

WORCESTER POLYTECHNIC INSTITUTE
MITRE CORPORATION

Project Number PA11

Element to Element Coupling Analysis in Transmit and Receive Phased Array

A Major Qualifying Project

Daphne Gorman

Project Sponsor: MITRE Corporation

Janet Werth
Frank Kolak
Steven Best

Approved by:

Professor Reinhold Ludwig _____

Professor Michael Gennert _____

Professor Sergey Makarov _____

Abstract

Despite an abundance of commercial simulation software, there is a need for a dedicated computational tool that can calculate the coupling between individual elements in two spatially separate phased arrays. The use of commercial simulation software is limited by high cost, resource availability and/or long simulation times. The objective of this MQP was the implementation of a MATLAB program that performs this coupling calculation. In addition to designing, developing and implementing the calculations in MATLAB, the code's predictions were validated against analytical and commercially available codes. Furthermore, this project also included an analysis on the accuracy and runtime of the software.

Relevant Definitions and Abbreviations

Coupling Energy transfer between two arrays or array elements

Insertion Loss Power transmitted by one element and received by a separate element given by the equation:

$$IL(dB) = 10 \log_{10} \frac{P_{transmitted}}{P_{received}} \quad (1)$$

Return Loss Power reflected back at the transmitting element given by the equation:

$$RL(dB) = 10 \log_{10} \frac{P_{incident}}{P_{reflected}} \quad (2)$$

Rx Shorthand notation for “Receive” or “Receiver”

S₁₁ Voltage return loss as defined by the equation:

$$RL(dB) = |20 \log_{10} |S_{11}| | \quad (3)$$

S₁₂ Voltage insertion loss as defined by the equation:

$$IL(dB) = -20 \log_{10} |S_{21}| \quad (4)$$

Tx Shorthand notation for “Transmit” or “Transmitter”

X-Band Frequency Range from 8.0 to 12.0 GHz

Table of Contents

Abstract	i
Relevant Definitions and Abbreviations	ii
Table of Contents	iii
List of Figures	iv
List of Tables	v
1. Introduction	6
1.1. Basic Problem Formulation	6
1.2. Problem Statement	8
2. Research Objectives	9
2.1. Goal 1: Create Program Calculator	9
2.2. Goal 2: Validate Program Accuracy	9
3. Approach	10
3.1. Theory	10
3.1.1. Friis' Transmission Equation.....	10
3.2. Methodology	11
3.2.1. Design and Coding of Program	12
3.2.2. Improving Program Runtime	13
4. Results and Observations	13
4.1. Results of CST Simulations	13
4.1.1. Single Isolated Radiating element	13
4.1.2. Single Radiating Element in 3x3 Transmit Array	16
4.1.3. Single Radiating Element in a 5x5 Tx Array.....	19
4.1.4. Results of Transmit and Receive Element Simulation	21
4.1.5. Results of Transmit and Receive Array Simulations.....	23
4.1.6. Summary of CST Results	25
4.2. Results of Code Validation	26
4.3. Results of Time Experiments	26
5. Conclusions	28
6. Future Work	28
Bibliography	29
Appendix I: Complete MATLAB Function	30
Appendix II: Functional GUI Code	32
Appendix III: Further Results from CST Simulations	33
Single Isolated Radiating Element.....	33
3x3 Array: Center Element Excited, Remaining Elements Are Open Circuited.....	34
3x3 Array: Center Element Excited, Remaining Elements Are Terminated to 50Ω.....	35
5x5 Array: Center Element Excited, Remaining Elements Are Open Circuited.....	37
3x3 Array: Center Element Excited, Remaining Elements Are Terminated to 50Ω.....	38

List of Figures

Figure 1: Undesirable Cross-Talk Between Antenna Elements	7
Figure 2: AdvHDR SATCOM System (3 Tx Arrays, 1 Rx Array) on a Submarine Mast To Be Mounted on a Submarine as Shown.....	8
Figure 3: Scalar Friis' Transmission Equation	11
Figure 4: User Interface for Input Parameters	12
Figure 5: Single Radiating Patch Element.....	14
Figure 6: Single Element Gain Pattern in a) an azimuth sweep at elevation = 0° and b) at all angles	15
Figure 7 - Comparison of Simulated and Approximated Gains for Single Element	16
Figure 8: Single Element in a 3x3 Array	16
Figure 9: Gain Pattern of Element in 3x3 Array, Remaining Elements Open Circuited..	17
Figure 10: Gain of Element in 3x3 Array, Remaining Elements Terminated to 50Ω	18
Figure 11: Single Element in a 5x5 Array	19
Figure 12: Gain Pattern of Element in 5x5 Array, Remaining Elements Open Circuited	20
Figure 13: Gain of Element in 5x5 Array, Remaining Elements Terminated to 50Ω	21
Figure 14: Transmit and Receive Element Simulated	22
Figure 15: Return and Insertion Loss of Single Tx and Rx Element Simulation	22
Figure 16: Full Array Geometry	23
Figure 17: S-Parameters of Tx and Rx Elements, Other Elements Open Circuited	24
Figure 18: S-Parameters of Tx and Rx Elements, Other Elements Terminated to 50Ω ...	25
Figure 19: Snippet of MatLab Program's Nested FOR-Loops	27
Figure 20: Return Loss, Center Frequency 8.14 GHz	33
Figure 21: Z-Smith Chart, $Z_1=(50.42-0.08j)\Omega$	33
Figure 22: Return Loss, Center Frequency 8.16 GHz	34
Figure 23: Z-Smith Chart, $Z_1=(51.67-3.92j)\Omega$	34
Figure 24: Return Loss, Center Frequency 8.15 GHz	35
Figure 25: All S-Parameters.....	35
Figure 26: Z-Smith Chart, $Z_1=(50.46-0.14j)\Omega$	36
Figure 27: Return Loss, Center Frequency 8.16 GHz	37
Figure 28: Z-Smith Chart, $Z_1=(51.51-4.11j)\Omega$	37
Figure 29: Return Loss, Center Frequency 8.15 GHz	38
Figure 30: All S-Parameters.....	38
Figure 31: Z-Smith Chart, $Z_1=(50.30+0.18j)\Omega$	40

List of Tables

Table 1: Single Element Radiation Pattern Observations.....	25
Table 2: Time Analysis Experiment Results	27
Table 3: Summary of Coupling, Power Coupled to Adjacent Elements is 5%	36
Table 4: Summary of Coupling, Power Coupled to Adjacent Elements is 6%	39

1. Introduction

The MITRE Corporation sponsored this Major Qualifying Project (MQP). The research was completed during A- and B-terms of 2011 at the MITRE Project Center in Bedford, MA. This project consisted of designing, validating, and optimizing a program that computes the coupling between an element of a transmit phased array and an element of a spatially separate receive phased array. The program will be used as a tool in developing techniques to reduce coupling between a transmit and a receive array that reside on the same geometric plane. Included in the final report of this project is an efficiency and accuracy analysis of the program. The completed MATLAB program was handed over to engineers of The MITRE Corporation.

1.1. Basic Problem Formulation

The focus of this project is the calculation of antenna coupling, or the transfer of energy from one antenna element to another. This concept is relevant in all applications of antenna arrays, from radio broadcasting, to warship detection and tracking, to satellite communications. In general, engineers expect that each individual antenna element communicates independently, but in practice there is cross-talk, often referred to as noise, between elements. This cross-talk negatively affects the effectiveness of the array, as it can cause harmonic distortion and phase shifts in the transmitted or received signals, among others. In order to observe the effects of this cross-talk, engineers simulate the phased arrays using commercial software such as CST MICROWAVE STUDIO[®] (CST MWS), and ANSYS' HFSS[®]. An image depicting undesirable cross talk between neighboring elements is provided in Figure 1.

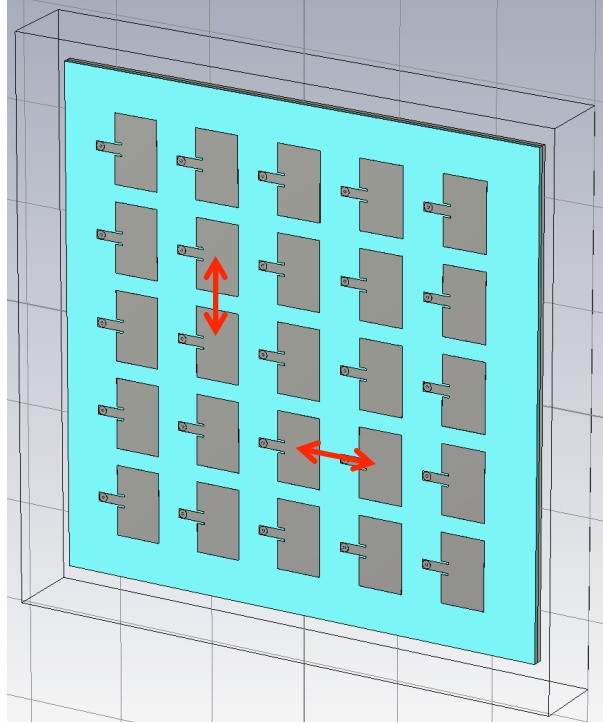


Figure 1: Undesirable Cross-Talk Between Antenna Elements

One of the most challenging aspects of simulating phased arrays for design engineers is the reliability and complexity of the commercially available software, specifically the accuracy and efficiency of different simulation software. The types of simulations performed can become very complex and different simulation software often yield dissimilar results given slight variation in mesh size, radiation boundary location, or solver type.

The particular application of phased-array coupling investigated in this project is satellite communication, particularly as it relates to the new Advanced High Data Rate (AdvHDR) system being investigated by the Navy. This SATCOM system involves a system of four phased arrays designed to communicate with satellites that would be mounted on a submarine mast. Three transmit arrays will be functioning at three separate frequencies, and the receive array will work for all three frequencies. This project focuses on the coupling from the X-band transmit array to the receive array. An image of the SATCOM system is provided in Figure 2. The arrow shows the cross-talk being investigated.

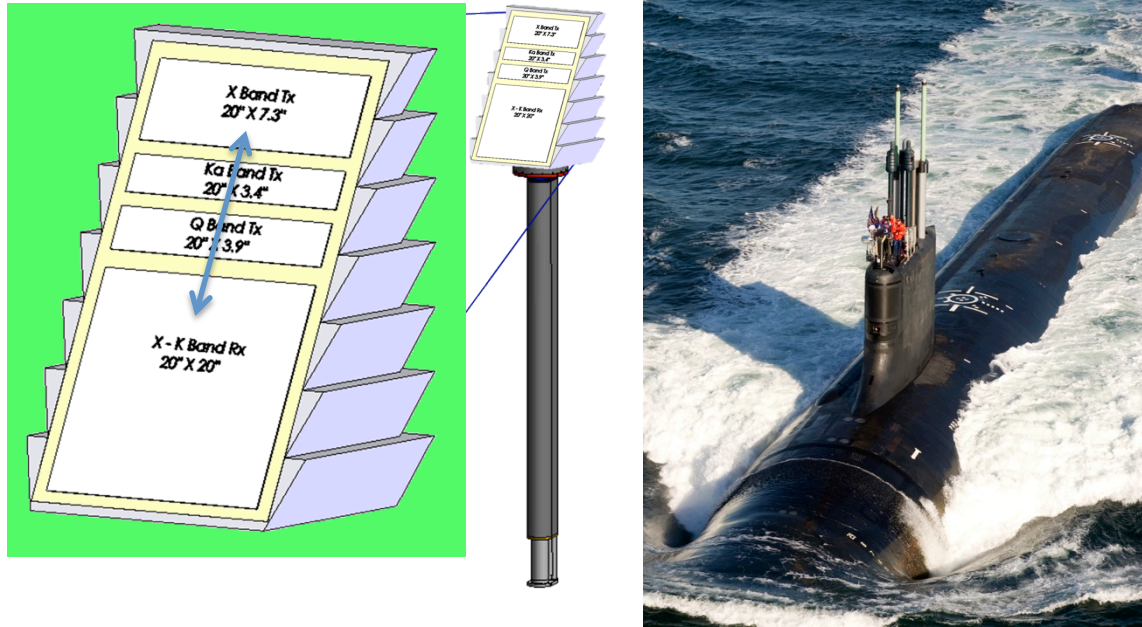


Image source¹

Figure 2: AdvHDR SATCOM System (3 Tx Arrays, 1 Rx Array) on a Submarine Mast To Be Mounted on a Submarine as Shown

1.2. Problem Statement

There is a need for a reliable, yet simple and flexible design tool that rapidly calculates the coupling between arrays in order to minimize cross-talk between elements in two different arrays. Transmit to receive isolation in a phased array antenna system is an important design consideration. Poor transmit to receive isolation causes the signal to be difficult to decipher due to interference and distortion. Therefore, many studies are being performed in order to determine how to best decrease the coupling between arrays, thereby decreasing the noise (Kolak, 2011). This effort focuses on the particular mitigation of the transmit to receive leakage/isolation as it applies to phased arrays. However the results apply to any physically separated transmit and receive antenna. In an effort to optimize system performance in satellite communication systems, the phased array systems integrator requires a method to reduce the RF coupling between the transmit and receive antenna systems. Implementation of a method to reduce coupling between arrays first requires the ability to accurately determine the coupling between the two arrays. Computing the coupling between two arrays is a challenging task, as many

¹ <http://www.navsea.navy.mil/ImageGallery/Forms/DispForm.aspx?ID=68>

commercial simulation tools have limitations with respect to low coupling values on electrically large structures. Current methods to compute coupling between arrays/elements utilize simple antenna models (dipoles, patches, slots) in place of full arrays, as these structures are easier and faster to simulate. Simulating these models provides an understanding of the coupling that can be expected and allow for the development of techniques to increase isolation (Best, 2011). This project included development of MATLAB code that calculates the coupling between two phased array elements as well as an efficiency analysis of this code.

2. Research Objectives

The purpose of this project was to provide my sponsor with a software tool capable of determining the coupling between elements of two spatially separate but coplanar phased antenna arrays. In order to prove that the program is sufficiently accurate, a comparison of the program's results with output from commercially available tools was completed. Comparable models in commercial simulation software were simulated and a comparison was made between the results of those simulations (coupling, runtime) and the output of my program. Finally, an analysis on the effectiveness and accuracy of this program was performed. This overall research objective can be divided into two smaller goals.

2.1. Goal 1: Create Program Calculator

The first goal of this project is to create a program in MATLAB that determines the coupling between the elements of a transmit and receive array. To do this, it is required to design a simple-to-use and versatile tool that quickly and efficiently calculates the transfer of energy between the phased antenna arrays. Upon completion, the program will be scrutinized such that it is possible to quantify the code's efficiency by computing resource usage and execution time.

2.2. Goal 2: Validate Program Accuracy

The second goal consists of validating the accuracy of the MATLAB code. This involves comparison of the results obtained by the MATLAB tool to the results of a full-wave numerical analysis code for two phased arrays. The full-wave simulation tool used to achieve this goal was CST MWS.

3. Approach

3.1. Theory

In order to write a program that accurately determines the coupling between elements of transmit and receive phased arrays, research of the current methods used in the field was performed. This research provided insight into the theory of the antenna-antenna coupling. The most relevant theoretical tool used in the field to this project is Friis' Transmission Equation (Kolak, 2011).

3.1.1. Friis' Transmission Equation

The software tool that was composed for this project utilizes the scalar Friis' Transmission Equation to determine the coupling between two array elements. Friis' Transmission Equation states that the ratio between the receive power and the transmit power is given by the equation:

$$\frac{P_r}{P_t} = G_t G_r \left(\frac{\lambda}{4\pi R} \right)^2 \quad (5)$$

where:

- P_r = Power on the receive array
- P_t = Power transmitted from the transmit array
- G_t = transmit gain
- G_r = receive gain
- λ = wavelength
- R = distance between the two antennas

A pictorial representation of the application of Friis' equation is shown in Figure 3.

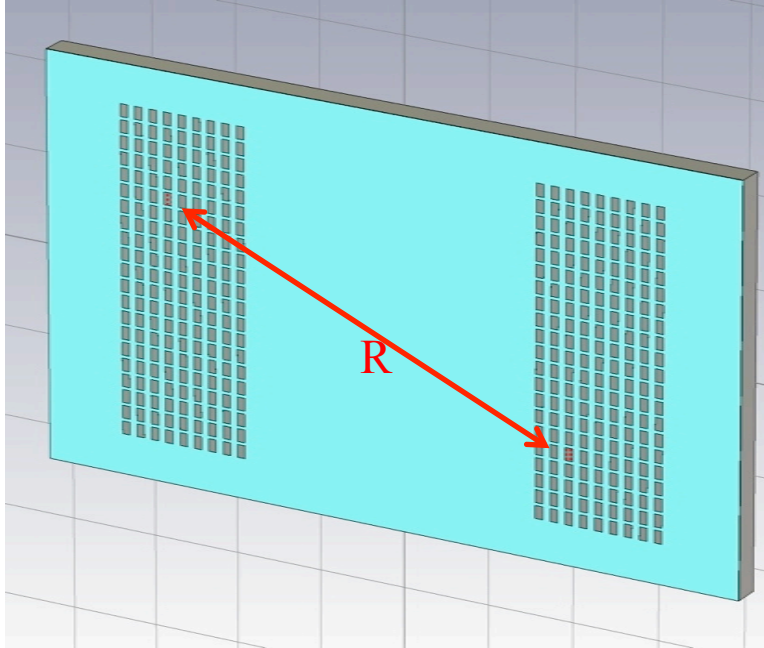


Figure 3: Scalar Friis' Transmission Equation

When the gain is provided in decibels, the equation for computing the power incident upon the receive antenna, when launched by the transmit antenna, is as follows:

$$P_r = P_t + G_t + G_r + 20 * \log_{10} \left(\frac{\lambda}{4\pi R} \right)^2 \quad [\text{dBm}] \quad (6)$$

3.2. Methodology

In order to provide my sponsor The MITRE Corporation a MATLAB program that efficiently and accurately computes the coupling between the elements of a transmit phased array and a spatially separate receive phased array located on the same plane with input parameters that include:

- Distance between arrays in inches
- Frequency of operation in gigahertz
- Element spacing in inches
- Transmit array size [# elements in x, # elements in y]
- Transmit array size [# elements in x, # elements in y]
- Transmit element position [x index, y index]
- Receive element position [x index, y index]
- Radiation pattern of transmit element (text file)
- Radiation pattern of receive element (text file)

A user interface is designed so that an engineer can input these parameters easily. An image of the interface is depicted in Figure 4.

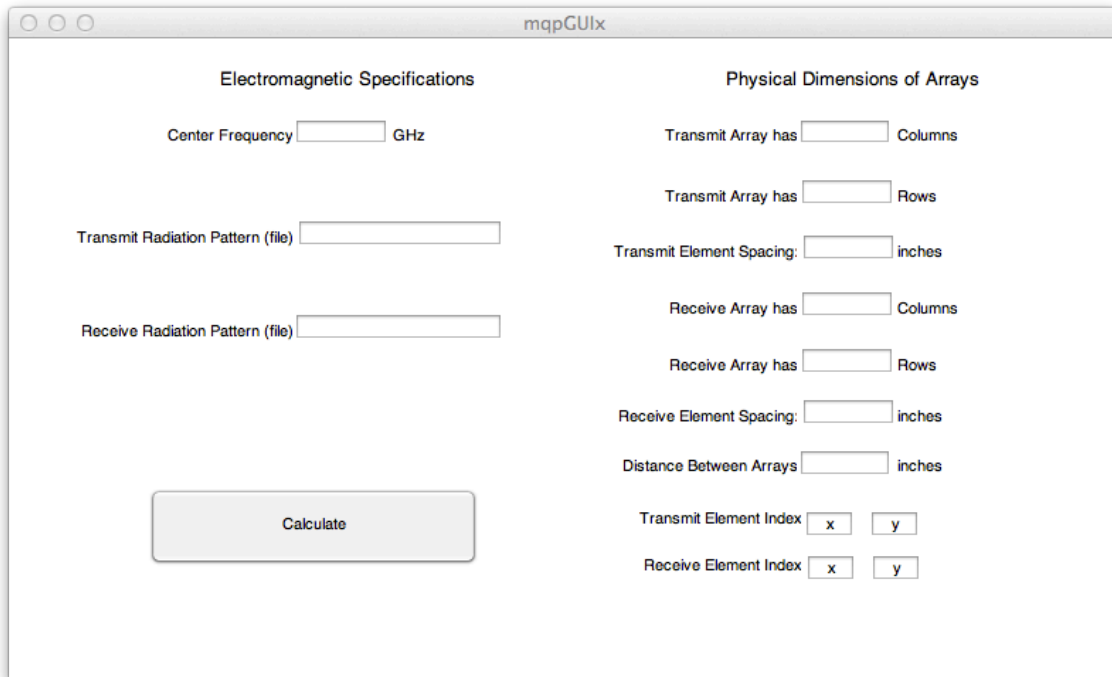


Figure 4: User Interface for Input Parameters

From these input parameters, the program calculates and outputs the coupling or energy transfer between two spatially separated elements (one transmit element and one receive element). The parameters shown in Figure 4 are explained in subsequent sections.

3.2.1. Design and Coding of Program

Creating the algorithm used in this program consisted of three main steps. The first step involves analyzing the radiation pattern (specifically, the realized gain) of a single element in an array with CST MWS and storing the results in such a way that they can be analyzed later. The second step initially involved analyzing the radiation pattern of a single radiating element embedded in a planar transmit array and quantifying the cross-talk to non-radiating elements. This step was performed on two small arrays: a 3x3 element array and a 5x5 element array. The third and final step was to analyze the radiating between full transmit- and receive arrays via Friis' Transmission Equation. The results of these simulations were used to develop the algorithm.

3.2.2. Improving Program Runtime

The program was further optimized so that the runtime was reduced. In order to do this, the MATLAB profiler was utilized, as was the “unroll” function that unrolls for-loops in order to lessen the number of time-inefficient jumps in the code. However, there is a trade-off between reducing the number of jumps and increasing the size of the code so much that there are adverse effects on the runtime. Every combination of the rolled and unrolled for-loops was run enough times to find a mean and standard deviation runtime for each piece of code. The results of this code analysis were analyzed, and the combination with the fastest runtime was kept. The machine used for testing purposes was a 2.66 GHz Intel Core i7 with 8 GB of RAM. The operating system used was Macintosh OS Lion.

4. Results and Observations

Chapter 4 of this report is dedicated to detailing the findings of this project including the full-wave simulations, the calculated predictions, and the execution time testing. In section 4.1, the results of the simulations performed in CST MWS will be described and analyzed.

4.1. Results of CST Simulations

In order to determine how the MATLAB program deals with the gain elements of Friis’ Transmission Equation, three steps were taken. First, a single, isolated patch element was simulated in CST and using the field monitors tool, the gain patterns were found. After this, the same method was utilized to observe the gain patterns on single elements in single 3x3 and 5x5 transmit array environments. The third and final step was determining the coupling in a similar manner on two full co-planar Tx and Rx arrays. The run time of all simulations were observed.

4.1.1. Single Isolated Radiating element

To accurately determine the coupling between elements of the transmit and receive phased arrays, it is necessary to determine the gain radiation patterns of an element. Therefore, CST was utilized to simulate a single isolated radiating antenna element. The element modeled is shown in Figure 5.

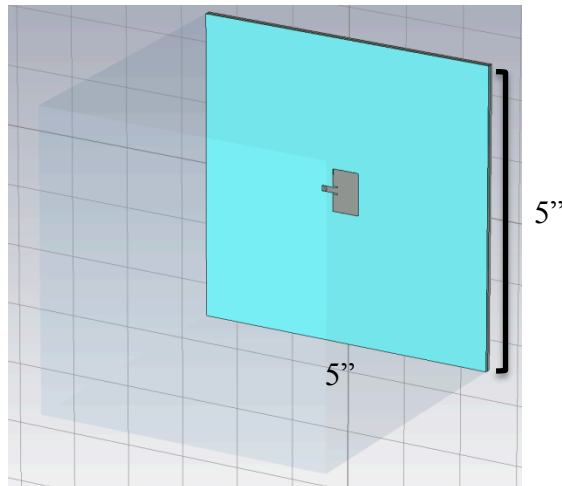


Figure 5: Single Radiating Patch Element

This particular element is fed through a coaxial line and has the dimensions specified in Figure 5. Conventionally, the elements are fed on the transmission line side with a coax cable; however long transmission lines could potentially excite external radiation and surface waves. Both of these potential effects of longer feed lines would not be representative of a typical array. Therefore, this element is fed through the bottom in a manner designed to minimize the amount of transmission line in the substrate. The element is on a 5" x 5" plate with 5" of vacuum between the patch and the radiation boundary in order to maximize the accuracy of the simulation. The substrate had a relative dielectric constant (ϵ_r) of 2.2, and the thickness of the substrate was 0.050". The Z-Smith Chart, the return-loss, and the far-field gain patterns resulting from this CST simulation were utilized in the design of the MATLAB code in order to increase accuracy.

Simulating the single isolated patch element yielded the results one would expect with a main lobe projecting almost directly away from the ground plate and a side lobe of interest at azimuth and elevation of 0° . The total simulation time for this single element was thirteen minutes and one second on a specialized machine. The resonant frequency of the element was found to be 8.14 GHz, and had a relatively broad-band frequency response (.7 GHz bandwidth at -5 dB coupling). Additionally, the resonant frequency was matched to almost exactly 50Ω . At elevation $= 0^\circ$, a cross section of the gain pattern is shown in Figure 6.

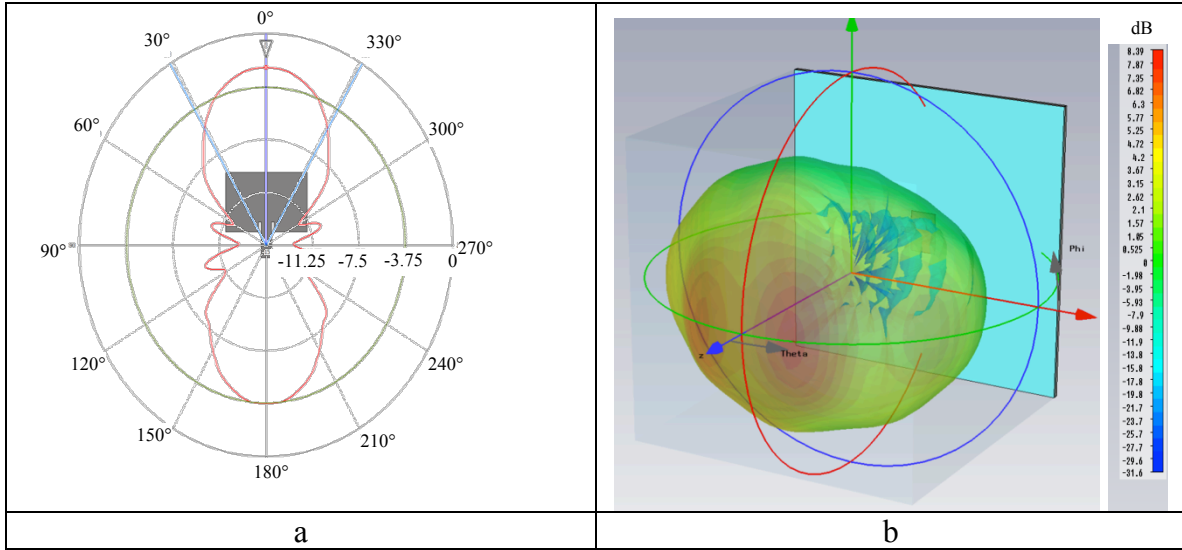


Figure 6: Single Element Gain Pattern in a) an azimuth sweep at elevation = 0° and b) at all angles

As seen in Figure 6, the gain at the 0° azimuth angle (or straight ahead) is -2.4 dB. Another interesting feature of this gain is that within the 3 dB beamwidth that spans 61.8°, the main beam can be approximated using a parabola. Specifically, it can be approximated by the parabola described in the equation:

$$G = -12 \cdot \left(\frac{\varphi}{61.8^\circ}\right)^2 - 2.4 \quad (7)$$

where φ is the azimuth angle (Markina-Khusid, 2011). A comparison of the antenna patterns is shown in Figure 7.

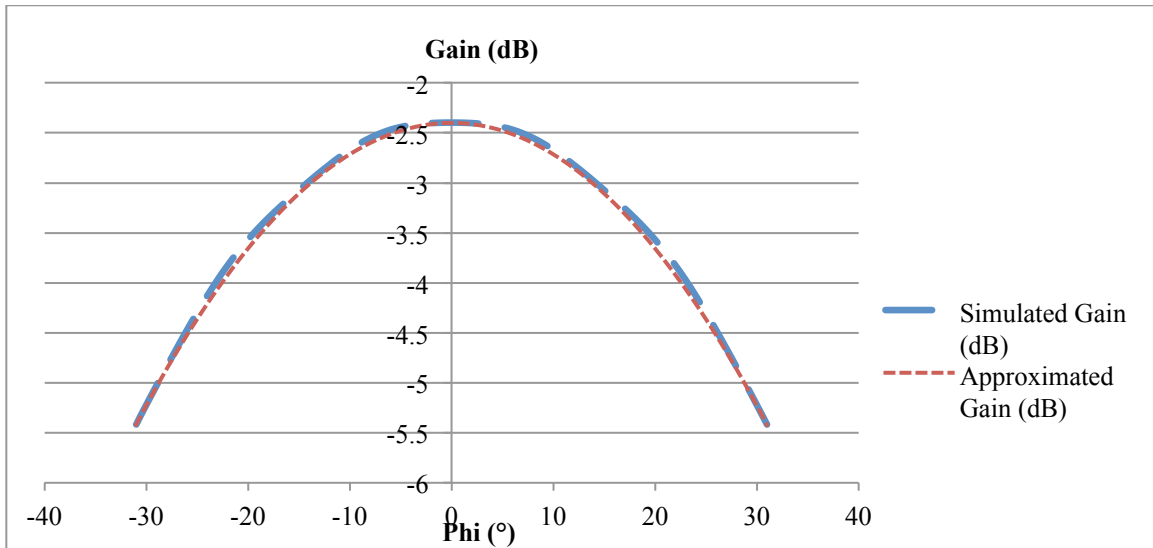


Figure 7 - Comparison of Simulated and Approximated Gains for Single Element

This is a useful approximation to utilize in the coupling calculation program, as it helps to increase accuracy of the result without any considerable effect on the runtime of the software.

4.1.2. Single Radiating Element in 3x3 Transmit Array

After the single isolated element was simulated and analyzed, the same element was simulated surrounded by other elements as shown in Figure 8. In the figure, the center element (circled in red) was excited, while the surrounding elements were not excited. The geometries in the figure was simulated twice: once with the surrounding elements open-circuited, and once with the surrounding elements terminated by 50Ω load impedances.

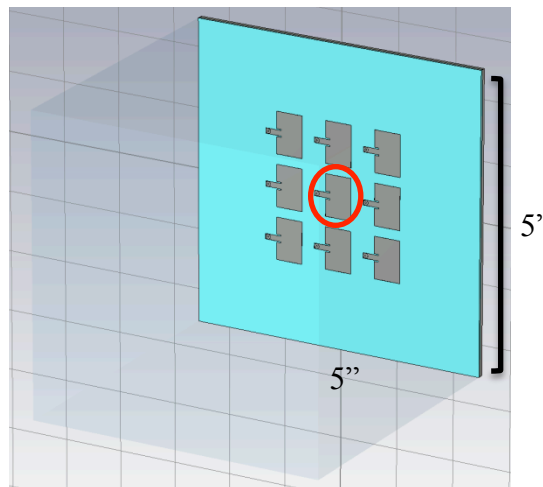


Figure 8: Single Element in a 3x3 Array

The Z-Smith Chart (provided in Appendix III), return-loss, insertion-loss, and the far-field gain patterns resultant of these two CST simulations were utilized in the design of the MATLAB code for optimization of accuracy.

Simulating this 3x3 array of patch elements, simulating only the center patch and leaving the remaining eight open-circuited had a total runtime of twenty-seven minutes and forty-four seconds. The resonant frequency shifted a negligible 20 MHz up to 8.16 GHz. The impedance also remained relatively unchanged. This was due to the minimal effect of the surrounding elements. However, the gain pattern at 8.15 GHz looked noticeably different, as shown in Figure 9.

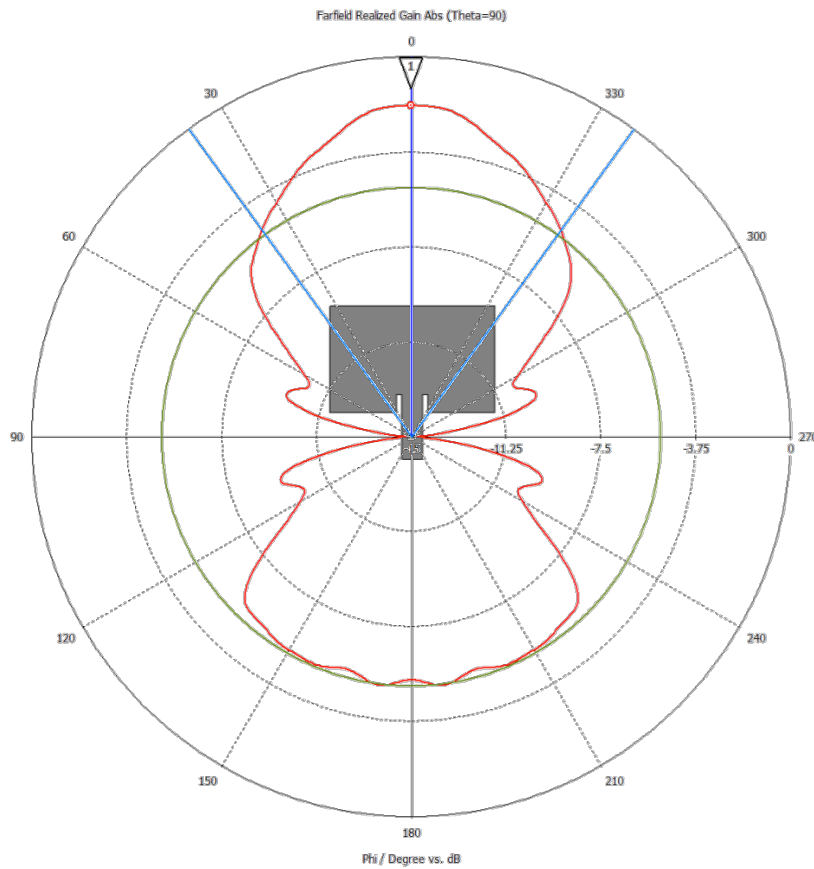


Figure 9: Gain Pattern of Element in 3x3 Array, Remaining Elements Open Circuited

The gain at 0° azimuth is -1.9 dB; this is .5 dB less than the isolated patch element. Also, the 3 dB beamwidth is wider than the single element: 71.7°. Within the 3 dB beamwidth, the gain can still be approximated by a parabola:

$$G = -12 \cdot \left(\frac{\varphi}{71.7^\circ}\right)^2 - 1.9 \quad (8)$$

An additional simulation of the 3x3 array was run, but this time all eight surrounding elements were terminated to 50Ω. The results of this simulation indicate that utilizing a parabola to approximate the gain for the software would be inaccurate (shown in Figure 10).

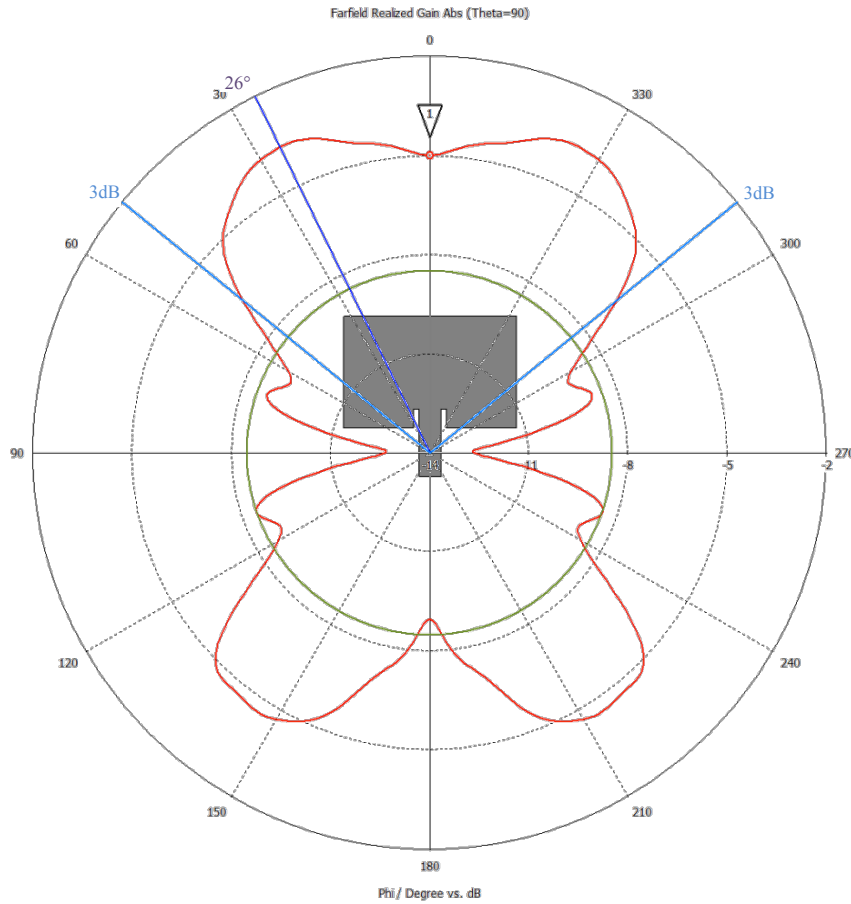


Figure 10: Gain of Element in 3x3 Array, Remaining Elements Terminated to 50Ω

The main lobe of the gain pattern is at 26°, with a maximum of -3.7 dB. At 0° azimuth, the gain is -5.0 dB. However, the range of angles within 3dB beam increased to 101.5°. Despite the changes in gain pattern, the resonant frequency and matched impedance remain at 8.15 GHz and 50Ω due to the additional elements. This simulation took twenty minutes and fifty-three seconds.

4.1.3. Single Radiating Element in a 5x5 Tx Array

Similarly to the 3x3 array, a 5x5 transmit array was simulated in the same fashion. The geometry for the 5x5 array is provided in Figure 11. Once again the Z-Smith Chart, return-loss, and far-field gain patterns were utilized in the design of the MATLAB code.

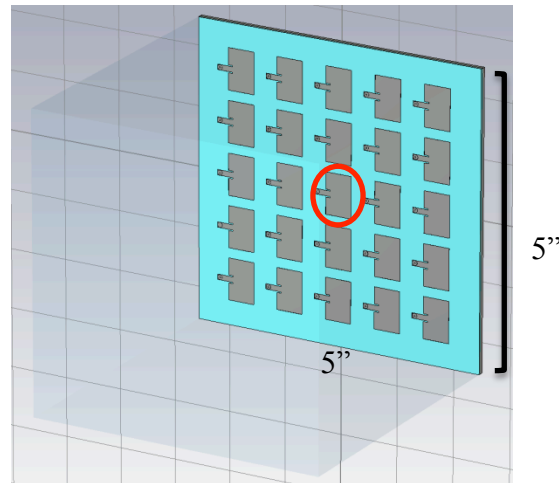


Figure 11: Single Element in a 5x5 Array

The simulations of the 5x5 arrays were performed in the same way as the 3x3 array simulations. The first simulation was of the single, center element excited and the remaining twenty-four open circuited. This took thirteen minutes and twenty-two seconds. The center frequency remained at 8.16 GHz, and the impedance was matched to 51Ω . The gain pattern at 8.15 GHz and elevation = 0° is shown in Figure 12.

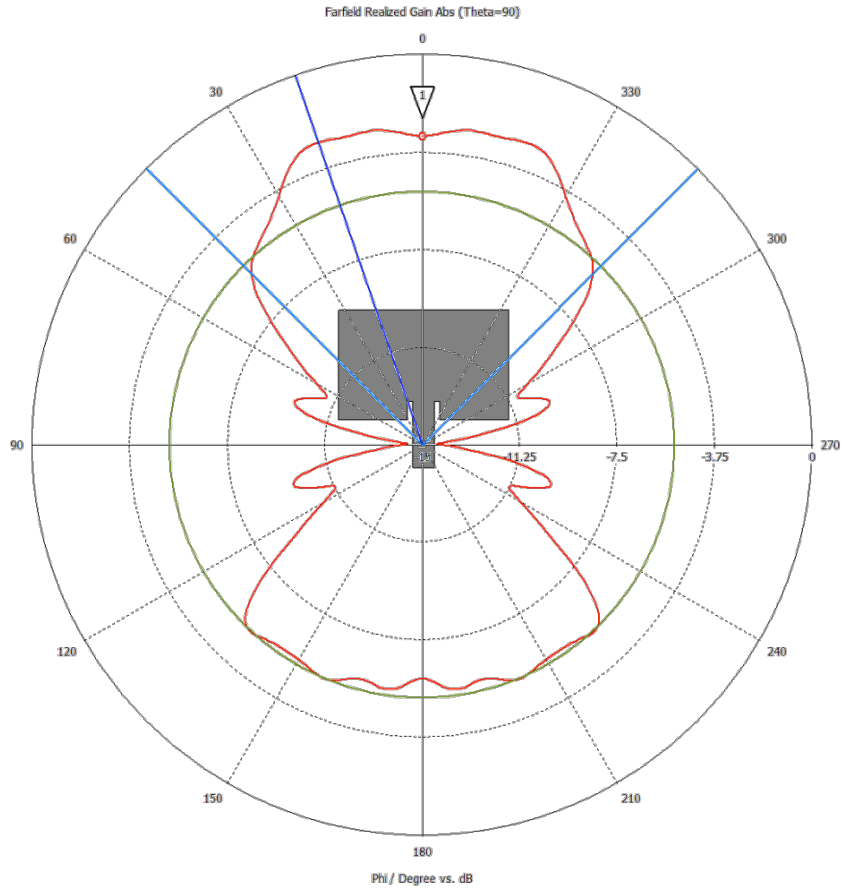


Figure 12: Gain Pattern of Element in 5x5 Array, Remaining Elements Open Circuited

Once again, a parabolic approximation would be inappropriate for this gain pattern. The peak gain is at $\varphi = 19.0^\circ$ at -2.7 dB. At $\varphi = 0^\circ$, the gain is -3.1 dB.

As with the 3x3 transmit array, the 5x5 array was also simulated with the center patch excited and the surrounding elements terminated by 50Ω . This simulation would provide gain pattern results that are the most similar to the actual gain patterns of elements in the full arrays.

As with the other simulations, the resonant frequency remained at 8.16 GHz with a impedance matched to about 50Ω . The resultant gain pattern from the simulation is shown in Figure 13.

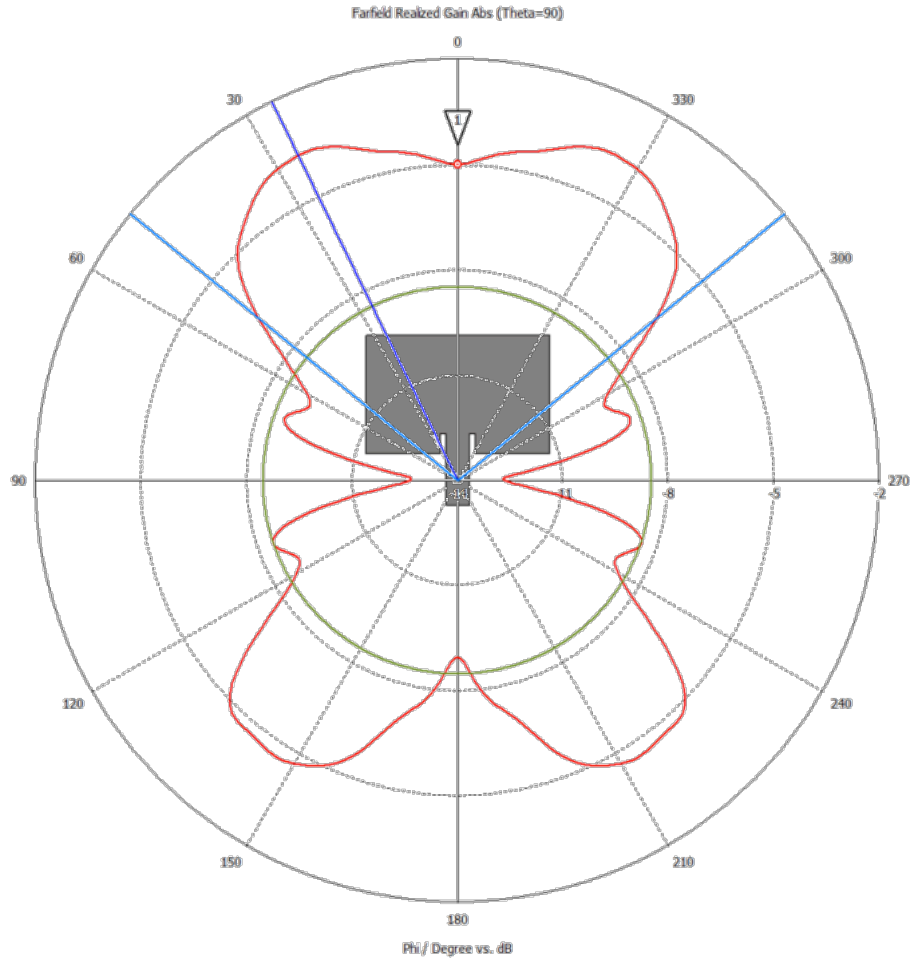


Figure 13: Gain of Element in 5x5 Array, Remaining Elements Terminated to 50Ω

As can be observed in Figure 13, this gain pattern would clearly not be accurately approximated by a parabola. The peak gain is -3.7 dB at 26° azimuth, and directly ahead there is a gain of -5.0 dB. This simulation took twelve minutes and twenty-nine seconds.

4.1.4. Results of Transmit and Receive Element Simulation

Due to the limited computing resources, the first simulation ran of the coupling between a transmit and receive element was just two elements in an environment of the same dimensions as the full arrays as shown in Figure 14. The transmit element with a center frequency of 8.15 GHz matched to 50Ω was excited, and the receive element was terminated at 50Ω.

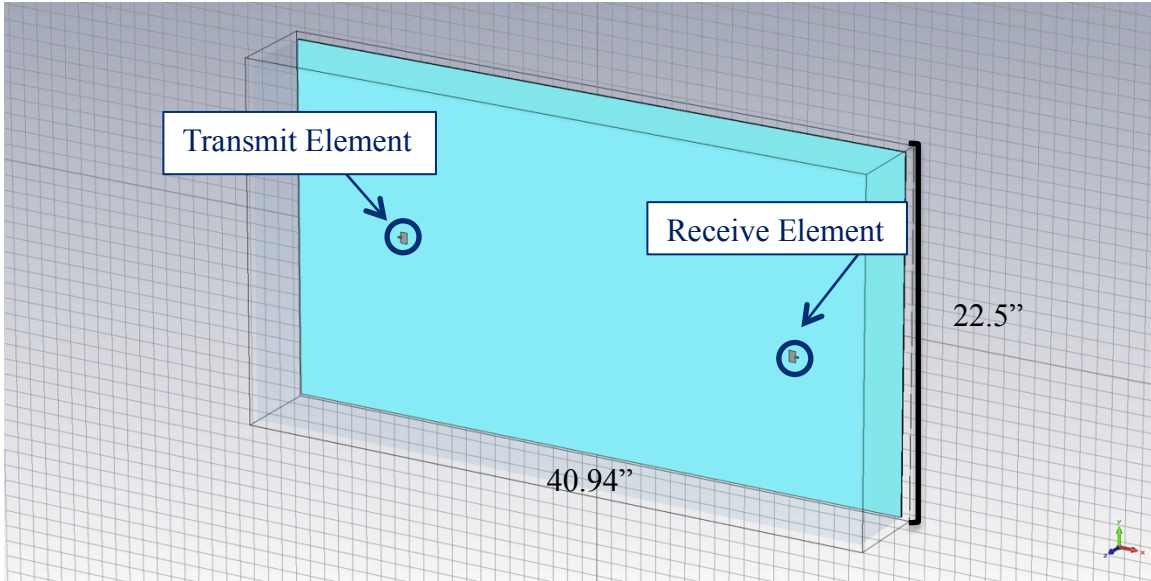


Figure 14: Transmit and Receive Element Simulated

Using simple mathematics, the free space path loss was calculated to be 41.5 dB. The full-wave simulator (CST) showed that there was -40 dB coupling at the 8.15 GHz, the frequency of interest. A graph created by CST of the losses is provided in Figure 15.

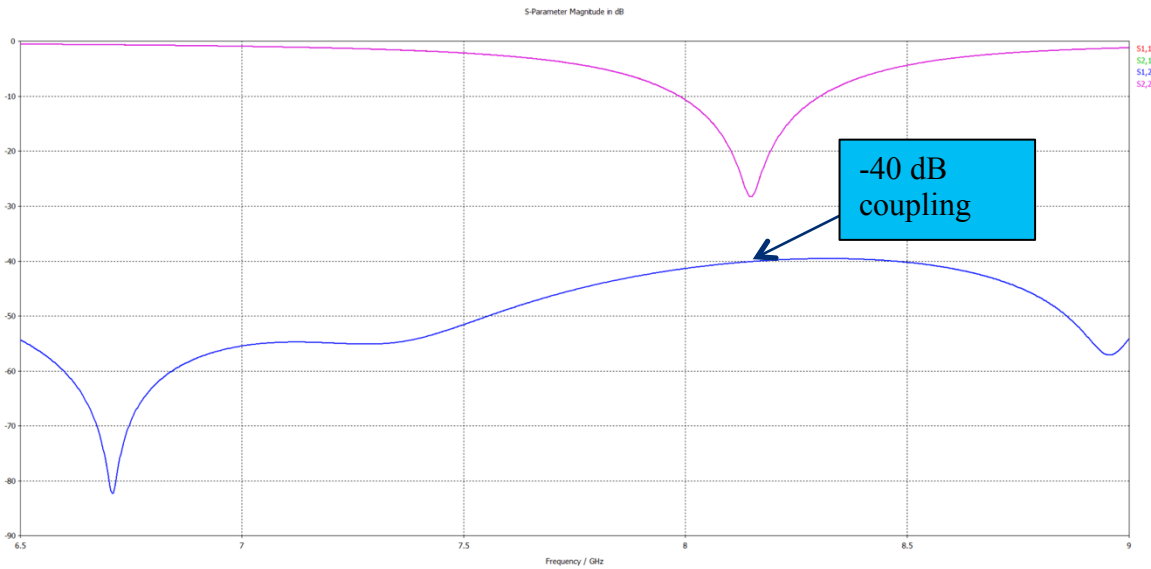


Figure 15: Return and Insertion Loss of Single Tx and Rx Element Simulation

In total, this simulation took nine hours, fifty-seven minutes, and forty-eight seconds.

4.1.5. Results of Transmit and Receive Array Simulations

After the single element-element simulation was completed, full array simulations were performed. The geometry of these simulations is provided in Figure 16.

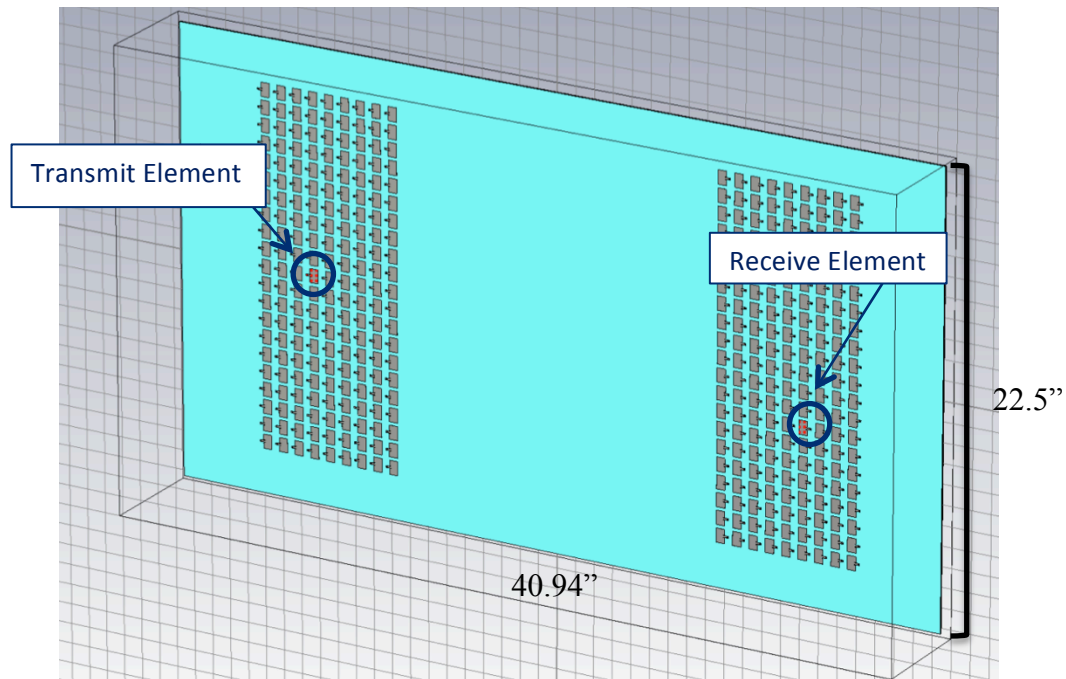


Figure 16: Full Array Geometry

As shown in Figure 16, the transmit element and receive element of interest remain in the same locations as the elements; they are simulated as described in section 4.1.4 of this report. Therefore, the free space path loss remains unchanged at 41.5 dB.

When the arrays were simulated with the transmit element excited, receive element terminated to 50Ω and the remaining elements left as an open circuit, the simulation took forty-one hours, thirteen minutes, and twenty-eight seconds. The results of this simulation are shown in Figure 17.

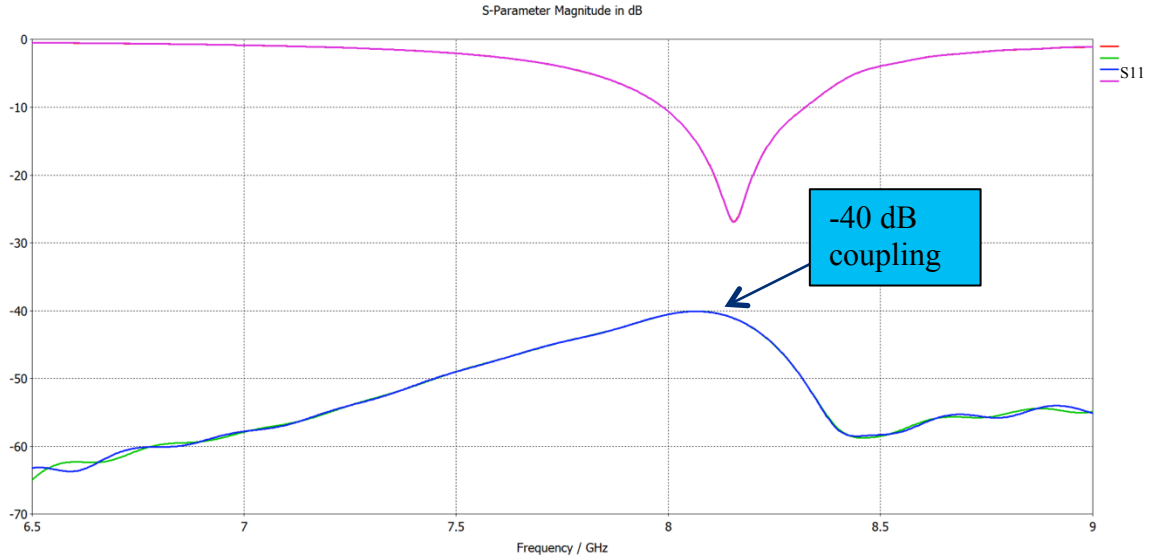


Figure 17: S-Parameters of Tx and Rx Elements, Other Elements Open Circuited

The return loss of the single element and this open circuited full array appear quite similar. However, the return loss has a largely dissimilar shape. Despite the difference in return loss overall, the element-element coupling at the frequency of interest (8.15 GHz) is unchanged at -40 dB.

After performing the above simulation, a simulation of the two full arrays with all elements terminated to 50Ω was performed, with the results as shown in Figure 18 below.

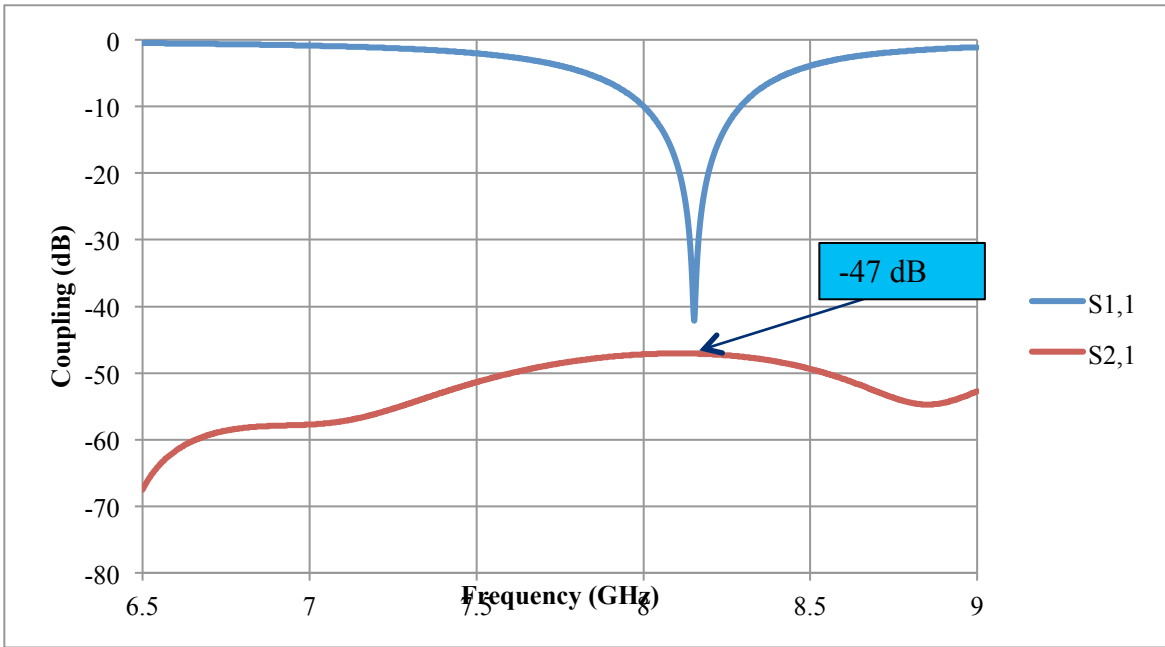


Figure 18: S-Parameters of Tx and Rx Elements, Other Elements Terminated to 50Ω

Once again, the return loss remains largely unchanged, but the insertion loss is notably different. Instead of -40 dB that the previous two simulations yielded, this simulation indicates that the coupling at 8.15 GHz for the two arrays is -47 dB, a 7 dB difference.

4.1.6. Summary of CST Results

After running the CST simulations, the results were compiled. The major results of the Single Element Radiation Pattern Simulations are provided in Table 1.

Table 1: Single Element Radiation Pattern Observations

	Peak Gain (dB)	Gain at Azimuth, Elevation=0° (dB)
Single Isolated Element	8	-2.3
8 Surrounding Elements (not terminated)	8.4	-1.9
8 Surrounding Elements (terminated)	7.1	-5.0
24 Surrounding Elements (not terminated)	9.1	-3.1
24 Surrounding Elements (terminated)	7.2	-4.6

As indicated in Table 1, the gain values at bore sight and elevation = 90° converge sufficiently between the 3x3 and 5x5 arrays. Additionally, the input impedance and resonant frequencies are consistent, showing independence from the surrounding elements.

4.2. Results of Code Validation

Two generic arrays were modeled on a single plate and simulated with commercial simulation software, specifically CST. In order to provide results to verify the accuracy of the MATLAB coupling tool developed for this project, the results of the full-wave simulation software was compared with the MATLAB code predictions.

To generate the element patterns that would be utilized in the calculation program, a single patch antenna was simulated in CST. The results were compared to the element pattern used in the MATLAB code. Additionally, two separate phased arrays of patch antennas on a single plate were modeled in CST. Using Microwave studio, individual elements of each array were electrically stimulated, and the results of these simulations were used as a comparison to verify the accuracy of the MATLAB code.

As shown in Figure 18, the CST simulation yielded a value for the return loss between two elements of -47 dB. To validate the program's accuracy, it was run for the same two elements with all the same specifications, and it provided a result of -47.7071 dB. Consequentially, my sponsor and I decided that these results were sufficiently similar to validate the code to be accurate enough for use in the field.

4.3. Results of Time Experiments

For reference, a code snippet of the important nested for-loops of the MATLAB function written is included in Figure 19.

```

for i=1:numReceiveColumns
    %x position of receive element
    xReceive=xElementSpacing*(i-1);
    for j=1:numReceiveRows
        %y position of receive element
        yReceive=yElementSpacing*(j-1);
        for k=1:numTransmitColumns
            %x position of transmit element
            xTransmit=-(arraySpacing+xElementSpacing*(k-1));
            for l=1:numTransmitRows
                %y position of transmit element
                yTransmit=yElementSpacing*(l-1);
            end
        end
    end
end

```

Figure 19: Snippet of MatLab Program's Nested FOR-Loops

In order to decrease the execution time taken for the program to run, the four for-loops of the program were unrolled. However, unrolling the for-loops sometimes causes the MATLAB function to take longer to execute. Results of the time experiments, as provided by the MATLAB profiler, are included in Table 2.

Table 2: Time Analysis Experiment Results

loops unrolled	number of calls	total time taken (s)	mean time taken (s)
i	100	15.691	0.15691
i, j	100	31.359	0.31359
i, j, k	100	17.022	0.17022
i, j, k, l	100	33.784	0.33784
i, j, l	100	33.867	0.33867
i, k	100	16.907	0.16907
i, l	100	33.871	0.33871
j	100	15.721	0.15721
j, k	100	17.086	0.17086
j, k, l	100	33.918	0.33918
j, l	100	33.978	0.33978
k	100	16.678	0.16678
k, l	100	33.943	0.33943
l	100	33.712	0.33712
none	100	15.712	0.15712

From these results, it appears that when the inner-most loop (loop l) is unrolled, the function call takes longer, and when the other loops are unrolled, there is little change from the original time taken. The only loop that caused a decrease in runtime when unrolled was the outer-most loop (loop i). However, these savings in time were minimal.

Furthermore, these runtimes were obtained on a computer with much less computing resources than what is typically used for the CST simulations. It was run on a 2.66 GHz Intel Core i7 Processor with 8GB of RAM.

5. Conclusions

In its current state, this MATLAB tool will serve as an asset for my sponsor despite the tradeoffs made during its creation. The most notable advantage this program provides is that it saves the engineers a considerable amount of simulation time. Instead of waiting for a full-wave simulation to complete over multiple days, a MITRE engineer can use this tool and have a result almost instantaneously. Additionally, full-wave simulations take time to model prior to simulation, whereas this tool has an easy to use graphical user interface as shown in Figure 4.

Admittedly, the MATLAB tool only provides an approximation of the coupling between the elements as opposed to the full-wave solution. Nonetheless, the program validation indicated that the approximation is sufficiently accurate. Additionally, this program does not provide field pattern interpolation or prediction functionalities as does a full-wave simulation software. However, in many applications, radiation patterns are largely irrelevant. Furthermore, in applications where field patterns are required, this tool provides the engineer with the element-to-element coupling that can be used as a starting point while concurrently the full-wave simulations are completing.

6. Future Work

My program is designed to be agile such that it can be adapted easily to fit different applications. For example, an easy expansion of this tool would be to incorporate the ability to perform frequency sweeps. Another modification that would be simple to implement might include the ability to change the azimuth and elevation angles of either array. This future work would be able to provide the sponsor with a more powerful tool with minimal work.

Bibliography

Ulaby, F. T., Michielssen, E., & Ravaioli, U. (2010). *Fundamentals of Applied Electromagnetics* (6th Edition ed.). Boston, Massachusetts: Prentice Hall.

Visser, H. J. (2005). *Array and Phased Array Antenna Basics*. Chichester, West Sussex, England: John Wiley & Sons.

Balanis, C. A. (1982). *Antenna Theory: Analysis and Design*. New York: Harper & Row.

Best, S. (2011, August 26). (D. Gorman, Interviewer)

CST Microwave Studio (R). (2011). Computer Simulation Technology.

Eswarappa, C., Anderson, R., & Kolak, F. Periodic Filters for Performance Enhancement of Millimeter-Wave Microstrip Antenna Arrays. *Microwave Symposium Digest, 2004. 1*, pp. 353-356. IEEE MTT-S International.

Kolak, F. (2011, June 20). (D. Gorman, Interviewer)

Markina-Khusid, A. (2011, September 1). (D. Gorman, Interviewer)

Stimson, G. W. (1998). *Introduction to Airborne Radar* (2nd Edition ed.). Mendham, New Jersey: SciTech Pub.

Radisic, V., Qian, Y., Coccioli, R., & Itoh, T. (1998). Novel 2-D Photonic Bandgap Structure for Microstrip Lines. *Microwave and Guided Wave letters* , 8, 69-71.

Rizzi, P. A. (1988). *Microwave Engineering: Passive Circuits*. Englewood Cliffs, New Jersey: Prentice Hall.

Toomay, J., & Hannen, P. j. (2004). *Radar Principles for the Non-Specialist* (3rd Edition ed.). Raleigh, North Carolina: SciTech Pub.

Appendix I: Complete MATLAB Function

```
function [ elementElementCoupling, elementCoupling, totalCoupling ] = couple(
frequency, transmitGaindB, transmitBeamwidthDegrees, receiveGaindB,
receiveBeamwidthDegrees, numTransmitColumns, numTransmitRows,
numReceiveColumns, numReceiveRows, arraySpacingInches, receiveXIndex,
receiveYIndex, transmitXIndex, transmitYIndex )

lambda = 3e8/(frequency * 1e9); %calculation and storage of wavelength
xElementSpacing = lambda * .6; %Spacing between elements in the X direction
yElementSpacing = lambda * .6; %Spacing between elements in the Y direction

transmitGain = 10^(transmitGaindB/10); %calculation of transmit gain
transmitBeamwidthRadians = transmitBeamwidthDegrees* pi/180; %calculation of
transmit beamwidth in radians
receiveGain = 10^(receiveGaindB/10); %calculation of receive gain
receiveBeamwidthRadians = receiveBeamwidthDegrees* pi/180; %calculation of
receive beamwidth in radians

arraySpacing = arraySpacingInches * 0.0254; %Space between arrays in
meters

power = 200; %arbitrary power value
radiatedPower = power * numTransmitColumns * numTransmitRows * transmitGain *
receiveGain; %radiated power calculation

%Variable Initialization
totalVoltage = complex(0,0); %total voltage (phasor) per receive
element
totalPower = 0;
maxElementPower = 0; %The highest power coupling to a single
receive element
fullArrayPower = 0; %total coupling between the two arrays
powerOnElements=zeros(numReceiveColumns, numReceiveRows);

couplingBetweenElements=zeros(numReceiveColumns, numReceiveRows, numTransmitCol
umns, numTransmitRows);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% Calculation of Power transferred to %%%
%%% Element of the Receive Array %%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=1:numReceiveColumns
xReceive=xElementSpacing*(i-1); %x position of receive element
for j=1:numReceiveRows
yReceive=yElementSpacing*(j-1); %y position of receive element
for k=1:numTransmitColumns
xTransmit=-(arraySpacing+xElementSpacing*(k-1)); %x position of
transmit element
for l=1:numTransmitRows
yTransmit=yElementSpacing*(l-1); %y position of transmit
element

distance=sqrt((xReceive-xTransmit)*(xReceive-
xTransmit)+(yReceive-yTransmit)*(yReceive-yTransmit)); %distance between
elements

angle = atan((yTransmit-yReceive)/(xTransmit-xReceive));
```



```

%angle between transmit and receive elements

        gainTransmit = transmitGain;
        gainReceive = receiveGain;

        elementPower=power * gainTransmit * gainReceive * lambda *
lambda / (4 * pi * distance) / (4 * pi * distance); %calculation of power
between these two elements (use of Friis' Transmission Equation)
        couplingBetweenElements(i,j,k,l) = elementPower;

        voltageMagnitude=sqrt(2*elementPower);           %voltage
magnitude
        phase=2 * pi * distance / lambda;               %voltage phase
        VR = voltageMagnitude * cos(phase);             %real part of
voltage phasor
        VI = voltageMagnitude * sin(phase);             %imaginary part
of voltage phasor
        V = complex(VR, VI);                            %voltage phasor
        totalVoltage = totalVoltage + V;               %add total
voltage transferred to receive element (i,j) from transmit array
        totalPower = totalPower + elementPower;
    end

        powerOnElements(i,j) = totalPower; %calculate and store the total
power transferred from the transmit array for receive element (i,j)
        totalPower = 0; %reinitialize vTotal variable for use in next
iteration

        if(powerOnElements(i,j) > maxElementPower)
            maxElementPower = powerOnElements(i,j); %if
this is the element recieving the most power, set MAX as this value
        end
        fullArrayPower = fullArrayPower + powerOnElements(i,j);
%Add this received power to the total coupling value
    end
end

elementElementCoupling = 10 *
log10(couplingBetweenElements(receiveXIndex, receiveYIndex, transmitXIndex, tran
smitYIndex)/power);
elementCoupling = 10 * log10(maxElementPower/radiatedPower);
totalCoupling = 10 * log10(fullArrayPower/radiatedPower);
%calculate the total coupling between the full arrays

end

```

Appendix II: Functional GUI Code

```
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
    frequency = str2double(get(handles.freq, 'String'));
    transmitGaindB = str2double(get(handles.transmitGainMag, 'String'));
    transmitBeamwidthDegrees = str2double(get(handles.transmitBeamwidth,
'String'));
    receiveGaindB = str2double(get(handles.receiveGainMag, 'String'));
    receiveBeamwidthDegrees = str2double(get(handles.receiveBeamwidth,
'String'));
    numTransmitColumns = str2double(get(handles.transmitColumns, 'String'));
    numTransmitRows = str2double(get(handles.transmitRows, 'String'));
    numReceiveColumns = str2double(get(handles.receiveColumns, 'String'));
    numReceiveRows = str2double(get(handles.receiveRows, 'String'));
    arraySpacingInches = str2double(get(handles.arrayDistance, 'String'));
    receiveXIndex = str2double(get(handles.receiveXIndex, 'String'));
    receiveYIndex = str2double(get(handles.receiveYIndex, 'String'));
    transmitXIndex = str2double(get(handles.transmitXIndex, 'String'));
    transmitYIndex = str2double(get(handles.transmitYIndex, 'String'));

    [elementElementCoupling, elementCoupling, totalCoupling] = couple(
frequency, transmitGaindB, transmitBeamwidthDegrees, receiveGaindB,
receiveBeamwidthDegrees, numTransmitColumns, numTransmitRows,
numReceiveColumns, numReceiveRows, arraySpacingInches, receiveXIndex,
receiveYIndex, transmitXIndex, transmitYIndex );

    set(handles.elementElementResult, 'String', strcat('element-element
coupling: ', num2str(elementElementCoupling), ' dB'));
    set(handles.elementResult, 'String', strcat('array-element coupling: ',
num2str(elementCoupling),
' dB'));
    set(handles.arrayResult, 'String', strcat('array-array coupling: ',
num2str(totalCoupling), ' dB'));
end
```

Appendix III: Further Results from CST Simulations

Single Isolated Radiating Element

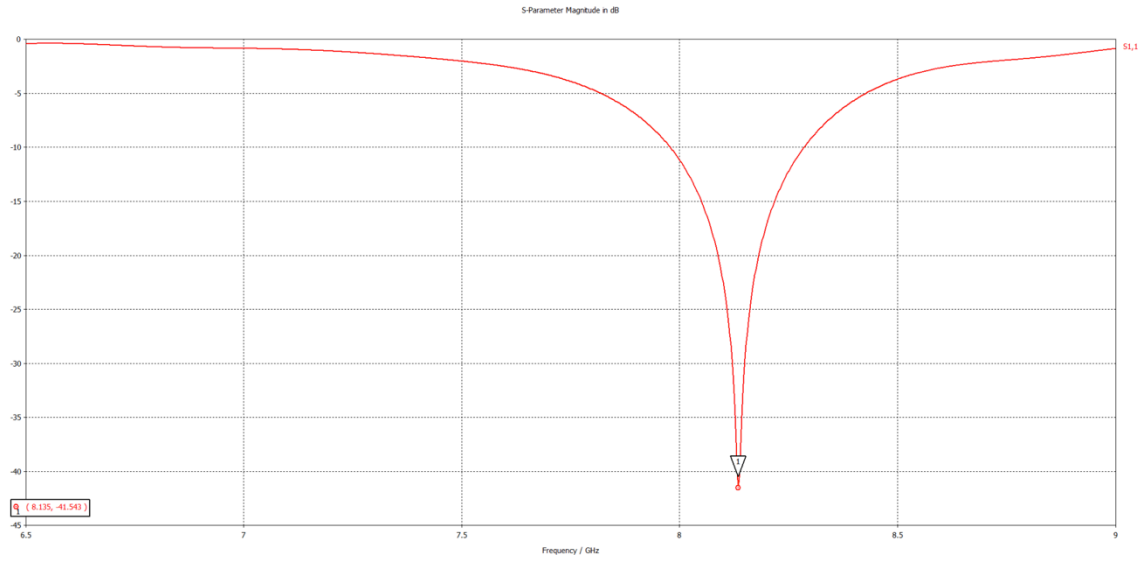


Figure 20: Return Loss, Center Frequency 8.14 GHz

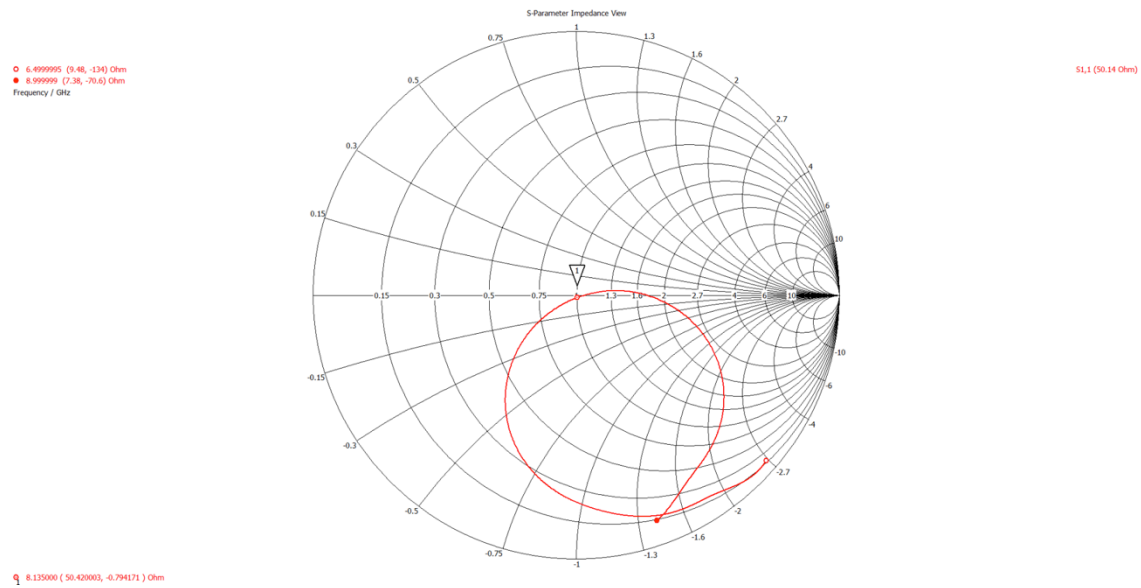


Figure 21: Z-Smith Chart, $Z_1=(50.42-0.08j)\Omega$

3x3 Array: Center Element Excited, Remaining Elements Are Open Circuited

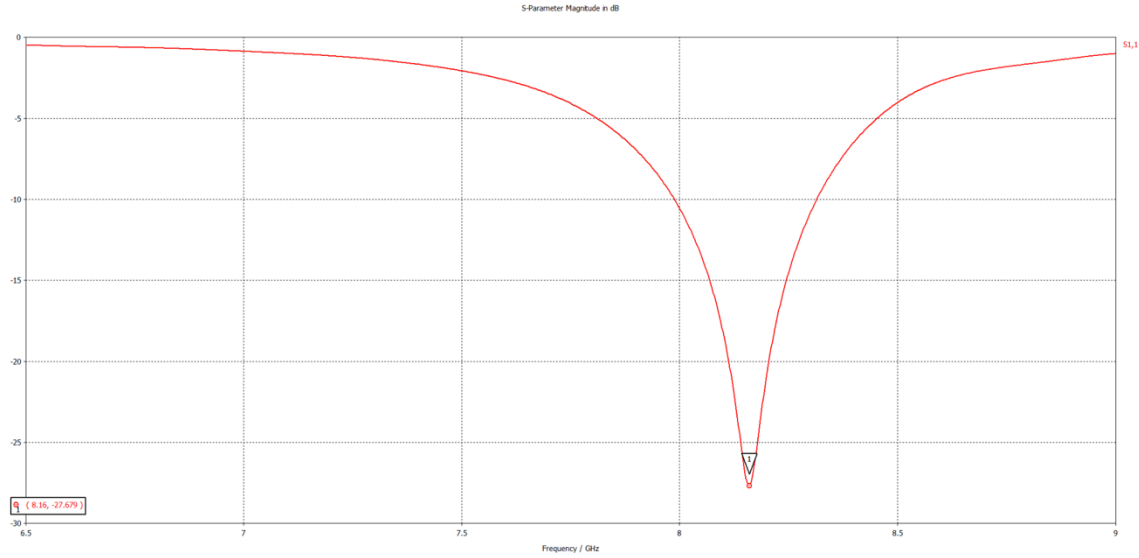


Figure 22: Return Loss, Center Frequency 8.16 GHz

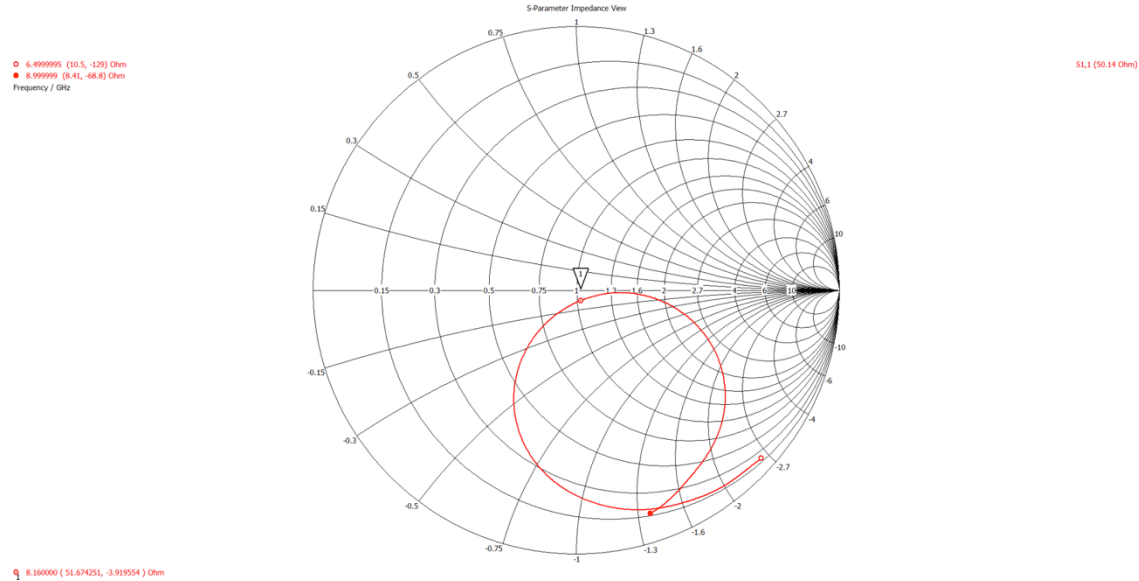


Figure 23: Z-Smith Chart, $Z_1=(51.67-3.92j)\Omega$

3x3 Array: Center Element Excited, Remaining Elements Are Terminated to 50Ω

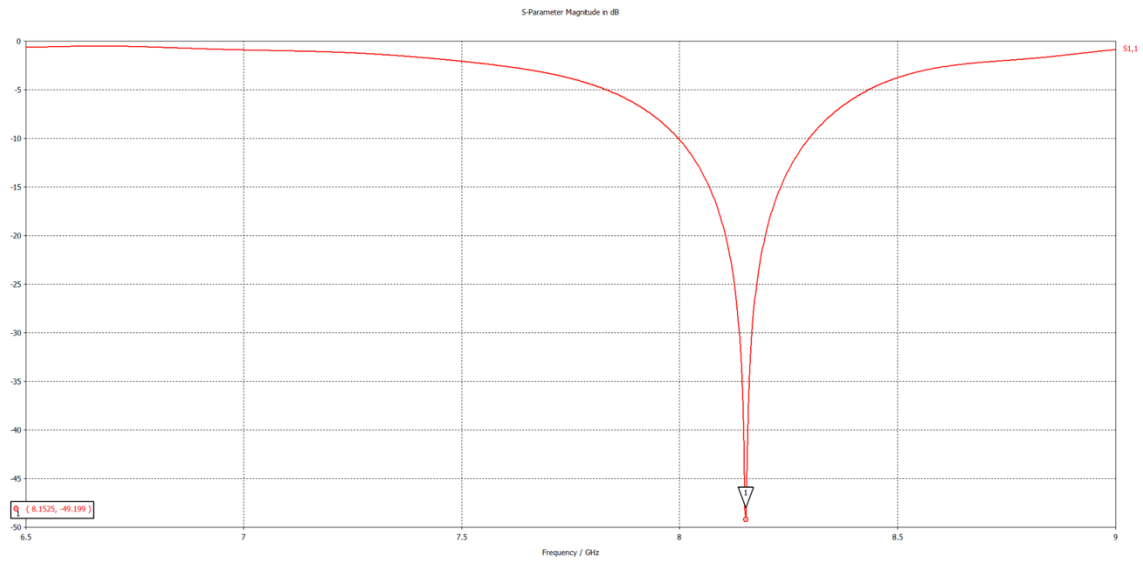


Figure 24: Return Loss, Center Frequency 8.15 GHz

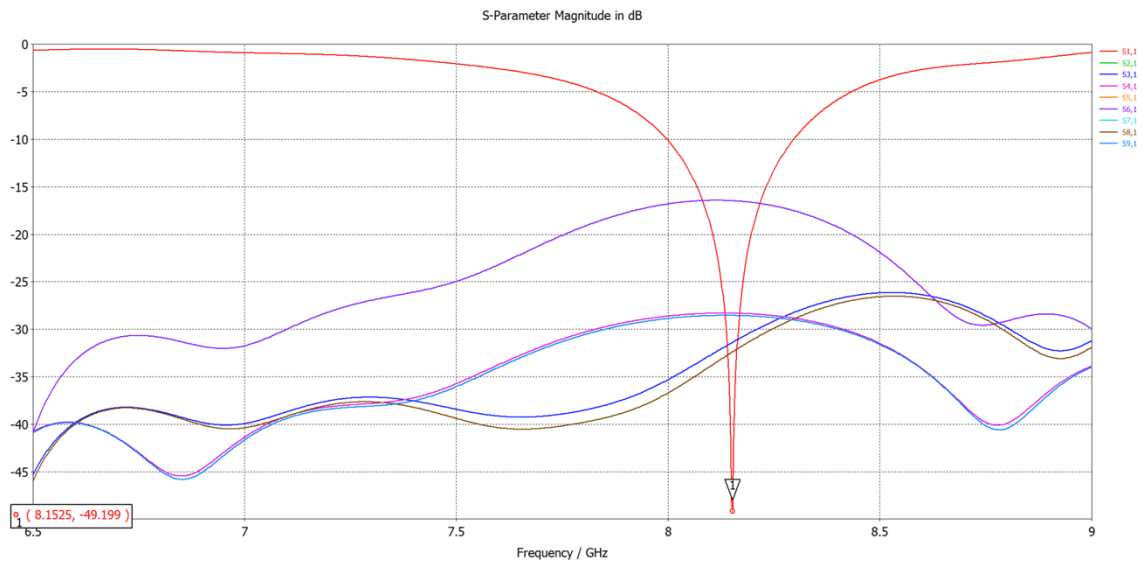


Figure 25: All S-Parameters

Table 3: Summary of Coupling, Power Coupled to Adjacent Elements is 5%

F=8.15 GHz	Coupling (dB)	Coupled Power (%)
S1,1	-48.19	n/a
S2,1	-28.32	0.15%
S3,1	-31.43	0.07%
S4,1	-28.32	0.15%
S5,1	-16.47	2.25%
S6,1	-16.47	2.25%
S7,1	-28.56	0.14%
S8,1	-32.44	0.06%
S9,1	-28.56	0.14%
		$\Sigma=5.21\%$

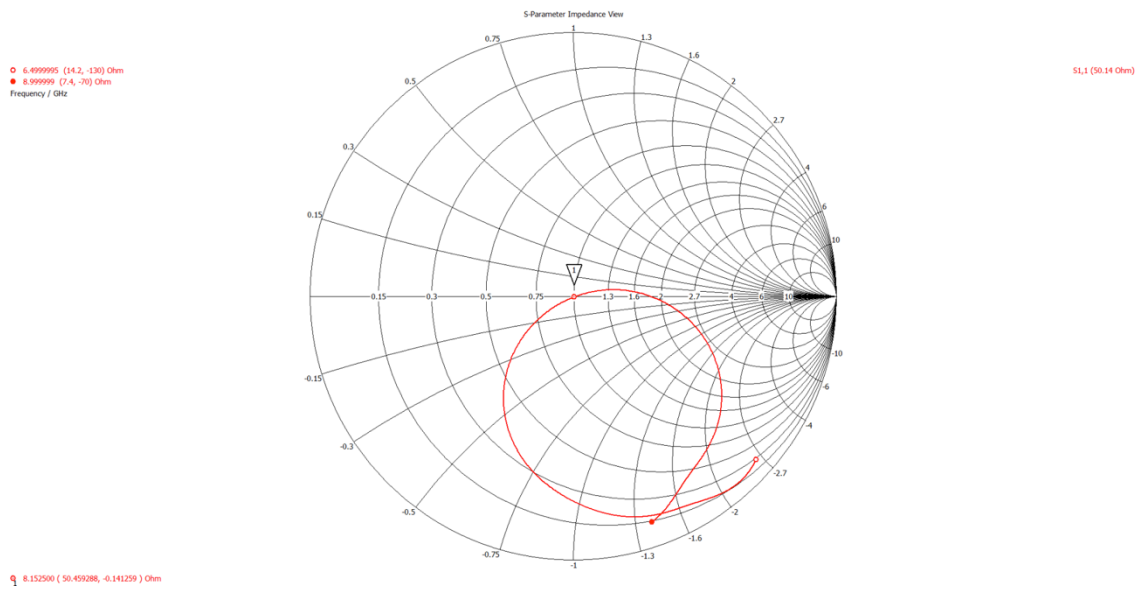


Figure 26: Z-Smith Chart, $Z_1=(50.46-0.14j)\Omega$

5x5 Array: Center Element Excited, Remaining Elements Are Open Circuited

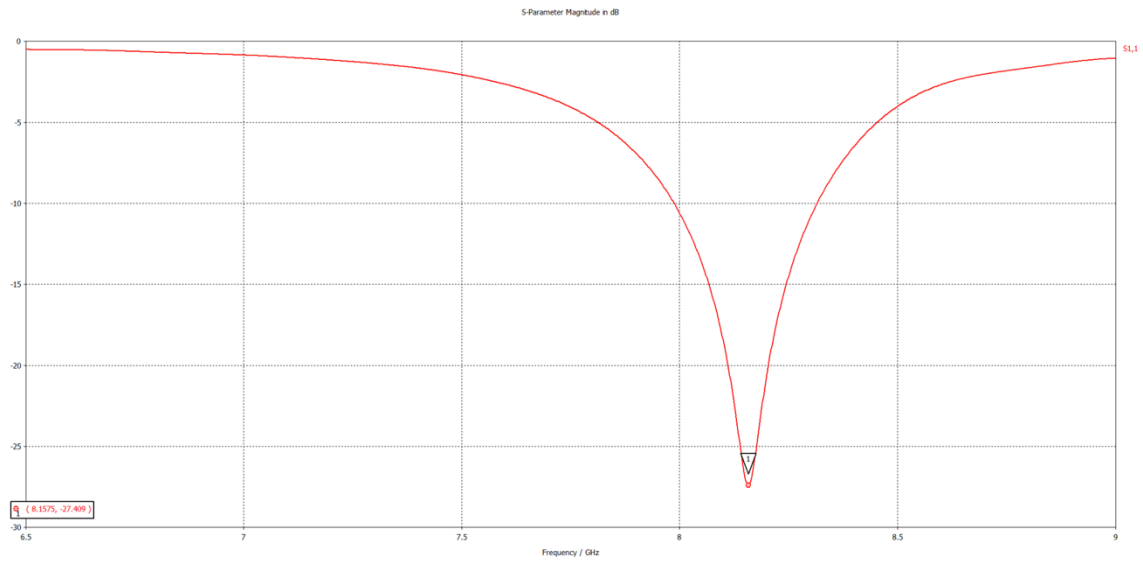


Figure 27: Return Loss, Center Frequency 8.16 GHz

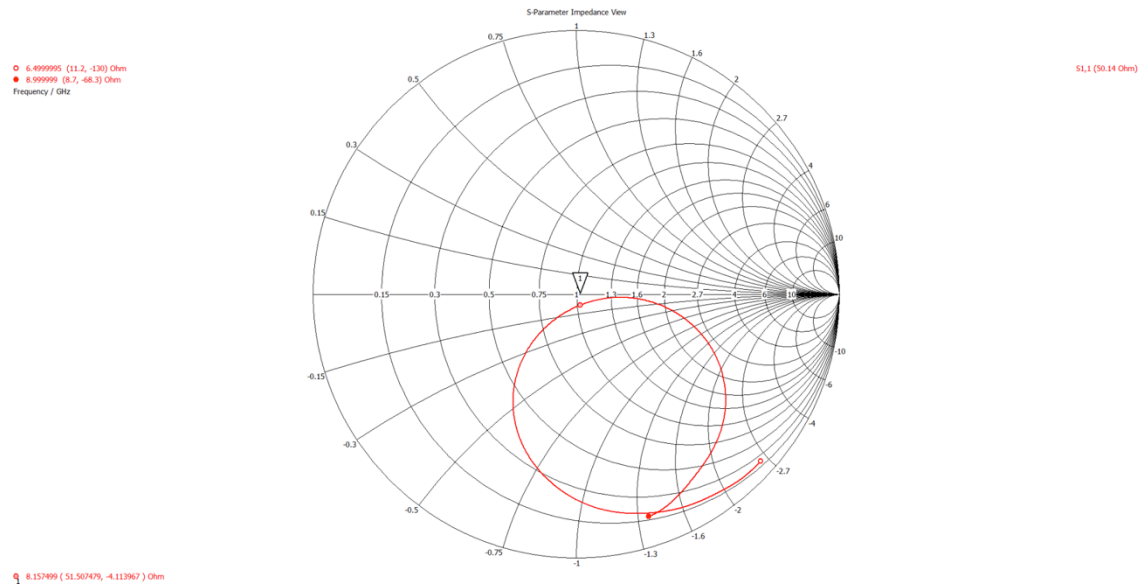


Figure 28: Z-Smith Chart, $Z_1 = (51.51 - 4.11j)\Omega$

3x3 Array: Center Element Excited, Remaining Elements Are Terminated to 50Ω

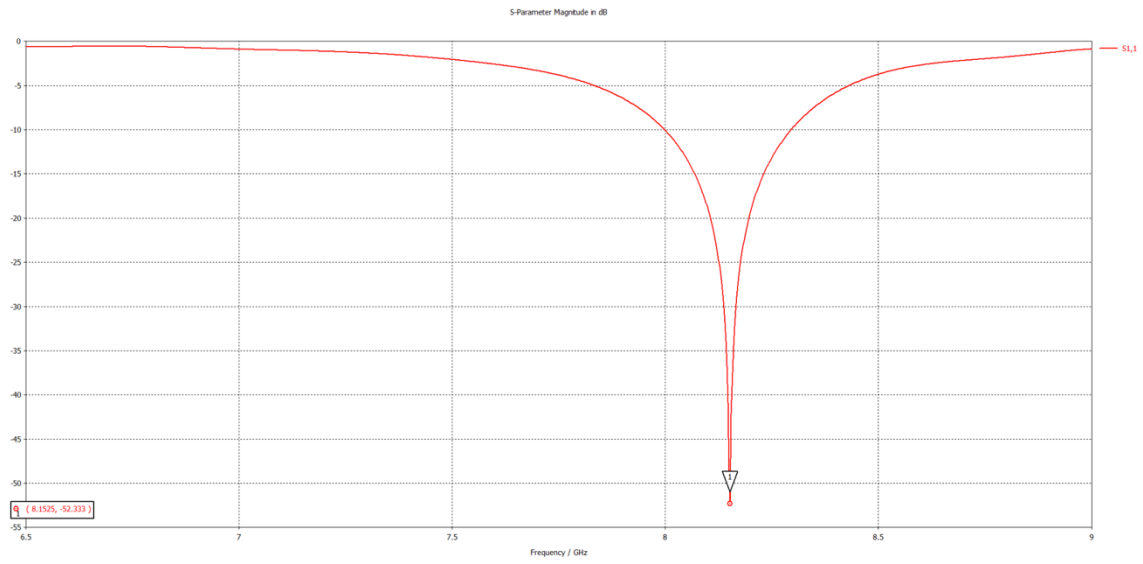


Figure 29: Return Loss, Center Frequency 8.15 GHz

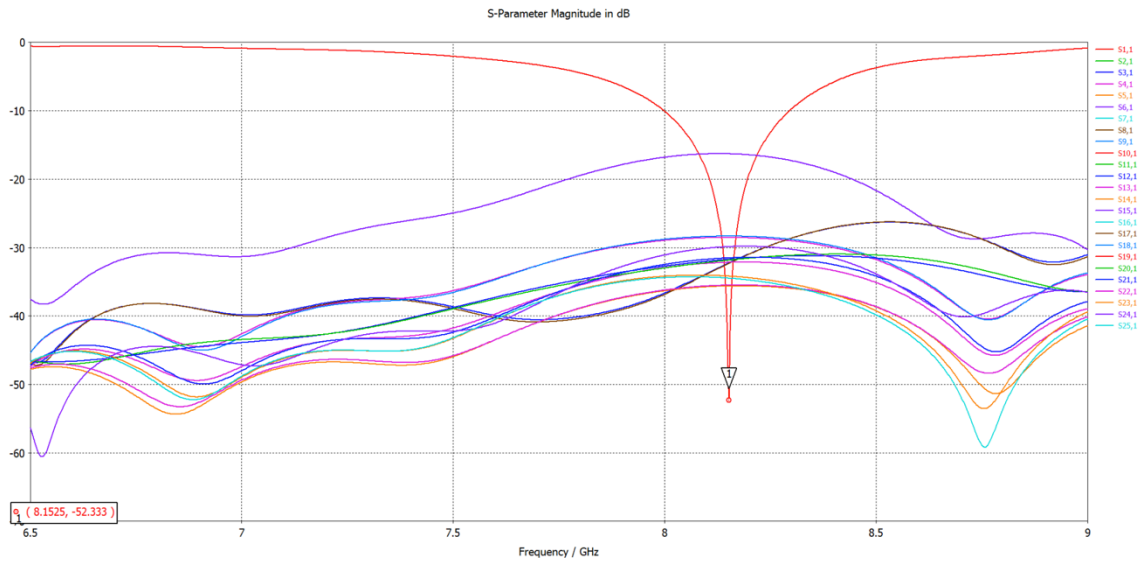


Figure 30: All S-Parameters

Table 4: Summary of Coupling, Power Coupled to Adjacent Elements is 6%

F = 8.15GHz	Coupling (dB)	Coupled Power (%)
S1,1	-48.91	n/a
S2,1	-28.52	0.14%
S3,1	-32.28	0.06%
S4,1	-28.52	0.14%
S5,1	-16.30	2.34%
S6,1	-16.30	2.34%
S7,1	-28.33	0.15%
S8,1	-32.36	0.06%
S9,1	-28.33	0.15%
S10,1	-35.63	0.03%
S11,1	-32.16	0.06%
S12,1	-31.73	0.07%
S13,1	-32.16	0.06%
S14,1	-35.63	0.03%
S15,1	-34.48	0.04%
S16,1	-29.88	0.10%
S17,1	-34.17	0.04%
S18,1	-35.50	0.03%
S19,1	-31.52	0.07%
S20,1	-31.85	0.07%
S21,1	-31.52	0.07%
S22,1	-35.50	0.03%
S23,1	-34.17	0.04%
S24,1	-29.88	0.10%
S25,1	-34.48	0.04%
		$\Sigma=6.24\%$

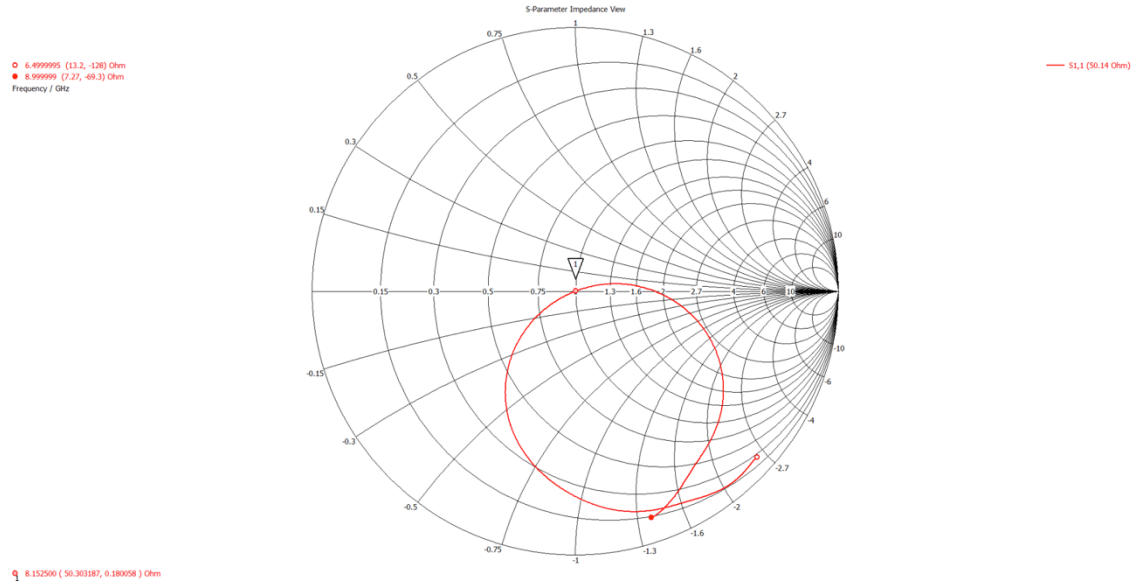


Figure 31: Z-Smith Chart, $Z_1=(50.30+0.18j)\Omega$