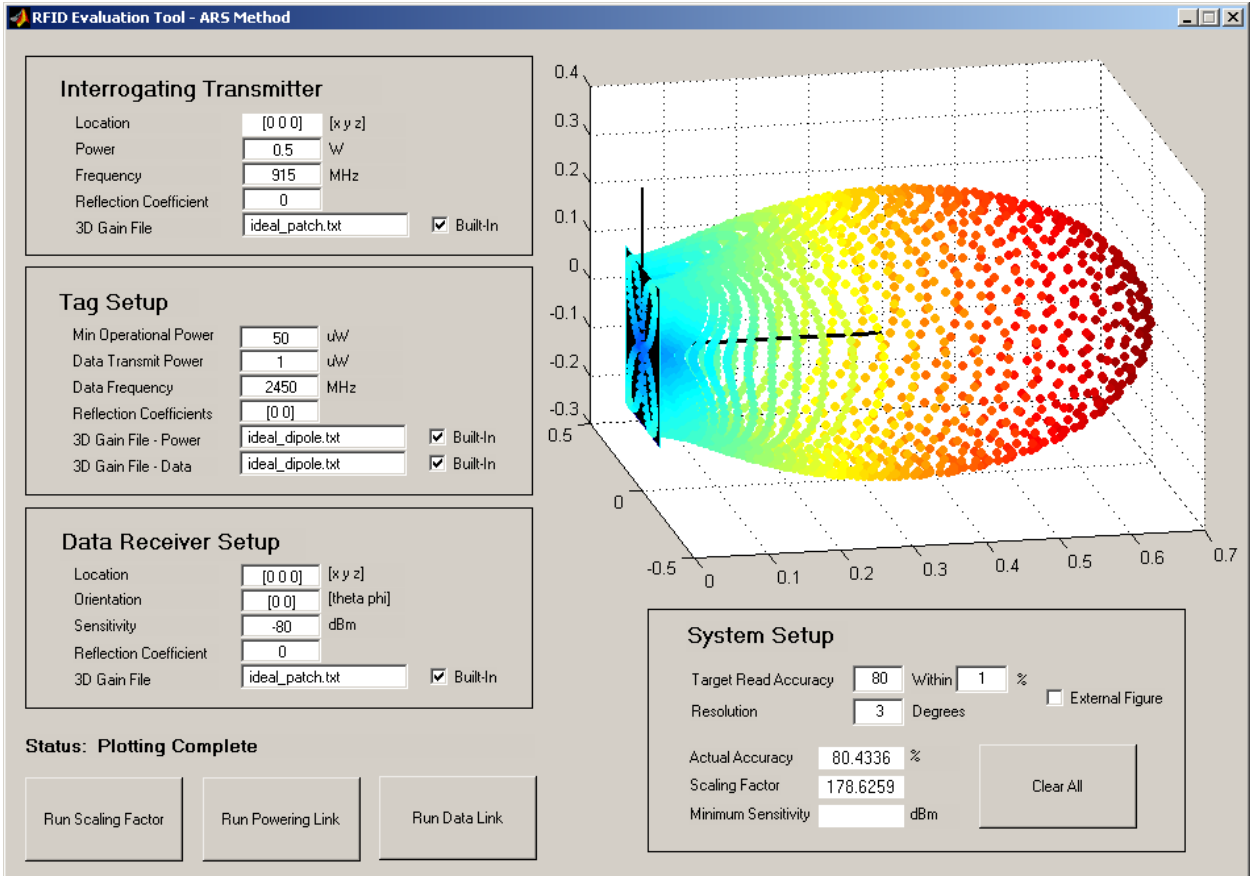

Figure 41 - Screen Shot of the ARS RFID Evaluation Tool after Running the Powering Link

As Figure 41 shows, the constant read accuracy contour extends to approximately 0.7 meters. The distance of the points are color-coded to enable clear viewing of all the data points. The program also allows rotation of the graphing window in order to see the area of operation from different angles. An end view can be seen in the following figure. As the figure shows, the width and height of the functional volume are approximately 0.5 meter each.

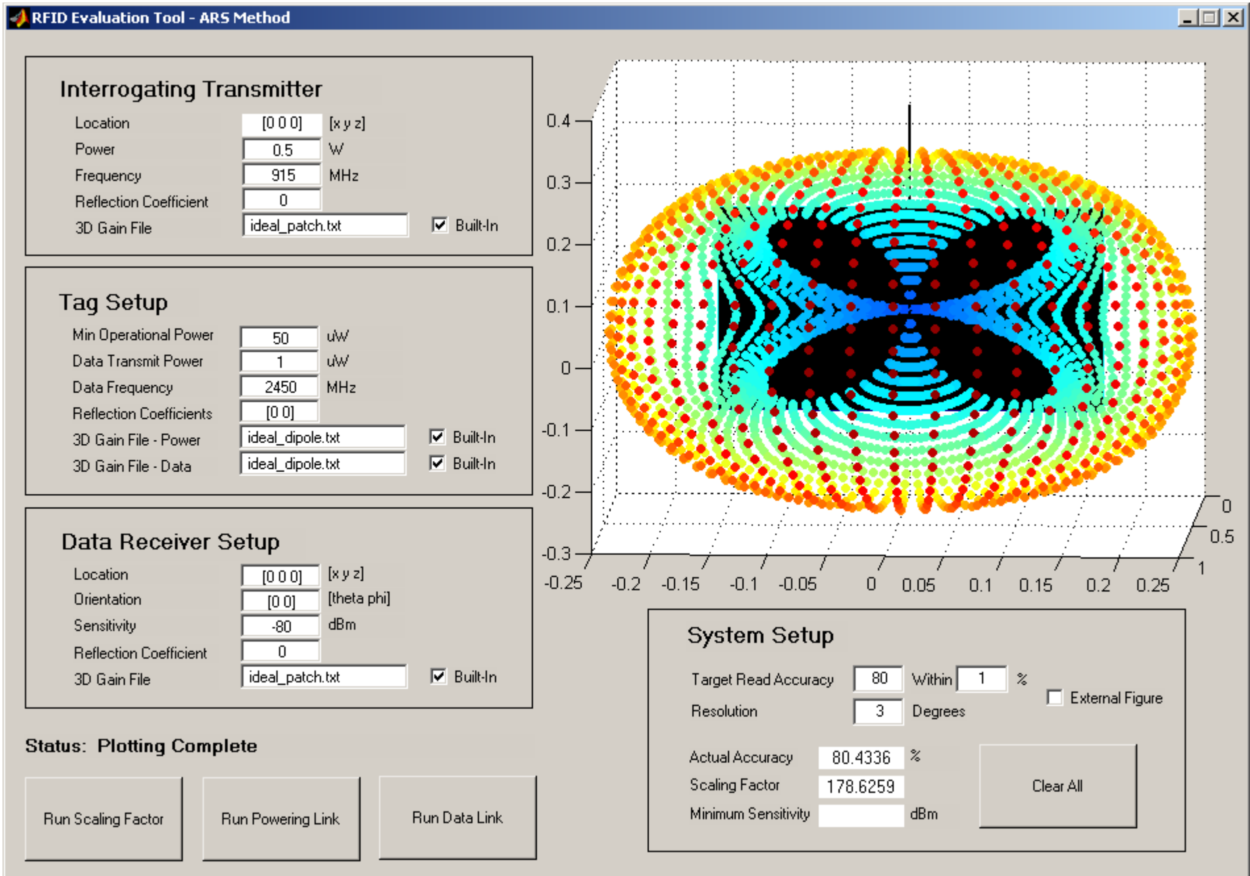

Figure 42 - Screen Shot of the ARS RFID Evaluation Tool after Running the Powering Link - Rotated

The final step in evaluating the performance of an RFID system is determining where the RFID tag can successfully transmit its data back to the receiving base station. This step is executed by clicking the "Run Data Link" button. The following figure shows the output from this step.

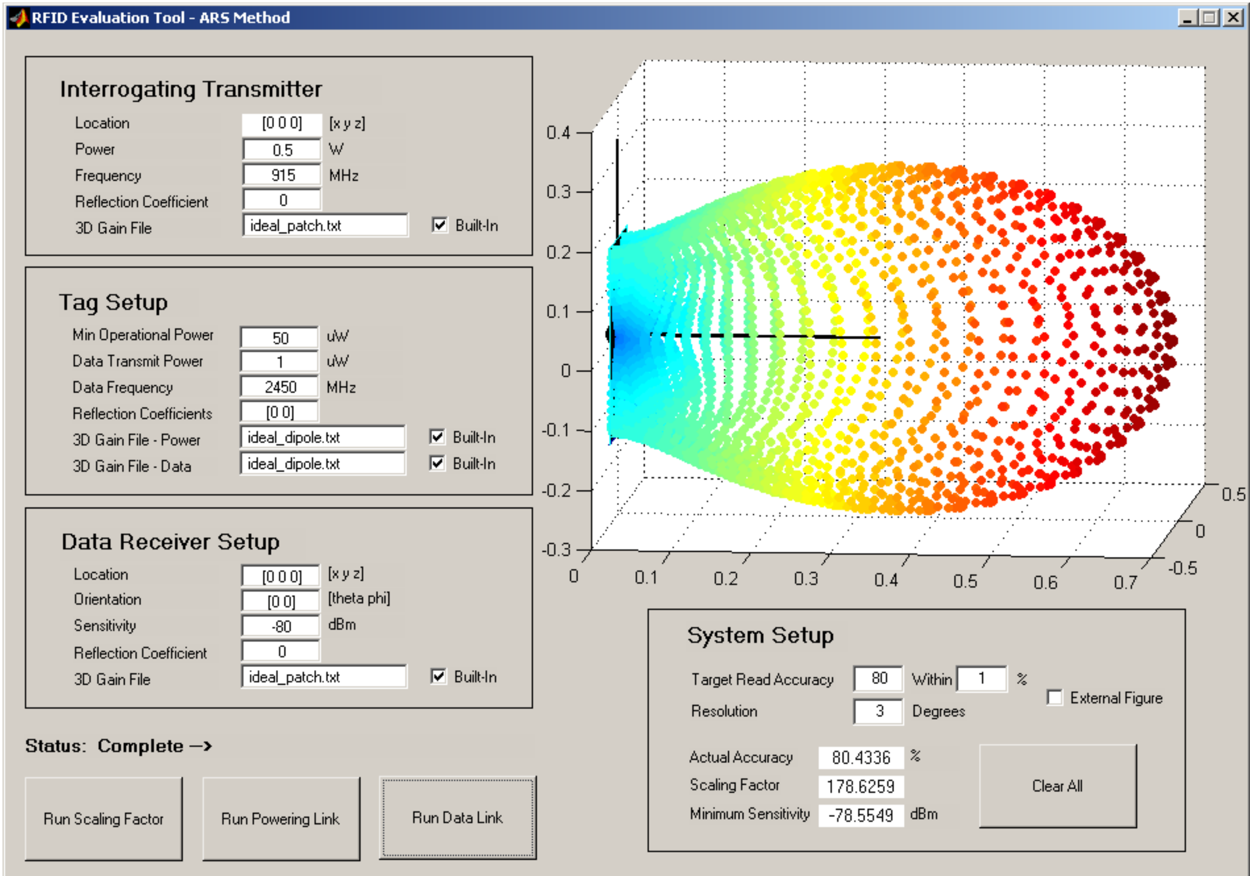

Figure 43 - Screen Shot of the ARS RFID Evaluation Tool after Running the Data Link

As shown in Figure 43, the program has returned a value of the minimum sensitivity. This means the data receiver must have a sensitivity less than -78.55dBm in order to read 80.43% of the data points. The program has also plotted the location of the data base station although it is not visible because its location is the same as the interrogating transmitter. This is not always advantageous and therefore it is possible to move its location. The following figure shows that the data receiver has been moved to [1 0 0] or at one meter on the X-axis. This is done by clicking the "Clear All" button and changing any desired system parameters. It should be noted that the orientation has been changed to have a phi of 180 degrees in order to inform the program that the maximum gain of the data receiver should point back at the interrogating transmitter rather than in the same direction. Additionally, the new minimum sensitivity is now less than the -80dBm specified and therefore, the evaluation tool has colored unreadable points in magenta meaning they are able to receive operation power from the interrogating transmitter but they cannot transmit back to the data receiver at an accuracy of 80.43%.

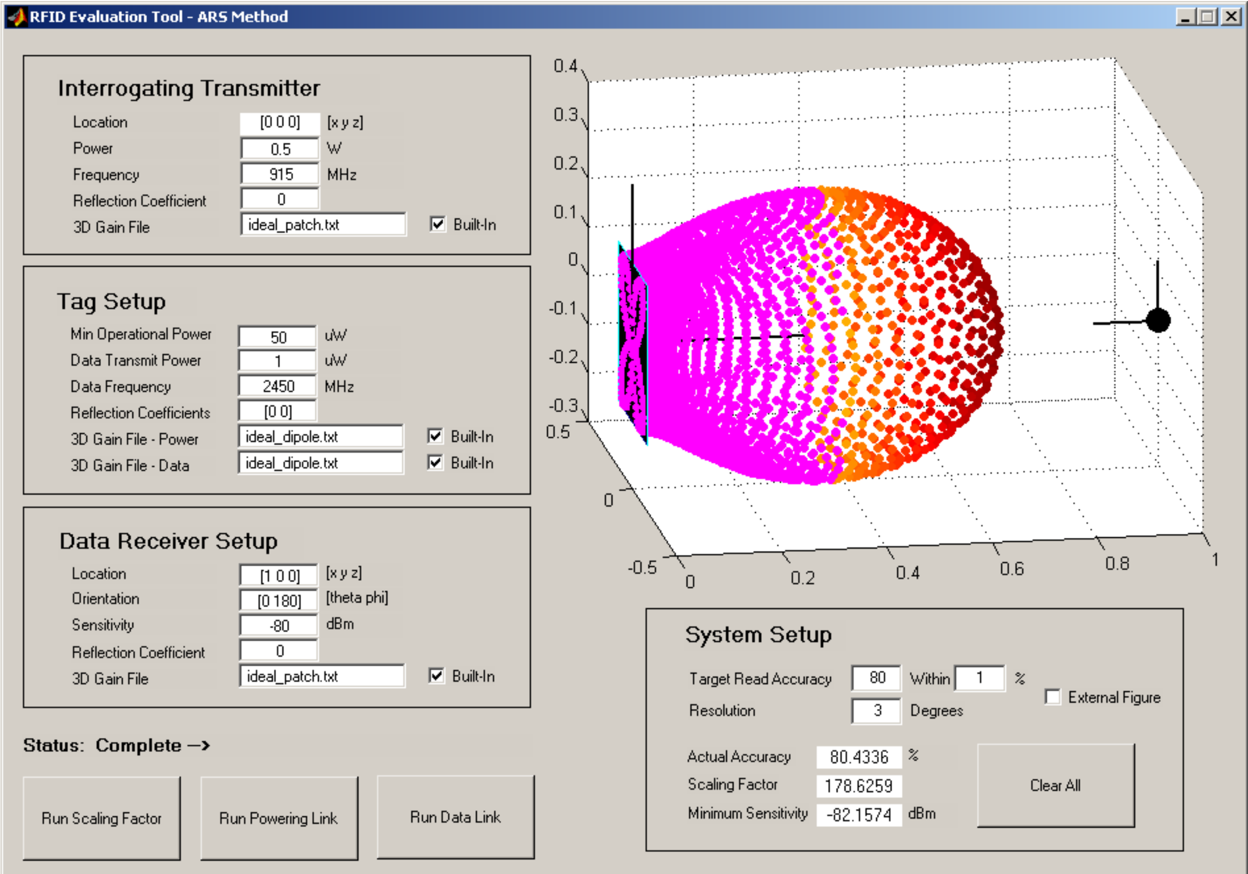

Figure 44 - Screen Shot of the ARS RFID Evaluation Tool after Running the Data Link with the Data Receiver at [1 0 0]

As has been shown in this section, the scaling factor technique can be applied to an ARS RFID system to allow fast calculations of the area of operation where an RFID tag can successfully receive operational power and transmit its data back to the receiving base station at a desired accuracy when in any orientation and polarization.

5.3 BACKSCATTER RFID SYSTEM TOOL

It is also possible to implement the scaling factor method with a backscatter RFID system. The mathematical model for a backscatter RFID system was developed in Section 3.4.2 and the equation was given in (3.162) and is repeated below for convenience.

$$P_R = P_T \frac{G_{TAG}^2(\theta,\phi)G_T(\theta_T,\phi_T)G_R(\theta_R,\phi_R)\lambda^4}{(4\pi)^4(r_1r_2)^2} \times \left|C+\Gamma^*\right|^2(1-\left|\Gamma_T\right|^2)(1-\left|\Gamma_R\right|^2)\left|\hat{\mathbf{p}}_T\cdot\hat{\mathbf{p}}_{TAG}\right|^2\left|\hat{\mathbf{p}}_{TAG}\cdot\hat{\mathbf{p}}_R\right|^{22} \tag{5.3}$$

This mathematical model was implemented in Matlab in conjunction with the scaling factor method described in the Section 4. A screen shot of the ARS Evaluation Tool GUI can be see in the following figure. The Matlab Code is given in Appendix F.

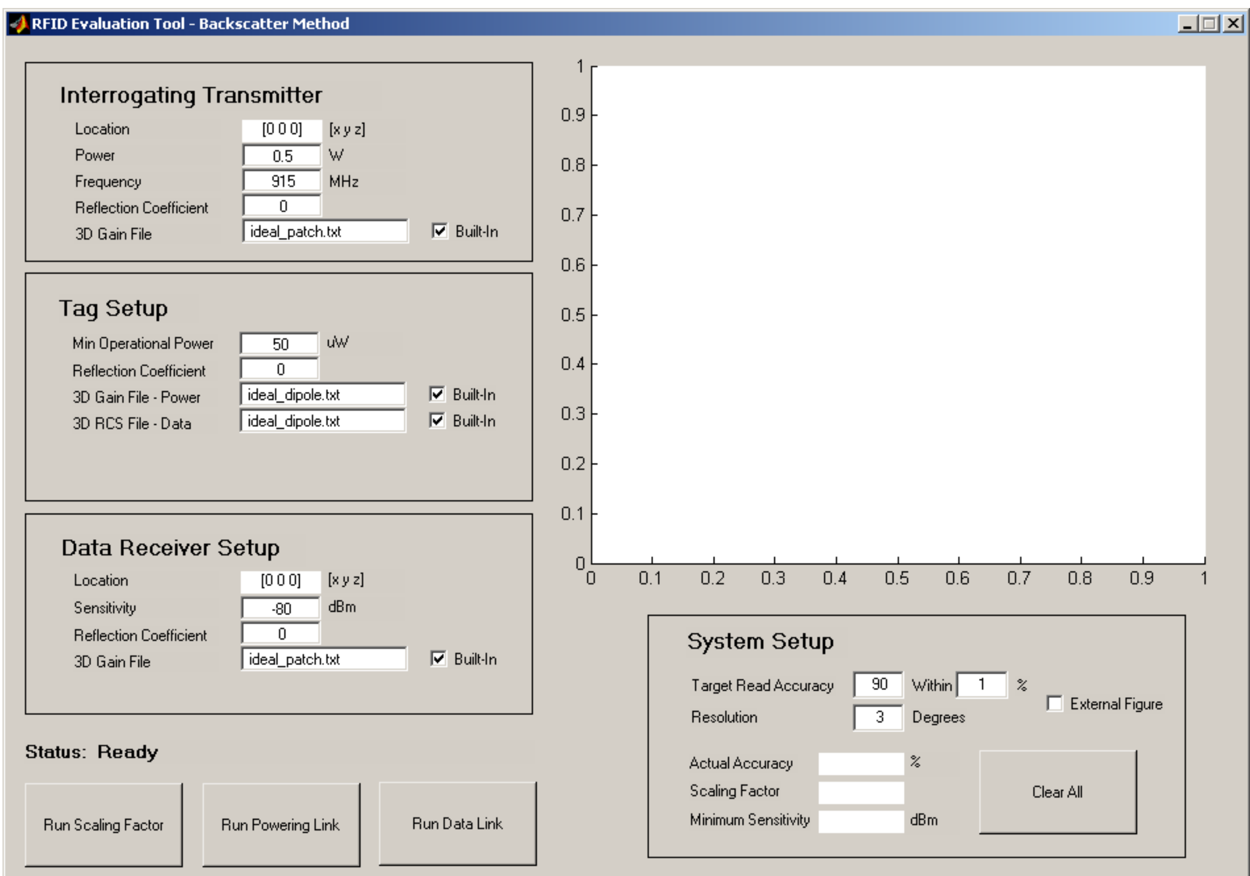

Figure 45 - Screen Shot of the Backscatter RFID Evaluation Tool

The backscatter evaluation tool, like the ARS version, requires detailed input from the user in order to accurately model the RFID system. The evaluation tool also outputs data that can be used to graphically and numerically evaluate the system performance. All of these parameters are outlined below for each of the four input sections.

Interrogating Transmitter

- **Location** - is a fixed value for the location of the interrogating transmitter. [0,0,0] is used as the reference location for the entire system.
- **Power** - is the power level of the interrogating transmitter. This is not the effective radiated power but the power delivered to the antenna.
- **Frequency** - is the fundamental frequency of the interrogating transmitter.
- **Reflection Coefficient** - is the reflection coefficient describing the impedance mismatch between the interrogating transmitter and the antenna.
- **3D Gain File** - is the three-dimensional gain file for the interrogating transmitter's antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave patch antenna by clicking the checkbox. The default is the built-in gain file. The resolution of a user defined gain file must match the resolution given in the System Setup box to avoid an error message.

Tag Setup

- **Min Operational Power** - is the minimum power required for the tag to operate its logic circuits. It should be noted that this value includes the power lost when the tag is modulating its RCS. Therefore, it is important to specify a power level, which can sustain the tag performance during the periods where the antenna is the most detuned.
- **Reflection Coefficient** - is the reflection coefficient describing the impedance mismatch between the tag's antenna and circuitry. This value is used not only for the powering link but also for the backscattering. It has been shown in Section 3.3 that the minimum backscattered power will be transmitted when the antenna is most closely matched. Therefore, it is not necessary to know the reflection coefficient when the RCS is modulated in order to calculate the minimum sensitivity.

- **3D Gain File - Power** - is the three-dimensional gain file for the tag's antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave dipole antenna by clicking the checkbox. The default is the built-in gain file. The resolution of a user defined gain file must match the resolution given in the System Setup box to avoid an error message.
- **3D RCS File - Data** - is the three-dimensional radar cross-section file for the tag's antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave dipole RCS by clicking the checkbox. The default is the built-in RCS file. The resolution of a user defined RCS file must match the resolution given in the System Setup box to avoid an error message.

Data Receiver Setup

- **Location** - is a fixed value for the location of the data receiver.
- **Sensitivity** - is the sensitivity of the data receiver used to determine whether the tag can backscatter sufficient power to communicate.
- **Reflection Coefficient** - is the reflection coefficient describing the impedance mismatch between the data receiver and the antenna.
- **3D Gain File** - is the three-dimensional gain file for data receiver's antenna that will be used for the system calculations. This file can be user defined or the user can enable the built-in ideal half-wave patch antenna by clicking the checkbox. The default is the built-in gain file. The resolution of a user defined gain file must match the resolution given in the System Setup box to avoid an error message.

System Setup

- **Target Read Accuracy** - is the read accuracy for which the user wishes to plot a constant read accuracy contour. The user can also specify a range used by the program when searching for the target accuracy.
- **Resolution** - is the resolution used by the program when calculating and plotting the data. Finer resolutions increase the number of data points exponentially and therefore increase the processing time. The resolution specified here must match the resolution of any user defined gain files to avoid an error message.

- **External Figure** - is used to plot the data in an external window for user manipulation.
- **Actual Accuracy** - is an output from the program specifying the actual accuracy, within the user-defined range, of the contour that will be plotted.
- **Scaling Factor** - is an output from the program specifying the power-scaling factor that is required to plot the constant accuracy contour.
- **Minimum Sensitivity** - is the minimum sensitivity needed by the data receiver in order to read all tag positions and orientations at the specified target read accuracy.

Once the user has entered the system parameters, the evaluation tool is ready for operation. The first step is to click the "Run Scaling Factor" button that enables the built-in scaling factor tool that was described in the Section 4. The program will search for the required Target Read Accuracy and will display the Actual Accuracy and Scaling Factor when completed. Additionally, the interrogating transmitter's antenna will be added to the plot with the X- and Z-axes for reference. This is shown in the following figure.

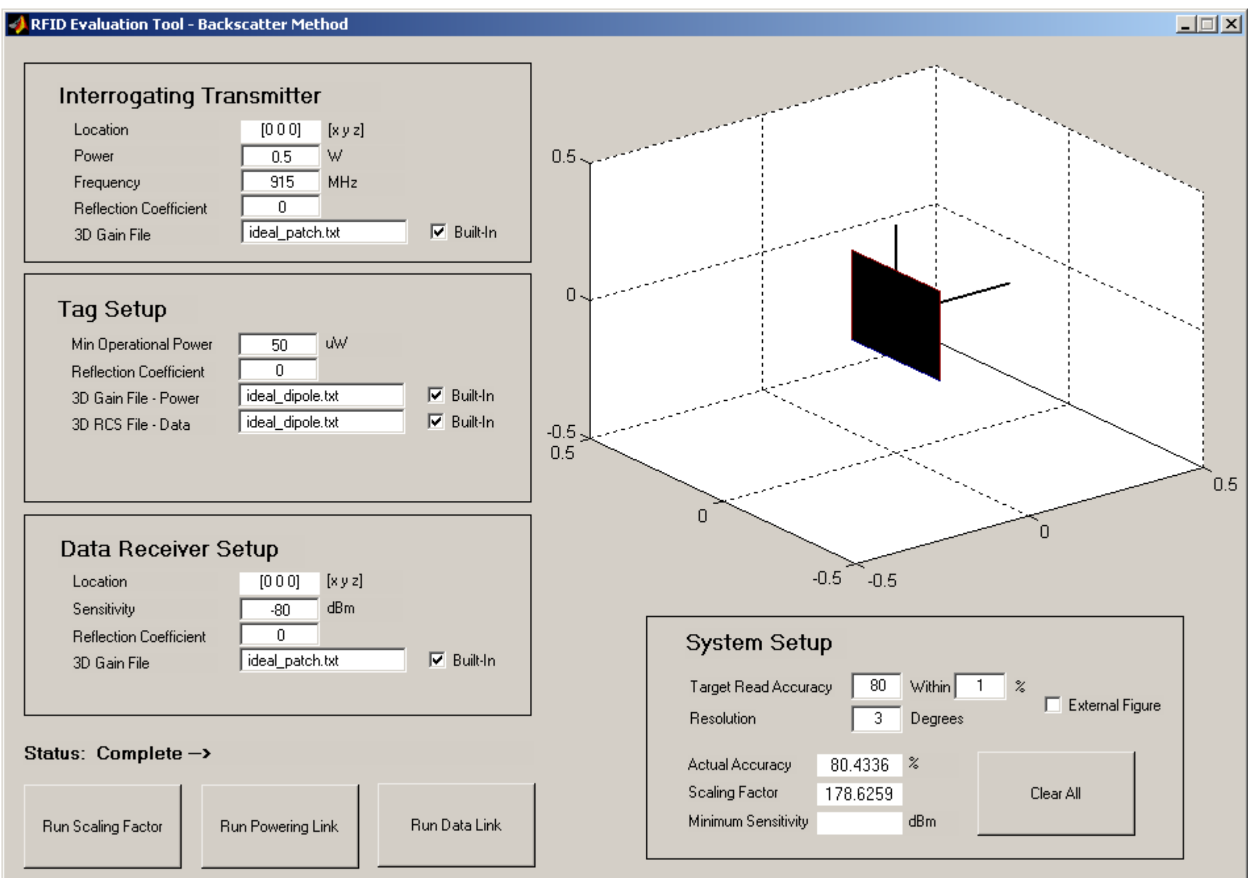

Figure 46 - Screen Shot of the Backscatter RFID Evaluation Tool after Running the Scaling Factor

Comparing Figure 46 to the ARS case shown in Figure 40 leads to the conclusion that the actual accuracy and the scaling factor are the same. This should be of no surprise because both methods obtain their power in the same fashion and only the communicating is done differently. This realization leads to the conclusion that the powering area of operation for a backscatter tag should also be the same as the ARS method. As can be seen in the following figure, the maximum read range is approximately 0.7 meters, which is the same as the ARS case.

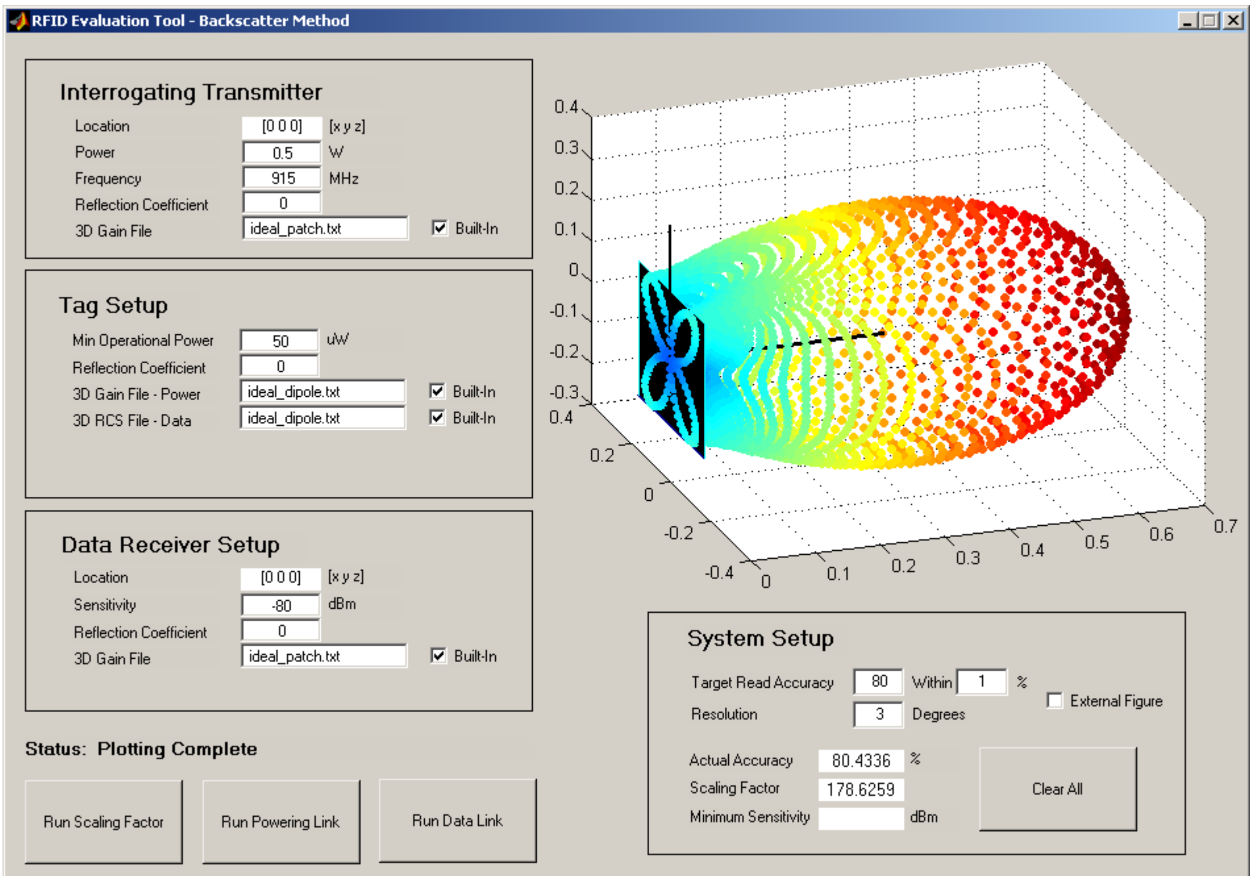

Figure 47 - Screen Shot of the Backscatter RFID Evaluation Tool after Running the Powering Link

The next step is to run the communication link to determine if 80.43% of the points can communicate. It should be noted that the data receiver location has been fixed at [0 0 0] to enable the use of the scaling factor method. For backscatter tags, the scattered or transmitted power is proportional to the captured power, and therefore, each point transmits a different power level at each position and orientation making it cumbersome to implement the base station at a different location. Running the communication link produces the output shown in the following figure.

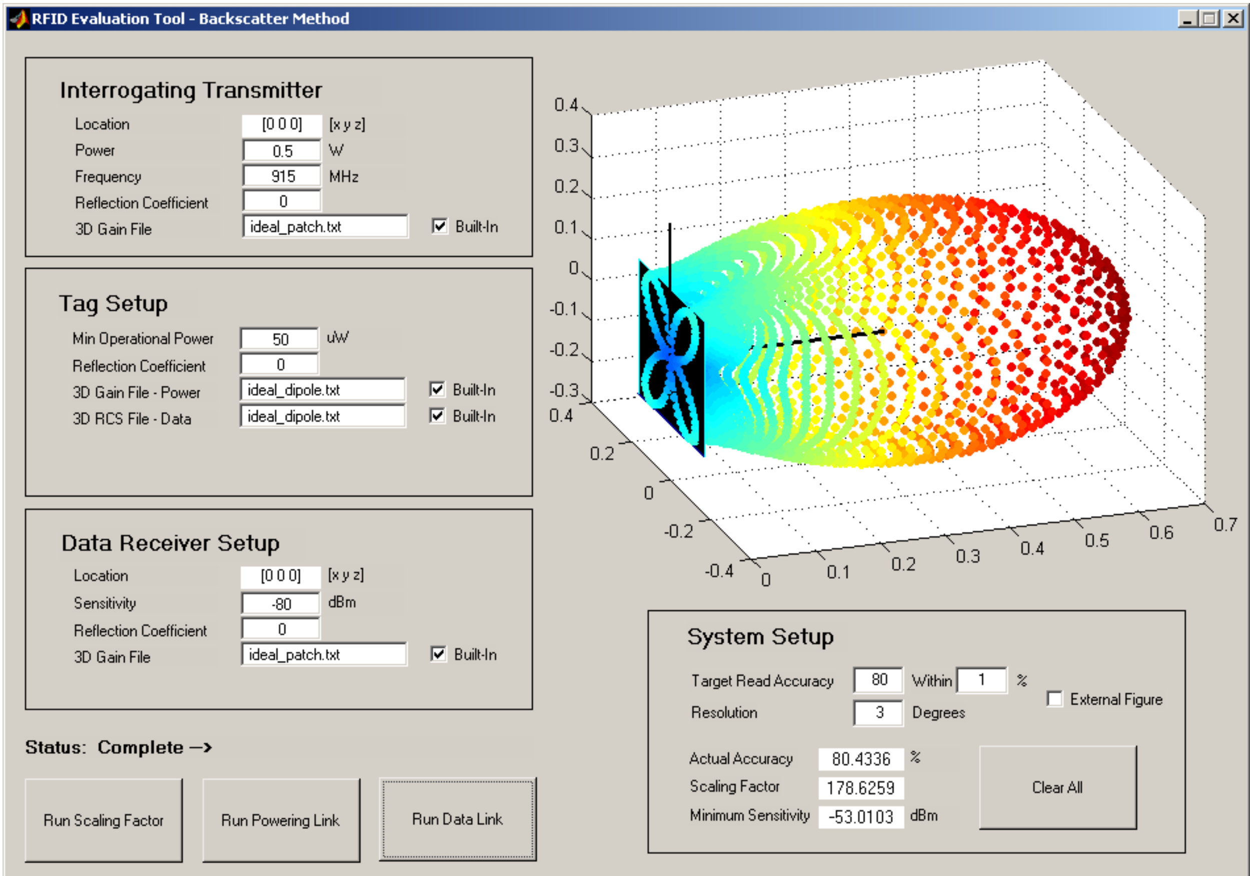

Figure 48 - Screen Shot of the Backscatter RFID Evaluation Tool after Running the Data Link

An interesting fact can be seen by examining Figure 48. The required sensitivity is much less than in the ARS case. This is due to the fact that a backscatter tag can transmit all, and under perfect conditions, four times the captured power by modulating its RCS. The result is an increase in the amount of power at the data receiver meaning it can be less sensitive.

6.0 CONCLUSIONS

6.1 RESEARCH RESULTS OVERVIEW

The presented research has developed mathematical models that can be used in evaluating the operation of RFID systems. Two different methods were presented including the ARS method developed at the University of Pittsburgh and the traditional backscatter technique used in most RFID systems. The resulting mathematical models were then examined with the intent of developing a technique for solution that did not require supercomputing capabilities. The resulting technique, termed the scaling factor method, provides a solution to the mathematical models that requires a large amount of calculations upfront, however, the resulting savings when solving the system is immense. The scaling factor technique was developed by the realization that the gain and polarization matrices can be combined to form a very large three-dimensional matrix that when solved with an arbitrary RFID system can produce a three-dimensional power matrix. The power matrix can then be searched in order to determine the percentage of points above a certain threshold, which is itself arbitrary when compared to the maximum value in the matrix. The resulting percentage is the corresponding read accuracy of the RFID system based on the ratio of the maximum power in the matrix to the threshold value. This ratio, termed the scaling factor, can then be applied to an RFID tag in its optimum position at a specific point to determine the read accuracy of all possible orientations and polarizations in a single calculation. Once the scaling factor is determined, a power threshold can be calculated by multiplying the scaling factor times the minimum power required for tag operation. The solution to the problem is then obtained by plotting a surface of constant power around the interrogating transmitter at the power threshold. This area describes a boundary between positions that can read above and below the specified read accuracy. This allows a visualization of where a tag must be in order to read accurately when in any orientation or polarization.

As was shown, the scaling factor method can also be applied to the communication links for both ARS and backscatter tags under certain conditions. Combining both the powering and communicating links led to the development of an evaluation tool, which can be used to implement the mathematical models and the scaling factor method to graphically display a tag's area of operation. This area accurately describes not only where the tag can receive operational power, but also where the tag can successfully transmit its data back to a receiving base station at a specified accuracy.

6.2 SPECIFIC CONTRIBUTIONS

As per the objectives outlined in Section 1.3, the research has provided mathematical models that can be used to describe the operation of ARS and backscatter RFID tags in a three-dimensional space. The analytical expressions developed for an ARS and backscatter RFID system are shown in (6.1) and (6.2), respectively.

$$P_R = \alpha P_T \frac{G_R(\theta_R,\phi_R)G_{TAG-TX}(\theta_{TAG},\phi_{TAG})G_T(\theta_T,\phi_T)G_{TAG-RX}(\theta_{TAG},\phi_{TAG})\lambda_R^2\lambda_T^2}{(4\pi)^4(r_1r_2)^2} \\ \times(1-|\Gamma_T|^2)(1-|\Gamma_R|^2)(1-|\Gamma_{TAG-TX}|^2)(1-|\Gamma_{TAG-RX}|^2)|\hat{\mathbf{p}}_R\bullet\hat{\mathbf{p}}_{TAG-TX}|^2|\hat{\mathbf{p}}_T\bullet\hat{\mathbf{p}}_{TAG-RX}|^2 \tag{6.1}$$

$$P_R = P_T \frac{G_{TAG}^2(\theta,\phi)G_T(\theta_T,\phi_T)G_R(\theta_R,\phi_R)\lambda^4}{(4\pi)^4(r_1r_2)^2} \\ \times|C+\Gamma^*|^2(1-|\Gamma_T|^2)(1-|\Gamma_R|^2)|\hat{\mathbf{p}}_T\bullet\hat{\mathbf{p}}_{TAG}|^2|\hat{\mathbf{p}}_{TAG}\bullet\hat{\mathbf{p}}_R|^2 \tag{6.2}$$

In order to demonstrate the functionality of the mathematical models in (6.1) and (6.2), the three-dimensional gains for two commonly used antennas in RFID systems, the half-wave dipole and the half-wave square patch, were mathematically evaluated to show the origins of the accepted analytical expressions. The expressions for the half-wave dipole and the half-wave square patch are shown in (6.3) and (6.4), respectively.

$$G(\theta,\phi) = 1.641\left[\frac{\cos\left(\frac{\pi}{2}\cos\theta\right)}{\sin\theta}\right]^2 \tag{6.3}$$

$$G(\theta,\phi) = 3.136\left\{\sin\theta\,\frac{\sin\left(\frac{\pi}{2}\cos\theta\right)}{\cos\theta}\cos\left(\frac{\pi}{2}\sin\theta\,\sin\phi\right)\right\}^2 \tag{6.4}$$

To develop the mathematical model for a backscatter RFID system, a relationship between the three-dimensional gain of the antenna and the three-dimensional radar cross-section of the antenna including the effects of antenna polarization was developed. The developed analytical expression can be seen in the following equation.

$$\sigma(Z_T,\theta,\phi,\hat{\mathbf{p}}_I,\hat{\mathbf{p}}_A) = \frac{\lambda^2}{4\pi}G_t^2(\theta,\phi)\left|C+\Gamma^*\right|^2\left|\hat{\mathbf{p}}_I \bullet \hat{\mathbf{p}}_A\right|^2 \tag{6.5}$$

In order to display the area of operation of an RFID system, a method for solving the mathematical model was developed. The method allows the determination of the percentage of reads possible for a given tag location in a three-dimensional space with respect to the interrogating transmitter and receiving base station when all possible orientations and polarizations are examined. The developed method is termed the scaling factor method due to its ability to quickly solve an RFID system by solving the tag's operation in the optimum position using a scaled minimum operational power value. The scaling factor method gives the ability to solve the area of operation for an RFID system in terms of constant read accuracy contours. The read accuracy contours graphically show where an RFID tag can be successfully powered and also where it can successfully transmit its data to a receiving base station at that specific accuracy.

To demonstrate the functionality of the mathematical models and the scaling factor method, several evaluation tools were developed. The first tool allows the determination of the scaling factor for a given antenna gain. The second and third tools were used to plot the area of operation for an RFID tag at a specific read accuracy for ARS and backscatter systems.

6.3 AREAS AFFECTED BY RESEARCH

The ability to display the area of operation of an RFID tag allows the easy comparison of different RFID tags. Generally, tags are compared on the basis of the maximum read range, however this gives little information on how the tag will perform in real-life situations that require the tag to work in a random position. As an extreme example, a patch antenna could be used as the tag's antenna to increase the read range of the tag due to the increase in gain and effective area. The problem here is that the tag will only work when it is pointed at the interrogating transmitter which may be impractical for certain applications.

The use of the mathematical models and evaluation tools developed through this research will allow quick comparison of different RFID tags on the basis of their areas of operation. As with the patch example, trying to plot the area of operation that would give 80% read accuracy would lead to the conclusion that the maximum obtainable accuracy is actually 50% (assuming zero back lobes). Tags can also be compared on the value of their scaling factor. The scaling factor tells the evaluator the amount of power the tag must receive in order to be read at a specific accuracy. Additionally, the scaling factor can be used to calculate the maximum read range for a given read accuracy. This allows evaluators to compare tags in the common maximum read range method and still obtain data about how the tag will perform in different orientation and polarizations. As an example, a tag may have a nine-meter maximum read range while the ninety-five percent maximum read range is only one meter. Therefore, the scaling factor can be used as a figure of merit to evaluate the performance of various RFID tags in the far-field and can be used to calculate another figure of merit; the maximum read range for a given read accuracy.

7.0 FUTURE RESEARCH

The goal of the presented research was to solve the area of operation of an RFID tag without assumptions. This goal was achieved and mathematical models where developed that could solve the problem without assumptions. However, when the mathematical models were implemented in an evaluation tool, certain design parameters needed to be set in order for the mathematical models to be implemented. As an example, the evaluation tools were implemented under the premise that the interrogating transmitter and data receiver had linear polarization. This implementation of an evaluation tool is mearly one of many different RFID systems. Future research could evaluate the performance of other RFID systems such as those with circularly polarized interrogating transmitters and/or data receivers. It may also be beneficial to implement an evaluation tool that includes multiple data receivers due to its wide use in the RFID industry.

Another area of particular interest is the RCS of an antenna interrogated by an in-band frequency. There have been numerous authors that have described the theoretical operation of antennas as scatters, however there is little research displaying measured results. Specifically, only one reference was found that gave measured data for different incident angles of the incoming wave front. And, interesting enough, the measured data only matched the theory for about ±30 degrees around the broadside case. Additionally, no references were found that described the RCS of an antenna for different polarizations. Therefore, research needs to be performed that provides measured data about the three-dimensional RCS of an antenna when oriented in different polarizations.

APPENDIX A

STANDARD RFID CALCULATIONS

Given parameters [2]:

$$\begin{aligned}
&\text{Minimum tag power, } P_{R\min} = 50\mu W \\
&\text{Effective Isotropic Radiated Power, } EIRP = P_T G_T = 4 \\
&\text{Tag Gain, } G_{TAG} = 1.64 \\
&\text{Frequency, } f = 915MHz \rightarrow \lambda = \frac{300}{915} = 0.328m
\end{aligned} \tag{8.1}$$

As was shown in (1.2),

$$P_R = P_T \frac{G_T G_R \lambda^2}{(4\pi r)^2} \tag{8.2}$$

Rearranging terms yields

$$r = \frac{\lambda}{4\pi}\sqrt{\frac{P_T G_T G_R}{P_R}} \tag{8.3}$$

Substituting in the known values gives

$$r_{max} = \frac{\lambda}{4\pi}\sqrt{\frac{EIRP\ G_{TAG}}{P_{R\min}}}$$
$$r_{max} = \frac{0.328}{4\pi}\sqrt{\frac{4\,(1.64)}{50\mu}} \tag{8.4}$$
$$r_{max} = 9.454m$$

Therefore, the maximum distance that the tag can receive the minimum power of 50μW when optimally positioned given the parameters in (8.1) is 9.454 meters.

The previous calculation only provided information on where the tag could receive operational power. There also needs to be a calculation to determine if the transmitted signal (backscatter in this case) can successfully be read by the reader, i.e. the reader sensitivity is low enough to sense the data. This calculation is done using the Radar Equation shown in (1.1), which is also repeated below.

$$P_R = P_T \frac{\sigma G_T G_R}{4\pi}\left[\frac{\lambda}{4\pi R_1 R_2}\right]^2 \tag{8.5}$$

The following parameters were given for this calculation.

$$\begin{aligned}
&\text{Minimum tag power, } P_{R\min} = 50\mu W\\
&\text{Effective Isotropic Radiated Power, } EIRP = P_T G_T = 4\\
&\text{Tag Gain, } G_{TAG} = 1.64\\
&\text{Frequency, } f = 915MHz \rightarrow \lambda = \frac{300}{915} = 0.328m\\
&\text{Distance between tag and reader, } R_1 = R_2 = r_{max} = 9.454m\\
&\text{Reader Gain, } G_R = 6dB = 4\\
&\text{RCS, } \sigma = 10cm^2 = 0.001m^2
\end{aligned} \tag{8.6}$$

Rewriting (8.5) in terms of the given parameters yields

$$P_R = EIRP\frac{\sigma G_R}{(4\pi)^3}\frac{\lambda^2}{r_{max}^4} \tag{8.7}$$

Substituting the parameters in (8.6) into (8.7) gives

$$
\begin{aligned}
P_R &= 4\frac{(0.001)\,4}{(4\pi)^3}\frac{(0.328)^2}{(9.454)^4} \\
P_R &= 1.086\times10^{-10}W \qquad (8.8) \\
P_R &= 10\log(1000\times1.086\times10^{-10}) \\
P_R &= -69.64dBm
\end{aligned}
$$

Therefore, the receiver must have a sensitivity of less than –69.64dBm in order to receive the data from a tag optimally positioned at 9.454 meters from the collocated interrogating transmitter and data receiver.

APPENDIX B

PLF MEASURED DATA

Angle	Measured Data (-dBm)	Power (uW)	Normalized Power to Obtain PLF	Calculated PLF
0	15.5	0.028183829	0.957194071	1
7.2	15.56	0.027797133	0.944060876	0.984291607
14.4	15.81	0.026242185	0.891250938	0.938153442
21.6	16.2	0.023988329	0.814704284	0.864484532
28.8	16.74	0.021183611	0.719448978	0.767913756
36	17.44	0.018030177	0.612350392	0.654509002
43.2	18.37	0.014554591	0.494310687	0.531395895
50.4	19.59	0.010990058	0.373250158	0.406310073
57.6	21.16	0.007655966	0.260015956	0.287111123
64.8	23.28	0.004698941	0.159587915	0.181288741
72	26.26	0.00236592	0.080352612	0.095492127
79.2	31.11	0.000774462	0.02630268	0.035112187
86.4	41.15	7.67361E-05	0.002606154	0.003942809
93.6	35	0.000316228	0.010739894	0.003942476
100.8	28.18	0.001520548	0.051641637	0.03511121
108	24.54	0.003515604	0.11939881	0.095490567
115.2	22.06	0.006223003	0.211348904	0.181286697
122.4	20.22	0.009506048	0.322849412	0.287108722
129.6	18.81	0.013152248	0.446683592	0.406307466
136.8	17.72	0.016904409	0.574116462	0.531393247
144	16.89	0.020464446	0.695024318	0.654506478
151.2	16.62	0.021777098	0.739605275	0.767911515
158.4	15.8	0.02630268	0.893305484	0.864482715
165.6	15.5	0.028183829	0.957194071	0.938152164
172.8	15.34	0.029241524	0.993116048	0.984290947
180	15.31	0.029444216	1	1
187.2	15.44	0.028575905	0.970509967	0.984292267
194.4	15.66	0.027164393	0.922571427	0.938154721
201.6	16.05	0.024831331	0.843334758	0.864486348
208.8	16.57	0.022029265	0.748169501	0.767915996
216	17.31	0.018578045	0.630957344	0.654511526
223.2	18.23	0.01503142	0.510505	0.531398544
230.4	19.43	0.011402498	0.387257645	0.406312679

Angle	Measured Data (-dBm)	Power (uW)	Normalized Power to Obtain PLF	Calculated PLF
237.6	20.97	0.007998343	0.271643927	0.287113524
244.8	23.06	0.004943107	0.167880402	0.181290786
252	25.9	0.002570396	0.087297137	0.095493686
259.2	30.47	0.000897429	0.03047895	0.035113164
266.4	40.66	8.59014E-05	0.002917427	0.003943142
273.6	37.55	0.000175792	0.005970353	0.003942144
280.8	29.36	0.001158777	0.039355008	0.035110233
288	25.25	0.002985383	0.101391139	0.095489007
295.2	22.63	0.005457579	0.185353162	0.181284652
302.4	20.67	0.008570378	0.291071712	0.28710632
309.6	19.21	0.011994993	0.407380278	0.406304859
316.8	18.08	0.015559656	0.528445252	0.531390599
324	17.21	0.019010783	0.645654229	0.654503954
331.2	16.58	0.021978599	0.746448758	0.767909275
338.4	16.11	0.024490632	0.831763771	0.864480899
345.6	15.82	0.02618183	0.889201118	0.938150886
352.8	15.68	0.027039584	0.918332596	0.984290287

APPENDIX C

HALF-WAVE DIPOLE MEASURED DATA

Angle	Measured Power (-dBm)	Power (uW)	Measured Gain	Calculated Gain
0	17.09	1.95434E-05	1.694374226	1.64
6	17.19	1.90985E-05	1.674978885	1.6138
12	17.23	1.89234E-05	1.667283056	1.5377
18	17.5	1.77828E-05	1.616252972	1.4185
24	17.93	1.61065E-05	1.538187708	1.2663
30	18.56	1.39316E-05	1.430570735	1.0933
36	19.34	1.16413E-05	1.307703652	0.9117
42	20.3	9.33254E-06	1.170871774	0.7325
48	21.53	7.03072E-06	1.016270541	0.5646
54	23	5.01187E-06	0.858043561	0.4145
60	24.91	3.22849E-06	0.688667101	0.2863
66	27.22	1.89671E-06	0.527848554	0.1817
72	30.29	9.35406E-07	0.370688691	0.1013
78	34.91	3.22849E-07	0.217775659	0.0447
84	44.1	3.89045E-08	0.075597831	0.0111
90	43.75	4.21697E-08	0.078706271	0
96	35	3.16228E-07	0.215530798	0.0111
102	30.6	8.70964E-07	0.35769209	0.0447
108	27.81	1.65577E-06	0.493184406	0.1013
114	25.75	2.66073E-06	0.625186133	0.1817
120	24	3.98107E-06	0.764732129	0.2863
126	22.73	5.33335E-06	0.885134638	0.4145
132	21.64	6.85488E-06	1.003481422	0.5646
138	20.73	8.45279E-06	1.114318489	0.7325
144	19.97	1.00693E-05	1.216212147	0.9117
150	19.4	1.14815E-05	1.298701483	1.0933
156	18.9	1.28825E-05	1.375654384	1.2663
162	18.48	1.41906E-05	1.443807643	1.4185
168	18.15	1.53109E-05	1.499717076	1.5377
174	17.94	1.60694E-05	1.536417823	1.6138
180	17.79	1.66341E-05	1.563181248	1.64
186	17.82	1.65196E-05	1.557791524	1.6138
192	18	1.58489E-05	1.525841197	1.5377
198	18.33	1.46893E-05	1.468957857	1.4185
204	18.75	1.33352E-05	1.399617411	1.2663

Angle	Measured Power (-dBm)	Power (uW)	Measured Gain	Calculated Gain
210	19.35	1.16145E-05	1.306198969	1.0933
216	20.05	9.88553E-06	1.205061847	0.9117
222	20.91	8.10961E-06	1.091463801	0.7325
228	22	6.30957E-06	0.96274071	0.5646
234	23.14	4.85289E-06	0.844324393	0.4145
240	24.55	3.50752E-06	0.717809723	0.2863
246	26.26	2.36592E-06	0.589534651	0.1817
252	28.44	1.43219E-06	0.458679506	0.1013
258	31.24	7.51623E-07	0.332283929	0.0447
264	34.89	3.2434E-07	0.218277684	0.0111
270	38.01	1.58125E-07	0.152408552	0
276	36.12	2.44343E-07	0.189456424	0.0111
282	32.26	5.94292E-07	0.295467241	0.0447
288	29.07	1.2388E-06	0.42658869	0.1013
294	26	2.51189E-06	0.607448322	0.1817
300	24.23	3.77572E-06	0.744747984	0.2863
306	22.7	5.37032E-06	0.888197071	0.4145
312	21.53	7.03072E-06	1.016270541	0.5646
318	20.44	9.03649E-06	1.152150829	0.7325
324	19.59	1.09901E-05	1.270601421	0.9117
330	18.88	1.2942E-05	1.378825595	1.0933
336	18.31	1.47571E-05	1.472344155	1.2663
342	17.92	1.61436E-05	1.539959632	1.4185
348	17.69	1.70216E-05	1.581282034	1.5377
354	17.55	1.75792E-05	1.6069758	1.6138

APPENDIX D

MATLAB CODE FOR SCALING FACTOR EVALUATION TOOL

```
%Written by Charles Greene
%University of Pittsburgh
%11-27-05
%
function varargout = scaling_factor(varargin)
% SCALING_FACTOR Application M-file for scaling_factor.fig
%    FIG = SCALING_FACTOR launch scaling_factor GUI.
%    SCALING_FACTOR('callback_name', ...) invoke the named callback.

% Last Modified by GUIDE v2.0 17-Nov-2005 18:17:54

if nargin == 0  % LAUNCH GUI

   fig = openfig(mfilename,'reuse');

   % Use system color scheme for figure:
   set(fig,'Color',get(0,'defaultUicontrolBackgroundColor'));

   % Generate a structure of handles to pass to callbacks, and store it.
   handles = guihandles(fig);
   guidata(fig, handles);

   if nargout > 0
      varargout{1} = fig;
   end

elseif ischar(varargin{1}) % INVOKE NAMED SUBFUNCTION OR CALLBACK

   try
      if (nargout)
         [varargout{1:nargout}] = feval(varargin{:}); % FEVAL switchyard
      else
         feval(varargin{:}); % FEVAL switchyard
      end
   catch
      disp(lasterr);
   end

end
```

```
%| ABOUT CALLBACKS:
%| GUIDE automatically appends subfunction prototypes to this file, and
%| sets objects' callback properties to call them through the FEVAL
%| switchyard above. This comment describes that mechanism.
%|
%| Each callback subfunction declaration has the following form:
%| <SUBFUNCTION_NAME>(H, EVENTDATA, HANDLES, VARARGIN)
%|
%| The subfunction name is composed using the object's Tag and the
%| callback type separated by '_', e.g. 'slider2_Callback',
%| 'figure1_CloseRequestFcn', 'axis1_ButtondownFcn'.
%|
%| H is the callback object's handle (obtained using GCBO).
%|
%| EVENTDATA is empty, but reserved for future use.
%|
%| HANDLES is a structure containing handles of components in GUI using
%| tags as fieldnames, e.g. handles.figure1, handles.slider2. This
%| structure is created at GUI startup using GUIHANDLES and stored in
%| the figure's application data using GUIDATA. A copy of the structure
%| is passed to each callback.  You can store additional information in
%| this structure at GUI startup, and you can change the structure
%| during callbacks.  Call guidata(h, handles) after changing your
%| copy to replace the stored original so that subsequent callbacks see
%| the updates. Type "help guihandles" and "help guidata" for more
%| information.
%|
%| VARARGIN contains any extra arguments you have passed to the
%| callback. Specify the extra arguments by editing the callback
%| property in the inspector. By default, GUIDE sets the property to:
%| <MFILENAME>('<SUBFUNCTION_NAME>', gcbo, [], guidata(gcbo))
%| Add any extra arguments after the last argument, before the final
%| closing parenthesis.



% --------------------------------------------------------------------
function varargout = slider1_Callback(h, eventdata, handles, varargin)


% --------------------------------------------------------------------
function varargout = pushbutton1_Callback(h, eventdata, handles, varargin)

global tagpol

slider_value = get(handles.slider1,'Value');

set(handles.text2,'String','Status:  Running');

%Set polarization
if get(handles.checkbox1,'Value')==1
    tagpol=1;
else
    tagpol=2;
end
```

```
%Read in tag antenna from user's file, assumes a theta x phi matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit1,'String'),'r');
text_gain=fscanf(fid,'%c');
gm2=str2num(text_gain);
fclose(fid);

%Check size of read matrix, should be 180/N+1 X 360/N
[rows,cols]=size(gm2);
step=360/cols;
if (rows-1)*2~=cols
    set(handles.text2,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');     %Write error to the screen
    set(handles.edit1,'String','ERROR');
    break;
end

gm3=gm2;                                %Initialize variable

%Generate 3D gain matrix with polarization data
if tagpol==1
    for pol=1:((90/step)+1)             %Point 1 is zero degrees, 91 is therefore
90 degrees
        gm3(:,:,pol)=gm2.*((cos((pol-1)*step*pi/180))^2);
    end
else
    gm3=gm2;            %No polarization dependence, only need 2D matrix
end

r=0.01;                                  %start distance
done=0;
rinc=1/(slider_value);                               %distance increment
lastread=0;
acctoohigh=0;
lam=0.328;
ptx=0.5;
pmin=50*10^-6;
gi=1;                               %Dummy gain used to calculate scaling
factor, divides out in the end, value does not matter


factor=ptx*gi*lam*lam/(4*pi)^2;     %Used to increase program speed, will
divide out at the end but makes easier to understand
a=180/step+1;
b=360/step;
c=90/step+1;
inc=1;

while done==0;                      %Use iteration to find scaling factor

    %Find maximum gain of input file
    gmax=max(max(gm2));

    %Maximum power
    prmax=factor*gmax/(r)^2;
%Friis Equation
```

```
    %Power Matrix
    pr=factor.*gm3./(r)^2;
%Friis Equation

    %Determine number of points over threshold for all possible gains and
polarizations at a point in space
    yes=0;
    no=0;

    if tagpol==1                    %Search 3D matrix
      for row=1:a
         for col=1:b
            for dep=1:c
                if pr(row,col,dep)>pmin
                    yes=yes+1;
                else
                    no=no+1;
                end
            end
         end
      end

    else                            %Search 2D matrix
        for row=1:a
           for col=1:b
              if pr(row,col)>pmin
                 yes=yes+1;
              else
                 no=no+1;
              end
           end
        end
    end


    total=yes+no;

    read_accuracy=yes/total*100;

    set(handles.text2,'String',strcat(['Status:  Current accurracy
',num2str(read_accuracy),'%']));      %Write status to the screen

    scaling_factor=prmax/pmin;

    x(inc)=read_accuracy;
    y(inc)=scaling_factor;
    inc=inc+1;

    if r>=10
        done=1;
    elseif read_accuracy<0.1
        done=1;
    elseif read_accuracy>85
        r=r+rinc/3;             %Increase resolution in nonlinear section
    else
        r=r+rinc;
    end
```

```
end

semilogy(x,y,'bo-')
axis([0 100 0 10000])

set(handles.text2,'String','Status:  Done');

format bank
x'
y'

hold off
% --------------------------------------------------------------------
function varargout = edit1_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = checkbox1_Callback(h, eventdata, handles, varargin)

global tagpol;

if get(handles.checkbox1,'Value')==1
    set(handles.checkbox2,'Value',0)
    tagpol=1;
else
    set(handles.checkbox2,'Value',1)
    tagpol=2;
end

% --------------------------------------------------------------------
function varargout = checkbox2_Callback(h, eventdata, handles, varargin)

global tagpol;

if get(handles.checkbox2,'Value')==1
    set(handles.checkbox1,'Value',0)
    tagpol=2;
else
    set(handles.checkbox1,'Value',1)
    tagpol=1;
end
```

APPENDIX E

MATLAB CODE FOR ARS EVALUATION TOOL

Files
evaltoolgui_ars.m
sfsolve_builtin.m
sfsolve_load.m
guiplot_builtin.m
guiplot_load.m
datalink_builtin.m
datalink_load.m

evaltoolgui_ars.m

```
%Written by Charles Greene
%University of Pittsburgh
%11-27-05
%
function varargout = evaltoolgui_ars(varargin)
% EVALTOOLGUI_ARS Application M-file for evaltoolgui_ars.fig
%    FIG = EVALTOOLGUI_ARS launch evaltoolgui_ars GUI.
%    EVALTOOLGUI_ARS('callback_name', ...) invoke the named callback.

% Last Modified by GUIDE v2.0 22-Oct-2005 10:31:47

if nargin == 0  % LAUNCH GUI

   fig = openfig(mfilename,'reuse');
   %set(fig,'Color',get(0,'defaultUicontrolBackgroundColor'));

   % Generate a structure of handles to pass to callbacks, and store it.
   handles = guihandles(fig);
   guidata(fig, handles);

   if nargout > 0
      varargout{1} = fig;
   end

elseif ischar(varargin{1}) % INVOKE NAMED SUBFUNCTION OR CALLBACK

   try
```

```
      if (nargout)
         [varargout{1:nargout}] = feval(varargin{:}); % FEVAL switchyard
      else
         feval(varargin{:}); % FEVAL switchyard
      end
   catch
      disp(lasterr);
   end

end


%| ABOUT CALLBACKS:
%| GUIDE automatically appends subfunction prototypes to this file, and
%| sets objects' callback properties to call them through the FEVAL
%| switchyard above. This comment describes that mechanism.
%|
%| Each callback subfunction declaration has the following form:
%| <SUBFUNCTION_NAME>(H, EVENTDATA, HANDLES, VARARGIN)
%|
%| The subfunction name is composed using the object's Tag and the
%| callback type separated by '_', e.g. 'slider2_Callback',
%| 'figure1_CloseRequestFcn', 'axis1_ButtondownFcn'.
%|
%| H is the callback object's handle (obtained using GCBO).
%|
%| EVENTDATA is empty, but reserved for future use.
%|
%| HANDLES is a structure containing handles of components in GUI using
%| tags as fieldnames, e.g. handles.figure1, handles.slider2. This
%| structure is created at GUI startup using GUIHANDLES and stored in
%| the figure's application data using GUIDATA. A copy of the structure
%| is passed to each callback.  You can store additional information in
%| this structure at GUI startup, and you can change the structure
%| during callbacks.  Call guidata(h, handles) after changing your
%| copy to replace the stored original so that subsequent callbacks see
%| the updates. Type "help guihandles" and "help guidata" for more
%| information.
%|
%| VARARGIN contains any extra arguments you have passed to the
%| callback. Specify the extra arguments by editing the callback
%| property in the inspector. By default, GUIDE sets the property to:
%| <MFILENAME>('<SUBFUNCTION_NAME>', gcbo, [], guidata(gcbo))
%| Add any extra arguments after the last argument, before the final
%| closing parenthesis.



% --------------------------------------------------------------------
function varargout = pushbutton1_Callback(h, eventdata, handles, varargin)

warning off                                           %Suppress divide by
zero warnings

global reqacc peracc step pmin ptx f lam prmax powered_positions gm3
scaling_factor
global gammai gammatrx gammattx gammabs
```

```
set(handles.text60,'String','Status:  Finding Scaling Factor...');     %Write
status to the screen

%Read in user entered data
reqacc=str2num(get(handles.edit1,'String'));            %Data from system
boxes
peracc=str2num(get(handles.edit5,'String'));
step=str2num(get(handles.edit6,'String'));

%step must be a factor of 90
steptest=round(90/step);
if steptest*step~=90
    set(handles.text60,'String','Status:  Abort --> Step must be a factor of
90');     %Write error to the screen
    break;
end

%Reflection Coefficients
gammai=str2num(get(handles.edit11,'String'));            %Interrogator
gamma_matrix=str2num(get(handles.edit23,'String'));
gammatrx=gamma_matrix(1);                               %Tag RX
gammattx=gamma_matrix(2);                                 %Tag TX
gammabs=str2num(get(handles.edit28,'String'));           %Data Base Station

pmin=(str2num(get(handles.edit25,'String')))*10^-6;     %Min power level need
by the tag
ptx=str2num(get(handles.edit9,'String'));               %Interrogating
transmitter power
f=str2num(get(handles.edit10,'String'));;               %Interrogating
frequency in MHz
lam=300/f;                                              %Free space
wavelength

%Plot a square halfwave length antenna, x-axis, and z-axis
x=[0 0 0 0];
y=[-lam/2 lam/2 -lam/2 lam/2];
z=[lam/2 lam/2 -lam/2 -lam/2;lam/2 lam/2 -lam/2 -lam/2;lam/2 lam/2 -lam/2 -
lam/2;lam/2 lam/2 -lam/2 -lam/2];
mesh(x,y,z,'FaceColor',[0 0 0])
hold on
x=[0 lam];
y=[0 0];
z=y;
plot3(x,y,z,'k','LineWidth',2)
hold on
z=[0 lam];
x=[0 0];
plot3(x,y,z,'k','LineWidth',2)
hold on
axis([-0.5 0.5 -0.5 0.5 -0.5 0.5])
rotate3d

if get(handles.checkbox5,'Value')==1                    %Is the user
requesting the built-in tag gain
    sfsolve_builtin                                     %Solve the scaling
factor and actual read accuracy
```

```
else
    sfsolve_load                                        %Load user defined
gain file and solve scaling factor
end

set(handles.text9,'String',num2str(read_accuracy));     %Return data to gui
set(handles.text15,'String',num2str(scaling_factor));

if acctoohigh==1
    set(handles.text60,'String',strcat(['Status:  Maximum obtainable
accurracy ',num2str(read_accuracy),'%']));
else
    set(handles.text60,'String','Status:  Complete -->');     %Write status
to the screen
end

% --------------------------------------------------------------------
function varargout = pushbutton2_Callback(h, eventdata, handles, varargin)

global reqacc peracc step pmin ptx f lam prmax x y z r
global gammai gammatrx gammattx gammabs

set(handles.text60,'String','Status:  Plotting Data');     %Write status to
the screen

if get(handles.checkbox2,'Value')==1                    %returns a one when
box is checked, zero otherwise
    guiplot_builtin                                             %Plot the
data
else
    guiplot_load                                        %Plot the data using
the user loaded gain
end

set(handles.text60,'String','Status:  Plotting Complete');     %Write status
to the screen

% --------------------------------------------------------------------
function varargout = pushbutton3_Callback(h, eventdata, handles, varargin)

global reqacc peracc step pmin ptx f lam prmax
global x y z powered_positions gm3 r scaling_factor
global gammai gammatrx gammattx gammabs

%These link calculations assume that the tag's antennas are at the same
location and each antenna has the same axes
%i.e. theta=0, phi=0 is the z-axis for both antennas, etc.
%This assumption is needed to transfer the points from the first link in
order to know what point will not be
%able to communicate due the lack of power.

%Data from tag and data receiver boxes
tagtxpower=str2num(get(handles.edit21,'String'))*10^-6;       %Tag
transmitted power from uW to watts
tagtxfreq=str2num(get(handles.edit22,'String'));        %Tag transmitted
frequency
taglam=300/tagtxfreq;                                 %Tag wavelength
```

```
bsloc=str2num(get(handles.edit30,'String'));            %Location of the base
station
bssenlog=str2num(get(handles.edit26,'String'));         %Base station
sensitivity

xdelta=bsloc(1);
ydelta=bsloc(2);
zdelta=bsloc(3);
bssen=10^(bssenlog/10)/1000;                            %Convert dBm to
linear sensitivity in watts

if get(handles.checkbox6,'Value')==1                    %returns a one when
box is checked, zero otherwise
    datalink_builtin                                    %Plot the data
else
    datalink_load                                       %Plot the data using
the user loaded gain
end

% --------------------------------------------------------------------
function varargout = pushbutton4_Callback(h, eventdata, handles, varargin)
hold off
plot3(0,0,0,'Color',[0 0 0])
hold on
grid on
rotate3d

set(handles.text60,'String','Status:  Ready');      %Write status to the
screen
set(handles.text9,'String',' ');                    %Clear data on gui
set(handles.text15,'String',' ');
set(handles.text64,'String',' ');


% --------------------------------------------------------------------
function varargout = checkbox1_Callback(h, eventdata, handles, varargin)

global x y z

if get(handles.checkbox1,'Value')==1                    %returns a one when
box is checked, zero otherwise

    figure(1)
    scatter3(x,y,z,30,sqrt(x.^2+y.^2+z.^2),'filled')    %Plots the data in an
external window for full screen view
    rotate3d
    title('External 3D Plot of the Area of Operation')
end

% --------------------------------------------------------------------
function varargout = checkbox2_Callback(h, eventdata, handles, varargin)

if get(handles.checkbox2,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit12,'String','ideal_patch.txt');
else
    [ifilename, ipathname] = uigetfile('*.txt', 'Choose a Gain File');
```

```
    if ifilename~=0
        set(handles.edit12,'String',ifilename);
    else
        set(handles.checkbox2,'Value',1)                   %Cancel button
pressed or error
        set(handles.edit12,'String','ideal_patch.txt');
    end
end

% --------------------------------------------------------------------
function varargout = checkbox5_Callback(h, eventdata, handles, varargin)

if get(handles.checkbox5,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit24,'String','ideal_dipole.txt');
else
    [trxfilename, trxpathname] = uigetfile('*.txt', 'Choose a Gain File');
    if trxfilename~=0
        set(handles.edit24,'String',trxfilename);
    else
        set(handles.checkbox5,'Value',1)                   %Cancel button
pressed or error
        set(handles.edit24,'String','ideal_dipole.txt');
    end
end

% --------------------------------------------------------------------
function varargout = checkbox6_Callback(h, eventdata, handles, varargin)

if get(handles.checkbox6,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit29,'String','ideal_patch.txt');
else
    [bsfilename, bspathname] = uigetfile('*.txt', 'Choose a Gain File');
    if bsfilename~=0
        set(handles.edit29,'String',bsfilename);
    else
        set(handles.checkbox6,'Value',1)                   %Cancel button
pressed or error
        set(handles.edit29,'String','ideal_patch.txt');
    end
end

% --------------------------------------------------------------------
function varargout = checkbox7_Callback(h, eventdata, handles, varargin)

if get(handles.checkbox7,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit31,'String','ideal_dipole.txt');
else
    [ttxfilename, ttxpathname] = uigetfile('*.txt', 'Choose a Gain File');
    if ttxfilename~=0
        set(handles.edit31,'String',ttxfilename);
    else
        set(handles.checkbox7,'Value',1)                   %Cancel button
pressed or error
        set(handles.edit31,'String','ideal_dipole.txt');
```

```
    end
end

% --------------------------------------------------------------------
function varargout = edit1_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit2_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit3_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit4_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit5_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit6_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit9_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit10_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit11_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));
```

```
% --------------------------------------------------------------------
function varargout = edit12_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit21_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit22_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit23_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit24_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit25_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit26_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit28_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit29_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit30_Callback(h, eventdata, handles, varargin)
```

```
set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit31_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit32_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));
```

sfsolve_builtin.m

```
gi=1;                               %Dummy gain used to calculate scaling
factor, divides out in the end, value does not matter
theta=[0:step:180];                 %Need both 0 and 180 degrees because gain
can be different
phi=[0:step:360-step];              %Subtract off step so 0 deg isn't counted
twice

%Generate 3D gain matrix
gt=1.64*(((cos(pi/2.*cos(theta*pi/180)))./sin(theta*pi/180)).^2);
gt(1)=0;                       %Set endpoints to zero, infinity and round off
problems
gt(length(gt))=0;
gm2=ones(180/step+1,360/step);
gm3=gm2;
for phi=1:360/step
    gm2(:,phi)=gt';            %Create 3D gain matrix
end

%Generate 3D gain matrix with polarization data
for pol=1:((90/step)+1)            %Point 1 is zero degrees, 91 is therefore 90
degrees
    gm3(:,:,pol)=gm2.*((cos((pol-1)*step*pi/180))^2);
end

r=0.1;                                 %distance
done=0;
rinc=0.1;                              %distance increment
lastread=0;
acctoohigh=0;

factor=ptx*gi*lam*lam/(4*pi)^2;        %Used to increase program speed, will
divide out at the end but makes easier to understand
a=180/step+1;
b=360/step;
c=90/step+1;
powered_positions=ones(a,b,c);         %Sets up a matrix to track points that
can receive enough power to operate, needed for next link

while done==0;                         %Use iteration to find scaling factor
```

```
    %Maximum power
    prmax=factor*1.64/(r)^2;
%Friis Equation

    %Power Matrix
    pr=factor.*gm3./(r)^2;
%Friis Equation

    %Determine number of points over threshold for all possible gains and
polarizations at a point in space
    yes=0;
    no=0;
    for row=1:a
        for col=1:b
            for dep=1:c
                if pr(row,col,dep)>pmin
                    yes=yes+1;
                else
                    no=no+1;
                    powered_positions(row,col,dep)=0;
                end
            end
        end
    end

    total=yes+no;

    read_accuracy=yes/total*100;

    set(handles.text60,'String',strcat(['Status:  Current accurracy
',num2str(read_accuracy),'%']));     %Write status to the screen

    scaling_factor=prmax/pmin;

    if (read_accuracy<=(reqacc+peracc) & read_accuracy>=(reqacc-peracc))
        done=1;
    elseif read_accuracy<(reqacc-peracc)
        if r==rinc
            rinc=rinc/2;                %so r can never equal zero
        end
        if round(lastread*1000)==round(read_accuracy*1000)      %Check to see
if accuracy is changing, if not, break
            done=1;
            acctoohigh=1;
        end
        r=r-rinc;
        rinc=rinc/2;
        lastread=read_accuracy;
    else
        r=r+rinc;
    end
end
```

sfsolve_load.m

```
gi=1;                                %Dummy gain used to calculate scaling
factor, divides out in the end, value does not matter

%Read in tag power antenna from user's file, assumes a theta x phi matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit24,'String'),'r');
text_gain=fscanf(fid,'%c');
gm2=str2num(text_gain);
fclose(fid);

%Check size of read matrix, should be 180/N+1 X 360/N
[rows,cols]=size(gm2);
stepcheck=360/cols;
if (rows-1)*2~=cols
    set(handles.text60,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');      %Write error to the screen
    set(handles.edit24,'String','ERROR');
    break;
end
if step~=stepcheck
    set(handles.text60,'String','Status:  Abort --> File Does Not Match Input
Resolution');      %Write error to the screen
    set(handles.edit24,'String','ERROR');
    break;
end

gm3=gm2;
%Generate 3D gain matrix with polarization data
for pol=1:((90/step)+1)          %Point 1 is zero degrees, 91 is therefore 90
degrees
    gm3(:,:,pol)=gm2.*((cos((pol-1)*step*pi/180))^2);
end

r=0.1;                                  %distance
done=0;
rinc=0.1;                               %distance increment
lastread=0;
acctoohigh=0;

factor=ptx*gi*lam*lam/(4*pi)^2;      %Used to increase program speed, will
divide out at the end but makes easier to understand
a=180/step+1;
b=360/step;
c=90/step+1;
powered_positions=ones(a,b,c);       %Sets up a matrix to track points that
can receive enough power to operate, needed for next link

while done==0;                       %Use iteration to find scaling factor

    %Find maximum gain of input file
    gmax=max(max(gm2));

    %Maximum power
    prmax=factor*gmax/(r)^2;
%Friis Equation
```

```
    %Power Matrix
    pr=factor.*gm3./(r)^2;
%Friis Equation

    %Determine number of points over threshold for all possible gains and
polarizations at a point in space
    yes=0;
    no=0;
    for row=1:a
        for col=1:b
            for dep=1:c
                if pr(row,col,dep)>pmin
                    yes=yes+1;
                else
                    no=no+1;
                    powered_positions(row,col,dep)=0;
                end
            end
        end
    end

    total=yes+no;

    read_accuracy=yes/total*100;

    set(handles.text60,'String',strcat(['Status:  Current accurracy
',num2str(read_accuracy),'%']));     %Write status to the screen

    scaling_factor=prmax/pmin;

    if (read_accuracy<=(reqacc+peracc) & read_accuracy>=(reqacc-peracc))
        done=1;
    elseif read_accuracy<(reqacc-peracc)
        if r==rinc
            rinc=rinc/2;                %so r can never equal zero
        end
        if round(lastread*1000)==round(read_accuracy*1000)      %Check to see
if accuracy is changing, if not, break
            done=1;
            acctoohigh=1;
        end
        r=r-rinc;
        rinc=rinc/2;
        lastread=read_accuracy;
    else
        r=r+rinc;
    end
end
```

guiplot_builtin.m

```
i=1;

%Create Interrogator gain matrix gi(theta,phi)
```

```
thetai=[0:step:180];
for phii=-90:step:90

gi=(3.136*((sin(pi/2.*cos(thetai*pi/180))./cos(thetai*pi/180).*cos(pi/2.*sin(
thetai*pi/180).*sin((phii)*pi/180))).^2).*(sin(thetai*pi/180)).^2);
%Patch Gain
    gmi(:,i)=gi';          %Create 3D gain matrix
    i=i+1;
end

%Solve the distances using scaling factor time min power required with
maximum tag gain
r=1/(4*pi)*sqrt(ptx*gmi*1.64*lam*lam*(1-abs(gammai)^2)*(1-
abs(gammatrx)^2)/prmax);        %3D distance matrix
i=1;

for theta=0:step:180
    j=1;
    for phi=-90:step:90

        x(i)=r(theta/step+1,j)*sin(theta*pi/180)*cos(phi*pi/180);
        y(i)=r(theta/step+1,j)*sin(theta*pi/180)*sin(phi*pi/180);
        z(i)=r(theta/step+1,j)*cos(theta*pi/180);
        j=j+1;
        i=i+1;

    end
end

scatter3(x,y,z,30,sqrt(x.^2+y.^2+z.^2),'filled')
rotate3d
axis auto
```

guiplot_load.m

```
%Read in interrogator antenna from user's file, assumes a theta x phi matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit12,'String'),'r');
text_gain=fscanf(fid,'%c');
gmi=str2num(text_gain);
fclose(fid);

%Check size of read matrix, should be 180/step+1 X 180/step+1
[rows,cols]=size(gmi);
stepcheck=180/(cols-1);
if rows~=cols
    set(handles.text60,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');      %Write error to the screen
    set(handles.edit12,'String','ERROR');
    break;
end
if step~=stepcheck
    set(handles.text60,'String','Status:  Abort --> File Does Not Match Input
Resolution');      %Write error to the screen
    set(handles.edit12,'String','ERROR');
```

```
    break;
end

gm3max=max(max(max(gm3)));
%Solve the distances using scaling factor time min power required with
maximum tag gain
r=1/(4*pi)*sqrt(ptx*gmi*gm3max*lam*lam*(1-abs(gammai)^2)*(1-
abs(gammatrx)^2)/prmax);        %3D distance matrix
i=1;

for theta=0:step:180
    j=1;
    for phi=-90:step:90

        x(i)=r(theta/step+1,j)*sin(theta*pi/180)*cos(phi*pi/180);
        y(i)=r(theta/step+1,j)*sin(theta*pi/180)*sin(phi*pi/180);
        z(i)=r(theta/step+1,j)*cos(theta*pi/180);
        j=j+1;
        i=i+1;

    end
end

scatter3(x,y,z,30,sqrt(x.^2+y.^2+z.^2),'filled')
rotate3d
axis auto
```

datalink_builtin.m

```
%Use reciprocity so the data can be determined in the same fashion as the
first link
%Transmit from the data base station to the tag which can be in any
orientation
%Valid when the interrogator and data base station have the same polarization

set(handles.text60,'String','Status:  Calculating Unreadable Points');
%Write status to the screen

%Determine distance between points and base station location
xdata=x-xdelta;
ydata=y-ydelta;
zdata=z-zdelta;
bsorient=str2num(get(handles.edit32,'String'));
%New axes for calculations
bstheta=bsorient(1)*pi/180;                             %Rotation of z-
axis in the x-z plane, 0 to 180 degrees, measured from z to x axis
bsphi=bsorient(2)*pi/180;                               %Rotation of x-
axis in the x-y plane, 0 to 360 degrees, measured from x to y axis

%Plot Data base station location
hold on
scatter3(xdelta,ydelta,zdelta,200,[0 0 0],'filled')
rotate3d
axis auto
```

```
%Plot antenna direction (x-axis)
rv=taglam;
xv=rv*sin(bstheta+pi/2)*cos(bsphi);
yv=rv*sin(bstheta+pi/2)*sin(bsphi);
zv=rv*cos(bstheta+pi/2);
xv=xv+xdelta;                           %Shift to antenna location
yv=yv+ydelta;
zv=zv+zdelta;
hold on
plot3([xdelta xv],[ydelta yv],[zdelta zv],'k','LineWidth',2)
rotate3d
hold on
%Plot antenna direction (z-axis)
xv=rv.*sin(bstheta)*cos(bsphi);
yv=rv.*sin(bstheta)*sin(bsphi);
zv=rv.*cos(bstheta);
xv=xv+xdelta;                           %Shift to antenna location
yv=yv+ydelta;
zv=zv+zdelta;
hold on
plot3([xdelta xv],[ydelta yv],[zdelta zv],'k','LineWidth',2)
rotate3d
hold on

i=sqrt(-1);
%Shift from new phi, z will be unchanged
r=sqrt(xdata.^2+ydata.^2+zdata.^2);
theta=angle(zdata+i*sqrt(xdata.^2+ydata.^2));       %use the angle command to
account for minus signs, rather than atan(sqrt(x^2+y^2)/z)
phiold=angle(xdata+i*ydata);                         %use the angle command to
account for minus signs, rather than atan(y/x)
phinew=phiold-bsphi;                                 %New phi angle with
respect to new x-axis

xnew1=r.*sin(theta).*cos(phinew);
ynew1=r.*sin(theta).*sin(phinew);
znew1=zdata;

%Shift from new theta, y will be unchanged
chiold=angle(znew1+i*xnew1);                         %Define new angle
measured from the z-axis toward the x-axis in the z-x plane
                                                     %use the angle command to
account for minus signs, rather than atan(xnew1/znew1)
psi=angle(ynew1+i*sqrt(xnew1.^2+znew1.^2));           %Define new angle
measured from the y-axis, atan(sqrt(x^2+z^2)/y)
chinew=chiold-bstheta;

%New data points with respect to data base station location, will remain
unchanged if colocated with interrogator
xnew2=r.*sin(psi).*sin(chinew);
ynew2=ynew1;
znew2=r.*sin(psi).*cos(chinew);
rnew2=sqrt(xnew2.^2+ynew2.^2+znew2.^2);

k=1;

%Do outside loop to increase speed
```

```
theta=angle(znew2+i*sqrt(xnew2.^2+ynew2.^2));
phi=angle(xnew2+i*ynew2);
factor=tagtxpower.*1.64*taglam*taglam/(4*pi)^2*(1-abs(gammattx)^2)*(1-
abs(gammabs)^2);

%Calculate for each xnew2, ynew2, znew2
for point=1:length(rnew2);
    thetacal=theta(point);
    phical=phi(point);                          %use the angle command to
account for minus signs, rather than atan(y/x)

    %Calculate gain at that point

gi=(3.136*((sin(pi/2.*cos(thetacal))./cos(thetacal).*cos(pi/2.*sin(thetacal).
*sin((phical)))).^2).*(sin(thetacal)).^2);        %Patch Gain

    %Calculate power received using optimum location, i.e. tag gain = 1.64
    prdata(point)=factor*gi/(rnew2(point))^2;
    if prdata(point)<(bssen*scaling_factor)
        xmark(k)=x(point);              %Point to mark as unreadable
        ymark(k)=y(point);
        zmark(k)=z(point);
        k=k+1;
    end
end

if k>1                  %only plot if there are unreadable points
    set(handles.text60,'String','Status:  Plotting Unreadable Points');
%Write status to the screen
    hold on
    scatter3(xmark,ymark,zmark,30,[1 0 1],'filled')
    rotate3d
end

minsen=min(prdata);
minsenlog=10*log10(minsen*1000/scaling_factor);                %minimum
sensitivity required for all points

set(handles.text64,'String',num2str(minsenlog));

set(handles.text60,'String','Status:  Complete -->');      %Write status to
the screen
```

datalink_load.m

```
%Use reciprocity so the data can be determined in the same fashion as the
first link
%Transmit from the data base station to the tag which can be in any
orientation
%Valid when the interrogator and data base station have the same polarization

set(handles.text60,'String','Status:  Calculating Unreadable Points');
%Write status to the screen

%Determine distance between points and base station location
```

```
xdata=x-xdelta;
ydata=y-ydelta;
zdata=z-zdelta;
bsorient=str2num(get(handles.edit32,'String'));
%New axes for calculations
bstheta=bsorient(1)*pi/180;                                %Rotation of z-
axis in the x-z plane, 0 to 180 degrees, measured from z to x axis
bsphi=bsorient(2)*pi/180;                                  %Rotation of x-
axis in the x-y plane, 0 to 360 degrees, measured from x to y axis

%Plot Data base station location
hold on
scatter3(xdelta,ydelta,zdelta,200,[0 0 0],'filled')
rotate3d
axis auto

%Plot antenna direction (x-axis)
rv=taglam;
xv=rv*sin(bstheta+pi/2)*cos(bsphi);
yv=rv*sin(bstheta+pi/2)*sin(bsphi);
zv=rv*cos(bstheta+pi/2);
xv=xv+xdelta;                       %Shift to antenna location
yv=yv+ydelta;
zv=zv+zdelta;
hold on
plot3([xdelta xv],[ydelta yv],[zdelta zv],'k','LineWidth',2)
rotate3d
hold on
%Plot antenna direction (z-axis)
xv=rv.*sin(bstheta)*cos(bsphi);
yv=rv.*sin(bstheta)*sin(bsphi);
zv=rv.*cos(bstheta);
xv=xv+xdelta;                       %Shift to antenna location
yv=yv+ydelta;
zv=zv+zdelta;
hold on
plot3([xdelta xv],[ydelta yv],[zdelta zv],'k','LineWidth',2)
rotate3d
hold on

i=sqrt(-1);
%Shift from new phi, z will be unchanged
r=sqrt(xdata.^2+ydata.^2+zdata.^2);
theta=angle(zdata+i*sqrt(xdata.^2+ydata.^2));       %use the angle command to
account for minus signs, rather than atan(sqrt(x^2+y^2)/z)
phiold=angle(xdata+i*ydata);                        %use the angle command to
account for minus signs, rather than atan(y/x)
phinew=phiold-bsphi;                                %New phi angle with
respect to new x-axis

xnew1=r.*sin(theta).*cos(phinew);
ynew1=r.*sin(theta).*sin(phinew);
znew1=zdata;

%Shift from new theta, y will be unchanged
chiold=angle(znew1+i*xnew1);                        %Define new angle
measured from the z-axis toward the x-axis in the z-x plane
```

```
                                                               %use the angle command to
account for minus signs, rather than atan(xnew1/znew1)
psi=angle(ynew1+i*sqrt(xnew1.^2+znew1.^2));           %Define new angle
measured from the y-axis, atan(sqrt(x^2+z^2)/y)
chinew=chiold-bstheta;

%New data points with respect to data base station location, will remain
unchanged if colocated with interrogator
xnew2=r.*sin(psi).*sin(chinew);
ynew2=ynew1;
znew2=r.*sin(psi).*cos(chinew);
rnew2=sqrt(xnew2.^2+ynew2.^2+znew2.^2);

%Read in tag data antenna from user's file, assumes a theta x phi matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit31,'String'),'r');
text_gain=fscanf(fid,'%c');
gtd=str2num(text_gain);
fclose(fid);

%Check size of read matrix, should be 180/N+1 X 360/N
[rows,cols]=size(gtd);
stepcheck=360/cols;
if (rows-1)*2~=cols
    set(handles.text60,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');     %Write error to the screen
    set(handles.edit31,'String','ERROR');
    break;
end
if step~=stepcheck
    set(handles.text60,'String','Status:  Abort --> File Does Not Match Input
Resolution');     %Write error to the screen
    set(handles.edit31,'String','ERROR');
    break;
end

gm3=gtd;
%Generate 3D gain matrix with polarization data
for pol=1:((90/step)+1)         %Point 1 is zero degrees, 91 is therefore 90
degrees
    gm3(:,:,pol)=gtd.*((cos((pol-1)*step*pi/180))^2);
end

%Calculate scaling factor for tag's data antenna
gi=1;                                %Dummy gain used to calculate scaling
factor, divides out in the end, value does not matter
r=0.1;                                  %distance
done=0;
rinc=0.1;                               %distance increment
lastread=0;
acctoohigh=0;

factor=ptx*gi*lam*lam/(4*pi)^2;      %Used to increase program speed, will
divide out at the end but makes easier to understand
a=180/step+1;
b=360/step;
c=90/step+1;
```

```
powered_positions=ones(a,b,c);      %Sets up a matrix to track points that
can receive enough power to operate, needed for next link

while done==0;                      %Use iteration to find scaling factor

    %Find maximum gain of input file
    gmax=max(max(gtd));                                     %Max gain of
tag's data antenna

    %Maximum power
    prmax=factor*gmax/(r)^2;
%Friis Equation

    %Power Matrix
    pr=factor.*gm3./(r)^2;
%Friis Equation

    %Determine number of points over threshold for all possible gains and
polarizations at a point in space
    yes=0;
    no=0;
    for row=1:a
        for col=1:b
            for dep=1:c
                if pr(row,col,dep)>pmin
                    yes=yes+1;
                else
                    no=no+1;
                    powered_positions(row,col,dep)=0;
                end
            end
        end
    end

    total=yes+no;

    read_accuracy=yes/total*100;

    set(handles.text60,'String',strcat(['Status:  Current accurracy
',num2str(read_accuracy),'%']));      %Write status to the screen

    scaling_factor=prmax/pmin;

    if (read_accuracy<=(reqacc+peracc) & read_accuracy>=(reqacc-peracc))
        done=1;
    elseif read_accuracy<(reqacc-peracc)
        if r==rinc
            rinc=rinc/2;                %so r can never equal zero
        end
        if round(lastread*1000)==round(read_accuracy*1000)      %Check to see
if accuracy is changing, if not, break
            done=1;
            acctoohigh=1;
        end
        r=r-rinc;
        rinc=rinc/2;
        lastread=read_accuracy;
```

```
    else
        r=r+rinc;
    end
end

if acctoohigh==1
    set(handles.text60,'String',strcat(['Status:  Maximum obtainable
accurracy ',num2str(read_accuracy),'%']));
else
    set(handles.text60,'String','Status:  Complete -->');     %Write status
to the screen
end

%Read in data base station antenna from user's file, assumes a theta x phi
matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit29,'String'),'r');
text_gain=fscanf(fid,'%c');
gi=str2num(text_gain);
fclose(fid);

%Check size of read matrix, should be 180/step+1 X 180/step+1
[rows,cols]=size(gi);
stepcheck=180/(cols-1);
if rows~=cols
    set(handles.text60,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');     %Write error to the screen
    set(handles.edit29,'String','ERROR');
    break;
end
if step~=stepcheck
    set(handles.text60,'String','Status:  Abort --> File Does Not Match Input
Resolution');     %Write error to the screen
    set(handles.edit29,'String','ERROR');
    break;
end

k=1;

%Do outside loop to increase speed
theta=angle(znew2+i*sqrt(xnew2.^2+ynew2.^2));
phi=angle(xnew2+i*ynew2);
factor=tagtxpower.*gmax*taglam*taglam/(4*pi)^2*(1-abs(gammattx)^2)*(1-
abs(gammabs)^2);

%Calculate for each xnew2, ynew2, znew2
for point=1:length(rnew2);
    thetacal=theta(point);
    phical=phi(point);                              %use the angle command to
account for minus signs, rather than atan(y/x)

    %Find gain point in user gain matrix using calculated angles
    thetaindex=round(thetacal/(step*pi/180))+1;
    phiindex=round((phical+pi/2)/(step*pi/180))+1;                  %need to
add pi/2 because phi is from -90 to 90 degrees
```

```
    giest=gi(thetaindex,phiindex);                        %Estimated gain
using nearest neighbor

    %Calculate power received using optimum location, i.e. tag gain = gmax
    prdata(point)=factor*giest/(rnew2(point))^2;
    if prdata(point)<(bssen*scaling_factor)
        xmark(k)=x(point);              %Point to mark as unreadable
        ymark(k)=y(point);
        zmark(k)=z(point);
        k=k+1;
    end
end

if k>1                  %only plot if there are unreadable points
    set(handles.text60,'String','Status:  Plotting Unreadable Points');
%Write status to the screen
    hold on
    scatter3(xmark,ymark,zmark,30,[1 0 1],'filled')
    rotate3d
end

minsen=min(prdata);
minsenlog=10*log10(minsen*1000/scaling_factor);                %minimum
sensitivity required for all points

set(handles.text64,'String',num2str(minsenlog));

set(handles.text60,'String','Status:  Complete -->');      %Write status to
the screen
```

APPENDIX F

MATLAB CODE FOR BACKSCATTER EVALUATION TOOL

Files

evaltoolgui_backscatter.m
sfsolve_builtin.m
sfsolve_load.m
guiplot_builtin.m
guiplot_load.m
datalink_builtin.m
datalink_load.m

evaltoolgui_backscatter.m

```
function varargout = evaltoolgui_backscatter(varargin)
% EVALTOOLGUI_BACKSCATTER Application M-file for evaltoolgui_backscatter.fig
%    FIG = EVALTOOLGUI_BACKSCATTER launch evaltoolgui_backscatter GUI.
%    EVALTOOLGUI_BACKSCATTER('callback_name', ...) invoke the named callback.

% Last Modified by GUIDE v2.0 22-Oct-2005 10:42:36

if nargin == 0  % LAUNCH GUI

   fig = openfig(mfilename,'reuse');

   % Generate a structure of handles to pass to callbacks, and store it.
   handles = guihandles(fig);
   guidata(fig, handles);

   if nargout > 0
      varargout{1} = fig;
   end

elseif ischar(varargin{1}) % INVOKE NAMED SUBFUNCTION OR CALLBACK

   try
      if (nargout)
         [varargout{1:nargout}] = feval(varargin{:}); % FEVAL switchyard
      else
         feval(varargin{:}); % FEVAL switchyard
```

```
      end
   catch
      disp(lasterr);
   end

end


%| ABOUT CALLBACKS:
%| GUIDE automatically appends subfunction prototypes to this file, and
%| sets objects' callback properties to call them through the FEVAL
%| switchyard above. This comment describes that mechanism.
%|
%| Each callback subfunction declaration has the following form:
%| <SUBFUNCTION_NAME>(H, EVENTDATA, HANDLES, VARARGIN)
%|
%| The subfunction name is composed using the object's Tag and the
%| callback type separated by '_', e.g. 'slider2_Callback',
%| 'figure1_CloseRequestFcn', 'axis1_ButtondownFcn'.
%|
%| H is the callback object's handle (obtained using GCBO).
%|
%| EVENTDATA is empty, but reserved for future use.
%|
%| HANDLES is a structure containing handles of components in GUI using
%| tags as fieldnames, e.g. handles.figure1, handles.slider2. This
%| structure is created at GUI startup using GUIHANDLES and stored in
%| the figure's application data using GUIDATA. A copy of the structure
%| is passed to each callback.  You can store additional information in
%| this structure at GUI startup, and you can change the structure
%| during callbacks.  Call guidata(h, handles) after changing your
%| copy to replace the stored original so that subsequent callbacks see
%| the updates. Type "help guihandles" and "help guidata" for more
%| information.
%|
%| VARARGIN contains any extra arguments you have passed to the
%| callback. Specify the extra arguments by editing the callback
%| property in the inspector. By default, GUIDE sets the property to:
%| <MFILENAME>('<SUBFUNCTION_NAME>', gcbo, [], guidata(gcbo))
%| Add any extra arguments after the last argument, before the final
%| closing parenthesis.



% --------------------------------------------------------------------
function varargout = pushbutton1_Callback(h, eventdata, handles, varargin)

warning off                                                %Suppress divide by
zero warnings

global reqacc peracc step pmin ptx f lam prmax powered_positions gm3
scaling_factor ptx
global gammai gammatag gammabs

set(handles.text60,'String','Status:  Finding Scaling Factor...');     %Write
status to the screen
```

```
%Read in user entered data
reqacc=str2num(get(handles.edit1,'String'));            %Data from system
boxes
peracc=str2num(get(handles.edit5,'String'));
step=str2num(get(handles.edit6,'String'));

%step must be a factor of 90
steptest=round(90/step);
if steptest*step~=90
    set(handles.text60,'String','Status:  Abort --> Step must be a factor of
90');     %Write error to the screen
    break;
end

%Reflection Coefficients
gammai=str2num(get(handles.edit11,'String'));           %Interrogator
gammatag=str2num(get(handles.edit23,'String'));         %Tag
gammabs=str2num(get(handles.edit28,'String'));          %Data Base Station

pmin=(str2num(get(handles.edit25,'String')))*10^-6;     %Min power level need
by the tag
ptx=str2num(get(handles.edit9,'String'));               %Interrogating
transmitter power
f=str2num(get(handles.edit10,'String'));;               %Interrogating
frequency in MHz
lam=300/f;                                              %Free space
wavelength

%Plot a square halfwave length antenna, x-axis, and z-axis
x=[0 0 0 0];
y=[-lam/2 lam/2 -lam/2 lam/2];
z=[lam/2 lam/2 -lam/2 -lam/2;lam/2 lam/2 -lam/2 -lam/2;lam/2 lam/2 -lam/2 -
lam/2;lam/2 lam/2 -lam/2 -lam/2];
mesh(x,y,z,'FaceColor',[0 0 0])
hold on
x=[0 lam];
y=[0 0];
z=y;
plot3(x,y,z,'k','LineWidth',2)
hold on
z=[0 lam];
x=[0 0];
plot3(x,y,z,'k','LineWidth',2)
hold on
axis([-0.5 0.5 -0.5 0.5 -0.5 0.5])
rotate3d

if get(handles.checkbox5,'Value')==1                    %Is the user
requesting the built-in tag gain
    sfsolve_builtin                                     %Solve the scaling
factor and actual read accuracy
else
    sfsolve_load                                        %Load user defined
gain file and solve scaling factor
end

set(handles.text9,'String',num2str(read_accuracy));     %Return data to gui
```

```
set(handles.text15,'String',num2str(scaling_factor));

if acctoohigh==1
    set(handles.text60,'String',strcat(['Status:  Maximum obtainable
accurracy ',num2str(read_accuracy),'%']));
else
    set(handles.text60,'String','Status:  Complete -->');     %Write status
to the screen
end

% ------------------------------------------------------------------
function varargout = pushbutton2_Callback(h, eventdata, handles, varargin)

global reqacc peracc step pmin ptx f lam prmax x y z r gmi gm3
global gammai gammatag gammabs

set(handles.text60,'String','Status:  Plotting Data');     %Write status to
the screen

if get(handles.checkbox2,'Value')==1                       %returns a one when
box is checked, zero otherwise
    guiplot_builtin                                                %Plot the
data
else
    guiplot_load                                           %Plot the data using
the user loaded gain
end

set(handles.text60,'String','Status:  Plotting Complete');     %Write status
to the screen

% ------------------------------------------------------------------
function varargout = pushbutton3_Callback(h, eventdata, handles, varargin)

global reqacc peracc step pmin ptx f lam prmax ptx gmi gm3
global x y z powered_positions gm3 r scaling_factor
global gammai gammatag gammabs

%These link calculations assume that the tag's antennas are at the same
location and each antenna has the same axes
%i.e. theta=0, phi=0 is the z-axis for both antennas, etc.
%This assumption is needed to transfer the points from the first link in
order to know what point will not be
%able to communicate due the lack of power.

%Data from tag and data receiver boxes
bssenlog=str2num(get(handles.edit26,'String'));         %Base station
sensitivity

xdelta=0;                                                  %Data base station
assumed to be colocated with interrogator
ydelta=0;
zdelta=0;
bssen=10^(bssenlog/10)/1000;                               %Convert dBm to
linear sensitivity in watts
```

```
if get(handles.checkbox6,'Value')==1                    %returns a one when
box is checked, zero otherwise
    datalink_builtin                                    %Plot the data
else
    datalink_load                                       %Plot the data using
the user loaded gain
end

% --------------------------------------------------------------------
function varargout = pushbutton4_Callback(h, eventdata, handles, varargin)
hold off
plot3(0,0,0,'Color',[0 0 0])
hold on
grid on
rotate3d

set(handles.text60,'String','Status:  Ready');      %Write status to the
screen
set(handles.text9,'String',' ');                    %Clear data on gui
set(handles.text15,'String',' ');
set(handles.text64,'String',' ');


% --------------------------------------------------------------------
function varargout = checkbox1_Callback(h, eventdata, handles, varargin)

global x y z

if get(handles.checkbox1,'Value')==1                    %returns a one when
box is checked, zero otherwise

    figure(1)
    scatter3(x,y,z,30,sqrt(x.^2+y.^2+z.^2),'filled')    %Plots the data in an
external window for full screen view
    rotate3d
    title('External 3D Plot of the Area of Operation')
end

% --------------------------------------------------------------------
function varargout = checkbox2_Callback(h, eventdata, handles, varargin)

if get(handles.checkbox2,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit12,'String','ideal_patch.txt');
else
    [ifilename, ipathname] = uigetfile('*.txt', 'Choose a Gain File');
    if ifilename~=0
        set(handles.edit12,'String',ifilename);
    else
        set(handles.checkbox2,'Value',1)                    %Cancel button
pressed or error
        set(handles.edit12,'String','ideal_patch.txt');
    end
end

% --------------------------------------------------------------------
function varargout = checkbox5_Callback(h, eventdata, handles, varargin)
```

```
if get(handles.checkbox5,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit24,'String','ideal_dipole.txt');
else
    [trxfilename, trxpathname] = uigetfile('*.txt', 'Choose a Gain File');
    if trxfilename~=0
        set(handles.edit24,'String',trxfilename);
    else
        set(handles.checkbox5,'Value',1)                    %Cancel button
pressed or error
        set(handles.edit24,'String','ideal_dipole.txt');
    end
end

% --------------------------------------------------------------------
function varargout = checkbox6_Callback(h, eventdata, handles, varargin)

if get(handles.checkbox6,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit29,'String','ideal_patch.txt');
else
    [bsfilename, bspathname] = uigetfile('*.txt', 'Choose a Gain File');
    if bsfilename~=0
        set(handles.edit29,'String',bsfilename);
    else
        set(handles.checkbox6,'Value',1)                    %Cancel button
pressed or error
        set(handles.edit29,'String','ideal_patch.txt');
    end
end

% --------------------------------------------------------------------
function varargout = checkbox7_Callback(h, eventdata, handles, varargin)

if get(handles.checkbox7,'Value')==1                    %returns a one when
box is checked, zero otherwise
    set(handles.edit31,'String','ideal_dipole.txt');
else
    [ttxfilename, ttxpathname] = uigetfile('*.txt', 'Choose a Gain File');
    if ttxfilename~=0
        set(handles.edit31,'String',ttxfilename);
    else
        set(handles.checkbox7,'Value',1)                    %Cancel button
pressed or error
        set(handles.edit31,'String','ideal_dipole.txt');
    end
end

% --------------------------------------------------------------------
function varargout = edit1_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit2_Callback(h, eventdata, handles, varargin)
```

```
set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit3_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit4_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit5_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit6_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit9_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit10_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit11_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit12_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));

% --------------------------------------------------------------------
function varargout = edit21_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));
```

```
% --------------------------------------------------------------------
function varargout = edit22_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit23_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit24_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit25_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit26_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit28_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit29_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit30_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit31_Callback(h, eventdata, handles, varargin)

set(h,'Value',str2num(get(h,'string')));


% --------------------------------------------------------------------
function varargout = edit32_Callback(h, eventdata, handles, varargin)
```

```
set(h,'Value',str2num(get(h,'string')));
```

sfsolve_builtin.m

```
gi=1;                            %Dummy gain used to calculate scaling
factor, divides out in the end, value does not matter
theta=[0:step:180];              %Need both 0 and 180 degrees because gain
can be different
phi=[0:step:360-step];           %Subtract off step so 0 deg isn't counted
twice

%Generate 3D gain matrix
gt=1.64*(((cos(pi/2.*cos(theta*pi/180)))./sin(theta*pi/180)).^2);
gt(1)=0;                    %Set endpoints to zero, infinity and round off
problems
gt(length(gt))=0;
gm2=ones(180/step+1,360/step);
gm3=gm2;
for phi=1:360/step
    gm2(:,phi)=gt';         %Create 3D gain matrix
end

%Generate 3D gain matrix with polarization data
for pol=1:((90/step)+1)         %Point 1 is zero degrees, 91 is therefore 90
degrees
    gm3(:,:,pol)=gm2.*((cos((pol-1)*step*pi/180))^2);
end

r=0.1;                              %distance
done=0;
rinc=0.1;                           %distance increment
lastread=0;
acctoohigh=0;

factor=ptx*gi*lam*lam/(4*pi)^2;     %Used to increase program speed, will
divide out at the end but makes easier to understand
a=180/step+1;
b=360/step;
c=90/step+1;
powered_positions=ones(a,b,c);      %Sets up a matrix to track points that
can receive enough power to operate, needed for next link

while done==0;                      %Use iteration to find scaling factor

    %Maximum power
    prmax=factor*1.64/(r)^2;
%Friis Equation

    %Power Matrix
    pr=factor.*gm3./(r)^2;
%Friis Equation

    %Determine number of points over threshold for all possible gains and
polarizations at a point in space
```

```
    yes=0;
    no=0;
    for row=1:a
        for col=1:b
            for dep=1:c
                if pr(row,col,dep)>pmin
                    yes=yes+1;
                else
                    no=no+1;
                    powered_positions(row,col,dep)=0;
                end
            end
        end
    end

    total=yes+no;

    read_accuracy=yes/total*100;

    set(handles.text60,'String',strcat(['Status:  Current accurracy
',num2str(read_accuracy),'%']));     %Write status to the screen

    scaling_factor=prmax/pmin;

    if (read_accuracy<=(reqacc+peracc) & read_accuracy>=(reqacc-peracc))
        done=1;
    elseif read_accuracy<(reqacc-peracc)
        if r==rinc
            rinc=rinc/2;                %so r can never equal zero
        end
        if round(lastread*1000)==round(read_accuracy*1000)      %Check to see
if accuracy is changing, if not, break
            done=1;
            acctoohigh=1;
        end
        r=r-rinc;
        rinc=rinc/2;
        lastread=read_accuracy;
    else
        r=r+rinc;
    end
end
```

sfsolve_load.m

```
gi=1;                               %Dummy gain used to calculate scaling
factor, divides out in the end, value does not matter

%Read in tag power antenna from user's file, assumes a theta x phi matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit24,'String'),'r');
text_gain=fscanf(fid,'%c');
gm2=str2num(text_gain);
fclose(fid);
```

```
%Check size of read matrix, should be 180/N+1 X 360/N
[rows,cols]=size(gm2);
stepcheck=360/cols;
if (rows-1)*2~=cols
    set(handles.text60,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');      %Write error to the screen
    set(handles.edit24,'String','ERROR');
    break;
end
if step~=stepcheck
    set(handles.text60,'String','Status:  Abort --> File Does Not Match Input
Resolution');      %Write error to the screen
    set(handles.edit24,'String','ERROR');
    break;
end

gm3=gm2;
%Generate 3D gain matrix with polarization data
for pol=1:((90/step)+1)         %Point 1 is zero degrees, 91 is therefore 90
degrees
    gm3(:,:,pol)=gm2.*((cos((pol-1)*step*pi/180))^2);
end

r=0.1;                              %distance
done=0;
rinc=0.1;                           %distance increment
lastread=0;
acctoohigh=0;

factor=ptx*gi*lam*lam/(4*pi)^2;     %Used to increase program speed, will
divide out at the end but makes easier to understand
a=180/step+1;
b=360/step;
c=90/step+1;
powered_positions=ones(a,b,c);      %Sets up a matrix to track points that
can receive enough power to operate, needed for next link

while done==0;                      %Use iteration to find scaling factor

    %Find maximum gain of input file
    gmax=max(max(gm2));

    %Maximum power
    prmax=factor*gmax/(r)^2;
%Friis Equation

    %Power Matrix
    pr=factor.*gm3./(r)^2;
%Friis Equation

    %Determine number of points over threshold for all possible gains and
polarizations at a point in space
    yes=0;
    no=0;
    for row=1:a
        for col=1:b
            for dep=1:c
```

```
                if pr(row,col,dep)>pmin
                    yes=yes+1;
                else
                    no=no+1;
                    powered_positions(row,col,dep)=0;
                end
            end
        end
    end

    total=yes+no;

    read_accuracy=yes/total*100;

    set(handles.text60,'String',strcat(['Status:  Current accurracy
',num2str(read_accuracy),'%']));      %Write status to the screen

    scaling_factor=prmax/pmin;

    if (read_accuracy<=(reqacc+peracc) & read_accuracy>=(reqacc-peracc))
        done=1;
    elseif read_accuracy<(reqacc-peracc)
        if r==rinc
            rinc=rinc/2;                %so r can never equal zero
        end
        if round(lastread*1000)==round(read_accuracy*1000)      %Check to see
if accuracy is changing, if not, break
            done=1;
            acctoohigh=1;
        end
        r=r-rinc;
        rinc=rinc/2;
        lastread=read_accuracy;
    else
        r=r+rinc;
    end
end
```

guiplot_builtin.m

```
i=1;

%Create Interrogator gain matrix gi(theta,phi)
thetai=[0:step:180];
for phii=-90:step:90

gi=(3.136*((sin(pi/2.*cos(thetai*pi/180))./cos(thetai*pi/180).*cos(pi/2.*sin(
thetai*pi/180).*sin((phii)*pi/180)))).^2).*(sin(thetai*pi/180)).^2);
%Patch Gain
    gmi(:,i)=gi';          %Create 3D gain matrix
    i=i+1;
end

%Solve the distances using scaling factor times min power required with
maximum tag gain
```

```
r=1/(4*pi)*sqrt(ptx*gmi*1.64*lam*lam*(1-
abs(gammai)^2)*((1+gammatag)^2)/prmax);        %3D distance matrix
i=1;

for theta=0:step:180
    j=1;
    for phi=-90:step:90

        x(i)=r(theta/step+1,j)*sin(theta*pi/180)*cos(phi*pi/180);
        y(i)=r(theta/step+1,j)*sin(theta*pi/180)*sin(phi*pi/180);
        z(i)=r(theta/step+1,j)*cos(theta*pi/180);
        j=j+1;
        i=i+1;

    end
end

scatter3(x,y,z,30,sqrt(x.^2+y.^2+z.^2),'filled')
rotate3d
axis auto
```

guiplot_load.m

```
%Read in interrogator antenna from user's file, assumes a theta x phi matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit12,'String'),'r');
text_gain=fscanf(fid,'%c');
gmi=str2num(text_gain);
fclose(fid);

%Check size of read matrix, should be 180/step+1 X 180/step+1
[rows,cols]=size(gmi);
stepcheck=180/(cols-1);
if rows~=cols
    set(handles.text60,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');     %Write error to the screen
    set(handles.edit12,'String','ERROR');
    break;
end
if step~=stepcheck
    set(handles.text60,'String','Status:  Abort --> File Does Not Match Input
Resolution');     %Write error to the screen
    set(handles.edit12,'String','ERROR');
    break;
end

gm3max=max(max(max(gm3)));
%Solve the distances using scaling factor times min power required with
maximum tag gain
r=1/(4*pi)*sqrt(ptx*gmi*gm3max*lam*lam*(1-
abs(gammai)^2)*((1+gammatag)^2)/prmax);        %3D distance matrix
i=1;

for theta=0:step:180
    j=1;
```

```
    for phi=-90:step:90

        x(i)=r(theta/step+1,j)*sin(theta*pi/180)*cos(phi*pi/180);
        y(i)=r(theta/step+1,j)*sin(theta*pi/180)*sin(phi*pi/180);
        z(i)=r(theta/step+1,j)*cos(theta*pi/180);
        j=j+1;
        i=i+1;

    end
end

scatter3(x,y,z,30,sqrt(x.^2+y.^2+z.^2),'filled')
rotate3d
axis auto
```

datalink_builtin.m

```
%Use reciprocity so the data can be determined in the same fashion as the
first link
%Transmit from the data base station to the tag which can be in any
orientation
%Valid when the interrogator and data base station have the same polarization

set(handles.text60,'String','Status:  Calculating Unreadable Points');
%Write status to the screen

r=sqrt(x.^2+y.^2+z.^2);
rmax=max(r);                                           %Distance to farthest point
gmimax=max(max(max(gmi)));                             %Maximum gain value of both
built-in antennas
gm3max=1.64;                                           %Maximum gain value for the tag
antenna
theta=angle(z+i*sqrt(x.^2+y.^2));
phi=angle(x+i*y);

%Calculate the minimum sensivity required
%Calculate sigma
o=lam*lam*(gm3max/scaling_factor).^2/(4*pi)*(1-abs(gammatag)^2)^2;

%Calculate the data base station received power matrix
prbs=pmin*o*scaling_factor/(gm3max*(1-
abs(gammatag)^2))*gmimax/(4*pi*rmax^2)*(1-abs(gammabs)^2);

if prbs<bssen
    scatter3(x,y,z,30,[1 0 1],'filled')              %Color all point, read
accuracy is not met
    rotate3d
end

minsen=prbs;
minsenlog=10*log10(minsen*1000);                       %minimum sensitivity required
for all points

set(handles.text64,'String',num2str(minsenlog));
```

```
set(handles.text60,'String','Status:  Complete -->');      %Write status to
the screen
```

datalink_load.m

```
%Use reciprocity so the data can be determined in the same fashion as the
first link
%Transmit from the data base station to the tag which can be in any
orientation
%Valid when the interrogator and data base station have the same polarization

set(handles.text60,'String','Status:  Calculating Unreadable Points');
%Write status to the screen

%Read in data base station antenna from user's file, assumes a theta x phi
matrix
%theta values are rows, phi values are columns
fid=fopen(get(handles.edit29,'String'),'r');
text_gain=fscanf(fid,'%c');
gbs=str2num(text_gain);
fclose(fid);

%Check size of read matrix, should be 180/step+1 X 180/step+1
[rows,cols]=size(gbs);
stepcheck=180/(cols-1);
if rows~=cols
    set(handles.text60,'String','Status:  Abort --> Invalid Gain Matrix
Dimensions');      %Write error to the screen
    set(handles.edit29,'String','ERROR');
    break;
end
if step~=stepcheck
    set(handles.text60,'String','Status:  Abort --> File Does Not Match Input
Resolution');      %Write error to the screen
    set(handles.edit29,'String','ERROR');
    break;
end

r=sqrt(x.^2+y.^2+z.^2);
rmax=max(r);                                  %Distance to farthest point
gmimax=max(max(max(gmi)));                    %Maximum gain value of
interrogating antenna
gm3max=max(max(max(gm3)));                    %Maximum gain value for the tag
antenna
gbsmax=max(max(max(gbs)));                    %Maximum gain value of the base
station antenna
theta=angle(z+i*sqrt(x.^2+y.^2));
phi=angle(x+i*y);

%Calculate the minimum sensivity required
%Calculate sigma
o=lam*lam*(gm3max/scaling_factor).^2/(4*pi)*(1-abs(gammatag)^2)^2;

k=1;
%Calculate the minimum data base station received power
```

```
for point=1:length(r)
    thetacal=theta(point);
    phical=phi(point);                          %use the angle command to
account for minus signs, rather than atan(y/x)

    %Find gain point in user gain matrix using calculated angles
    thetaindex=round(thetacal/(step*pi/180))+1;
    phiindex=round((phical+pi/2)/(step*pi/180))+1;                  %need to
add pi/2 because phi is from -90 to 90 degrees

    gbspoint=gbs(thetaindex,phiindex);

    prbs(point)=pmin*o*scaling_factor/(gm3max*(1-
abs(gammatag)^2))*gbspoint/(4*pi*r(point)^2)*(1-abs(gammabs)^2);
    if prbs(point)<bssen
        xmark(k)=x(point);              %Point to mark as unreadable
        ymark(k)=y(point);
        zmark(k)=z(point);
        k=k+1;
    end
end


if k>1                  %only plot if there are unreadable points
    set(handles.text60,'String','Status:  Plotting Unreadable Points');
%Write status to the screen
    hold on
    scatter3(xmark,ymark,zmark,30,[1 0 1],'filled')
    rotate3d
end


minsen=min(prbs);
minsenlog=10*log10(minsen*1000);                %minimum sensitivity required
for all points, 1000 converts to mW

set(handles.text64,'String',num2str(minsenlog));

set(handles.text60,'String','Status:  Complete -->');      %Write status to
the screen
```

BIBLIOGRAPHY

[1] Constantine A. Balanis, Antenna Theory: Analysis and Design. Second Edition. John Wiley & Sons, Inc. New York: New York, 1997.

[2] Kevin Powell. SG3_200207_324R_MatricsPres.ppt, Matrics, Inc., presented in Vienna, July 17, 2002.

[3] FCC Rules: CFR Title 47, Section 15.247.

[4] Klaus Finkenzeller, RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification. Second Edition. John Wiley & Sons, Inc. New York: New York, 2003.

[5] Harry Stockman, "Communications by Means of Reflected Power" Proceedings of the IRE, pp 1196-1204, October 1948.

[6] Jerry Landt, "Shrouds of Time: The History of RFID" An AIM Publication, 2001, www.aimglobal.org.

[7] Smail Tedjini, Tan-Phu Vuong, Vincent Beroulle, "Antennas for RFID Tags" Joint SOC-EUSAI Conference, Grenoble, October 2005.

[8] IEEE Transactions on Antennas and Propagation, Vols. AP-17, No. 3, May 1969; Vol. AP-22, No. 1, January 1974; and Vol. AP-31, No. 6, Part II, November 1983.

[9] Warren L. Stutzman and Gary A. Thiele, Antenna Theory and Design. John Wiley & Sons, Inc. New York: New York, 1981.

[10] John D. Kraus and Ronald J. Marhefka, Antennas: For All Applications. Third Edition, McGraw-Hill, 2002.

[11] P.R. Foster and R.A. Burberry, "Antenna Problems in RFID Systems," The Institute of Electrical Engineers, Savory Place, London, 1999.

[12] S.K. Padhi, N.C. Karmakar and C.L. Law, "Daul Polarized Reader Antenna Array for RFID Applications," IEEE, 2003.

[13] Daniel Stancil, "Antennas for Wireless Communications," Lectures at Carnegie-Mellon University, 2005.

[14] Matthew N. O. Sadiku, Elements of Electromagnetics Third Edition, Oxford University Press. NewYork: New York, 2001, pg. 66, 84.

[15] "SuperNEC: MOM Technical Reference Manual," Version 2.7, http://www.supernec.com/manuals/snmomtrm.

[16] Alan V. Oppenheim, Alan S. Willsky, Signals and Systems Second Edition, Prentice Hall, 1997, pg. 92.

[17] George B. Thomas, Jr. and Ross L. Finney, Calculus. 9th Edition. Addison-Wesley Publishing Company, 1996.

[18] S. K. Schelkunoff and H. T. Friis, Antennas: Theory and Practice, Wiley, New York, 1952, pg. 229-244, 351-353.

[19] Reimar Marg and Arne F. Jacob, "Efficiency Considerations in the Design of Aperture Coupled Patch Antennas with Stripline Feed," Annual Report 1995, Institut fur Hochfrequenztechnik, TU Braunschweig.

[20] Leo Mats, Antenna Simulation Data, University of Pittsburgh, 2005.

[21] "Automatic Identification - Radio Frequency Identification for Item Management Part 6: Mode 3 - Physical Layer, Anti Collision System and Protocols for Ultra High Frequency (UHF) Systems," ISO WD 18000-6 Mode 3, Annex 4, 2002.

[22] Robert K. Schneider, "A Re-look at Antenna In-band RCSR via Load Mismatching," IEEE, 1996.

[23] Johnson J.H. Wang, Chang W. Choi, and Rick L. Moore, "Precision Experimental Characterization of the Scattering and Radiation Properties of Antennas," IEEE Transactions on Antennas and Propagation, Vol. AP-30, No. 1, January 1982.

[24] W.K. Kahn, "Minimum-Scattering Antennas," IEEE APS, September 1965, pg. 671-675.

[25] R.C. Hansen, "Structural, Residual and Antenna Mode Scattering," TR-812-1, December 1986.

[26] Katina Michael, Luke McCathie, "The Pros and Cons of RFID in Supply Chain Management," Proceedings of the International Conference on Mobile Business (ICMB '05), 2005.

[27] Young M. Lee, Feng Cheng, and Ying Tat Leung, "Exploring the Impact of RFID on Supply Chain Dynamics," Proceedings of the 2004 Winter Simulation Conference, 2004.

[28] Mary Catherine O'Connor, "Integral RFID Looks for RF Hot Spot," RFID Journal. http://www.rfidjournal.com/article/articleview/1462/1/1/, March 23, 2005.

[29] "RFID Design Tools," Softronica, Inc. http://www.softronica.org/ShowProductInfo.php?product_id=ANTLR6000, ANTLR1.pdf.