Maximally-Flat and Genetic Algorithm Solutions to Achieve Wideband Jaumann Absorbers

by

Wiwi Samsul

Department of Electrical and Computer Engineering
Duke University

Date: ______________________

Approved:

____
William T. Joines, Supervisor

____
Martin Brooke

____
Steven Cummer

____
Stéphane Larouche

____
Angel V. Peterchev

Dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering in the Graduate School of Duke University
2017

Abstract

Maximally-Flat and Genetic Algorithm Solutions to Achieve Wideband Jaumann Absorbers

by

Wiwi Samsul

Department of Electrical and Computer Engineering
Duke University

Date: ____________________

Approved:

____
William T. Joines, Supervisor

____
Martin Brooke

____
Steven Cummer

____
Stéphane Larouche

____
Angel V. Peterchev

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Electrical and Computer Engineering
in the Graduate School of Duke University
2017

Abstract

This work revolves around the Jaumann wave absorber, its solutions (either obtained through maximally flat solutions and through genetic algorithms), its possible implementation for designing wave-absorbing and radar-cloaking systems, and measurements of ambient RF energy in four Duke buildings to gauge the feasibility of harvesting energy from the RF band.

The first part of the introduction provides a short recap of radar, electromagnetic wave absorbers, and their equivalence to transmission line systems. The next part describes a brief summary of the history of the Jaumann wave absorber, its development over the years, and the short comings of previous attempts of finding its solutions. Finally, we conclude the introduction with contributions of this work.

The second chapter presents the mathematical formulation of Jaumann absorbers which is crucial in deriving their maximally flat solutions. This chapter lays the foundation to explore other solutions through cut-and-try and genetic algorithms.

The third chapter expands the search for solutions through automated means. We first try the simple cut-and-try method before moving on to a more sophisticated genetic algorithms. The bulk of this chapter describes the details of genetic algorithm and how they are implemented into our Jaumann wave absorber problem.

The fourth chapter provides the results and analysis obtained from the algorithms described in the previous chapter. However, only samples of such results are presented in this chapter while the bulk of the results is moved to Appendix A.9 for the

sake of brevity.

The fifth chapter describes our experimental works in verifying a sample result of the Jaumann wave absorber and the measurement of ambient RF energy in four Duke buildings. We also include the measurement results and analysis in this chapter. A brief comparison for the ambient energy measurement to other studies in other locations and the environment is also included.

Finally, the sixth chapter concludes this work.

Contents

Abstract **iv**

List of Tables **x**

List of Figures **xii**

List of Abbreviations and Symbols **xiv**

Acknowledgements **xv**

1 Introduction **1**

1.1 RCS Fundamentals 2

1.1.1 One layer absorber 3

1.2 Background on Jaumann Wave Absorber 6

1.3 Contributions 8

2 Mathematical Formulation of Jaumann Absorbers **9**

2.1 Determining Maximally Flat Solutions Through ABCD Matrix 9

2.1.1 One section 9

2.1.2 Two sections 10

2.1.3 Three sections 11

2.2 Bandwidth Enlargement 14

2.2.1 Reflection coefficient for one section case 14

2.2.2 Reflection coefficient for two sections case 15

2.2.3 Reflection coefficient for three sections case 15

2.2.4 Reflection coefficient for four, five and six sections case 16
2.2.5 Plot of reflection coefficients 16
2.2.6 Proof of maximally flat response 17
2.3 Jaumann Absorbers with Varying Separation 20
3 Genetic Algorithms and Their Application to Jaumann Wave Absorber Problems 22
3.1 Motivation 22
3.2 Components of Genetic Algorithm 23
3.3 Translation of GA components into Jaumann Wave Absorber problem 24
3.3.1 Population 24
3.3.2 Fitness criteria 25
3.3.3 Initial population 25
3.3.4 Survivor selection 26
3.3.5 Spawning next generation 26
3.3.6 Mutation process 26
3.4 Genetic algorithm flowchart 27
4 Results and Analysis 29
4.1 Cut-and-try Algorithm 29
4.2 Genetic Algorithm 30
4.2.1 Standard Results 31
4.2.2 Expanded Results 31
4.3 Results Comparison and Analysis 33
4.3.1 Gain-Bandwidth Product 33
4.3.2 Comparison with other works. 35
5 Measurements 38
5.1 Realization in transmission line 38

5.2 Measurement of ambient RF energy 40
5.2.1 Locations Stats 40
5.2.2 Antennas Profiles 41
5.2.3 Spectrum Analyzer 41
5.2.4 Ambient Energy Calculations 42

6 Conclusions and Future Work **47**

6.1 Conclusions 47
6.2 Future Work 48

A Appendix List **49**

A.1 System of equations to obtain solutions for four, five, and six sections case 50
A.1.1 Four sections 50
A.1.2 Five sections 52
A.1.3 Six sections 56
A.2 Expression of $\frac{d^2\Gamma}{d\theta^2}$ for three sections case 61
A.3 General form of reflection coefficients for 4, 5, and 6 layers case . . . 62
A.4 MATLAB code of GA for variable length case 63
A.5 Supplemental MATLAB code of for GA 72
A.6 MATLAB code for plotting convenience 92
A.7 General form of reflection coefficients with variable transmission line length for 1, 2, 3, 4, 5, and 6 layers case 94
A.8 Cut-and-try Algorithm Results 95
A.9 Genetic Algorithm Expanded Results Tables 96
A.9.1 Threshold level: -10 dB 98
A.9.2 Threshold level: -20 dB 107
A.10 Transmission Line Design and Calculation 116

A.11 Details of Location Measurements 119

A.11.1 CIEMAS 119

A.11.2 Teer 119

A.11.3 Physics 120

A.11.4 French 121

Bibliography **122**

Biography **126**

List of Tables

2.1 Summary of the G values up to six sections. 13

2.2 Impedance values for the case $Z_0 = 50\ \Omega$ up to six sections. 13

2.3 Impedance values for the case $Z_0 = 377\ \Omega$ up to six sections. 14

4.1 Summary of the G values from cut-and-try algorithm up to five sections. 30

4.2 Summary of standard GA solutions (S_{11} cut-off: -20 dB). 31

4.3 Summary of expanded GA solutions for 3 sections case (S_{11} cut-off: -20 dB). 32

4.4 Comparison with ref [24]. 37

4.5 Comparison with ref [34]. 37

5.1 List of resistor values used in 3 layers MFS experiments. 39

5.2 Summary of measurement locations. 41

5.3 Setting details of the spectrum analyzer. 42

5.4 Parameters of spiral antenna. 44

5.5 Summary of average ambient energy by location category. 45

5.6 Summary of average ambient energy by buildings. 46

A.1 GA expanded result for 1 section with Γ threshold -10 dB. 98

A.2 GA expanded result for 2 sections with Γ threshold -10 dB. 99

A.3 GA expanded result for 3 sections with Γ threshold -10 dB. 100

A.4 GA expanded result for 4 sections with Γ threshold -10 dB. 102

A.5 GA expanded result for 5 sections with Γ threshold -10 dB. 104
A.6 GA expanded result for 6 sections with Γ threshold -10 dB. 106
A.7 GA expanded result for 1 section with Γ threshold -20 dB. 107
A.8 GA expanded result for 2 sections with Γ threshold -20 dB. 108
A.9 GA expanded result for 3 sections with Γ threshold -20 dB. 109
A.10 GA expanded result for 4 sections with Γ threshold -20 dB. 111
A.11 GA expanded result for 5 sections with Γ threshold -20 dB. 113
A.12 GA expanded result for 6 sections with Γ threshold -20 dB. 115
A.13 Details of measurement locations in CIEMAS. 119
A.14 Details of measurement locations in Teer. 119
A.15 Details of measurement locations in Physics. 120
A.16 Details of measurement locations in French. 121

List of Figures

1.1 Wave absorber placed quarter wavelength away from a conducting sheet. 4

1.2 Wave absorber layers (top) and transmission line equivalent (bottom). 5

1.3 Proposed circuit equivalent to one layer wave absorber model. 6

2.1 One section transmission line circuit 9

2.2 Two sections circuit layout 10

2.3 Three sections circuit layout 12

2.4 Bandwidth plot, y-axis in dB and x-axis in Hz 17

3.1 Cut-and-try algorithm flowchart 23

3.2 Genetic algorithm flowchart 28

4.1 Comparison of maximally flat and cut-and-try solutions for 3 section case. 30

4.2 S_{11} plot of GA expanded result for 3 sections case (theta limit = 20). 32

4.3 Gain-bandwidth product comparison between maximally flat solutions. 33

4.4 Gain-bandwidth product comparison between 3 layer maximally flat solution and GA. 34

4.5 Gain-bandwidth product comparison at 3 layers between our work (GA), ref. [20], and ref. [13]. 34

5.1 Experimental set up for three sections Jaumann wave absorber in transmission line. 39

5.2 Comparison between simulation with added stray inductances (solid line) and measured S_{11} (dotted line) for 3 section case. 39

5.3 S_{11} comparison between using lumped vs chip resistors. 40

5.4 S_{11} graphs of antennas used in experiment. 41
5.5 Sample of a spectrum taken in CIEMAS 3rd floor skyway. 42
5.6 Average power spectrum across all buildings and all location categories. 45
A.1 Cut-and-try result for 2 sections case 95
A.2 Cut-and-try result for 3 sections case 95
A.3 Cut-and-try result for 4 sections case 96
A.4 Cut-and-try result for 5 sections case 96
A.5 Microstrip line 117
A.6 Parallel plate line 117

List of Abbreviations and Symbols

Symbols

η_0	Characteristic impedance of free space/air.
Γ	Reflection coefficient.
Z_0	Characteristic impedance of transmission line.
σ	Conductance of material.
ϵ	Permittivity of material.
μ	Permeability of material.
ω	Angular frequency.

Abbreviations

RCS	Radar Cross Section
JWA	Jaumann Wave Absorber
GA	Genetic Algorithm
RBW	Relative Bandwidth

Acknowledgements

The author thanks prof. William T. Joines for providing the opportunity of working in this field of study. Thank you to Duke University for its continuous support in this work until the end. Thank you to the committees who have provided the author with positive feedback for this work. Thank you to my parents, family, and friends for their emotional support. The author is grateful for being alive and hopeful that we, as humanity, will achieve space travel technology within 5 billion years before our sun becomes a red giant and renders Earth uninhabitable and another few million years before we may have to leave our solar system permanently when the sun becomes a white dwarf. The author hopes this work adds another step in the development of technology towards that goal.

1

Introduction

After James Clerk Maxwell published his famous equations, scientists and engineers have predicted the existence of electromagnetic waves. Only a mere 8 years after Maxwell's departure, Heinrich Hertz made the first observation of microwave electromagnetic radiation [1]. Soon after, Nikola Tesla envisioned a connected world through wireless technology.

Development of wireless technology has brought us the invention of RADAR (RAdio Detection And Ranging) during World War I. Radar is mainly used for detection of an object, specifically to determine the distance, speed, and/or size of the object. A radar transmitter shoots high frequency radio waves and a receiver detects the reflection from an object. A radar antenna can act as both transmitter and receiver, switching between the two roles alternately. As a transmitter, the antenna is usually designed to focus on one direction to achieve a high gain but rotated around to achieve a wide detection area. As a receiver, it merely waits for the reflected wave to travel back. The time delay between transmitted and reflected wave determines the distance between the transmitter and the detected object. A more sophisticated radar system also detects the Doppler shift between transmitted and reflected wave to determine the speed and direction on a radial axis with respect to the transmitter. The detection of size, however, is tricky since a radar system relies on the strength of the reflected wave to figure out the "size" of the detected object which may not be equal to the object's actual physical size. In particular, the stronger the reflected wave, the bigger the "size" detected by the radar system. This "size" is called Radar Cross Section (RCS) which will be discussed briefly in Section 1.1.

1.1 RCS Fundamentals

RCS is defined as the effective area of a detected object as if it is the only area which receives the transmitted radar signal and perfectly reflects the radar signal back to the receiver. In other words, if we substitute the detected object with a perfectly reflecting sphere, the cross section of the sphere is the RCS. Mathematically speaking, the RCS is expressed as follows:

$$\text{RCS} = \sigma = \frac{P_r}{P_t G_t} \frac{(4\pi r^2)^2}{A_{eff}} \quad (1.1)$$

where

$$\begin{aligned}
\text{RCS} (\sigma) &= \text{Radar Cross Section } (\text{m}^2) \\
P_r &= \text{reflected power (W)} \\
P_t &= \text{transmitted power (W)} \\
G_t &= \text{gain of transmitter antenna ()} \\
r &= \text{distance between transmitter and receiver (m)} \\
A_{eff} &= \text{effective area of receiver antenna } (\text{m}^2)
\end{aligned}$$

In most cases, especially in the development of stealth technology, we are interested to reduce the RCS of detected objects. On the contrary, designs of the commercial airplane would seek to maximize the RCS of the airplane for example. Several methods of reducing RCS are:

- designing the surface of the object in such a way that the radar wave is reflected as much as possible to any direction except the incoming wave [2],
- guiding the incoming wave around the object so that the transmitter receives no reflected wave back [3, 4],
- coating the object with Radar Absorbent Material (RAM), [5, 2]
- transmitting a wave which destructively interferes with the incoming radar wave (active cancellation stealth) [6, 7]

In this project, we primarily focus on reducing RCS by designing an electromagnetic wave absorbing system.

1.1.1 One layer absorber

Consider a wave absorber placed $\frac{\lambda}{4}$ away from a conducting sheet as shown in Fig. 1.1. Using transmission line equation (Eq. 1.2) and setting $\alpha = 0$ for lossless condition in Eq. 1.3, we obtain Eq. 1.4.

$$Z_1 = Z_0 \frac{Z_L + Z_0 \tanh(\gamma_a t)}{Z_0 + Z_L \tanh(\gamma_a t)} \tag{1.2}$$

$$\gamma_a = \alpha_a + j\beta_a \tag{1.3}$$

$$Z_1 = Z_0 \tanh(j\beta_a t) = jZ_0 \tan(\beta_a t) = \infty \tag{1.4}$$

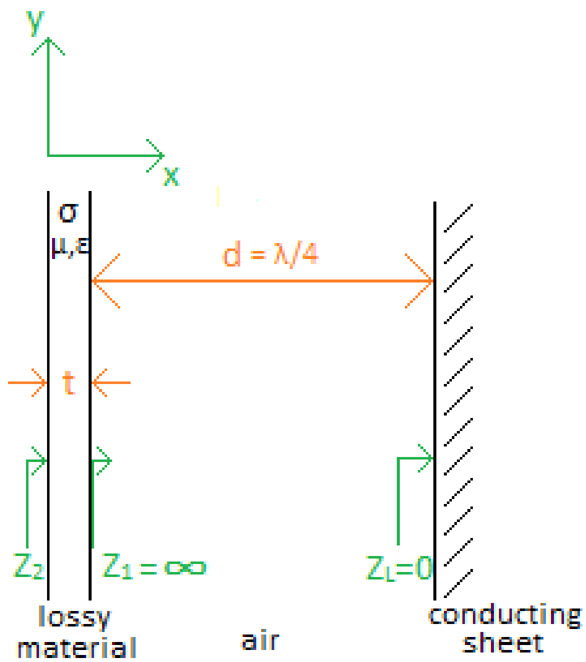

FIGURE 1.1: Wave absorber placed quarter wavelength away from a conducting sheet.

Similarly, for the case inside the lossy conductor (0 < x < t), which ABCD matrix is expressed in Eq. 1.5:

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}_{lossy} = \begin{bmatrix} \cosh(\gamma t) & Z_0 \sinh \gamma t \\ Y_0 \sinh \gamma t & \cosh \gamma t \end{bmatrix} \tag{1.5}$$

$$Z_{in} = \frac{AZ_L + B}{CZ_L + D} = \frac{A}{C} = \frac{Z_0}{\tanh(\gamma t)} \tag{1.6}$$

If (γt) is small :

$$\frac{Z_0}{\tanh(\gamma t)} \approx \frac{Z_0}{\gamma t}$$

$$= \frac{\sqrt{\frac{j\omega\mu}{\sigma + j\omega\epsilon}}}{\sqrt{j\omega\mu(\sigma + j\omega\epsilon)}\, t}$$

$$= \frac{1}{(\sigma + j\omega\epsilon)\, t} \tag{1.7}$$

If $\sigma \gg j\omega\epsilon$:

$$Z_{in} = \frac{1}{\sigma t} = 377\,\Omega \tag{1.8}$$

Thus, given a specific value for sigma, the thickness can be tailored to achieve maximum absorption. Similarly, for square resistance sheet (width (W) = length (L)) we obtain:

$$R = \frac{1}{\sigma}\frac{L}{A} = \frac{L}{\sigma W t} = \frac{1}{\sigma t} \equiv R_s \tag{1.9}$$

where R_s is sheet resistance. Therefore, we can model the wave absorber system with a transmission line circuit as shown in Fig. 1.2 where $R_{sn} = Z_n$.

However, doing it this way yields a narrow bandwidth of absorption, due to the requirement of $\lambda/4$ separation between the conducting sheet with the absorbing layer. In the interest to improve the bandwidth, we use Jaumann wave absorber system which is discussed more in section 1.2 [8, 9].

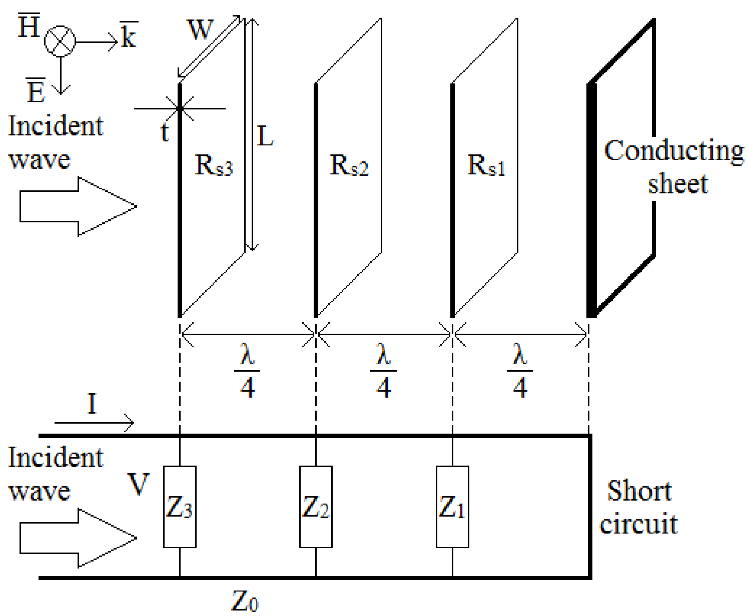

FIGURE 1.2: Wave absorber layers (top) and transmission line equivalent (bottom).

Finally, Fig. 1.3 shows our proposed circuit to one layer Jaumann wave absorber model. Incoming E-field induces voltage difference as given by V = EL. The diode bridge rectifies the induced current so that the resistor always have polarity as indicated in the figure. In addition, we can also tune the length of conducting connectors so that the amount of stray inductances cancels out the innate capacitance present in the diodes.

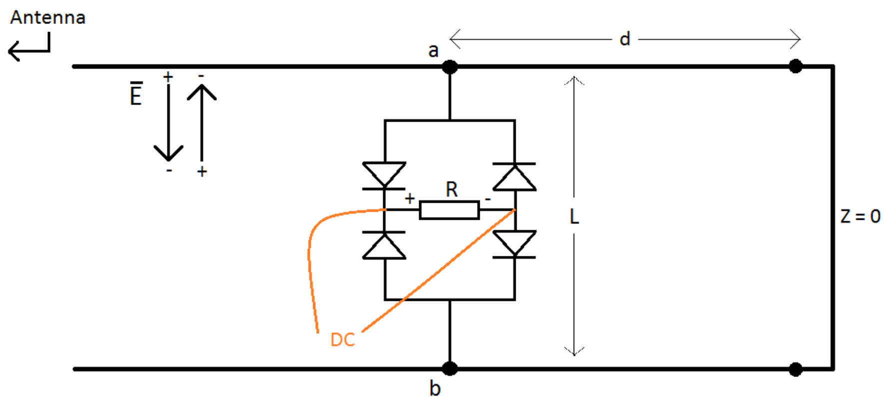

FIGURE 1.3: Proposed circuit equivalent to one layer wave absorber model.

1.2 Background on Jaumann Wave Absorber

Electromagnetic wave absorbing materials were first developed during WWII era when both the axis and allies were competing to build radar technology to detect enemies' submarines and prevent their own from being detected. Therefore, a wave absorbing layer which renders a metal surface undetectable by radar was considered crucial for winning the war. On the axis side, a German engineer named J. Jaumann designed a set of solutions of impedance values which absorbs at least 90% of radar signal power between 1 - 10 GHz [8]. Meanwhile on the allies side, an American inventor W. W. Salisbury patented his design of an electromagnetic absorbent body, an early idea from which wave absorber layers today are named as Salisbury screens [10].

Throughout the rest of the 20th century, development of radar absorbing materials (RAM) continued [9, 11, 12, 13, 14, 15, 16]. However, efforts in improving multiple layers of wave absorbers so far has been limited either by lack of computational power or the scope of its application. For instance, Jaumann produced his solutions through a trial-and-error process assisted by an American invention called the Smith chart. He started from the zero impedance point (short circuit) in the chart, rotated $\lambda/4$ and added in parallel an arbitrary impedance value Z_1, rotated $\lambda/4$ and added in parallel another arbitrary impedance value Z_2, and repeated this process until he ended on the center point of the chart (no reflection). The set of solution $Z_1, Z_2, ..., Z_n$ was then tested for its reflection coefficient. If the reflection coefficient was unsatisfactory, then the whole process was repeated to search for a new solution set. In another example, Knott et al [12] presents a more systematic approach by using an S-matrix to solve for maximally-flat response of the Jaumann absorber. However, they only provided the solution for 2 layers. Meanwhile, Fante and McCormack [14] used the multiple wave reflections approach to calculate the admittance for 1, 2, and 3 layers of electric screens, but their approach became incredibly complicated for more than three layers. Another work by Nortier et al [13] provides a design table for up to 7 layers, but their paper did not explain how they arrived at their values. Moreover, their table entries produce S_{11} plots with significant ripples in the pass band (not maximally-flat).

In this work, we have successfully expanded the maximally-flat solutions of Jaumann wave absorbers up to 6 layers or more, an improvement over previous attempts recorded in the literature [9, 11, 12, 17, 13, 14, 15, 16]. Furthermore, our results not only produce the maximum absorption at center frequency, but the mathematical procedure we developed can also be expanded to solutions of any number of layers, given enough computational resources. More details on this is presented in chapter 2.

In chapter 3, we describe how we use the result from mathematical procedure from chapter 2 and feed it into genetic algorithm to search for more solutions. Moreover, we can fine tune the parameters in genetic algorithm to obtain solutions which yield minimum bandwidth, desired minimum reflection coefficient at center frequency, and/or center frequency of the bandwidth.

1.3 Contributions

Our contributions are summarized as follows:

- We produced a theoretical framework to obtain maximally flat solution of Jaumann wave problem.
- Using this theoretical framework, we explore other solutions to Jaumann wave problem using Genetic Algorithm. We compile 240 unique solutions, (120 each for -10 and -20 dB reflection coefficient limit; for 1 to 6 layers with 20 solutions each for frequency range which corresponds to f_0 - 20 f_0 for each solution).
- In order to test the viability of wave absorber system as energy harvester, we measured the RF ambient energy level on 4 Duke buildings (CIEMAS, Teer, French, and Physics). The report of which is under preparation.

2

Mathematical Formulation of Jaumann Absorbers

2.1 Determining Maximally Flat Solutions Through ABCD Matrix

2.1.1 One section

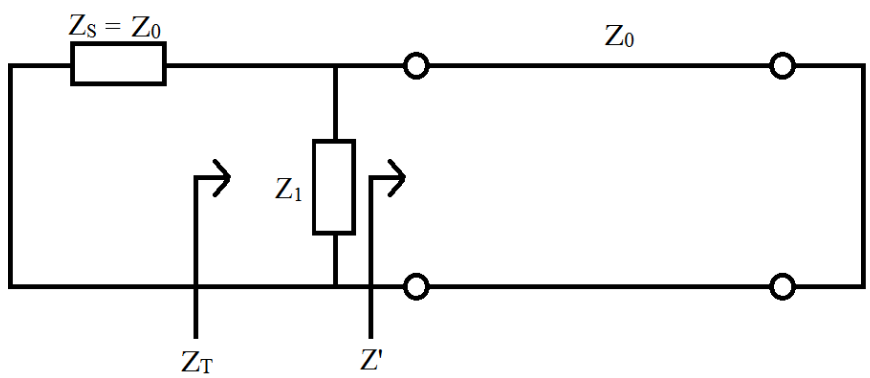

FIGURE 2.1: One section transmission line circuit

By using ABCD matrix, the circuit shown in Fig. 2.1 can be represented as:

$$\begin{aligned} ABCD &= \begin{bmatrix} 1 & 0 \\ G_1 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & jZ_0 \sin\theta \\ jY_0 \sin\theta & \cos\theta \end{bmatrix} \\ &= \begin{bmatrix} \cos\theta & jZ_0 \sin\theta \\ G_1 \cos\theta + jY_0 \sin\theta & jG_1 Z_0 \sin\theta + \cos\theta \end{bmatrix} \end{aligned} \tag{2.1}$$

Where $G_1 = \frac{1}{Z_1}$

And the total impedance of the circuit is:

$$
\begin{aligned}
Z_T &= \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D} \\
&= \frac{jZ_0 \sin\theta}{jG_1 Z_0 \sin\theta + \cos\theta}
\end{aligned} \tag{2.2}
$$

Setting $Z_0 = 1$ and $Z_T = 1$ (which means we normalize Z_0 and set $Z_T = Z_S$ for maximum power transfer), we obtain:

$$
\begin{aligned}
jG_1 \sin\theta + \cos\theta - j\sin\theta = 0 \\
j\sin\theta\,(G_1 - 1) + \cos\theta = 0
\end{aligned} \tag{2.3}
$$

When the length of transmission line is $\frac{\lambda}{4}$, we can solve for $G_1 = 1$, where G_1 is the normalized conductance.

2.1.2 Two sections

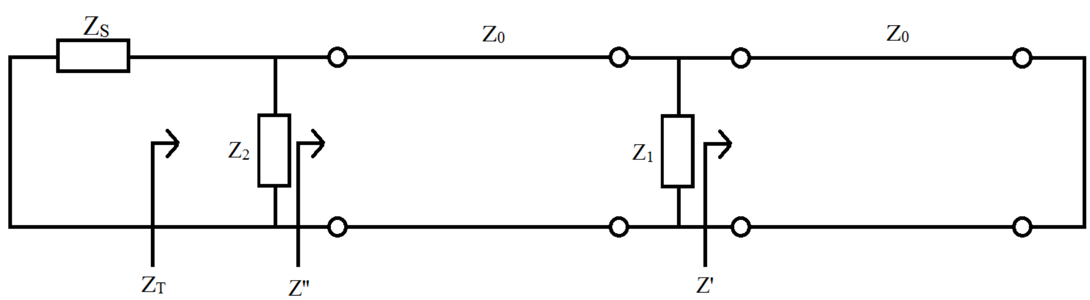

FIGURE 2.2: Two sections circuit layout

The ABCD matrix for the circuit shown in Fig. 2.2 is:

$$
\begin{aligned}
ABCD &= \begin{bmatrix} 1 & 0 \\ G_2 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & jZ_0 \sin\theta \\ jY_0 \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ G_1 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & jZ_0 \sin\theta \\ jY_0 \sin\theta & \cos\theta \end{bmatrix} \\
&= \begin{bmatrix} \cos 2\theta + jG_1 Z_0 \cos\theta \sin\theta & Z_0 \sin\theta\,(2j\cos\theta - G_1 Z_0 \sin\theta) \\ \frac{1}{2Z_0}\,(G_1 Z_0 + (G_1 + 2G_2) Z_0 \cos 2\theta + j(2 + G_1 G_2 Z_0^2) \sin 2\theta) & \cos^2\theta + j(G_1 + 2G_2) Z_0 \cos\theta \sin\theta - (1 + G_1 G_2 Z_0^2) \sin^2\theta \end{bmatrix}
\end{aligned} \tag{2.4}
$$

The load Z_T with $Z_L = 0$ (due to short circuit at the end) is:

$$\begin{aligned} Z_T &= \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D} \\ &= \frac{Z_0 \sin\theta\,(2j\cos\theta - G_1 Z_0 \sin\theta)}{\cos^2\theta + j(G_1+G_2)Z_0\cos\theta\sin\theta - (1+G_1G_2Z_0^2)\sin^2\theta} \div \frac{j\sin^2\theta}{j\sin^2\theta} \\ &= \frac{2Z_0\cot\theta + jG_1Z_0^2}{Z_0\cot\theta(G_1+2G_2) + j(1+G_1G_2Z_0^2 - \cot^2\theta)} \end{aligned} \tag{2.5}$$

Setting $Z_0 = 1$ and $Z_T = 1$, we obtain:

$$\begin{aligned} \cot\theta(G_1+2G_2) + j(1+G_1G_2-\cot^2\theta) &= 2\cot\theta + jG_1 \\ \cot\theta(G_1+2G_2-2) + j(1+G_1G_2-G_1-\cot^2\theta) &= 0 \end{aligned} \tag{2.6}$$

Note, make sure that all present trigonometric functions don't turn into an independent constant when $\theta \to \pi/2$ (any independent trigonometric function must go to zero, but trigonometric function which attached to a set of other parts are fine).

Since we have set $Z_T = 1$, setting all the coefficients equal to zero (in addition to $\theta \to \pi/2$) in Eq. 2.6 will give the G_1 and G_2 values which satisfies the impedance matching:

$$\begin{cases} G_1 + 2G_2 - 2 = 0 \\ 1 + G_1G_2 - G_1 = 0 \end{cases} \tag{2.7}$$

Solution is: $G_1 = 1.4142$, $G_2 = 0.29289$.

2.1.3 Three sections

$$ABCD = \begin{bmatrix} 1 & 0 \\ G_3 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & jZ_0\sin\theta \\ jY_0\sin\theta & \cos\theta \end{bmatrix} \cdots\cdots \begin{bmatrix} 1 & 0 \\ G_1 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta & jZ_0\sin\theta \\ jY_0\sin\theta & \cos\theta \end{bmatrix} \tag{2.8}$$

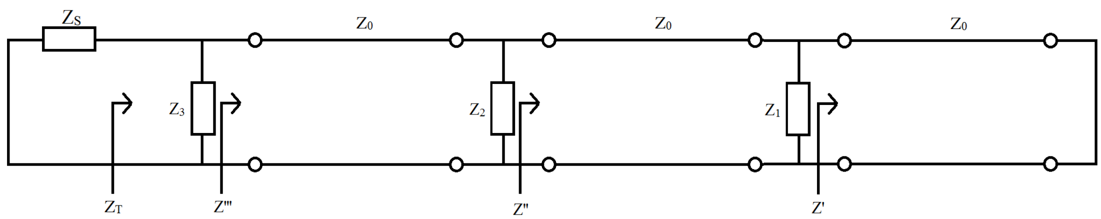

FIGURE 2.3: Three sections circuit layout

The result of matrix multiplication is:

$$
\begin{aligned}
A &= \cos^3\theta + j(2G_1+G_2)Z_0\cos^2\theta\sin\theta - (3+G_1G_2Z_0^2)\cos\theta\sin^2\theta - jG_2Z_0\sin^3\theta \\
B &= (1/2)jZ_0\sin\theta(2 - G_1G_2Z_0^2 + (4+G_1G_2Z_0^2)\cos 2\theta + 2j(G_1+G_2)Z_0\sin 2\theta) \\
C &= \frac{1}{4Z_0}\Big((2G_1Z_0 + 2G_2Z_0 - G_1G_2G_3Z_0^3)\cos\theta + Z_0(2G_1+2G_2+4G_3+G_1G_2G_3Z_0^2)\cos 3\theta + \\
&\quad 2j(2+G_1(G_2+2G_3)Z_0^2 + (4+2G_2G_3Z_0^2+G_1(G_2+2G_3)Z_0^2)\cos 2\theta)\sin\theta\Big) \\
D &= \cos^3\theta + j(G_1+2G_2+3G_3)Z_0\cos^2\theta\sin\theta - (3+2G_2G_3Z_0^2+G_1(G_2+2G_3)Z_0^2)\cos\theta\sin^2\theta - \\
&\quad jZ_0(G_1+G_3+G_1G_2G_3Z_0^2)\sin^3\theta
\end{aligned}
\tag{2.9}
$$

The load Z_T with short circuit load is:

$$
\begin{aligned}
Z_T &= \frac{AZ_L+B}{CZ_L+D} = \frac{B}{D} \\
&= \frac{Z_0\sin\theta(-3\cos^2\theta+\sin\theta(-2j(G_1+G_2)Z_0\cos\theta+(1+G_1G_2Z_0^2)\sin\theta))}{j\cos^3\theta-(G_1+2G_2+3G_3)Z_0\cos^2\theta\sin\theta-j(3+2G_2G_3Z_0^2+G_1(G_2+2G_3)Z_0^2)\cos\theta\sin^2\theta+Z_0(G_1+G3+G_1G_2G_3Z_0^2)\sin^3\theta}
\end{aligned}
\tag{2.10}
$$

Setting $Z_0 = 1$ and $Z_T = 1$ and cross multiply, we obtain:

$$
\begin{aligned}
&\cos^2\theta(-3+G_1+2G_2+3G_3-j\cot\theta) + j(3+2G_2(-1+G_3)+G_1(-2+G_2+2G_3))\cos\theta\sin\theta \\
&\qquad -(-1+G_1(1+G_2(-1+G_3))+G_3)\sin^2\theta = 0
\end{aligned}
\tag{2.11}
$$

Therefore, the system of equations that we need to solve are:

$$\begin{cases} -3+G_1+2G_2+3G_3=0 \\ 3+2G_2(-1+G_3)+G_1(-2+G_2+2G_3)=0 \\ -1+G_1(1+G_2(-1+G_3))+G_3=0 \end{cases} \tag{2.12}$$

And the solution is: $G_1 = 1.644$, $G_2 = 0.5143$, $G_3 = 0.1091$.

In the interest of brevity, list of system of equations for four, five, and six sections case are moved to Appendix A.1. Table 2.1 lists the summary of solutions and the resulting relative bandwidth (RBW).

Summary of the solutions

Table 2.1: Summary of the G values up to six sections.

# of sections	G_1	G_2	G_3	G_4	G_5	G_6	RBW
1	1.000	-	-	-	-	-	0.253
2	1.414	0.293	-	-	-	-	0.673
3	1.644	0.514	0.109	-	-	-	0.926
4	1.782	0.670	0.234	0.044	-	-	1.089
5	1.868	0.778	0.343	0.114	0.019	-	1.203
6	1.921	0.853	0.430	0.188	0.057	0.008	1.288

Table 2.2: Impedance values for the case $Z_0 = 50\ \Omega$ up to six sections.

# of sections	Z_1 (Ω)	Z_2 (Ω)	Z_3 (Ω)	Z_4 (Ω)	Z_5 (Ω)	Z_6 (Ω)
1	50.000	-	-	-	-	-
2	35.356	170.713	-	-	-	-
3	30.413	97.220	458.295	-	-	-
4	28.052	74.683	213.950	1128.668	-	-
5	26.768	64.292	145.900	440.141	2659.574	-
6	26.029	58.644	116.198	266.382	883.392	6097.561

Table 2.3: Impedance values for the case $Z_0 = 377\ \Omega$ up to six sections.

# of sections	Z_1 (Ω)	Z_2 (Ω)	Z_3 (Ω)	Z_4 (Ω)	Z_5 (Ω)	Z_6 (Ω)
1	377.000	-	-	-	-	-
2	266.582	1287.172	-	-	-	-
3	229.318	733.035	3455.545	-	-	-
4	211.513	563.107	1613.179	8510.158	-	-
5	201.831	484.763	1100.088	3318.662	20053.191	-
6	196.262	442.177	876.133	2008.524	6660.777	45975.610

2.2 Bandwidth Enlargement

2.2.1 Reflection coefficient for one section case

Earlier we found that the Z_T for one section is:

$$\begin{aligned} Z_T &= \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D} \\ &= \frac{jZ_0 \sin\theta}{jG_1 Z_0 \sin\theta + \cos\theta} \end{aligned} \tag{2.13}$$

Then, the reflection coefficient is:

$$\begin{aligned} \Gamma_1 &= \frac{Z_T - Z_S}{Z_T + Z_S} \\ &= \frac{-(Z_S \cos\theta + jZ_0(-1 + G_1 Z_S)\sin\theta)}{Z_S \cos\theta + jZ_0(1 + G_1 Z_S)\sin\theta} \end{aligned} \tag{2.14}$$

Assuming the case where $Z_S = Z_0 = 50\,\Omega$. Substituting the de-normalized solution for one section ($G_1 = \frac{1}{Z_0}$), we have:

$$\Gamma_1 = \frac{-\cos\theta}{\cos\theta + 2j\sin\theta} \tag{2.15}$$

2.2.2 *Reflection coefficient for two sections case*

$$\begin{aligned} Z_T &= \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D} \\ &= \frac{-2Z_0 \sin\theta(-2j\cos\theta + G_1 Z_0 \sin\theta)}{(2 + G_1 G_2 Z_0^2)\cos 2\theta + Z_0(-G_1 G_2 Z_0 + j(G_1 + 2G_2)\sin 2\theta)} \end{aligned} \tag{2.16}$$

Then, the reflection coefficient is:

$$\begin{aligned} \Gamma_2 &= \frac{Z_T - Z_S}{Z_T + Z_S} \\ &= \frac{(-2Z_S + G_1 Z_0^2(1 - G_2 Z_S))\cos 2\theta + Z_0(G_1 Z_0(-1 + G_2 Z_S) - j(-2 + G_1 Z_S + 2G_2 Z_S)\sin 2\theta)}{((2Z_S + G_1 Z_0^2(1 + G_2 Z_S))\cos 2\theta + Z_0((-G_1) Z_0(1 + G_2 Z_S) + j(2 + G_1 Z_S + 2G_2 Z_S)\sin 2\theta))} \end{aligned} \tag{2.17}$$

Assuming the case where $Z_S = Z_0 = 50\,\Omega$. Substituting the de-normalized solution for two sections $\left(G_1 = \frac{1.414}{Z_0}, G_2 = \frac{0.293}{Z_0}\right)$, we obtain:

$$\Gamma_2 = 0.079 + \frac{0.447 - 2.605\cos^2\theta - 0.632j\cos\theta\sin\theta}{-1.828 + 3.829\cos 2\theta + 8.000j\cos\theta\sin\theta} \tag{2.18}$$

2.2.3 *Reflection coefficient for three sections case*

$$\begin{aligned} Z_T &= \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D} \\ &= \frac{Z_0 \sin\theta(-3\cos^2\theta + \sin\theta(-2j(G_1 + G_2)Z_0\cos\theta + (1 + G_1 G_2 Z_0^2)\sin\theta))}{j\cos^3\theta - (G_1 + 2G_2 + 3G_3)Z_0\cos^2\theta\sin\theta - j(3 + 2G_2 G_3 Z_0^2 + G_1(G_2 + 2G_3)Z_0^2)\cos\theta\sin^2\theta + Z_0(G_1 + G_3 + G_1 G_2 G_3 Z_0^2)\sin^3\theta} \end{aligned} \tag{2.19}$$

Then, the reflection coefficient is:

$$\begin{aligned} \Gamma_3 &= \frac{Z_T - Z_S}{Z_T + Z_S} \\ &= \frac{(-Z_S\cos^3\theta - jZ_0(-3 + (G_1 + 2G_2 + 3G_3)Z_S)\cos^2\theta\sin\theta + (3Z_S + Z_0^2(-2(G_1 + G_2) + (G_1 G_2 + 2(G_1 + G_2)G_3)Z_S))\cos\theta\sin\theta^2 + jZ_0(-1 + G_3 Z_S + G_1(Z_S + G_2 Z_0^2(-1 + G_3 Z_S)))\sin\theta^3)}{(Z_S\cos^3\theta + jZ_0(3 + (G_1 + 2G_2 + 3G_3)Z_S)\cos^2\theta\sin\theta - (3Z_S + Z_0^2(2(G_1 + G_2) + (G_1 G_2 + 2(G_1 + G_2)G_3)Z_S))\cos\theta\sin\theta^2 - jZ_0(1 + G_3 Z_S + G_1(Z_S + G_2 Z_0^2(1 + G_3 Z_S)))\sin\theta^3)} \end{aligned} \tag{2.20}$$

Assuming the case where $Z_S = Z_0 = 50\,\Omega$. Substituting the de-normalized solution for three sections $\left(G_1 = \frac{1.644}{Z_0}, G_2 = \frac{0.514}{Z_0},\ \ G_3 = \frac{0.109}{Z_0}\right)$, we obtain:

$$\Gamma_3 = \frac{-0.5j\cos\theta(1-\cos\theta)}{-1.408j\cos\theta + 2.408j\cos 3\theta + 1.268\sin\theta - 2.422\sin 3\theta} \tag{2.21}$$

2.2.4 *Reflection coefficient for four, five and six sections case*

The expressions of the reflection coefficients after de-normalization and substitution of the G values for each case:

$$\Gamma_4 = \frac{-\cos^4\theta}{-0.085 - 1.795\cos 2\theta + 2.880\cos 4\theta - 3.532j\cos\theta\sin\theta + 2.883j\sin 4\theta} \tag{2.22}$$

$$\Gamma_5 = \frac{0.625j\cos\theta + 0.312j\cos 3\theta + 0.063j\cos 5\theta}{0.112j\cos\theta + 2.243j\cos 3\theta - 3.356j\cos 5\theta - 0.072\sin\theta - 2.237\sin 3\theta + 3.356\sin 5\theta} \tag{2.23}$$

$$\Gamma_6 = \frac{0.312 + 0.469\cos 2\theta + 0.187\cos 4\theta + 0.031\cos 6\theta}{0.024 + 0.100\cos 2\theta + 2.714\cos 4\theta - 3.837\cos 6\theta + 0.177j\cos\theta\sin\theta + 2.712j\sin 4\theta - 3.838j\sin 6\theta} \tag{2.24}$$

2.2.5 *Plot of reflection coefficients*

Fig. 2.4 shows the plots of Eq. 2.15, 2.18, 2.21, 2.22, 2.23, and 2.24. The plots show that $\Gamma = 0$ when $\theta = \frac{\pi}{2} \pm n\pi$, $n = 0, 1, 2, ...$

where:

$$\theta = \frac{\pi}{2}\frac{f}{f_0} \tag{2.25}$$

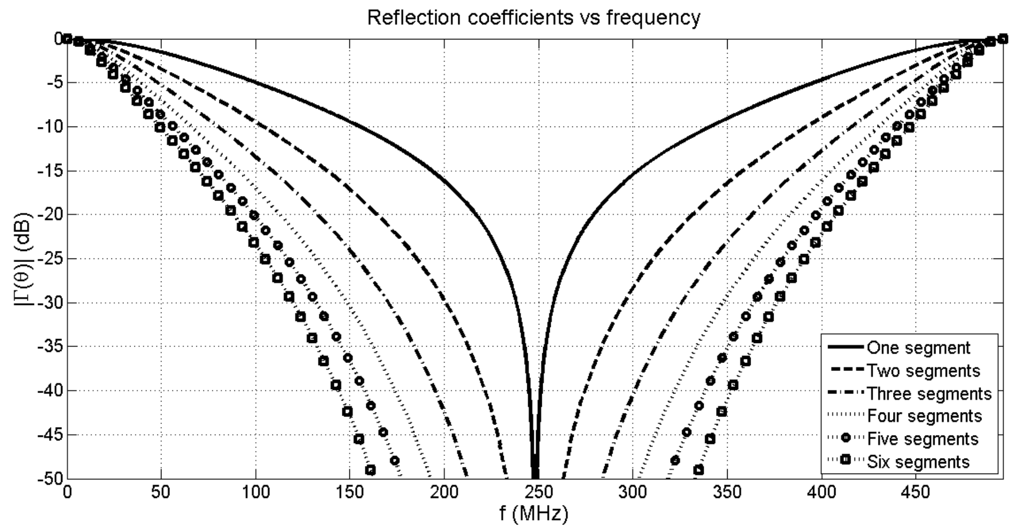

FIGURE 2.4: Bandwidth plot, y-axis in dB and x-axis in Hz

$$f = \text{frequency}$$
$$f_0 = \text{center frequency}$$

2.2.6 Proof of maximally flat response

Obtaining a solution which yields maximally flat response requires that the reflection coefficient and its derivatives be set to zero at the center frequency. Despite of much more algebraically tedious, we will see that this approach also leads us to the same solution as ABCD matrix approach.

One section

Starting from Eq. 2.14:

$$\Gamma_1 = \frac{-(Z_S \cos\theta + jZ_0(-1 + G_1 Z_S)\sin\theta)}{Z_S \cos\theta + jZ_0(1 + G_1 Z_S)\sin\theta}$$

Setting Z_S and $Z_0 = 1$:

$$\Gamma_1 = -\frac{\cos\theta + j(-1 + G_1)\sin\theta}{\cos\theta + j(1 + G_1)\sin\theta}$$

Setting $\theta \to \dfrac{\pi}{2}$:

$$\Gamma_1 = \frac{1 - G_1}{1 + G_1} \tag{2.26}$$

Solving for $\Gamma_1 = 0$ gives $G_1 = 1$.

Two sections

From Eq. 2.17 and setting Z_S and $Z_0 = 1$:

$$\Gamma_2 = \frac{(G_1(-1 + G_2) + (-2 + G_1 - G_1 G_2)\cos 2\theta - j(-2 + G_1 + 2G_2)\sin 2\theta)}{(-G_1(1 + G_2) + (2 + G_1 + G_1 G_2)\cos 2\theta + j(2 + G_1 + 2G_2)\sin 2\theta)} \tag{2.27}$$

$$\frac{d\Gamma_2}{d\theta} = \frac{(4j(4 - G_1^2 + G_1^2 \cos 2\theta + 2jG_1 \sin 2\theta))}{((-G_1)(1 + G_2) + (2 + G_1 + G_1 G_2)\cos 2\theta + j(2 + G_1 + 2G_2)\sin 2\theta)^2} \tag{2.28}$$

Setting $\theta \to \frac{\pi}{2}$:

$$\Gamma_2 = \frac{1 - G_1 + G_1 G_2}{1 + G_1 + G_1 G_2} \tag{2.29}$$

$$\frac{d\Gamma_2}{d\theta} = -\frac{(2j(-2 + G_1^2))}{(1 + G_1 + G_1 G_2)^2} \tag{2.30}$$

Solving for $\Gamma_2 = 0$ and $\frac{d\Gamma_2}{d\theta} = 0$ yields:

$$\begin{aligned} G_1 &= 1.414 \\ G_2 &= 0.293 \end{aligned}$$

Three sections

From Eq. 2.20:

$$\Gamma_3 = \frac{(-Z_S\cos^3\theta - jZ_0(-3+(G_1+2G_2+3G_3)Z_S)\cos^2\theta\sin\theta + (3Z_S+Z_0^2(-2(G_1+G_2)+(G_1G_2+2(G_1+G_2)G_3)Z_S))\cos\theta\sin\theta^2 + jZ_0(-1+G_3Z_S+G_1(Z_S+G_2Z_0^2(-1+G_3Z_S)))\sin^3\theta)}{(Z_S\cos^3\theta + jZ_0(3+(G_1+2G_2+3G_3)Z_S)\cos^2\theta\sin\theta - (3Z_S+Z_0^2(2(G_1+G_2)+(G_1G_2+2(G_1+G_2)G_3)Z_S))\cos\theta\sin^2\theta - jZ_0(1+G_3Z_S+G_1(Z_S+G_2Z_0^2(1+G_3Z_S)))\sin^3\theta)} \tag{2.31}$$

Setting Z_S and $Z_0 = 1$:

$$\Gamma_3 = \frac{-\cos^3\theta - j(-3+G_1+2G_2+3G_3)\cos^2\theta\sin\theta + (3+G_1G_2-2(G_1+G_2)+2(G_1+G_2)G_3)\cos\theta\sin^2\theta + j(-1+G_1(1+G_2(-1+G_3))+G_3)\sin^3\theta}{\cos^3\theta + j(3+G_1+2G_2+3G_3)\cos^2\theta\sin\theta - (3+2G_2(1+G_3)+G_1(2+G_2+2G_3))\cos\theta\sin^2\theta - j(1+G_3+G_1(1+G_2+G_2G_3))\sin^3\theta} \tag{2.32}$$

$$\frac{d\Gamma_3}{d\theta} = \frac{(j(24-8G_1^2-4G_2^2+3G_1^2G_2^2-4G_1(2G_2+G_1(-2+G_2^2)))\cos 2\theta + G_2(8G_1+4G_2+G_1^2G_2)\cos 4\theta + 16jG_1\sin 2\theta - 4jG_1^2G_2\sin 2\theta - 8jG_1G_2^2\sin 2\theta + 8jG_2\sin 4\theta + 2jG_1^2G_2\sin 4\theta + 4jG_1G_2^2\sin 4\theta))}{(4(\cos^3\theta + j(3+G_1+2G_2+3G_3)\cos^2\theta\sin\theta - (3+2G_2(1+G_3)+G_1(2+G_2+2G_3))\cos\theta\sin^2\theta - j(1+G_3+G_1(1+G_2+G_2G_3))\sin^3\theta)^2)} \tag{2.33}$$

(The expression for $\frac{d^2\Gamma_3}{d\theta^2}$ is moved to Appendix A.2 since it's too long.)

Setting $\theta \to \frac{\pi}{2}$:

$$\Gamma_3 = \frac{-1+G_1(1+G_2(-1+G_3))+G_3}{1+G_1+G_1G_2+G_3+G_1G_2G_3} \tag{2.34}$$

$$\frac{d\Gamma_3}{d\theta} = -\frac{(2j(3+2G_1G_2+G_1^2(-2+G_2^2)))}{(1+G_3+G_1(1+G_2+G_2G_3))^2} \tag{2.35}$$

$$\frac{d^2\Gamma_3}{d\theta^2} = -\frac{(4(9+4G_2(1+G_3)+G_1(8+7G_2+8G_3)+G_1^2(-4+G_2(1+3G_2+G_3))+G_1^3(-4(1+G_3)+G_2(-3+G_2(1+G_2+G_3)))))}{(1+G_3+G_1(1+G_2+G_2G_3))^3} \tag{2.36}$$

Solving for $\Gamma_3 = 0$, $\frac{d\Gamma_3}{d\theta} = 0$ and $\frac{d^2\Gamma_3}{d\theta^2} = 0$ gives:

$$\begin{aligned} G_1 &= 1.644 \\ G_2 &= 0.514 \\ G_3 &= 0.109 \end{aligned}$$

Since the results from maximally flat approach and ABCD matrix approach are exactly the same, this proves that the results from section 2.1 gives maximally flat response.

2.3 Jaumann Absorbers with Varying Separation

In section 2.1 we have already discussed on deriving the mathematical expression of Jaumann absorber problem which we will use in Genetic Algorithm (GA) in chapter 3. However, we can step further into the search of optimal solution by generalizing the length of transmission line separation between the absorbers (which was previously assumed to be $\lambda/4$).

On the first thought, varying the length of transmission line simply means assigning different θ_n to each section. However, separating the θ into θ_n while simultaneously varying θ_n to plot S_{11} would add complication in the algorithm. To avoid this problem, we introduce a new variable X_n defined as:

$$\theta_n = X_n \theta \tag{2.37}$$

$$X_n = \frac{L}{\lambda/4} \tag{2.38}$$

Where L is the length of the separation between absorbers. We define X_n and L in this way for the sake of consistency with our preceding calculations. We maintain $X_n = 1$ for the quarter wavelength sections as the default and let the rest scale with respect to quarter wavelength. This idea allows us to simply add X_n in front of every θ_n inside the trigonometric functions to control the varying separation length.

In the interest of maintaining the brevity of this chapter, we place the full expression of the reflection coefficient Γ for varying separation case in Appendix A.4 along with the MATLAB code.

3

Genetic Algorithms and Their Application to Jaumann Wave Absorber Problems

3.1 Motivation

Although maximally flat solutions yield minimum reflection at the center frequency, it does not necessarily yield the most desired result. From an engineering perspective, once a certain reflection threshold is satisfied then it is a good enough solution. It would make sense then, to search for other possible solutions which simultaneously satisfy the reflection threshold and yield the widest bandwidth possible. Our effort in deriving the full mathematical expression, however, would prove to be useful since it enables us to automate extensive searches for other solutions. In doing so, we first try a simple cut-and-try algorithm which flowchart is shown in Fig. 3.1. The search space mentioned in the chart consists of equally-spaced 20 points centered around the maximally flat solution. The lower and upper limit of the search, however, are manually set and changed with each iteration. We repeat the search until the results stay fixed within 4 decimal points (default numerical precision in Matlab).

This algorithm is the result of collaboration with William Kim, a junior at the time in Pratt School of Engineering, who worked for a year as research assistant under supervision of Wiwi Samsul. The algorithm, while successfully discovers better solutions, but also has limitations in scope of how far it can go. To address this issue, we develop Genetic Algorithm to further extend the searching quality and flexibility.

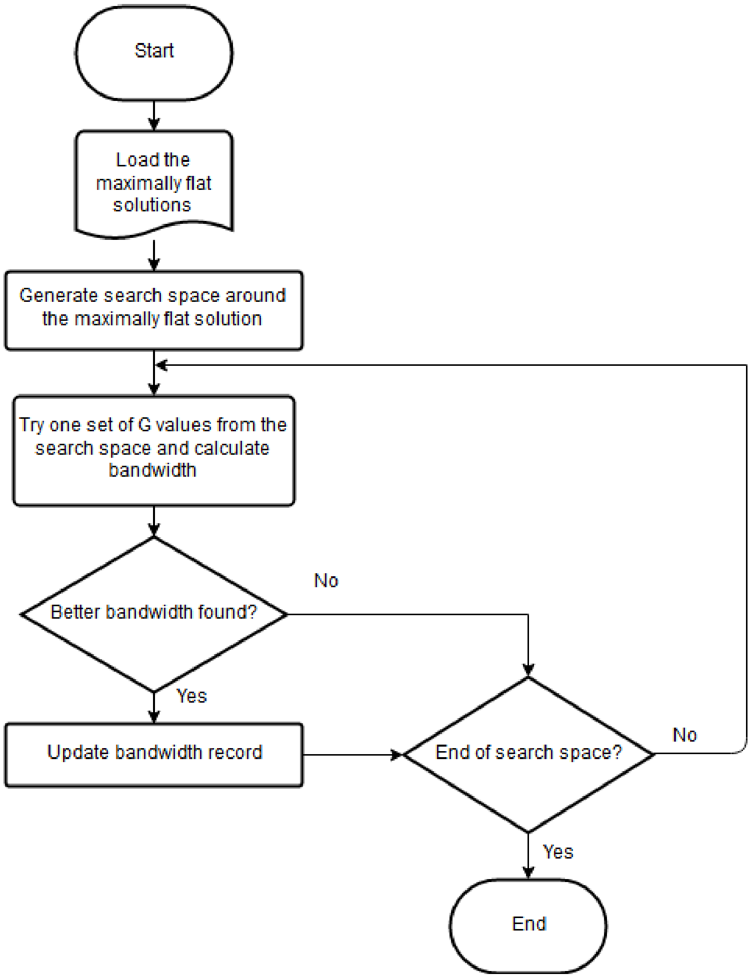

FIGURE 3.1: Cut-and-try algorithm flowchart

3.2 Components of Genetic Algorithm

Genetic algorithm (GA) is an optimization algorithm which mimics the way natural selection process sorts out individuals who survive or do not survive by their fitness

[18, 19]. Through iterations of natural selection, we hope to find individuals which genes produces a desired fitness.

In this work, the key components of GA are:

- Population and individual
- Each individual's genes
- Fitness criteria and selection threshold
- Probability of gene mutation

While the key processes include:

- Generating initial population
- Evaluating each individual's fitness and selecting survivors
- Mixing of survivor's genes to produce new generation
- Introducing mutation into the population's genes

In the following sections we describe the translation of GA key components and the implementation of each components on solving Jaumann wave absorber problem.

3.3 Translation of GA components into Jaumann Wave Absorber problem

3.3.1 Population

We define the population as the set of all possible solutions. In addition, we present two types of solutions: solving only for the G values and solving for both G and X values. An individual consists of either a set of G values $(G_1, G_2, ..., G_n)$ or G and X values $(G_1, G_2, ..., G_n, X_1, X_2, ..., X_n)$. Each G or X value corresponds to one gene. In other words, each individual in n layer case has n genes (if we only solve for G) or $2n$ genes (if we solve for both G and X).

3.3.2 Fitness criteria

The fitness of each individual is defined as the relative bandwidth of the resulting Γ plot of its genes below a predetermined dB level (either -10 or -20 dB):

$$\text{Fitness} = \frac{f_2 - f_1}{f_0} \tag{3.1}$$

f_0 = center frequency

f_1 = frequency when the first time Γ plot dips below the dB level

f_2 = frequency when the first time Γ plot rises above the dB level after f_1 has been recorded

We only include the first time Γ plot dips below and rises above the dB level in order to ensure every individual which Γ plot oscillates at the dB level has low fitness and gets eliminated from gene pool.

For the case of fixed quarter wavelength separation case, the Γ curve is calculated using Eq. 2.14, 2.17, 2.20, A.14, A.15, and A.16 (with Z_0 and Z_S are set equal to 1) for 1, 2, 3, 4, 5, and 6 layers case respectively.

For the case of varying separation case, the Γ curve is calculated using equations presented in Appendix A.4.

3.3.3 Initial population

The initial population is generated with uniform random distribution between value 0 to 2. The limit was set as such because our previous maximally flat result shown in Table 2.1 provides reasonable ground to expect the optimal values for 1 to 6 layers case are between 0 and 2.

3.3.4 Survivor selection

After each individual's fitness is calculated, the population is then sorted in descending order. Then we select the better half of the population as survivors. This threshold is arbitrarily chosen.

3.3.5 Spawning next generation

We randomly choose 2 individuals from the survivor pool as the parents and randomly select the genes from each parent to spawn a child. The probability of selecting each gene is 50% regardless of the parents' fitness. The gene selection process also follows the proper sequence. For example, suppose parent A has gene $G_1a, G_2a, G_3a, ..., G_na$, parent B has gene $G_1b, G_2b, G_3b, ..., G_nb$, and the resulting child has gene $G_1c, G_2c, G_3c, ..., G_nc$. Then gene G_kc is randomly picked between G_ka and G_kb for all $k \in [1, n]$.

3.3.6 Mutation process

The mutation process happens after the next generation is spawned. The rules of mutation are:

- Mutation happens with equal probability (p) for each gene and independent of each event. We set probability of mutation $p = 0.2$.
- Value of mutation (v) is determined by random uniform distribution with predetermined boundary (b). In mathematical notation: $|v| \leq b$. We set $b = 0.5$.
- If mutation occurs, the value of mutation is added algebraically into the gene. Otherwise, $v = 0$.

The mutation process may cause a gene to turn into a negative value. We do not check for negative genes during the search process. However, if the algorithm produces final result with negative G or X values then that result is discarded.

3.4 Genetic algorithm flowchart

Figure 3.2 shows the algorithm flowchart with additional specifications as follows:

- Initial population is arbitrarily set to be 100.
- Number of genes is equal to number of variables being searched.
- Number of generation limit is arbitrarily set to be 200. This condition prevents infinite loop.
- The saved population data comprises of the genes, relative bandwidth, bandwidth start, bandwidth end, theta limit, and the generation number when the best solution is found.
- The fitness is evaluated at the specified dB level (either -10 dB or -20 dB in this work, but the algorithm can work for an arbitrary dB level).

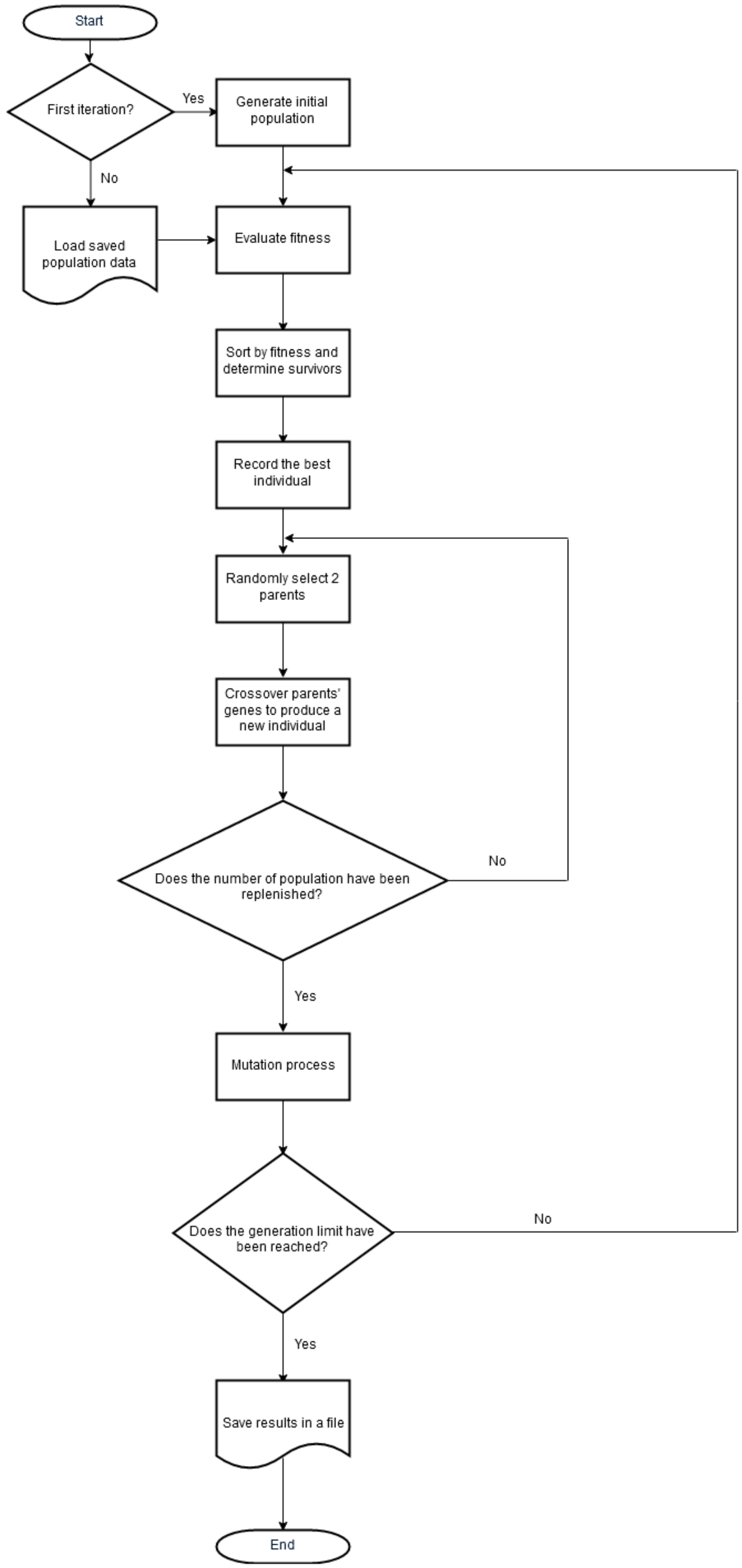

FIGURE 3.2: Genetic algorithm flowchart

4

Results and Analysis

We structure this chapter by first laying out all the results then comparing and analyzing them in later in section 4.3.

4.1 Cut-and-try Algorithm

Table 4.1 presents the result from cut-and-try algorithm up to 5 layers case. The RBW increase is compared to the maximally flat solution. The bandwidth increase peaks at 3 layers case and then declines. We never managed to find the solutions for 6 layers case with this algorithm because of lack of computational power. The computers we use keep crashing in the middle of the searches. We think the main problem in this way of searching is because the algorithm has to keep track of every permutation of previous loops. Therefore, each additional layer increases the search space exponentially. On the other hand, GA does not encounter the same problem because older genes are erased and replaced with new genes on each generation.

For the sake of brevity, we only show a sample plot of reflection coefficient from 3 layers case. All other plots are available on Appendix A.8.

Table 4.1: Summary of the G values from cut-and-try algorithm up to five sections.

# of sections	G_1	G_2	G_3	G_4	G_5	RBW	RBW increase (%)
1	1.000	-	-	-	-	0.253	0
2	1.253	0.423	-	-	-	0.815	21.034
3	1.368	0.514	0.316	-	-	1.214	31.106
4	1.500	0.605	0.290	0.290	-	1.402	28.691
5	1.211	0.653	0.321	0.226	0.226	1.523	26.589

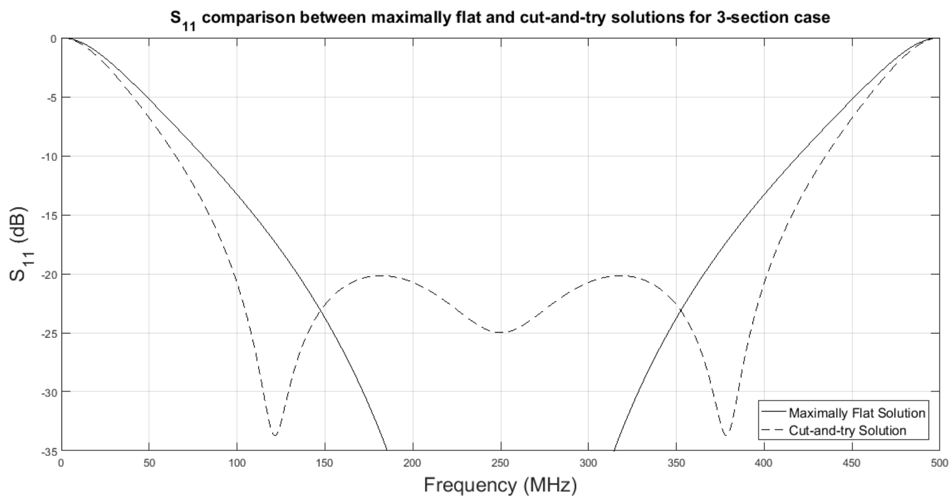

FIGURE 4.1: Comparison of maximally flat and cut-and-try solutions for 3 section case.

4.2 Genetic Algorithm

We separate this section into two subsections: first subsection is comprised of results in the standard format like the preceding section and second subsection is comprised of results in expanded format while introducing a new variable we call "theta limit" t (where $t = k\ f_0$, or the upper frequency limit of search range). The reason we introduce this new variable is because we realize that GA is able to efficiently search for solutions in any multiples of bandwidth range with respect to f_0. In other words, for any given f_0, GA can search for any solution which produce any relative bandwidth (RBW) we want by specifying the theta limit.

4.2.1 Standard Results

The standard results of GA solution is shown in Table 4.2. The '% increase' column shows the RBW increase compared to the cut-and-try algorithm. The two right most columns are the relative bandwidth start and end. In addition, we provide the start and end of bandwidth to highlight that the standard solutions only searches for bandwidth around f_0 while the expanded search (described below) can search for any bandwidth even outside of f_0.

Table 4.2: Summary of standard GA solutions (S_{11} cut-off: -20 dB).

#	G_1	G_2	G_3	G_4	G_5	G_6	RBW	% increase	BW Start	BW End
1	1.016	-	-	-	-	-	0.254	0.416	0.873	1.127
2	1.256	0.426	-	-	-	-	0.872	6.963	0.565	1.436
3	1.295	0.532	0.305	-	-	-	1.218	0.284	0.391	1.609
4	1.365	0.593	0.302	0.274	-	-	1.408	0.432	0.296	1.704
5	1.250	0.637	0.347	0.217	0.238	-	1.525	0.098	0.238	1.763
6	1.221	0.610	0.399	0.256	0.164	0.222	1.604	N/A	0.198	1.802

4.2.2 Expanded Results

We take results from 3 layers case as an example of the expanded solution and show them in Figure 4.2 and Table 4.3 below. In the last column, "T" refers to "Theta limit". As usual, we put the rest of the results in Appendix for the sake of brevity. Note that the ability of GA to search for solutions outside of f_0 gives us a huge flexibility in designing a system with any bandwidth in any frequency range.

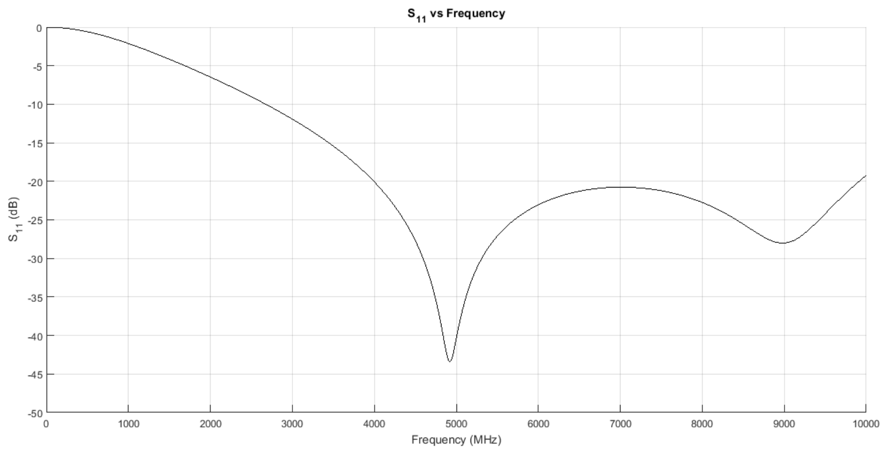

FIGURE 4.2: S_{11} plot of GA expanded result for 3 sections case (theta limit = 20).

Table 4.3: Summary of expanded GA solutions for 3 sections case (S_{11} cut-off: -20 dB).

G_1	G_2	G_3	X_1	X_2	X_3	RBW	BW Start	BW End	T
1.377	0.533	0.321	0.784	0.791	0.813	1.506	0.495	2.000	1
1.365	0.508	0.313	0.402	0.405	0.400	3.015	0.976	3.991	2
1.410	0.576	0.317	0.253	0.265	0.262	4.467	1.533	6.000	3
1.230	0.541	0.273	0.201	0.198	0.192	5.995	2.005	8.000	4
1.376	0.573	0.273	0.155	0.151	0.164	7.359	2.641	10.000	5
1.385	0.480	0.300	0.131	0.129	0.125	8.907	3.093	12.000	6
1.522	0.489	0.261	0.107	0.114	0.109	10.199	3.802	14.000	7
1.462	0.507	0.309	0.102	0.100	0.095	11.935	4.065	16.000	8
1.181	0.518	0.252	0.088	0.087	0.087	13.444	4.556	18.000	9
1.263	0.518	0.284	0.077	0.076	0.076	14.836	5.164	20.000	10
1.280	0.496	0.273	0.065	0.066	0.064	15.906	6.095	22.000	11
1.270	0.499	0.274	0.068	0.067	0.064	18.030	5.970	24.000	12
1.281	0.523	0.273	0.061	0.062	0.063	19.485	6.476	25.961	13
1.428	0.561	0.334	0.057	0.057	0.057	20.942	7.059	28.000	14
1.313	0.513	0.261	0.051	0.052	0.057	22.272	7.690	29.962	15
1.445	0.497	0.266	0.049	0.047	0.053	23.475	8.525	32.000	16
1.357	0.487	0.270	0.045	0.047	0.043	25.058	8.942	34.000	17
1.410	0.548	0.341	0.042	0.042	0.043	26.718	9.283	36.000	18
3.654	0.964	0.380	0.014	0.034	0.038	23.548	14.452	38.000	19
0.541	1.229	0.417	0.007	0.028	0.036	23.780	16.006	39.786	20

4.3 Results Comparison and Analysis

4.3.1 Gain-Bandwidth Product

Gain-bandwidth product (GBWP) is the product of gain and bandwidth at which the gain is measured. In this work, we use this attribute to compare our result with previous works particularly from ref. [20] and [13]. Since our goal is to produce bandwidth as wide as possible at the desired gain, the closer the plot to upper left corner the better the result. All calculations of GBWP in this section is performed at $f_0 = 250$ MHz.

First, we show the comparison of maximally flat solutions from 1 to 6 layers in Fig. 4.3. As the number of layers increases, the the plot moves closer to the upper left corner which indicates a bigger bandwidth for the same range of frequency.

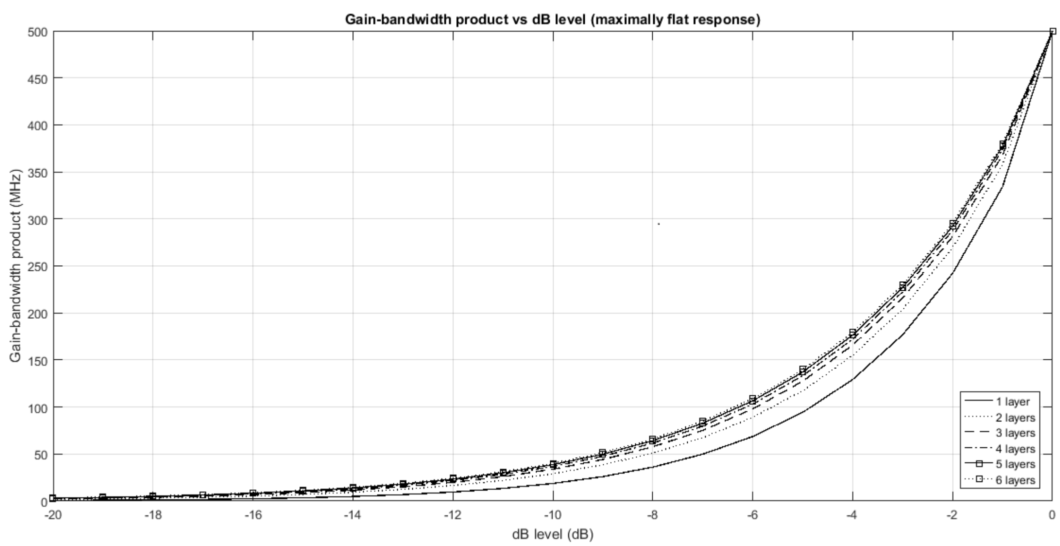

FIGURE 4.3: Gain-bandwidth product comparison between maximally flat solutions.

Next, Fig. 4.4 shows the comparison between maximally flat solution with GA expanded result (theta limit set to 1) at 3-layer case.

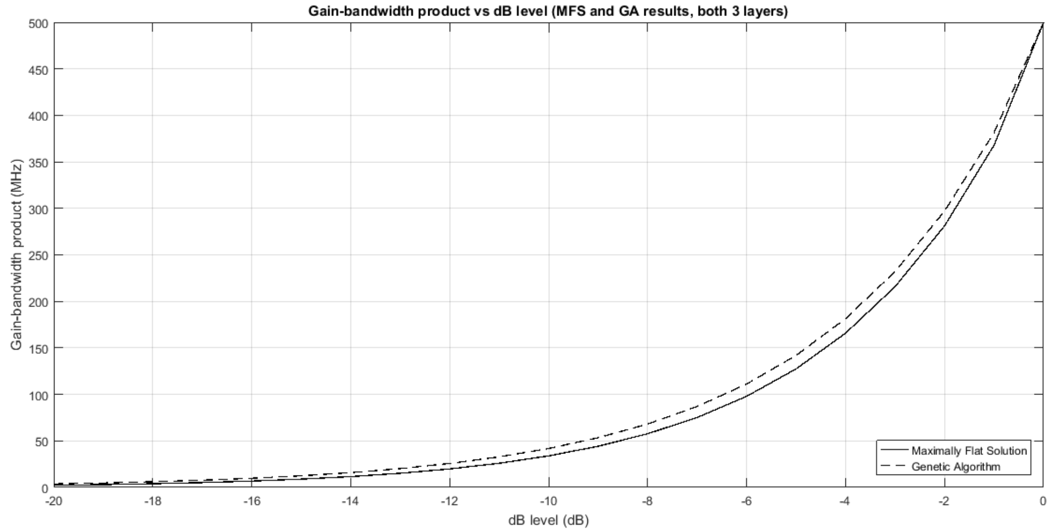

FIGURE 4.4: Gain-bandwidth product comparison between 3 layer maximally flat solution and GA.

Finally, Fig. 4.5 shows the sample comparison of GBWP between our result (GA expanded result and theta limit set to 1) and results from ref. [20] (Kraus-Fleisch in the plot legend) and [13] (Nortier in the plot legend). All comparisons are performed at 3-layer case.

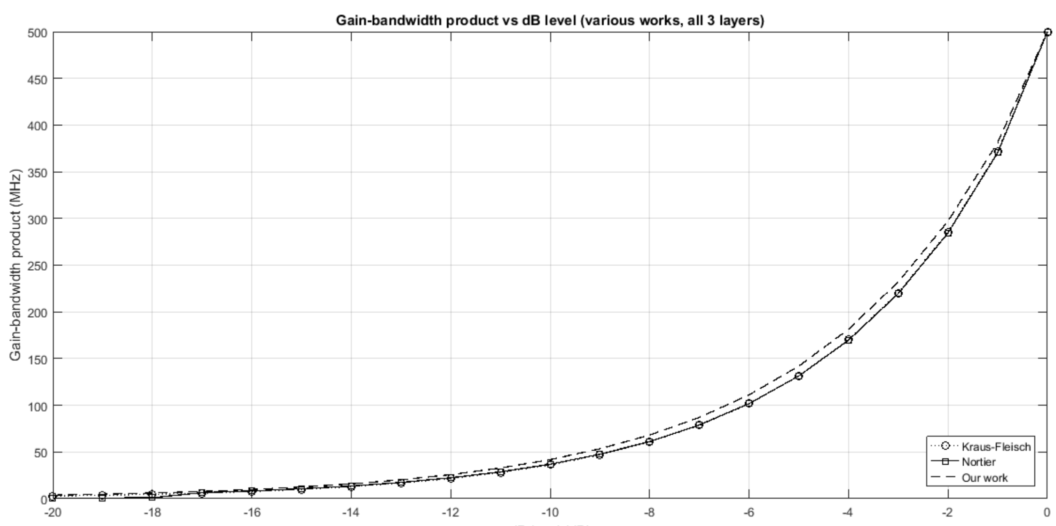

FIGURE 4.5: Gain-bandwidth product comparison at 3 layers between our work (GA), ref. [20], and ref. [13].

4.3.2 Comparison with other works.

Other works on Jaumann absorbers can be found in ref [21, 17, 15, 13, 12, 22, 23, 24, 25, 26, 27, 28] while optimization of radar absorbers with genetic algorithm includes ref [24, 29, 30, 31, 32, 26, 33]. However, only ref [24, 31, 26] involve both GA and Jaumann absorbers so we will compare our work with the latter three works.

Table 4.4 lists our result with Chambers et al [24]'s results for one, two, and three layers case. "NBW" column stands for "Normalized Bandwidth", which is NBW = $2\frac{f_H - f_L}{f_H + f_L}$. The fact that both our results has the same NBW suggests that they are the maximum NBW that can be achieved in the respective number of layers. However, Chambers et al's GA still uses the default quarter wavelength spacings while our GA varies the spacings and finds solutions which give shorter design. A shorter design is considered better to save space and materials.

Both ref [31] and [26] incorporate spacing variation into their GA. However, result from ref [31] achieves 95% absorption at maximum while all our results achieve 99% absorption (-20 dB) which is usually the standard absorption level. Furthermore, the way our GA is implemented makes the algorithm flexible and can be set to search for an even higher absorption level. Finally, ref [26] maintains to achieve absorption level of -35 dB with total thickness of 2.10 cm (or 7.10 λ as indicated by their classic Jaumann comparison) in 5 layers case. We set our GA to search for the same absorption level and produce the following gene: [0.46, 0.89, 1.81, 1.24, 0.40, 2.50, 1.69, 1.67, 1.64, 1.55]. The first five genes are the G values while the latter five genes are X values which is a multiple of quarter wavelength and gives us a total length of 2.26 λ (or 68% shorter).

Finally, we also compare our result with ref [34] in more detail. Two takeaways from this reference are that any results in normal incidence perform best compared to any other angle, and any result that is specifically optimized in specific incident angle perform best when subjected to normal angle. Thus, we only need to compare our results in normal incidence scenario. While their work include spacing variation, all of their results state the spacing with only a single number which must mean all of the spacings have equal length and they can not mean the total length because their result would have become absurdly improbable while taking it as spacing between each layers yields results which makes sense. We will compare our result with their best result at 4-layers case as shown in Table 4.5. Since their report only provides figures instead of numerical value, we have to approximate (generously) their NBW based on their reflection coefficient plots. Their best 4-layers result has spacing of 7.15 mm each with $\epsilon_r = 1.1$. Assuming $f_0 = 10GHz$ (their center frequency plot), this results in a thickness of 1.0006 λ with NBW (generously) approximated at 1.40. In contrast, our best 4-layers result is [2.40, 0.90, 0.44, 0.28, 0.40, 0.73, 0.77, 0.77] with total thickness of 0.67 λ. It appears varying the spacings while limiting them all to be equal would yield results that are not far from the default quarter-wavelength ones. In contrast, our GA is able to vary all of them independently of each other which results in much bigger search space.

In conclusion, our work produces results which are:

- 44%/28%/20% shorter in design length than ref [24],
- at least 4% higher absorption level than ref [31],
- 68% shorter in design length than ref [26], and
- 33% shorter in design length than ref [34].

Table 4.4: Comparison with ref [24].

	Ref [24]		Our work		% Shorter
	NBW	Length (λ)	NBW	Length (λ)	
1 layer	0.25	0.25	0.25	0.14	44
2 layers	0.87	0.50	0.87	0.36	28
3 layers	1.22	0.75	1.21	0.60	20

Table 4.5: Comparison with ref [34].

	Ref [34]		Our work		% Shorter
	NBW	Length (λ)	NBW	Length (λ)	
4 layers	1.40	1.00	1.24	0.67	33

5

Measurements

5.1 Realization in transmission line

We experimentally verify our 3-layer result for the Maximally Flat Solution (MFS) in transmission line setting. First we build a 50 Ω transmission line as shown in Fig. 5.1, which —for the sake of brevity—calculation details and dimensions are provided in Appendix A.10. Then we pick the available off-the-shelf resistors which values are the closest to the theoretical solutions as listed in Table 5.1 and build the Jaumann absorber circuit. We first perform the experiment using lumped resistors, but the result is distorted such that the S_{11} plot fluctuates around the -20 dB line (dotted line in Fig. 5.2). We hypothesize that stray inductances is the main culprit for the distorted result, so we simulate the theoretical solution with added 8 nH to all the resistors and we obtain good agreement with the experiment (Fig. 5.2). In addition, we also redo the experiment by replacing the lumped elements with chip resistors in order to obtain an improved result as shown in Fig. 5.3.

In conclusion, we have experimentally proven a sample solution of Jaumann wave absorber problem. Any difficulty in real life implementation mainly revolves around producing precise resistive value either in transmission line model or wave absorber model.

Table 5.1: List of resistor values used in 3 layers MFS experiments.

	$Z_1(\Omega)$	$Z_2(\Omega)$	$Z_3(\Omega)$	RBW	
				at -10 dB	at -20 dB
Theoretical	30.414	97.220	458.295	1.356	0.926
Lumped	30	100	470	1.523	N/A
Chip	30.1	100	475	1.625	0.975

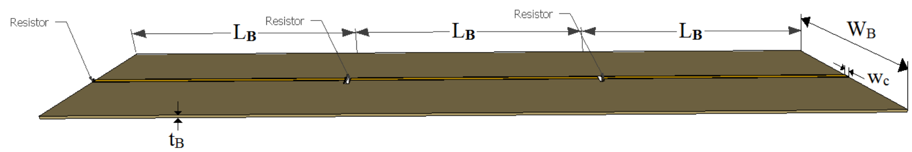

FIGURE 5.1: Experimental set up for three sections Jaumann wave absorber in transmission line.

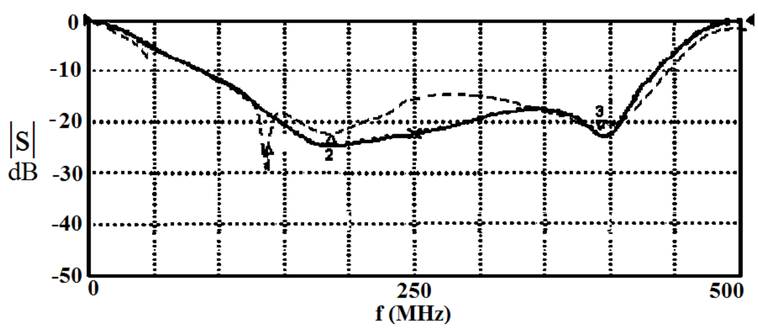

FIGURE 5.2: Comparison between simulation with added stray inductances (solid line) and measured S_{11} (dotted line) for 3 section case.

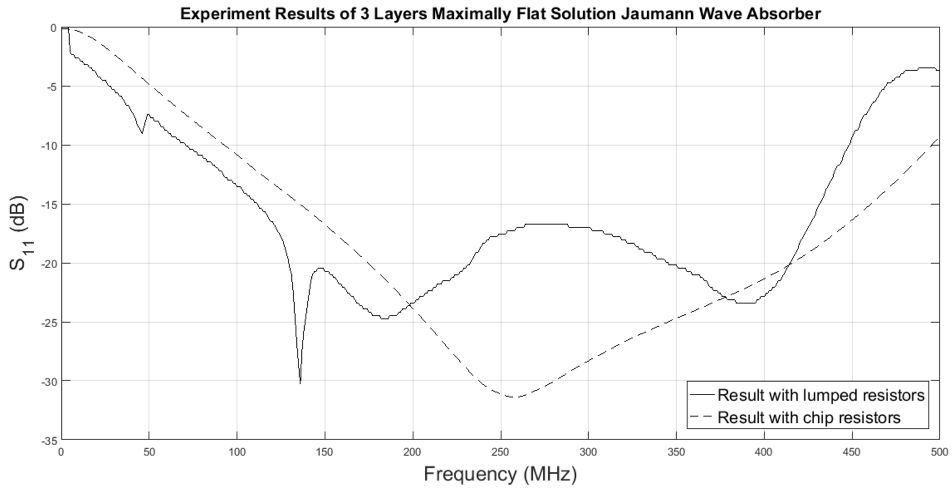

FIGURE 5.3: S_{11} comparison between using lumped vs chip resistors.

5.2 Measurement of ambient RF energy

As part of examining the viability of Jaumann wave absorber on energy harvesting technology, we measure the availability of ambient energy that can be harvested especially in RF frequency. We perform this measurement of ambient RF power in four Duke buildings: CIEMAS, Teer, Physics, and French. Energy detection is performed with two dipoles, bowtie, and spiral antennas connected to spectrum analyzer. We record the spectral energy density in various locations throughout the buildings. We categorize the locations as: corners, halls, rooms, near router/repeater, and near window. Note that some locations fall into more than one category (e.g: corners which have routers nearby). Experiment details and results are summarized in the following subsections.

5.2.1 Locations Stats

Table 5.2 lists the summary of measurement locations. More details on these locations is provided in Appendix A.11.

Table 5.2: Summary of measurement locations.

Building	Category					# of Data Points	# of Locations
	Corner	Hall	Room	Near Router	Near Window		
CIEMAS	3	2	2	4	2	13	8
Teer	2	0	3	4	2	11	9
Physics	4	6	1	10	1	22	12
French	7	8	3	7	3	28	18

5.2.2 Antennas Profiles

Fig. 5.4 shows the S_{11} profile of each antenna used in the experiment.

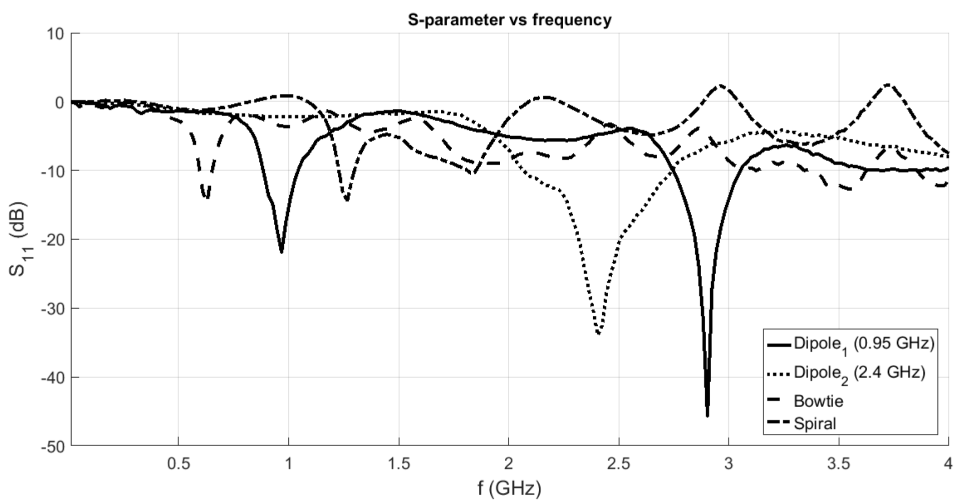

FIGURE 5.4: S_{11} graphs of antennas used in experiment.

5.2.3 Spectrum Analyzer

We measure the ambient spectrum with Spectrum Analyzer from Agilent Model E4446A (serial number MY46180443) which settings of the analyzer is detailed in Table 5.3. A sample of spectrum measurement taken at CIEMAS 3rd floor skyway is shown in Fig. 5.5. We pick this spectrum to show because it has the highest detected peak (at -21.22 dBm around 0.9 GHz) compared all other measurements even surpassing the ones taken near router/signal repeater. The spikes around 0.9

GHz indicate presence of cellphone signal while the spikes around 2.4 GHz indicate presence of WiFi. We hypothesize this is because of the skyway lacks concrete walls which is the main culprit of attenuation of cellphone signals.

Table 5.3: Setting details of the spectrum analyzer.

Min. Frequency	0.01 GHz
Cent. Frequency	2.01 GHz
Max. Frequency	4.01 GHz
Resolution bandwidth	3 MHz
Video bandwidth	3 MHz
Sweep time	3 s
Num of data points	601
Ref level	0 dBm
Data format	.csv

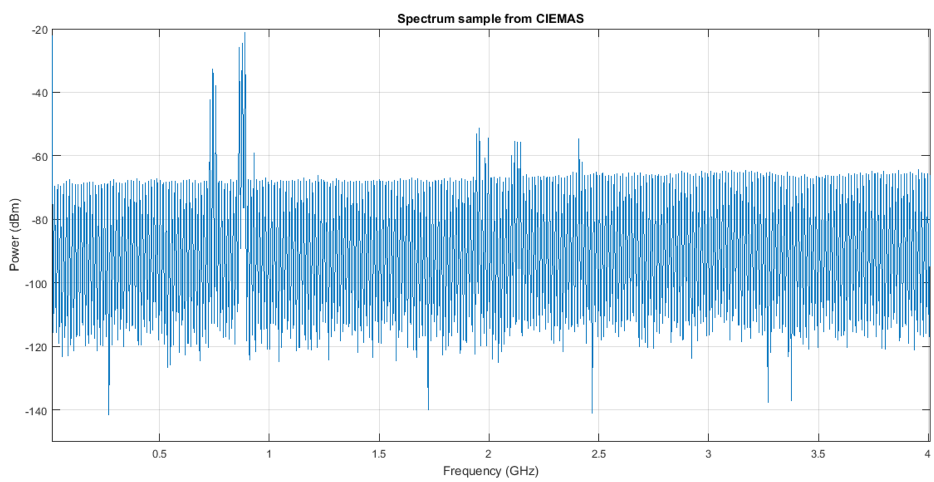

FIGURE 5.5: Sample of a spectrum taken in CIEMAS 3rd floor skyway.

5.2.4 Ambient Energy Calculations

Our goal is to obtain a figure of power-to-area of ambient field which Eq.5.1 and 5.2 show the general calculation process. However, the details of the calculation is not as straightforward as it seems. Referring to table 5.3, we notice that while the resolution bandwidth is 3 MHz, the spectrum analyzer records the measurement in 601 data points over the span of 4 GHz which means there is a 6.6 MHz gap between data points. In order to counter this mismatch, we first interpolate the spectrum to match the resolution bandwidth of 3 MHz gap between data points.

$$P_{amb.} = \frac{P_{meas.}}{1 - 10^{S_{11}/10}} \tag{5.1}$$

$$\frac{P_{amb.}}{\text{unit area}} = \frac{P_{amb.}}{A_e} \tag{5.2}$$

where:

$$\begin{aligned} P_{amb.} &= \text{ambient power (W)} \\ P_{meas.} &= \text{measured power (W)} \\ S_{11} &= \text{reflection coefficient (dB)} \\ A_e &= \text{effective area of antenna (cm}^2\text{)} \end{aligned}$$

Furthermore, we need to obtain the effective area of the spiral antenna which does not have a straightforward formula to calculate. Hence, we opt to simulate the spiral antenna in CST and obtain its maximum gain then calculate the effective area using Eq. 5.3. Table 5.4 shows the parameters of the spiral antenna. Inner diameter refers to the spacing between spirals or in other words the difference of diameter between a spiral and its subsequent spiral. Outer diameter refers to the total diameter of the outermost spiral.

$$A_e = \frac{\lambda^2}{4\pi} G \tag{5.3}$$

Table 5.4: Parameters of spiral antenna.

Name	Description	Value
D_i	Inner diameter	1.5 mm
D_o	Outer diameter	143 mm
N	Number of turns	96
h	Handedness	Right handed

Putting all of them together, we obtain the average power spectrum across all buildings and all location categories as shown in Fig. 5.6. We also integrate the spectrum to obtain the total power between 0-4 GHz as 0.9263 μW/cm^2. For comparison, other studies conducted in Belgium [35], UK [36, 37], Netherlands [38], and Japan [39] on ambient RF spectrum maintain that the energy level varies between 0.3 μW/cm^2 - 3.6 mW/cm^2 where the lower and upper limit were obtained from measurements in rural and urban area respectively. Those other studies are implied to be conducted in outdoor environment while our measurement is in indoor setting.

Detailed breakdowns of average ambient energy by location categories and buildings are shown in Table 5.5. Surprisingly, 'corner' sites has the highest concentration of RF energy, almost 5 times higher compared to 'near router' sites. In addition, 'hall' also has higher energy than 'near router'. Combined with the fact that 'room' has the lowest energy, this phenomenon suggests that ambient energy tend to concentrate near walls (despite the definition of 'room' as space contained within 4 walls, none of the rooms can be considered claustrophobic; width of the rooms tend to be wider than halls). It is also worth noting that 'near window' has higher energy concentration than 'near router'. We suspect it is because of contribution from cellphone signals from base stations which takes attenuation only from a window compared to multiple walls in 'halls' or 'rooms'. Finally, Table 5.6 shows the average by buildings. Since category of 'corner' holds the most concentrated energy, we take a closer look at the proportion of corners in each building. CIEMAS has 2 out of 11 (18.18%), Teer has 2 out of 11 (18.18%), Physics has 4 out of 21 (19.05%), and French has 7 out of 26 (26.92%). Physics and French have greater proportion of corners compared to Teer yet Teer still holds a higher ambient RF energy concentration. Therefore, it is not because of the corners that makes Teer and CIEMAS have higher concentration than Physics and French. We hypothesize the reason is due to the number of students around. Both Teer and CIEMAS have dedicated sites for students to work and the student's laptops and cellphones increase the ambient RF energy concentration.

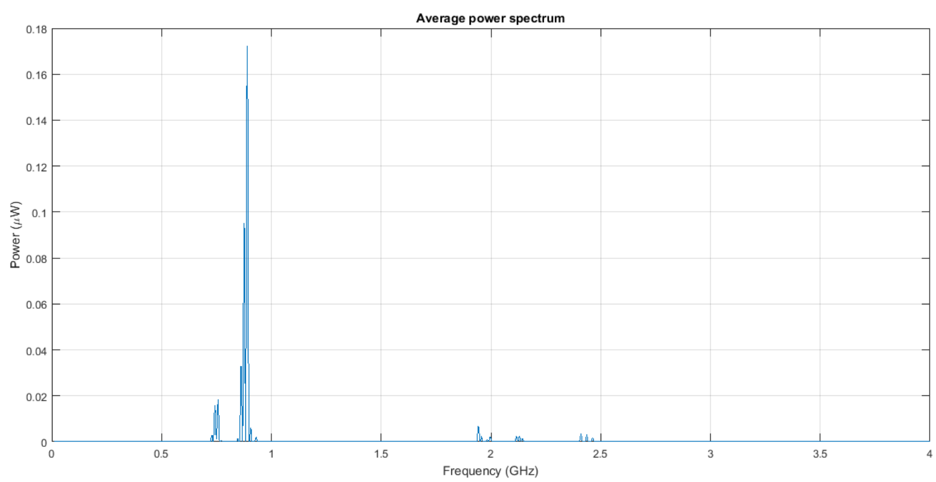

FIGURE 5.6: Average power spectrum across all buildings and all location categories.

Table 5.5: Summary of average ambient energy by location category.

Category	Average ambient energy (μW/cm^2)
Corner	5.2031
Hall	1.1915
Room	0.2877
Near router	1.1120
Near window	1.4031

Table 5.6: Summary of average ambient energy by buildings.

Building	Average ambient energy (μW/cm^2)
CIEMAS	1.2234
Teer	1.6808
Physics	0.3576
French	0.8480

6

Conclusions and Future Work

6.1 Conclusions

We have briefly explored the origin of Jaumann wave absorber and provided a solid mathematical framework to solve for its maximally flat solution for any number of layers while past attempts to do so were merely blind trial-and-error [8], stopped at 2 layers [17], impractical to go beyond 3 layers [14], did not produce a reliable absorption spectrum [13], did not optimize for absorber spacings [21, 17, 15, 13, 12, 22, 23, 24, 25, 27, 28], or still has room for improvement [21, 31, 26]. In comparison to the last three references, we surmise that our GA gives better results because the way we design our GA by starting from the framework of maximally flat solution and automate the search for other solutions seem to provide a greater flexibility in searching with any absorption level, varying both resistances and spacings, and finding a more, if not the most, optimum solution compared to other works.

In order to support the feasibility of building a wave absorber system, we perform measurements in four Duke buildings and compare our result with other studies. We conclude that the average ambient RF energy in indoor environment ranges between

0.2877 - 5.2031 μW/cm^2. We also conclude that the best spots to install wave absorbing device would be in corners since it is the category in which the highest ambient RF energy concentrates.

6.2 Future Work

This work sees possibility of usage for constructing energy harvesting devices through absorbing electromagnetic wave. By implementing the right values of resistances and their spacings, one can engineer the absorption level and bandwidth to suit one's needs. However, given the measurement results of ambient RF energy in buildings, such devices must operate in μW level unless we artificially pump the ambient RF energy, effectively turning the whole system into wireless charging. Another possible usage comes in radar cloaking devices. By engineering the wave absorber to absorb in targeted bandwidth with certain absorption level, one could render radars ineffective in detecting the cloaked item. This idea offers another approach to the existing stealth systems.

Appendix A

Appendix List

A.1 System of equations to obtain solutions for four, five, and six sections case

A.1.1 Four sections

The ABCD matrix for four sections case is:

$$
\begin{aligned}
A &= \frac{1}{8}\Big(-4G_2G_3Z_0^2\cos 2\theta + 2(4 + G_2G_3Z_0^2 + G_1(G_2+G_3)Z_0^2)\cos 4\theta + \\
&\quad Z_0(-2((-G_2)G_3 + G_1(G_2+G_3))Z_0 - 2j(2G_3 + G_1(-2 + G_2G_3Z_0^2))\sin 2\theta + \\
&\quad j(4(G_2+G_3) + G_1(4 + G_2G_3Z_0^2))\sin 4\theta)\Big) \\
B &= Z_0\sin\theta\Big(4j\cos^3\theta - (3G_1 + 4G_2 + 3G_3)Z_0\cos^2\theta\sin\theta - \\
&\quad 2j(2 + G_2G_3Z_0^2 + G_1(G_2+G_3)Z_0^2)\cos\theta\sin^2\theta + Z_0(G_1 + G_3 + G_1G_2G_3Z_0^2)\sin^3\theta\Big) \\
C &= \frac{1}{8Z_0}\Big(4Z_0(G_1 + G_3 - G_2G_3G_4Z_0^2)\cos 2\theta + Z_0(4G_2 + 4G_3 + 8G_4 + 2G_2G_3G_4Z_0^2 + \\
&\quad G_1(4 + 2G_3G_4Z_0^2 + G_2(G_3 + 2G_4)Z_0^2))\cos 4\theta - j((-j)(2G_1G_3G_4Z_0^3 + G_2Z_0(G_1(G_3 + \\
&\quad 2G_4)Z_0^2 - 2(2 + G_3G_4Z_0^2))) + 2Z_0^2(2G_3G_4 + G_1(-2G_4 + G_3(-2 + G_2G_4Z_0^2)))\sin 2\theta - \\
&\quad (8 + 4G_3G_4Z_0^2 + 2G_2(G_3 + 2G_4)Z_0^2 + G_1Z_0^2(2G_2 + 2G_3 + 4G_4 + G_2G_3G_4Z_0^2))\sin 4\theta)\Big) \\
D &= \cos^4\theta + j(G_1 + 2G_2 + 3G_3 + 4G_4)Z_0\cos^3\theta\sin\theta - (6 + 2G_2G_3Z_0^2 + 3G_3G_4Z_0^2 + \\
&\quad G_1(G_2 + 2G_3 + 3G_4)Z_0^2)\cos^2\theta\sin^2\theta - jZ_0(G_3 + 4G_4 + 2G_2(1 + G_3G_4Z_0^2) + \\
&\quad G_1(3 + 2G_3G_4Z_0^2 + G_2(G_3 + 2G_4)Z_0^2))\cos\theta\sin^3\theta + \\
&\quad (1 + G_3G_4Z_0^2 + G_1Z_0^2(G_2 + G_4 + G_2G_3G_4Z_0^2))\sin^4\theta - G_2G_4Z_0^2\sin^2 2\theta
\end{aligned}
\tag{A.1}
$$

The load Z_T with short circuit load is:

$$
Z_T = \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D} \tag{A.2}
$$

Setting $Z_0 = 1$ and $Z_T = 1$ and cross multiply, we obtain:

$$\begin{aligned} j\cos^3\theta(-4 + G_1 + 2G_2 + 3G_3 + 4G_4 - j\cot\theta) - \\ (6 + 3G_3(-1 + G_4) + 2G_2(-2 + G_3 + 2G_4) + G_1(-3 + G_2 + 2G_3 + 3G_4))\cos^2\theta\sin\theta - \\ j(-4 + G_3 + 2G_2(1 + G_3(-1 + G_4)) + 4G_4 + \\ G_1(3 + 2G_3(-1 + G_4) + G_2(-2 + G_3 + 2G_4)))\cos\theta\sin^2\theta + \\ (1 + G_3(-1 + G_4) + G_1(-1 + G_2(1 + G_3(-1 + G_4)) + G_4))\sin^3\theta = 0 \end{aligned} \tag{A.3}$$

Therefore, the system of equations that we need to solve are:

$$\begin{cases} -4 + G_1 + 2G_2 + 3G_3 + 4G_4 = 0 \\ 6 + 3G_3(-1 + G_4) + 2G_2(-2 + G_3 + 2G_4) + G_1(-3 + G_2 + 2G_3 + 3G_4) = 0 \\ -4 + G_3 + 2G_2(1 + G_3(-1 + G_4)) + 4G_4 + G_1(3 + 2G_3(-1 + G_4) + ... \\ G_2(-2 + G_3 + 2G_4)) = 0 \\ (1 + G_3(-1 + G_4) + G_1(-1 + G_2(1 + G_3(-1 + G_4)) + G_4)) = 0 \end{cases} \tag{A.4}$$

And the solution is: $G_1 = 1.7824$, $G_2 = 0.6695$, $G_3 = 0.2337$, $G_4 = 0.0443$.

A.1.2 Five sections

The ABCD matrix for five sections case is:

$$
\begin{aligned}
A &= \cos^5\theta + j(4G_1 + 3G_2 + 2G_3 + G_4)Z_0\cos^4\theta\sin\theta - (10 + G_3G_4Z_0^2 + 2G_2(G_3 + G_4)Z_0^2 + \\
&\quad G_1(3G_2 + 4G_3 + 3G_4)Z_0^2)\cos^3\theta\sin^2\theta - jZ_0(6(G_3 + G_4) + G_2(4 + G_3G_4Z_0^2) + 2G_1(2 + \\
&\quad G_3G_4Z_0^2 + G_2(G_3 + G_4)Z_0^2))\cos^2\theta\sin^3\theta + (5 + 3G_3G_4Z_0^2 + 2G_2(G_3 + G_4)Z_0^2 + \\
&\quad G_1Z_0^2(G_2 + G_4 + G_2G_3G_4Z_0^2))\cos\theta\sin^4\theta + jZ_0(G_2 + G_4 + G_2G_3G_4Z_0^2)\sin^5\theta \\
B &= Z_0\sin\theta\Big(5j\cos^4\theta - 2(2G_1 + 3G_2 + 3G_3 + 2G_4)Z_0\cos^3\theta\sin\theta - 3j(G_3G_4 + \\
&\quad G_1(G_2 + G_4))Z_0^2\cos^2\theta\sin^2\theta + 2Z_0(G_2 + G_3 + 2G_4 + G_2G_3G_4Z_0^2 + \\
&\quad G_1(2 + G_3G_4Z_0^2 + G_2(G_3 + G_4)Z_0^2))\cos\theta\sin^3\theta + (1/2)j(2(1 + G_3G_4Z_0^2 + \\
&\quad G_1Z_0^2(G_2 + G_4 + G_2G_3G_4Z_0^2))\sin^4\theta - (5 + 2G_1G_3Z_0^2 + 2G_2(G_3 + G_4)Z_0^2)\sin 2\theta^2)\Big)
\end{aligned}
$$

$$
\begin{aligned}
C \;=\; & \frac{1}{16Z_0}\Big(2Z_0(G_2(4+G_1Z_0^2(-2G_4-2G_5+G_3G_4G_5Z_0^2))-2(G_1G_4G_5Z_0^2+G_3(-2- \\
& G_4G_5Z_0^2+G_1(G_4+2G_5)Z_0^2)))\cos\theta - Z_0(4G_2G_3G_5Z_0^2+2G_4(-4+4G_3G_5Z_0^2+ \\
& G_2(G_3+2G_5)Z_0^2)+G_1(-2(4+G_3(G_4+2G_5)Z_0^2)+ \\
& G_2Z_0^2(-2G_4+G_3(2+3G_4G_5Z_0^2))))\cos 3\theta + \\
& 8G_1Z_0\cos 5\theta+8G_2Z_0\cos 5\theta+8G_3Z_0\cos 5\theta+8G_4Z_0\cos 5\theta+16G_5Z_0\cos 5\theta+ \\
& 2G_1G_2G_3Z_0^3\cos 5\theta+2G_1G_2G_4Z_0^3\cos 5\theta+2G_1G_3G_4Z_0^3\cos 5\theta+2G_2G_3G_4Z_0^3\cos 5\theta+ \\
& 4G_1G_2G_5Z_0^3\cos 5\theta+4G_1G_3G_5Z_0^3\cos 5\theta+4G_2G_3G_5Z_0^3\cos 5\theta+4G_1G_4G_5Z_0^3\cos 5\theta+ \\
& 4G_2G_4G_5Z_0^3\cos 5\theta+4G_3G_4G_5Z_0^3\cos 5\theta+G_1G_2G_3G_4G_5Z_0^5\cos 5\theta-4jG_1G_2Z_0^2\sin\theta+ \\
& 8jG_2G_3Z_0^2\sin\theta+4jG_1G_4Z_0^2\sin\theta-4jG_3G_4Z_0^2\sin\theta+8jG_2G_5Z_0^2\sin\theta- \\
& 8jG_3G_5Z_0^2\sin\theta-2jG_1G_2G_3G_4Z_0^4\sin\theta- \\
& 4jG_1G_2G_3G_5Z_0^4\sin\theta-4jG_1G_2G_4G_5Z_0^4\sin\theta-4jG_1G_3G_4G_5Z_0^4\sin\theta+ \\
& 8jG_2G_3G_4G_5Z_0^4\sin\theta+4jG_1G_3Z_0^2\sin 3\theta-4jG_2G_3Z_0^2\sin 3\theta+ \\
& 8jG_1G_4Z_0^2\sin 3\theta+4jG_2G_4Z_0^2\sin 3\theta+8jG_1G_5Z_0^2\sin 3\theta-8jG_4G_5Z_0^2\sin 3\theta- \\
& jG_1G_2G_3G_4Z_0^4\sin 3\theta-2jG_1G_2G_3G_5Z_0^4\sin 3\theta-2jG_1G_2G_4G_5Z_0^4\sin 3\theta- \\
& 2jG_1G_3G_4G_5Z_0^4\sin 3\theta-6jG_2G_3G_4G_5Z_0^4\sin 3\theta+16j\sin 5\theta+4jG_1G_2Z_0^2\sin 5\theta+ \\
& 4jG_1G_3Z_0^2\sin 5\theta+4jG_2G_3Z_0^2\sin 5\theta+4jG_1G_4Z_0^2\sin 5\theta+ \\
& 4jG_2G_4Z_0^2\sin 5\theta+4jG_3G_4Z_0^2\sin 5\theta+8jG_1G_5Z_0^2\sin 5\theta+ \\
& 8jG_2G_5Z_0^2\sin 5\theta+8jG_3G_5Z_0^2\sin 5\theta+8jG_4G_5Z_0^2\sin 5\theta+jG_1G_2G_3G_4Z_0^4\sin 5\theta+ \\
& 2jG_1G_2G_3G_5Z_0^4\sin 5\theta+2jG_1G_2G_4G_5Z_0^4\sin 5\theta+2jG_1G_3G_4G_5Z_0^4\sin 5\theta+ \\
& 2jG_2G_3G_4G_5Z_0^4\sin 5\theta\Big)
\end{aligned}
\tag{A.5}
$$

$$
\begin{aligned}
D \;=\;& \cos^5\theta + j(G_1 + 2G_2 + 3G_3 + 4G_4 + 5G_5)Z_0\cos^4\theta\sin\theta - \\
& (10 + 3G_3G_4Z_0^2 + 6G_3G_5Z_0^2 + 4G_4G_5Z_0^2 + \\
& 2G_2(G_3 + 2G_4 + 3G_5)Z_0^2 + G_1(G_2 + 2G_3 + 3G_4 + 4G_5)Z_0^2)\cos^3\theta\sin^2\theta - \\
& jZ_0(4G_3 + 4G_4 + 10G_5 + 3G_3G_4G_5Z_0^2 + 2G_2(3 + 2G_4G_5Z_0^2 + G_3(G_4 + 2G_5)Z_0^2) + \\
& G_1(6 + 3G_4G_5Z_0^2 + 2G_3(G_4 + 2G_5)Z_0^2 + G_2(G_3 + 2G_4 + 3G_5)Z_0^2))\cos^2\theta\sin^3\theta + \\
& (5 + G_3G_4Z_0^2 + 2G_3G_5Z_0^2 + 4G_4G_5Z_0^2 + 2G_2Z_0^2(G_3 + G_5 + G_3G_4G_5Z_0^2) + \\
& G_1Z_0^2(2G_3 + G_4 + 4G_5 + 2G_3G_4G_5Z_0^2 + \\
& G_2(3 + 2G_4G_5Z_0^2 + G_3(G_4 + 2G_5)Z_0^2)))\cos\theta\sin^4\theta + \\
& jZ_0(G_3 + G_5 + G_3G_4G_5Z_0^2 + G_1(1 + G_4G_5Z_0^2 + G_2Z_0^2(G_3 + G_5 + G_3G_4G_5Z_0^2)))\sin^5\theta
\end{aligned}
\tag{A.6}
$$

The load Z_T with short circuit load is:

$$
Z_T = \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D}
$$

Setting $Z_0 = 1$ and $Z_T = 1$ and cross multiply, we obtain:

$$
\begin{aligned}
&\cos^4\theta(-5 + G_1 + 2G_2 + 3G_3 + 4G_4 + 5G_5 - j\cot\theta) + \\
&j(10 - 6G_3 - 4G_4 + 3G_3G_4 + 6G_3G_5 + 4G_4G_5 + \\
&2G_2(-3 + G_3 + 2G_4 + 3G_5) + G_1(-4 + G_2 + 2G_3 + 3G_4 + 4G_5))\cos^3\theta\sin\theta - \\
&(G_3(4 + 3G_4(-1 + G_5)) + 2(-5 + 2G_4 + 5G_5) + 2G_2(3 + 2G_4(-1 + G_5) + \\
&G_3(-2 + G_4 + 2G_5)) + G_1(6 + 3G_4(-1 + G_5) + 2G_3(-2 + G_4 + 2G_5) + \\
&G_2(-3 + G_3 + 2G_4 + 3G_5)))\cos^2\theta\sin^2\theta - j(5 + G_3(-2 + G_4) - 4G_4 + \\
&2(G_3 + 2G_4)G_5 + 2G_2(-1 + G_3 - G_3G_4 + G_5 + G_3G_4G_5) + G_1(-4 + G_4 + 2G_3(1 + \\
&G_4(-1 + G_5)) + 4G_5 + G_2(3 + G_3(-2 + G_4) - 2G_4 + 2(G_3 + G_4)G_5)))\cos\theta\sin^3\theta + \\
&(-1 + G_3 - G_1(-1 + G_2(1 + G_3(-1 + G_4)) + G_4) + \\
&G_3G_4(-1 + G_5) + G_5 + G_1(G_2 + G_4 + G_2G_3G_4)G_5)\sin^4\theta = 0
\end{aligned}
\tag{A.7}
$$

Therefore, the system of equations that we need to solve are:

$$
\begin{cases}
-5 + G_1 + 2G_2 + 3G_3 + 4G_4 + 5G_5 = 0 \\
(10 - 6G_3 - 4G_4 + 3G_3G_4 + 6G_3G_5 + 4G_4G_5 + \ldots \\
2G_2(-3 + G_3 + 2G_4 + 3G_5) + G_1(-4 + G_2 + 2G_3 + 3G_4 + 4G_5)) = 0 \\
(G_3(4 + 3G_4(-1 + G_5)) + 2(-5 + 2G_4 + 5G_5) + 2G_2(3 + 2G_4(-1 + G_5) + \ldots \\
G_3(-2 + G_4 + 2G_5)) + G_1(6 + 3G_4(-1 + G_5) + \ldots \\
2G_3(-2 + G_4 + 2G_5) + G_2(-3 + G_3 + 2G_4 + 3G_5))) = 0 \\
(5 + G_3(-2 + G_4) - 4G_4 + 2(G_3 + 2G_4)G_5 + \ldots \\
2G_2(-1 + G_3 - G_3G_4 + G_5 + G_3G_4G_5) + \ldots \\
G_1(-4 + G_4 + 2G_3(1 + G_4(-1 + G_5)) + 4G_5 + \ldots \\
G_2(3 + G_3(-2 + G_4) - 2G_4 + 2(G_3 + G_4)G_5))) = 0 \\
(-1 + G_3 - G_1(-1 + G_2(1 + G_3(-1 + G_4)) + G_4) + \ldots \\
G_3G_4(-1 + G_5) + G_5 + G_1(G_2 + G_4 + G_2G_3G_4)G_5) = 0
\end{cases}
\tag{A.8}
$$

And the solution is: $G_1 = 1.8679$, $G_2 = 0.7777$, $G_3 = 0.3427$, $G_4 = 0.1136$, $G_5 = 0.0188$.

A.1.3 Six sections

The ABCD matrix for six sections case is:

$$
\begin{aligned}
A &= \cos^6\theta + j(5G_1+4G_2+3G_3+2G_4+G_5)Z_0\cos^5\theta\sin\theta - (15+2G_3G_4Z_0^2+ \\
&\quad 2G_3G_5Z_0^2+G_4G_5Z_0^2+2G_1(2G_2+3G_3+3G_4+2G_5)Z_0^2+ \\
&\quad G_2(3G_3+4G_4+3G_5)Z_0^2)\cos^4\theta\sin^2\theta - j(G_3G_4G_5+ \\
&\quad 2G_2(G_4G_5+G_3(G_4+G_5))+G_1(3G_4G_5+4G_3(G_4+G_5)+G_2(3G_3+4G_4+ \\
&\quad 3G_5)))Z_0^3\cos^3\theta\sin^3\theta + (15+6G_3G_4Z_0^2+6G_3G_5Z_0^2+6G_4G_5Z_0^2+ \\
&\quad G_2Z_0^2(4(G_4+G_5)+G_3(4+G_4G_5Z_0^2))+2G_1Z_0^2(G_3+G_4+2G_5+G_3G_4G_5Z_0^2+ \\
&\quad G_2(2+G_4G_5Z_0^2+G_3(G_4+G_5)Z_0^2)))\cos^2\theta\sin^4\theta + jZ_0(3G_3+2G_4+5G_5+ \\
&\quad 3G_3G_4G_5Z_0^2+2G_2(2+G_4G_5Z_0^2+G_3(G_4+G_5)Z_0^2)+G_1(1+G_4G_5Z_0^2+G_2Z_0^2(G_3+ \\
&\quad G_5+G_3G_4G_5Z_0^2)))\cos\theta\sin^5\theta + (1/4)(-4(1+G_4G_5Z_0^2+G_2Z_0^2(G_3+G_5+ \\
&\quad G_3G_4G_5Z_0^2))\sin^6\theta - j(5G_1+4G_2+5G_3+6G_4+5G_5)Z_0\sin^3 2\theta)
\end{aligned}
$$

$$
\begin{aligned}
B &= Z_0\sin\theta(6j\cos^5\theta - (5G_1+8G_2+9G_3+8G_4+5G_5)Z_0\cos^4\theta\sin\theta - \\
&\quad 2j(10+3G_3G_4Z_0^2+3G_3G_5Z_0^2+2G_4G_5Z_0^2+G_1(2G_2+3G_3+3G_4+2G_5)Z_0^2+ \\
&\quad G_2(3G_3+4G_4+3G_5)Z_0^2)\cos^3\theta\sin^2\theta + Z_0(8G_2+6G_3+8G_4+10G_5+ \\
&\quad 3G_3G_4G_5Z_0^2+G_1(10+3G_4G_5Z_0^2+3G_2(G_3+G_5)Z_0^2))\cos^2\theta\sin^3\theta + \\
&\quad 2j(3+G_3G_4Z_0^2+G_3G_5Z_0^2+2G_4G_5Z_0^2+G_2Z_0^2(G_3+G_5+G_3G_4G_5Z_0^2)+ \\
&\quad G_1Z_0^2(G_3+G_4+2G_5+G_3G_4G_5Z_0^2+G_2(2+G_4G_5Z_0^2+ \\
&\quad G_3(G_4+G_5)Z_0^2)))\cos\theta\sin^4\theta - Z_0\sin\theta((G_3+G_5+G_3G_4G_5Z_0^2+ \\
&\quad G_1(1+G_4G_5Z_0^2+G_2Z_0^2(G_3+G_5+G_3G_4G_5Z_0^2)))\sin^4\theta - \\
&\quad (G_1(G_2G_4+G_3(G_4+G_5))+G_2(G_4G_5+G_3(G_4+G_5)))Z_0^2\sin^2 2\theta))
\end{aligned}
\tag{A.9}
$$

$$
\begin{aligned}
C \;=\; & ((G_1+G_2+G_3+G_4+G_5+G_6)Z_0\cos^6\theta + j(6+G_3G_4Z_0^2+2G_3G_5Z_0^2+ \\
& G_4G_5Z_0^2+3G_3G_6Z_0^2+2G_4G_6Z_0^2+G_5G_6Z_0^2+G_2(G_3+2G_4+3G_5+4G_6)Z_0^2+ \\
& G_1(G_2+2G_3+3G_4+4G_5+5G_6)Z_0^2)\cos^5\theta\sin\theta - Z_0(6G_3+7G_4+10G_5+ \\
& 15G_6+G_3G_4G_5Z_0^2+2G_3G_4G_6Z_0^2+2G_3G_5G_6Z_0^2+G_4G_5G_6Z_0^2+G_2(7+ \\
& 3G_5G_6Z_0^2+2G_4(G_5+2G_6)Z_0^2+G_3(G_4+2G_5+3G_6)Z_0^2)+G_1(10+3G_4G_5Z_0^2+ \\
& 6G_4G_6Z_0^2+4G_5G_6Z_0^2+2G_3(G_4+2G_5+3G_6)Z_0^2+ \\
& G_2(G_3+2G_4+3G_5+4G_6)Z_0^2))\cos^4\theta\sin^2\theta - \\
& jZ_0^2(4G_3G_4+6G_3G_5+6G_4G_5+G_3G_4G_5G_6Z_0^2+G_2(4G_5+2G_4(2+G_5G_6Z_0^2)+ \\
& G_3(4+2G_5G_6Z_0^2+G_4(G_5+2G_6)Z_0^2))+G_1(4G_4+4G_5+3G_4G_5G_6Z_0^2+ \\
& 2G_3(3+2G_5G_6Z_0^2+G_4(G_5+2G_6)Z_0^2)+G_2(6+3G_5G_6Z_0^2+2G_4(G_5+2G_6)Z_0^2+ \\
& G_3(G_4+2G_5+3G_6)Z_0^2)))\cos^3\theta\sin^3\theta + Z_0(9G_3+7G_4+5G_5+15G_6+ \\
& 3G_3G_4G_5Z_0^2+6G_3G_4G_6Z_0^2+6G_3G_5G_6Z_0^2+6G_4G_5G_6Z_0^2+G_2(7+4G_5G_6Z_0^2+ \\
& 2G_4(G_5+2G_6)Z_0^2+G_3Z_0^2(2G_4+2G_5+4G_6+G_4G_5G_6Z_0^2))+G_1(5+G_4G_5Z_0^2+ \\
& 2G_4G_6Z_0^2+4G_5G_6Z_0^2+2G_3Z_0^2(G_4+G_6+G_4G_5G_6Z_0^2)+G_2Z_0^2(2G_4+G_5+ \\
& 4G_6+2G_4G_5G_6Z_0^2+G_3(3+2G_5G_6Z_0^2+G_4(G_5+2G_6)Z_0^2))))\cos^2\theta\sin^4\theta + \\
& j(6+3G_3G_4Z_0^2+G_4G_5Z_0^2+3G_3G_6Z_0^2+2G_4G_6Z_0^2+5G_5G_6Z_0^2+3G_3G_4G_5G_6Z_0^4+ \\
& G_2Z_0^2(2G_4+G_5+4G_6+2G_4G_5G_6Z_0^2+G_3(3+2G_5G_6Z_0^2+G_4(G_5+2G_6)Z_0^2))+ \\
& G_1Z_0^2(G_4+G_6+G_4G_5G_6Z_0^2+G_2(1+G_5G_6Z_0^2+ \\
& G_3Z_0^2(G_4+G_6+G_4G_5G_6Z_0^2))))\cos\theta\sin^5\theta + (1/4)(-4Z_0(G_4+ \\
& G_6+G_4G_5G_6Z_0^2+G_2(1+G_5G_6Z_0^2+G_3Z_0^2(G_4+G_6+G_4G_5G_6Z_0^2)))\sin^6\theta - \\
& j(10+5G_1G_6Z_0^2+4G_2G_6Z_0^2+5G_3G_6Z_0^2+6G_4G_6Z_0^2+5G_5G_6Z_0^2)\sin^3 2\theta))/Z_0
\end{aligned}
$$

$$
\begin{aligned}
D \;=\;& \cos^6\theta + j(G_1 + 2G_2 + 3G_3 + 4G_4 + 5G_5 + 6G_6)Z_0\cos^5\theta\sin\theta - (15 + 3G_3G_4Z_0^2 + \\
& 6G_3G_5Z_0^2 + 4G_4G_5Z_0^2 + 9G_3G_6Z_0^2 + 8G_4G_6Z_0^2 + 5G_5G_6Z_0^2 + \\
& 2G_2(G_3 + 2G_4 + 3G_5 + 4G_6)Z_0^2 + \\
& G_1(G_2 + 2G_3 + 3G_4 + 4G_5 + 5G_6)Z_0^2)\cos^4\theta\sin^2\theta - \\
& j(3G_3G_4G_5 + 6G_3G_4G_6 + 6G_3G_5G_6 + 4G_4G_5G_6 + \\
& 2G_2(G_5(2G_4 + 3G_6) + G_3(G_4 + 2G_5 + 3G_6)) + G_1(3G_4G_5 + 6G_4G_6 + 4G_5G_6 + \\
& 2G_3(G_4 + 2G_5 + 3G_6) + G_2(G_3 + 2G_4 + 3G_5 + 4G_6)))Z_0^3\cos^3\theta\sin^3\theta + \\
& (15 + 4G_3G_4Z_0^2 + 2G_3G_5Z_0^2 + 4G_4G_5Z_0^2 + 6G_3G_6Z_0^2 + 8G_4G_6Z_0^2 + \\
& 10G_5G_6Z_0^2 + 3G_3G_4G_5G_6Z_0^4 + 2G_2Z_0^2(G_5 + 4G_6 + 2G_4(1 + G_5G_6Z_0^2) + \\
& G_3(3 + 2G_5G_6Z_0^2 + G_4(G_5 + 2G_6)Z_0^2)) + G_1Z_0^2(4G_4 + 4G_5 + 10G_6 + 3G_4G_5G_6Z_0^2 + \\
& 2G_3(3 + 2G_5G_6Z_0^2 + G_4(G_5 + 2G_6)Z_0^2) + G_2(6 + 3G_5G_6Z_0^2 + \\
& 2G_4(G_5 + 2G_6)Z_0^2 + G_3(G_4 + 2G_5 + 3G_6)Z_0^2)))\cos^2\theta\sin^4\theta + jZ_0(3G_3 + 4G_4 + \\
& G_5 + 6G_6 + G_3G_4G_5Z_0^2 + 2G_3G_4G_6Z_0^2 + 2G_3G_5G_6Z_0^2 + 4G_4G_5G_6Z_0^2 + \\
& 2G_2(1 + G_5G_6Z_0^2 + G_3Z_0^2(G_4 + G_6 + G_4G_5G_6Z_0^2)) + \\
& G_1(5 + G_4G_5Z_0^2 + 2G_4G_6Z_0^2 + 4G_5G_6Z_0^2 + 2G_3Z_0^2(G_4 + \\
& G_6 + G_4G_5G_6Z_0^2) + G_2Z_0^2(2G_4 + G_5 + 4G_6 + 2G_4G_5G_6Z_0^2 + \\
& G_3(3 + 2G_5G_6Z_0^2 + G_4(G_5 + 2G_6)Z_0^2))))\cos\theta\sin^5\theta - \\
& (1 + G_5G_6Z_0^2 + G_3Z_0^2(G_4 + G_6 + G_4G_5G_6Z_0^2) + G_1Z_0^2(G_4 + G_6 + G_4G_5G_6Z_0^2 + \\
& G_2(1 + G_5G_6Z_0^2 + G_3Z_0^2(G_4 + G_6 + G_4G_5G_6Z_0^2))))\sin^6\theta - \\
& (1/4)jZ_0(5G_1 + 5G_3 + 4G_4 + 5G_5 + 10G_6 + G_2(6 + 4G_4G_6Z_0^2))\sin^3 2\theta
\end{aligned}
\tag{A.10}
$$

The load Z_T with short circuit load is:

$$
Z_T = \frac{AZ_L + B}{CZ_L + D} = \frac{B}{D}
$$

Setting $Z_0 = 1$ and $Z_T = 1$ and cross multiply, we obtain:

$$
\begin{aligned}
&(-j)\cos^5\theta(-6 + G_1 + 2G_2 + 3G_3 + 4G_4 + 5G_5 + 6G_6 - j\cot\theta) + \\
&(15 - 9G_3 - 8G_4 + 3G_3G_4 - 5G_5 + 6G_3G_5 + 4G_4G_5 + 9G_3G_6 + 8G_4G_6 + \\
&5G_5G_6 + 2G_2(-4 + G_3 + 2G_4 + 3G_5 + 4G_6) + G_1(-5 + G_2 + 2G_3 + 3G_4 + \\
&4G_5 + 5G_6))\cos^4\theta\sin\theta + j(-20 + 10G_3 + 8G_4 - 6G_3G_4 + 10G_5 - \\
&6G_3G_5 - 4G_4G_5 + 3G_3G_4G_5 + 2(10 + 2G_4G_5 + \\
&3G_3(G_4 + G_5))G_6 + 2G_2(6 + 3G_5(-1 + G_6) + 2G_4(-2 + G_5 + 2G_6) + \\
&G_3(-3 + G_4 + 2G_5 + 3G_6)) + G_1(10 - 6G_4 - 4G_5 + 3G_4G_5 + 6G_4G_6 + 4G_5G_6 + \\
&2G_3(-3 + G_4 + 2G_5 + 3G_6) + G_2(-4 + G_3 + 2G_4 + 3G_5 + 4G_6)))\cos^3\theta\sin^2\theta - \\
&(15 - 6G_3 - 8G_4 + 4G_3G_4 - 10G_5 + 2G_3G_5 + 4G_4G_5 - 3G_3G_4G_5 + \\
&(8G_4 + 10G_5 + 3G_3(2 + G_4G_5))G_6 + 2G_2(-4 + G_5 + 2G_4(1 + G_5(-1 + G_6)) + \\
&4G_6 + G_3(3 + 2G_5(-1 + G_6) + G_4(-2 + G_5 + 2G_6))) + \\
&G_1(G_4(4 + 3G_5(-1 + G_6)) + 2(-5 + 2G_5 + 5G_6) + \\
&2G_3(3 + 2G_5(-1 + G_6) + G_4(-2 + G_5 + 2G_6)) + G_2(6 + 3G_5(-1 + G_6) + \\
&2G_4(-2 + G_5 + 2G_6) + G_3(-3 + G_4 + 2G_5 + 3G_6))))\cos^2\theta\sin^3\theta - \\
&j(-6 + 3G_3 + 4G_4 - 2G_3G_4 + G_5 - 2G_3G_5 - 4G_4G_5 + \\
&G_3G_4G_5 - 2G_2(-1 + G_3(1 + G_4(-1 + G_5)) + G_5) + 2G_2(G_3 + G_5 + \\
&G_3G_4G_5)G_6 + 2(3 + 2G_4G_5 + G_3(G_4 + G_5))G_6 + G_1(5 + G_4(-2 + G_5) - 4G_5 + \\
&2(G_4 + 2G_5)G_6 + 2G_3(-1 + G_4 - G_4G_5 + G_6 + G_4G_5G_6) + \\
&G_2(-4 + G_5 + 2G_4(1 + G_5(-1 + G_6)) + 4G_6 + G_3(3 + G_4(-2 + G_5) - 2G_5 + \\
&2(G_4 + G_5)G_6))))\cos\theta\sin^4\theta + (1 + G_5(-1 + G_6) + G_3(-1 + G_4 - G_4G_5 + \\
&G_6 + G_4G_5G_6) + G_1(-1 + G_4 - G_2(-1 + G_3(1 + G_4(-1 + G_5)) + G_5) + \\
&G_4G_5(-1 + G_6) + G_6 + G_2(G_3 + G_5 + G_3G_4G_5)G_6))\sin^5\theta \;=\; 0
\end{aligned}
\tag{A.11}
$$

Therefore, the system of equations that we need to solve are:

$$\begin{cases} -6+G_1+2G_2+3G_3+4G_4+5G_5+6G_6=0 \\ (15-9G_3-8G_4+3G_3G_4-5G_5+6G_3G_5+4G_4G_5+\dots \\ 9G_3G_6+8G_4G_6+5G_5G_6+2G_2(-4+G_3+2G_4+\dots \\ 3G_5+4G_6)+G_1(-5+G_2+2G_3+3G_4+4G_5+5G_6))=0 \\ (-20+10G_3+8G_4-6G_3G_4+10G_5-6G_3G_5-4G_4G_5+\dots \\ 3G_3G_4G_5+2(10+2G_4G_5+3G_3(G_4+G_5))G_6+\dots \\ 2G_2(6+3G_5(-1+G_6)+2G_4(-2+G_5+2G_6)+\dots \\ G_3(-3+G_4+2G_5+3G_6))+G_1(10-6G_4-4G_5+\dots \\ 3G_4G_5+6G_4G_6+4G_5G_6+2G_3(-3+G_4+2G_5+3G_6)+\dots \\ G_2(-4+G_3+2G_4+3G_5+4G_6)))=0 \\ (15-6G_3-8G_4+4G_3G_4-10G_5+2G_3G_5+4G_4G_5-\dots \\ 3G_3G_4G_5+(8G_4+10G_5+3G_3(2+G_4G_5))G_6+\dots \\ 2G_2(-4+G_5+2G_4(1+G_5(-1+G_6))+4G_6+\dots \\ G_3(3+2G_5(-1+G_6)+G_4(-2+G_5+2G_6)))+\dots \\ G_1(G_4(4+3G_5(-1+G_6))+2(-5+2G_5+5G_6)+\dots \\ 2G_3(3+2G_5(-1+G_6)+G_4(-2+G_5+2G_6))+\dots \\ G_2(6+3G_5(-1+G_6)+2G_4(-2+G_5+2G_6)+\dots \\ G_3(-3+G_4+2G_5+3G_6))))=0 \\ (-6+3G_3+4G_4-2G_3G_4+G_5-2G_3G_5-4G_4G_5+\dots \\ G_3G_4G_5-2G_2(-1+G_3(1+G_4(-1+G_5))+G_5)+\dots \\ 2G_2(G_3+G_5+G_3G_4G_5)G_6+2(3+2G_4G_5+\dots \\ G_3(G_4+G_5))G_6+G_1(5+G_4(-2+G_5)-4G_5+\dots \\ 2(G_4+2G_5)G_6+2G_3(-1+G_4-G_4G_5+G_6+\dots \\ G_4G_5G_6)+G_2(-4+G_5+2G_4(1+G_5(-1+G_6))+\dots \\ 4G_6+G_3(3+G_4(-2+G_5)-2G_5+2(G_4+G_5)G_6))))=0 \\ (1+G_5(-1+G_6)+G_3(-1+G_4-G_4G_5+G_6+G_4G_5G_6)+\dots \\ G_1(-1+G_4-G_2(-1+G_3(1+G_4(-1+G_5))+G_5)+\dots \\ G_4G_5(-1+G_6)+G_6+G_2(G_3+G_5+G_3G_4G_5)G_6))=0 \end{cases} \tag{A.12}$$

And the solution is: $G_1 = 1.9209$, $G_2 = 0.8526$, $G_3 = 0.4303$, $G_4 = 0.1877$, $G_5 = 0.0566$, $G_6 = 0.0082$.

A.2 Expression of $\frac{d^2\Gamma}{d\theta^2}$ for three sections case

$$
\begin{aligned}
\frac{d^2\Gamma}{d\theta^2} = \Big[&(8G_2(-8+3G_2^2)+G_1^2(64+104G_2-30G_2^3+64G_3)+G_1^3(64+G_2(28(1+G_3)-5G_2(4+G_2+G_2G_3)))+4G_1(-88+G_2(-64(1+G_3)+3G_2(4+3G_2(1+G_3)))))\cos\theta+\\
&3(-8(-12+G_2^2)(2+G_2+2G_3)+4G_1(24-G_2(-16+3G_2^2)(1+G_3))+2G_1^2(-32(1+G_3)+3G_2(-8+G_2(2+3G_2+2G_3)))+G_1^3(-32+3G_2(-4(1+G_3)+G_2(4+G_2+G_2G_3))))\cos 3\theta+\\
&(-8G_2^2(-2+G_2-2G_3)-4G_1(-4+3G_2^2)(8+G_2+G_2G_3)+2G_1^2(64(1+G_3)-G_2(-8+15G_2(2+G_2+2G_3)))+G_1^3(32+G_2(4(1+G_3)-5G_2(4+G_2+G_2G_3))))\cos 5\theta+\\
&G_2((2+G_1)^3(2+G_2)^2+(32G_2+24G_1^2G_2+12G_1(4+G_2^2)+G_1^3(4+G_2^2))G_3)\cos 7\theta-\\
&j(8G_2(-8+3G_2^2)(1+G_3)+2G_1^2(32-G_2(-52+15G_2^2)(1+G_3))+4G_1(88(1+G_3)-G_2(-80+3G_2(4+5G_2+4G_3)))+G_1^3(-128(1+G_3)+G_2(-132+5G_2(4+7G_2+4G_3))))\sin\theta+\\
&3j(-8(-12+G_2^2)(2+G_2+G_2G_3)+4G_1(G_2(8+G_2(-4+G_2-4G_3))+24(1+G_3))+G_1^3(-32(1+G_3)+G_2(-20+G_2(8+7G_2+8G_3)))+2G_1^2(-32+G_2(-8(1+G_3)+G_2(14+G_2+G_2G_3))))\sin 3\theta-\\
&j(-8G_2^2(-2+G_2+G_2G_3)+4G_1(-32(1+G_3)+G_2(-4+3G_2(4+3G_2+4G_3)))+G_1^3(-32(1+G_3)+G_2(-4+G_2(16+7G_2+16G_3)))+2G_1^2(-64+G_2(-8(1+G_3)+3G_2(14+3G_2(1+G_3)))))\sin 5\theta+\\
&jG_2((2+G_1)^3(2+G_2)^2+2(24G_1G_2+2G_1^3G_2+4(4+G_2^2)+3G_1^2(4+G_2^2))G_3)\sin 7\theta\Big]/\\
&\Big[16(\cos^3\theta+j(3+G_1+2G_2+3G_3)\cos^2\theta\sin\theta-(3+2G_2(1+G_3)+G_1(2+G_2+2G_3))\cos\theta\sin^2\theta-j(1+G_3+G_1(1+G_2+G_2G_3))\sin^3\theta)^3\Big]
\end{aligned}
\tag{A.13}
$$

A.3 General form of reflection coefficients for 4, 5, and 6 layers case

Below are the general form of Eq. 2.22, 2.23, and 2.24 (before substitution of $Z_S = Z_0 = 1$ and maximally flat solutions) respectively.

$$\Gamma_4 = \frac{-Z_S + \frac{Z_0 \sin\theta(4j\cos^3\theta - (3G_1+4G_2+3G_3)Z_0\cos^2\theta\sin\theta - 2j(2+(G_2G_3+G_1(G_2+G_3))Z_0^2)\cos\theta\sin^2\theta + Z_0(G_1+G_3+G_1G_2G_3Z_0^2)\sin^3\theta)}{\cos^4\theta + j(G_1+2G_2+3G_3+4G_4)Z_0\cos^3\theta\sin\theta - (6+(G_1G_2+2G_1G_3+2G_2G_3+3(G_1+G_3)G_4)Z_0^2)\cos^2\theta\sin^2\theta - jZ_0(3G_1+2G_2+G_3+4G_4+(G_1G_2G_3+2G_2G_3G_4+2G_1(G_2+G_3)G_4)Z_0^2)\cos\theta\sin^3\theta + (1+Z_0^2(G_3G_4+G_1(G_2+G_4+G_2G_3G_4Z_0^2)))\sin^4\theta - G_2G_4Z_0^2\sin^2 2\theta}}{Z_S + \frac{Z_0 \sin\theta(4j\cos^3\theta - (3G_1+4G_2+3G_3)Z_0\cos^2\theta\sin\theta - 2j(2+(G_2G_3+G_1(G_2+G_3))Z_0^2)\cos\theta\sin^2\theta + Z_0(G_1+G_3+G_1G_2G_3Z_0^2)\sin^3\theta)}{\cos^4\theta + j(G_1+2G_2+3G_3+4G_4)Z_0\cos^3\theta\sin\theta - (6+(G_1G_2+2G_1G_3+2G_2G_3+3(G_1+G_3)G_4)Z_0^2)\cos^2\theta\sin^2\theta - jZ_0(3G_1+2G_2+G_3+4G_4+(G_1G_2G_3+2G_2G_3G_4+2G_1(G_2+G_3)G_4)Z_0^2)\cos\theta\sin^3\theta + (1+Z_0^2(G_3G_4+G_1(G_2+G_4+G_2G_3G_4Z_0^2)))\sin^4\theta - G_2G_4Z_0^2\sin^2 2\theta}} \tag{A.14}$$

$$\Gamma_5 = \text{[illegible]} \tag{A.15}$$

$$\Gamma_6 = \text{[illegible]} \tag{A.16}$$

A.4 MATLAB code of GA for variable length case

```
clear

for t = 1:1 % The theta limit to adjust for spacings

    close all

    %%
    %{
1. Define a single figure of merit which will be used to evaluate all the
solutions in the same manner.
2. In this case, we will be using S11 as the figure of merit (the
lower the S11 the better. (how this figure of merit will be calculated can
be changed to suit the search requirements).
3. S11 = (use the S11 formulas for each layer).
4. Start the population with random genes (random number or randomly
    manually set between 0-1)
5. Test their fitness (find their figure of merit).
6. Choose the best
    %}

    tic;

    %% Generating initial population

    number_of_population = 1000; % Arbitrarily set
    number_of_layers = 4;
    number_of_genes = 2*number_of_layers; % This equals the number of
        Jaumann layers AND the variable length

    number_of_generation = 1000;

    % Initialization
    population = zeros(number_of_population,number_of_genes);

    % Generating population
    % Genes vary from 0-2
    population = 0 + 2*rand(number_of_population,number_of_genes);


    %% Manually defining initial population with best survivors from
        previous search
    % The ones marked with '*' are updated with latest iteration
    %% 1 layer
```

```
    % population(1:50,1) = 1.0157;

    %% 2 layers
    %    population(1:50,1) = 1.486353;
    %    population(1:50,2) = 0.511298;
    %    population(1:50,3) = 0.040800;
    %    population(1:50,4) = 0.042000;

    %% 3 layers *
    %    population(1:25,1) = 1.6798;
    %    population(1:25,2) = 1.8442;
    %    population(1:25,3) = 0.7360;
    %
    %    population(1:25,4) = 0.0830;
    %    population(1:25,5) = 0.0135;
    %    population(1:25,6) = 0.0284;

%    population(1:25,1) = 1.377;
%    population(1:25,2) = 0.533;

%    population(1:25,3) = 0.321;
%    population(1:25,4) = 0.784;
%    population(1:25,5) = 0.791;
%    population(1:25,6) = 0.813;

%    population(1,1:6) = [1.345680  0.565396      0.353227      0.773810
         0.801603      0.820233      ];

    %% 4 layers *
    % population(1:25,1) = 0.5636;
    % population(1:25,2) = 3.4802;
    % population(1:25,3) = 1.1403;
    % population(1:25,4) = 0.4061;
    %
    % population(1:25,5) = 0.9573;
    % population(1:25,6) = 1.1320;
    % population(1:25,7) = 0.2257;
    % population(1:25,8) = 0.2014;

%    population(1,1:8) = [1.238353  0.600010      0.330545      0.237590
         8.589016      8.448836      8.666715      8.253772      ];

%    population(1,1:8) = [2.397     0.899   0.437   0.277 0.400   0.733
    0.765  0.765];
```

```
%% 5 layers *
% population(1:25,1) = 0.6800;
% population(1:25,2) = 2.3352;
% population(1:25,3) = 1.8294;
% population(1:25,4) = 0.7995;
% population(1:25,5) = 0.2149;

% population(1:25,6) = 0.1212;
% population(1:25,7) = 1.4906;
% population(1:25,8) = 0.1198;
% population(1:25,9) = 0.1355;
% population(1:25,10) = 0.1806;

%% 6 layers
%     population(1:25,1) = 1.160500;
%     population(1:25,2) = 1.405500;
%     population(1:25,3) = 1.981900;
%     population(1:25,4) = 2.343700;
%     population(1:25,5) = 0.978500;
%     population(1:25,6) = 0.350083;
%
%     population(1:25,7) = 0.302400;
%     population(1:25,8) = 0.773300;
%     population(1:25,9) = 0.025700;
%     population(1:25,10) = 0.428900;
%     population(1:25,11) = 0.040500;
%     population(1:25,12) = 0.042700;

%% Loading the entire table
%{
fID = fopen('C:\Users\Wiwi Samsul\Documents\MATLAB\GA_results.txt');
B = textscan(fID,'%s','HeaderLines',1,'delimiter','\t');
fclose(fID);

TableSize = length(B{1});
NumberOfLayers = zeros(1,round(TableSize/360));
MaxNumberOfLayers = 6;
NumberOfEntriesPerLayer = 20;
NumberOfGenes = 6;

% test = cell2mat(B{1}(1))
for n = 1:round(TableSize/360)
    NumberOfLayers(1,n) = str2double(cell2mat(B{1}(1 + (n-1)*360)));
end
```

```
GValues = zeros(NumberOfEntriesPerLayer,NumberOfGenes,length(
    NumberOfLayers));
XValues = zeros(NumberOfEntriesPerLayer,NumberOfGenes,length(
    NumberOfLayers));
RelativeBandwidthValues = zeros(NumberOfEntriesPerLayer,1,length(
    NumberOfLayers));
StartBandwidthValues = zeros(NumberOfEntriesPerLayer,1,length(
    NumberOfLayers));
EndBandwidthValues = zeros(NumberOfEntriesPerLayer,1,length(
    NumberOfLayers));

for m = 1:NumberOfEntriesPerLayer
    for n = 1:NumberOfGenes
        for p = 1:length(NumberOfLayers)
            GNaNCheck = isnan(str2double(cell2mat(B{1}(18*(m-1) + (2+(n
                -1)) + 360*(p-1)))));
            if(~GNaNCheck)
                GValues(m,n,p) = str2double(cell2mat(B{1}(18*(m-1) +
                    (2+(n-1)) + 360*(p-1))));
            end
            XNaNCheck = isnan(str2double(cell2mat(B{1}(18*(m-1) + (8+(n
                -1)) + 360*(p-1)))));
            if(~XNaNCheck)
                XValues(m,n,p) = str2double(cell2mat(B{1}(18*(m-1) +
                    (8+(n-1)) + 360*(p-1))));
            end
            RelativeBandwidthValues(m,1,p) = str2double(cell2mat(B
                {1}(18*(m-1) + (14) + 360*(p-1))));
            StartBandwidthValues(m,1,p) = str2double(cell2mat(B{1}(18*(m
                -1) + (15) + 360*(p-1))));
            EndBandwidthValues(m,1,p) = str2double(cell2mat(B{1}(18*(m
                -1) + (16) + 360*(p-1))));
        end
    end
end

% Automatically populate with the newest result taken from the table
for m = 1:number_of_layers
    population(1:25,m) = GValues(t,m,number_of_layers);
    population(1:25,(m + number_of_layers)) = XValues(t,m,
        number_of_layers);
end

%}
%%
```

```
fitness_collection = zeros(1,number_of_population); % Initialization
all_history_fitness_collection = zeros(number_of_generation,
    number_of_population); % For collecting all history of fitness over
     generations
all_history_generation_collection = zeros(number_of_generation,
    number_of_population,number_of_genes); % For collecting all history
     of population
%    best_fitness_gene = zeros(1,number_of_genes); % To store the best
gene
best_fitness_record = 0; % To store the best recorded fitness
generation_number_record = 0; % To record at which generation the value
     was found

n = 0;
for n = 1:number_of_generation
    if(mod(n,100)==0)
        fprintf('%d th gen finished. \n',n);
    end
    % while ((max(fitness_collection) < 1.4018) && (n <
        number_of_generation))
    %    n = n + 1;
    for m = 1:number_of_population

        fitness_collection(1,m) = S11Fitness_with_var_L_var_theta(
            population(m,:),t);

    end

    all_history_fitness_collection(n,:) = fitness_collection(1,:);
    all_history_generation_collection(n,:,:) = population(:,:);

    [fitness_sorted, original_index_in_sorted_order] = sort(
        fitness_collection,'descend');

    % Take the better half/quarter of population based on its fitness
    survivors = zeros(ceil(0.5*number_of_population),number_of_genes);

    % Record the best fitness
    if(best_fitness_record < fitness_sorted(1))
        best_fitness_record = fitness_sorted(1);
        best_fitness_gene = population(original_index_in_sorted_order
            (1),:);
        generation_number_record = n;
    end
```

```
    for m = 1:length(survivors)
        survivors(m,:) = population(original_index_in_sorted_order(m)
            ,:);
    end

    new_gen = zeros(number_of_population,number_of_genes);

    %{
% COMMENT BLOCK SPECIAL CASE FOR 1 GENE POPULATIONS
% Crossover process (by averaging)
% Select two surviving parents randomly and average their values
% This only works for population with 1 gene
for m = 1:number_of_population

    % Create a random number between 1-length(surviving gen)
    % To randomly take 2 parents
    parent1_index = round(1 + (length(survivors) - 1).*rand(1,1));
    parent2_index = round(1 + (length(survivors) - 1).*rand(1,1));

    new_gen(m) = 0.5*(survivors(parent1_index)+survivors(parent2_index)
        );
end
    %}

    % Cross over genes
    for m = 1:number_of_population

        % Randomly take 2 parents
        parent1_index = round(1 + (length(survivors) - 1).*rand(1,1));
        parent2_index = round(1 + (length(survivors) - 1).*rand(1,1));

        % Send for cross over
        new_gen(m,:) = crossover(survivors(parent1_index,:),survivors(
            parent2_index,:));
    end

    % Mutation
    mutation_magnitude = 0.5;
    mutation_chance = 0.2;
    population = abs(mutation(new_gen,mutation_magnitude,
        mutation_chance));

end
```

```
x_axis = 1:number_of_population;

fig1 = figure(1);
for m = 1:number_of_generation
    hold on
    plot(x_axis,all_history_fitness_collection(m,:));
    hold on
end
grid on
% axis([1 number_of_population floor(min(min(
    all_history_fitness_collection))) ceil(max(max(
    all_history_fitness_collection)))])
axis([1 number_of_population 0 ceil(max(max(
    all_history_fitness_collection)))])
xlabel('Individual number / Solutions')
ylabel('Fitness / Relative bandwidth at -10 dB')
title('Evolution of solutions')
% legend('Gen 1','Gen 2','Gen 3','Gen 4','Gen 5','Gen 6','Gen 7','Gen
    8','Gen 9','Gen 10')

image_filename = sprintf('Genetic_algorithm_result_%
    d_layers_with_var_L_population_%d_theta.png',number_of_layers,t);
%     saveas(gcf,image_filename);

% Line markers
lineMarkers = ['-k ';':k ';'--k ';'-.k ';'-sk ';':sk ';'--sk';'-.sk
    ';'-+k ';':+k ';'--+k';'-.+k';'-ok ';':ok ';'--ok';'-.ok';'-xk ';':
    xk ';'--xk';'-.xk';'-dk ';':dk ';'--dk';'-.dk'];
lineMarkers = cellstr(lineMarkers);

fig2 = figure(2);
hold on
for m = 1:number_of_genes
    plot(all_history_generation_collection(number_of_generation,:,m),
        char(lineMarkers(m)));
end
grid on
% axis([1 number_of_population floor(min(min(min(
    all_history_generation_collection(number_of_generation,:,:)))))
    ceil(max(max(max((all_history_generation_collection(
    number_of_generation,:,:)))))]);
axis([1 number_of_population 0 ceil(max(max(max((
    all_history_generation_collection(number_of_generation,:,:))))))]);
xlabel('Individual number / Solutions')
ylabel('Gene value')
```

```
    title('Last generation')
    legend('1st gene','2nd gene','3rd gene','4th gene','5th gene','6th gene
       ','7th gene','8th gene','9th gene','10th gene','11th gene','12th
       gene')

    image_filename = sprintf('Genetic_algorithm_result_%
       d_layers_with_var_L_genes_%d_theta.png',number_of_layers,t);
%     saveas(gcf,image_filename);

    maximum_fitness = max(max(all_history_fitness_collection));
    display_text = sprintf('Maximum fitness found: %f',maximum_fitness);
    disp(display_text)

    display_text = sprintf('Best fitness record: %f',best_fitness_record);
    disp(display_text)
    display_text = sprintf('Best fitness gene: \n');
    disp(display_text)
    display_text = '';
    add_gene = sprintf('%f',best_fitness_gene(1));
    for n = 2:number_of_genes
       display_text = [display_text add_gene '\t'];
       add_gene = sprintf('%f',best_fitness_gene(n));
    end
    display_text = [display_text add_gene '\t'];
    display_text = sprintf(display_text);
    disp(display_text)

    double_check = S11Fitness_with_var_L_var_theta(best_fitness_gene,t);
    display_text = sprintf('Double checking the best fitness gene: %f',
       double_check);
    disp(display_text)

    display_text = sprintf('Solution found at generation #: %d',
       generation_number_record);
    disp(display_text)

    % Calling in the function to graph the result with best gene:
    [ fitness,theta_in_Hz,gamma ] = S11Fitness_with_var_L_var_theta(
       best_fitness_gene,t );
    S11FitnessGraph_with_var_L( theta_in_Hz, gamma )
    image_filename = sprintf('Genetic_algorithm_result_%
       d_layers_with_var_L_genes_S11_plot_%d_theta.png',number_of_layers,t
       );
%     saveas(gcf,image_filename);
```

```
    toc;

    total_search_time = toc;
    effective_search_time = toc*generation_number_record/
        number_of_generation;
    display_text = sprintf('Estimated search time: %f',
        effective_search_time);
    disp(display_text)

end
```

A.5 Supplemental MATLAB code of for GA

```
function [ fitness,theta_in_Hz,gamma ] = S11Fitness_with_var_L_var_theta(
    individual,t )
% fitness = S11Fitness(individual, numOfGenes)
% S11Fitness calculates the relative bandwidth of S11 at dB_level from
    individual solutions
% individual must be in normalized G
% Converting between L and x :
% L \lambda = 0.25 * x \theta

ZS = 1;
Z0 = 1;

f0 = 2.50e2;
theta = [0:pi/8000:t*pi];
theta_in_Hz = 2*theta*f0/pi;

numOfGenes = round(length(individual)/2);

dB_level = -10;

switch numOfGenes
    case 1
        G1 = individual(1);
        x1 = individual(2);
        gamma = mag2db(abs(-((ZS.*cos(x1.*theta) + 1i.*Z0.*(-1 + G1.*ZS).*
            sin(x1.*theta))./(ZS.*cos(x1.*theta) + 1i.*Z0.*(1 + G1.*ZS).*sin
            (x1.*theta)))));
    case 2
        G1 = individual(1);
        G2 = individual(2);
        x1 = individual(3);
        x2 = individual(4);
        gamma = mag2db(abs((cos(x1.*(theta)).*(ZS.*cos(x2.*(theta)) + 1i.*
            Z0.*(-1 + G2.*ZS).*sin(x2.*(theta))) + sin(x1.*(theta)).*(1i.*Z0
            .*(-1 + G1.*ZS + G2.*ZS).*cos(x2.*(theta)) - (ZS + G1.*Z0^2.*(-1
             + G2.*ZS)).*sin(x2.*(theta))))./ ((-cos(x1.*(theta))).*(ZS.*cos
            (x2.*(theta)) + 1i.*Z0.*(1 + G2.*ZS).*sin(x2.*(theta))) + sin(x1
            .*(theta)).*((-1i).*Z0.*(1 + G1.*ZS + G2.*ZS).*cos(x2.*(theta))
            + (ZS + G1.*Z0^2.*(1 + G2.*ZS)).*sin(x2.*(theta))))));
    case 3
        G1 = individual(1);
        G2 = individual(2);
        G3 = individual(3);
```

```
    x1 = individual(4);
    x2 = individual(5);
    x3 = individual(6);
    gamma = mag2db(abs((cos(x1.*(theta)).*(cos(x2.*(theta)).*((-1i).*ZS
        .*cos(x3.*(theta)) + Z0.*(-1 + G3.*ZS).*sin(x3.*(theta))) + sin(
        x2.*(theta)).*(Z0.*(-1 + G2.*ZS + G3.*ZS).*cos(x3.*(theta)) + 1i
        .*(ZS + G2.*Z0^2.*(-1 + G3.*ZS)).*sin(x3.*(theta)))) + sin(x1.*(
        theta)).*(cos(x2.*(theta)).*(Z0.*(-1 + G1.*ZS + G2.*ZS + G3.*ZS)
        .*cos(x3.*(theta)) + 1i.*(ZS + G1.*Z0^2.*(-1 + G3.*ZS) + G2.*Z0
        ^2.*(-1 + G3.*ZS)).*sin(x3.*(theta))) + sin(x2.*(theta)).*(1i.*(
        ZS + G1.*Z0^2.*(-1 + G2.*ZS + G3.*ZS)).*cos(x3.*(theta)) - Z0
        .*(-1 + G3.*ZS + G1.*(ZS + G2.*Z0^2.*(-1 + G3.*ZS))).*sin(x3.*(
        theta))))./ (1i.*cos(x1.*(theta)).*(cos(x2.*(theta)).*(ZS.*cos(
        x3.*(theta)) + 1i.*Z0.*(1 + G3.*ZS).*sin(x3.*(theta))) - sin(x2
        .*(theta)).*((-1i).*Z0.*(1 + G2.*ZS + G3.*ZS).*cos(x3.*(theta))
        + (ZS + G2.*Z0^2.*(1 + G3.*ZS)).*sin(x3.*(theta)))) + sin(x1.*(
        theta)).*((-1i).*cos(x2.*(theta)).*((-1i).*Z0.*(1 + G1.*ZS + G2
        .*ZS + G3.*ZS).*cos(x3.*(theta)) + (ZS + G1.*Z0^2.*(1 + G3.*ZS)
        + G2.*Z0^2.*(1 + G3.*ZS)).*sin(x3.*(theta))) + sin(x2.*(theta))
        .*((-1i).*(ZS + G1.*Z0^2.*(1 + G2.*ZS + G3.*ZS)).*cos(x3.*(theta
        )) + Z0.*(1 + G3.*ZS + G1.*(ZS + G2.*Z0^2.*(1 + G3.*ZS))).*sin(
        x3.*(theta)))))));
case 4
    G1 = individual(1);
    G2 = individual(2);
    G3 = individual(3);
    G4 = individual(4);
    x1 = individual(5);
    x2 = individual(6);
    x3 = individual(7);
    x4 = individual(8);
    gamma = mag2db(abs((-ZS + (Z0.*(cos(x2.*(theta)).*(cos(x3.*(theta))
        .*((-1i).*cos(x4.*(theta)).*sin(x1.*(theta)) + ((-1i).*cos(x1.*(
        theta)) + (G1 + G2 + G3).*Z0.*sin(x1.*(theta))).*sin(x4.*(theta)
        )) + sin(x3.*(theta)).*(cos(x1.*(theta)).*((-1i).*cos(x4.*(theta
        )) + G3.*Z0.*sin(x4.*(theta))) + sin(x1.*(theta)).*((G1 + G2).*
        Z0.*cos(x4.*(theta)) + 1i.*(1 + G1.*G3.*Z0^2 + G2.*G3.*Z0^2).*
        sin(x4.*(theta))))) + sin(x2.*(theta)).*(cos(x1.*(theta)).*(cos(
        x3.*(theta)).*((-1i).*cos(x4.*(theta)) + (G2 + G3).*Z0.*sin(x4
        .*(theta))) + 1i.*sin(x3.*(theta)).*((-1i).*G2.*Z0.*cos(x4.*(
        theta)) + (1 + G2.*G3.*Z0^2).*sin(x4.*(theta)))) + sin(x1.*(
        theta)).*((-sin(x3.*(theta))).*((-1i).*(1 + G1.*G2.*Z0^2).*cos(
        x4.*(theta)) + Z0.*(G1 + G3 + G1.*G2.*G3.*Z0^2).*sin(x4.*(theta)
        )) +
```

```
1i.*cos(x3.*(theta)).*((-1i).*G1.*Z0.*cos(x4.*(theta)) + (1 + G1.*(G2 +
    G3).*Z0^2).*sin(x4.*(theta))))))./ (cos(x1.*(theta)).*(cos(x2.*(theta
    )).*((-cos(x3.*(theta))).*(cos(x4.*(theta)) + 1i.*G4.*Z0.*sin(x4.*(
    theta))) + sin(x3.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x4.*(theta)) +
     (1 + G3.*G4.*Z0^2).*sin(x4.*(theta)))) + sin(x2.*(theta)).*(cos(x3.*(
    theta)).*((-1i).*(G2 + G3 + G4).*Z0.*cos(x4.*(theta)) + (1 + G2.*G4.*
    Z0^2 + G3.*G4.*Z0^2).*sin(x4.*(theta))) + sin(x3.*(theta)).*((1 + G2
    .*(G3 + G4).*Z0^2).*cos(x4.*(theta)) + 1i.*Z0.*(G2 + G4 + G2.*G3.*G4.*
    Z0^2).*sin(x4.*(theta))))) + sin(x1.*(theta)).*(cos(x2.*(theta)).*(cos
    (x3.*(theta)).*((-1i).*(G1 + G2 + G3 + G4).*Z0.*cos(x4.*(theta)) + (1
    + G1.*G4.*Z0^2 + G2.*G4.*Z0^2 + G3.*G4.*Z0^2).*sin(x4.*(theta))) + sin
    (x3.*(theta)).*((1 + G1.*(G3 + G4).*Z0^2 + G2.*(G3 + G4).*Z0^2).*cos(
    x4.*(theta)) + 1i.*Z0.*(G1 + G2 + G4 +
G1.*G3.*G4.*Z0^2 + G2.*G3.*G4.*Z0^2).*sin(x4.*(theta)))) + sin(x2.*(theta
    )).*(cos(x3.*(theta)).*((1 + G1.*(G2 + G3 + G4).*Z0^2).*cos(x4.*(theta
    )) + 1i.*Z0.*(G1 + G4 + G1.*G2.*G4.*Z0^2 + G1.*G3.*G4.*Z0^2).*sin(x4
    .*(theta))) - sin(x3.*(theta)).*((-1i).*Z0.*(G3 + G4 + G1.*(1 + G2.*(
    G3 + G4).*Z0^2)).*cos(x4.*(theta)) + (1 + G3.*G4.*Z0^2 + G1.*Z0^2.*(G2
     + G4 + G2.*G3.*G4.*Z0^2)).*sin(x4.*(theta)))))))./ (ZS + (Z0.*(cos(x2
    .*(theta)).*(cos(x3.*(theta)).*((-1i).*cos(x4.*(theta)).*sin(x1.*(
    theta)) + ((-1i).*cos(x1.*(theta)) + (G1 + G2 + G3).*Z0.*sin(x1.*(
    theta))).*sin(x4.*(theta))) + sin(x3.*(theta)).*(cos(x1.*(theta))
    .*((-1i).*cos(x4.*(theta)) + G3.*Z0.*sin(x4.*(theta))) + sin(x1.*(
    theta)).*((G1 + G2).*Z0.*cos(x4.*(theta)) + 1i.*(1 + G1.*G3.*Z0^2 + G2
    .*G3.*Z0^2).*sin(x4.*(theta))))) +
sin(x2.*(theta)).*(cos(x1.*(theta)).*(cos(x3.*(theta)).*((-1i).*cos(x4.*(
    theta)) + (G2 + G3).*Z0.*sin(x4.*(theta))) + 1i.*sin(x3.*(theta))
    .*((-1i).*G2.*Z0.*cos(x4.*(theta)) + (1 + G2.*G3.*Z0^2).*sin(x4.*(
    theta)))) + sin(x1.*(theta)).*((-sin(x3.*(theta))).*((-1i).*(1 + G1.*
    G2.*Z0^2).*cos(x4.*(theta)) + Z0.*(G1 + G3 + G1.*G2.*G3.*Z0^2).*sin(x4
    .*(theta))) + 1i.*cos(x3.*(theta)).*((-1i).*G1.*Z0.*cos(x4.*(theta)) +
     (1 + G1.*(G2 + G3).*Z0^2).*sin(x4.*(theta)))))))./ (cos(x1.*(theta))
    .*(cos(x2.*(theta)).*((-cos(x3.*(theta))).*(cos(x4.*(theta)) + 1i.*G4
    .*Z0.*sin(x4.*(theta))) + sin(x3.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos
    (x4.*(theta)) + (1 + G3.*G4.*Z0^2).*sin(x4.*(theta)))) + sin(x2.*(
    theta)).*(cos(x3.*(theta)).*((-1i).*(G2 + G3 + G4).*Z0.*cos(x4.*(theta
    )) + (1 + G2.*G4.*Z0^2 + G3.*G4.*Z0^2).*sin(x4.*(theta))) + sin(x3.*(
    theta)).*((1 + G2.*(G3 + G4).*Z0^2).*cos(x4.*(theta)) +
1i.*Z0.*(G2 + G4 + G2.*G3.*G4.*Z0^2).*sin(x4.*(theta))))) + sin(x1.*(
    theta)).*(cos(x2.*(theta)).*(cos(x3.*(theta)).*((-1i).*(G1 + G2 + G3 +
     G4).*Z0.*cos(x4.*(theta)) + (1 + G1.*G4.*Z0^2 + G2.*G4.*Z0^2 + G3.*G4
    .*Z0^2).*sin(x4.*(theta))) + sin(x3.*(theta)).*((1 + G1.*(G3 + G4).*Z0
    ^2 + G2.*(G3 + G4).*Z0^2).*cos(x4.*(theta)) + 1i.*Z0.*(G1 + G2 + G4 +
    G1.*G3.*G4.*Z0^2 + G2.*G3.*G4.*Z0^2).*sin(x4.*(theta)))) + sin(x2.*(
    theta)).*(cos(x3.*(theta)).*((1 + G1.*(G2 + G3 + G4).*Z0^2).*cos(x4.*(
```

```
    theta)) + 1i.*Z0.*(G1 + G4 + G1.*G2.*G4.*Z0^2 + G1.*G3.*G4.*Z0^2).*sin
    (x4.*(theta))) - sin(x3.*(theta)).*((-1i).*Z0.*(G3 + G4 + G1.*(1 + G2
    .*(G3 + G4).*Z0^2)).*cos(x4.*(theta)) + (1 + G3.*G4.*Z0^2 + G1.*Z0
    ^2.*(G2 + G4 + G2.*G3.*G4.*Z0^2)).*sin(x4.*(theta))))))));
   case 5
      G1 = individual(1);
      G2 = individual(2);
      G3 = individual(3);
      G4 = individual(4);
      G5 = individual(5);
      x1 = individual(6);
      x2 = individual(7);
      x3 = individual(8);
      x4 = individual(9);
      x5 = individual(10);
      gamma = mag2db(abs((-ZS + (1i.*Z0.*(cos(x1.*theta).*(cos(x2.*theta)
          .*(cos(x4.*theta).*(cos(x5.*theta).*sin(x3.*theta) + (cos(x3.*
          theta) + 1i.*(G3 + G4).*Z0.*sin(x3.*theta)).*sin(x5.*theta)) +
          sin(x4.*theta).*(cos(x3.*theta).*(cos(x5.*theta) + 1i.*G4.*Z0.*
          sin(x5.*theta)) - sin(x3.*theta).*((-1i).*G3.*Z0.*cos(x5.*theta)
           + (1 + G3.*G4.*Z0^2).*sin(x5.*theta)))) + sin(x2.*theta).*(cos(
          x3.*theta).*(cos(x4.*theta).*(cos(x5.*theta) + 1i.*(G2 + G3 + G4
          ).*Z0.*sin(x5.*theta)) - sin(x4.*theta).*((-1i).*(G2 + G3).*Z0.*
          cos(x5.*theta) + (1 + G2.*G4.*Z0^2 + G3.*G4.*Z0^2).*sin(x5.*
          theta))) + sin(x3.*theta).*((-sin(x4.*theta)).*((1 + G2.*G3.*Z0
          ^2).*cos(x5.*theta) + 1i.*Z0.*(G2 + G4 + G2.*G3.*G4.*Z0^2).*sin(
          x5.*theta)) - cos(x4.*theta).*((-1i).*G2.*Z0.*cos(x5.*theta) +
          (1 + G2.*(G3 + G4).*Z0^2).*sin(x5.*theta))))) + sin(x1.*theta)
          .*((-sin(x2.*theta)).*(cos(x4.*theta).*(cos(x5.*theta).*sin(x3.*
          theta) + (cos(x3.*theta) +
1i.*(G3 + G4).*Z0.*sin(x3.*theta)).*sin(x5.*theta)) + sin(x4.*theta).*(
    cos(x3.*theta).*(cos(x5.*theta) + 1i.*G4.*Z0.*sin(x5.*theta)) - sin(x3
    .*theta).*((-1i).*G3.*Z0.*cos(x5.*theta) + (1 + G3.*G4.*Z0^2).*sin(x5
    .*theta)))) + cos(x2.*theta).*(cos(x3.*theta).*(cos(x4.*theta).*(cos(
    x5.*theta) + 1i.*(G2 + G3 + G4).*Z0.*sin(x5.*theta)) - sin(x4.*theta)
    .*((-1i).*(G2 + G3).*Z0.*cos(x5.*theta) + (1 + G2.*G4.*Z0^2 + G3.*G4.*
    Z0^2).*sin(x5.*theta))) + sin(x3.*theta).*((-sin(x4.*theta)).*((1 + G2
    .*G3.*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G2 + G4 + G2.*G3.*G4.*Z0^2).*
    sin(x5.*theta)) - cos(x4.*theta).*((-1i).*G2.*Z0.*cos(x5.*theta) + (1
    + G2.*(G3 + G4).*Z0^2).*sin(x5.*theta)))) + 1i.*G1.*Z0.*(cos(x2.*theta
    ).*(cos(x4.*theta).*(cos(x5.*theta).*sin(x3.*theta) + (cos(x3.*theta)
    + 1i.*(G3 + G4).*Z0.*sin(x3.*theta)).*sin(x5.*theta)) + sin(x4.*theta)
    .*(cos(x3.*theta).*(cos(x5.*theta) + 1i.*G4.*Z0.*sin(x5.*theta)) - sin
    (x3.*theta).*((-1i).*G3.*Z0.*cos(x5.*theta) + (1 +
```

```
G3.*G4.*Z0^2).*sin(x5.*theta)))) + sin(x2.*theta).*(cos(x3.*theta).*(cos(
    x4.*theta).*(cos(x5.*theta) + 1i.*(G2 + G3 + G4).*Z0.*sin(x5.*theta))
    - sin(x4.*theta).*((-1i).*(G2 + G3).*Z0.*cos(x5.*theta) + (1 + G2.*G4
    .*Z0^2 + G3.*G4.*Z0^2).*sin(x5.*theta))) + sin(x3.*theta).*((-sin(x4.*
    theta)).*((1 + G2.*G3.*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G2 + G4 + G2.*
    G3.*G4.*Z0^2).*sin(x5.*theta)) - cos(x4.*theta).*((-1i).*G2.*Z0.*cos(
    x5.*theta) + (1 + G2.*(G3 + G4).*Z0^2).*sin(x5.*theta))))))))./(cos(x1
    .*theta).*(cos(x2.*theta).*(cos(x3.*theta).*(cos(x4.*theta).*(cos(x5.*
    theta) + 1i.*G5.*Z0.*sin(x5.*theta)) - sin(x4.*theta).*((-1i).*(G4 +
    G5).*Z0.*cos(x5.*theta) + (1 + G4.*G5.*Z0^2).*sin(x5.*theta))) + 1i.*
    sin(x3.*theta).*(cos(x4.*theta).*((G3 + G4 + G5).*Z0.*cos(x5.*theta) +
     1i.*(1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x5.*theta)) + 1i.*sin(x4.*
    theta).*((1 + G3.*(G4 + G5).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G3 + G5
    +
G3.*G4.*G5.*Z0^2).*sin(x5.*theta)))) + 1i.*sin(x2.*theta).*(cos(x3.*theta
    ).*(cos(x4.*theta).*((G2 + G3 + G4 + G5).*Z0.*cos(x5.*theta) + 1i.*(1
    + G2.*G5.*Z0^2 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x5.*theta)) + 1i.*
    sin(x4.*theta).*((1 + G2.*(G4 + G5).*Z0^2 + G3.*(G4 + G5).*Z0^2).*cos(
    x5.*theta) + 1i.*Z0.*(G2 + G3 + G5 + G2.*G4.*G5.*Z0^2 + G3.*G4.*G5.*Z0
    ^2).*sin(x5.*theta))) + 1i.*sin(x3.*theta).*(cos(x4.*theta).*((1 + G2
    .*(G3 + G4 + G5).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G2 + G5 + G2.*G3.*
    G5.*Z0^2 + G2.*G4.*G5.*Z0^2).*sin(x5.*theta)) - sin(x4.*theta).*((-1i)
    .*Z0.*(G4 + G5 + G2.*(1 + G3.*(G4 + G5).*Z0^2)).*cos(x5.*theta) + (1 +
     G4.*G5.*Z0^2 + G2.*Z0^2.*(G3 + G5 + G3.*G4.*G5.*Z0^2)).*sin(x5.*theta
    ))))) + 1i.*sin(x1.*theta).*(cos(x2.*theta).*(cos(x3.*theta).*(cos(x4
    .*theta).*((G1 + G2 + G3 + G4 +
G5).*Z0.*cos(x5.*theta) + 1i.*(1 + G1.*G5.*Z0^2 + G2.*G5.*Z0^2 + G3.*G5.*
    Z0^2 + G4.*G5.*Z0^2).*sin(x5.*theta)) + 1i.*sin(x4.*theta).*((1 + G3.*
    G4.*Z0^2 + G3.*G5.*Z0^2 + G1.*(G4 + G5).*Z0^2 + G2.*(G4 + G5).*Z0^2).*
    cos(x5.*theta) + 1i.*Z0.*(G1 + G2 + G3 + G5 + G1.*G4.*G5.*Z0^2 + G2.*
    G4.*G5.*Z0^2 + G3.*G4.*G5.*Z0^2).*sin(x5.*theta))) + 1i.*sin(x3.*theta
    ).*(cos(x4.*theta).*((1 + G1.*(G3 + G4 + G5).*Z0^2 + G2.*(G3 + G4 + G5
    ).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G1 + G2 + G5 + G1.*G3.*G5.*Z0^2 +
    G2.*G3.*G5.*Z0^2 + G1.*G4.*G5.*Z0^2 + G2.*G4.*G5.*Z0^2).*sin(x5.*theta
    )) - sin(x4.*theta).*((-1i).*Z0.*(G2 + G4 + G5 + G2.*G3.*G4.*Z0^2 + G2
    .*G3.*G5.*Z0^2 + G1.*(1 + G3.*(G4 + G5).*Z0^2)).*cos(x5.*theta) + (1 +
     G4.*G5.*Z0^2 + G1.*Z0^2.*(G3 + G5 + G3.*G4.*G5.*Z0^2) + G2.*Z0^2.*(G3
     + G5 + G3.*G4.*G5.*Z0^2)).*sin(x5.*theta)))) - sin(x2.*theta).*(cos(
    x3.*theta).*((-1i).*cos(x4.*theta).*((1 + G1.*(G2 + G3 + G4 +
G5).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G5 + G1.*(1 + G2.*G5.*Z0^2 + G3.*G5
    .*Z0^2 + G4.*G5.*Z0^2)).*sin(x5.*theta)) + sin(x4.*theta).*(Z0.*(G4 +
    G5 + G1.*(1 + G2.*(G4 + G5).*Z0^2 + G3.*(G4 + G5).*Z0^2)).*cos(x5.*
    theta) + 1i.*(1 + G4.*G5.*Z0^2 + G1.*Z0^2.*(G2 + G3 + G5 + G2.*G4.*G5
    .*Z0^2 + G3.*G4.*G5.*Z0^2)).*sin(x5.*theta))) + sin(x3.*theta).*(cos(
    x4.*theta).*(Z0.*(G3 + G4 + G5 + G1.*(1 + G2.*(G3 + G4 + G5).*Z0^2)).*
```

```
cos(x5.*theta) + 1i.*(1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2 + G1.*Z0^2.*(G2
+ G5 + G2.*G3.*G5.*Z0^2 + G2.*G4.*G5.*Z0^2)).*sin(x5.*theta)) + 1i.*
sin(x4.*theta).*((1 + G3.*(G4 + G5).*Z0^2 + G1.*Z0^2.*(G4 + G5 + G2
.*(1 + G3.*(G4 + G5).*Z0^2))).*cos(x5.*theta) + 1i.*Z0.*(G3 + G5 + G3
.*G4.*G5.*Z0^2 + G1.*(1 + G4.*G5.*Z0^2 + G2.*Z0^2.*(G3 + G5 + G3.*G4.*
G5.*Z0^2))).*sin(x5.*theta))))))./(ZS + (1i.*Z0.*(cos(x1.*theta).*(
cos(x2.*theta).*(cos(x4.*theta).*(cos(x5.*theta).*sin(x3.*theta) +
(cos(x3.*theta) + 1i.*(G3 + G4).*Z0.*sin(x3.*theta)).*sin(x5.*theta)) +
sin(x4.*theta).*(cos(x3.*theta).*(cos(x5.*theta) + 1i.*G4.*Z0.*sin(x5
.*theta)) - sin(x3.*theta).*((-1i).*G3.*Z0.*cos(x5.*theta) + (1 + G3.*
G4.*Z0^2).*sin(x5.*theta)))) + sin(x2.*theta).*(cos(x3.*theta).*(cos(
x4.*theta).*(cos(x5.*theta) + 1i.*(G2 + G3 + G4).*Z0.*sin(x5.*theta))
- sin(x4.*theta).*((-1i).*(G2 + G3).*Z0.*cos(x5.*theta) + (1 + G2.*G4
.*Z0^2 + G3.*G4.*Z0^2).*sin(x5.*theta))) + sin(x3.*theta).*((-sin(x4.*
theta)).*((1 + G2.*G3.*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G2 + G4 + G2.*
G3.*G4.*Z0^2).*sin(x5.*theta)) - cos(x4.*theta).*((-1i).*G2.*Z0.*cos(
x5.*theta) + (1 + G2.*(G3 + G4).*Z0^2).*sin(x5.*theta))))) + sin(x1.*
theta).*((-sin(x2.*theta)).*(cos(x4.*theta).*(cos(x5.*theta).*sin(x3.*
theta) + (cos(x3.*theta) + 1i.*(G3 + G4).*Z0.*sin(x3.*theta)).*sin(x5
.*theta)) + sin(x4.*theta).*(cos(x3.*theta).*(cos(x5.*theta) +
1i.*G4.*Z0.*sin(x5.*theta)) - sin(x3.*theta).*((-1i).*G3.*Z0.*cos(x5.*
theta) + (1 + G3.*G4.*Z0^2).*sin(x5.*theta)))) + cos(x2.*theta).*(cos(
x3.*theta).*(cos(x4.*theta).*(cos(x5.*theta) + 1i.*(G2 + G3 + G4).*Z0
.*sin(x5.*theta)) - sin(x4.*theta).*((-1i).*(G2 + G3).*Z0.*cos(x5.*
theta) + (1 + G2.*G4.*Z0^2 + G3.*G4.*Z0^2).*sin(x5.*theta))) + sin(x3
.*theta).*((-sin(x4.*theta)).*((1 + G2.*G3.*Z0^2).*cos(x5.*theta) + 1i
.*Z0.*(G2 + G4 + G2.*G3.*G4.*Z0^2).*sin(x5.*theta)) - cos(x4.*theta)
.*((-1i).*G2.*Z0.*cos(x5.*theta) + (1 + G2.*(G3 + G4).*Z0^2).*sin(x5.*
theta)))) + 1i.*G1.*Z0.*(cos(x2.*theta).*(cos(x4.*theta).*(cos(x5.*
theta).*sin(x3.*theta) + (cos(x3.*theta) + 1i.*(G3 + G4).*Z0.*sin(x3.*
theta)).*sin(x5.*theta)) + sin(x4.*theta).*(cos(x3.*theta).*(cos(x5.*
theta) + 1i.*G4.*Z0.*sin(x5.*theta)) - sin(x3.*theta).*((-1i).*G3.*Z0
.*cos(x5.*theta) + (1 + G3.*G4.*Z0^2).*sin(x5.*theta)))) + sin(x2.*
theta).*(cos(x3.*theta).*(cos(x4.*theta).*(cos(x5.*theta) +
1i.*(G2 + G3 + G4).*Z0.*sin(x5.*theta)) - sin(x4.*theta).*((-1i).*(G2 +
G3).*Z0.*cos(x5.*theta) + (1 + G2.*G4.*Z0^2 + G3.*G4.*Z0^2).*sin(x5.*
theta))) + sin(x3.*theta).*((-sin(x4.*theta)).*((1 + G2.*G3.*Z0^2).*
cos(x5.*theta) + 1i.*Z0.*(G2 + G4 + G2.*G3.*G4.*Z0^2).*sin(x5.*theta))
 - cos(x4.*theta).*((-1i).*G2.*Z0.*cos(x5.*theta) + (1 + G2.*(G3 + G4)
.*Z0^2).*sin(x5.*theta)))))))./(cos(x1.*theta).*(cos(x2.*theta).*(cos
(x3.*theta).*(cos(x4.*theta).*(cos(x5.*theta) + 1i.*G5.*Z0.*sin(x5.*
theta)) - sin(x4.*theta).*((-1i).*(G4 + G5).*Z0.*cos(x5.*theta) + (1 +
 G4.*G5.*Z0^2).*sin(x5.*theta))) + 1i.*sin(x3.*theta).*(cos(x4.*theta)
.*((G3 + G4 + G5).*Z0.*cos(x5.*theta) + 1i.*(1 + G3.*G5.*Z0^2 + G4.*G5
.*Z0^2).*sin(x5.*theta)) + 1i.*sin(x4.*theta).*((1 + G3.*(G4 + G5).*Z0
```

```
    ^2).*cos(x5.*theta) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x5.*
    theta)))) + 1i.*sin(x2.*theta).*(cos(x3.*theta).*(cos(x4.*theta).*((G2
     + G3 + G4 + G5).*Z0.*cos(x5.*theta) + 1i.*(1 + G2.*G5.*Z0^2 + G3.*G5
    .*Z0^2 +
G4.*G5.*Z0^2).*sin(x5.*theta)) + 1i.*sin(x4.*theta).*((1 + G2.*(G4 + G5)
    .*Z0^2 + G3.*(G4 + G5).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G2 + G3 + G5
    + G2.*G4.*G5.*Z0^2 + G3.*G4.*G5.*Z0^2).*sin(x5.*theta))) + 1i.*sin(x3
    .*theta).*(cos(x4.*theta).*((1 + G2.*(G3 + G4 + G5).*Z0^2).*cos(x5.*
    theta) + 1i.*Z0.*(G2 + G5 + G2.*G3.*G5.*Z0^2 + G2.*G4.*G5.*Z0^2).*sin(
    x5.*theta)) - sin(x4.*theta).*((-1i).*Z0.*(G4 + G5 + G2.*(1 + G3.*(G4
    + G5).*Z0^2)).*cos(x5.*theta) + (1 + G4.*G5.*Z0^2 + G2.*Z0^2.*(G3 + G5
     + G3.*G4.*G5.*Z0^2)).*sin(x5.*theta))))) + 1i.*sin(x1.*theta).*(cos(
    x2.*theta).*(cos(x3.*theta).*(cos(x4.*theta).*((G1 + G2 + G3 + G4 + G5
    ).*Z0.*cos(x5.*theta) + 1i.*(1 + G1.*G5.*Z0^2 + G2.*G5.*Z0^2 + G3.*G5
    .*Z0^2 + G4.*G5.*Z0^2).*sin(x5.*theta)) + 1i.*sin(x4.*theta).*((1 + G3
    .*G4.*Z0^2 + G3.*G5.*Z0^2 + G1.*(G4 +
G5).*Z0^2 + G2.*(G4 + G5).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G1 + G2 + G3
    + G5 + G1.*G4.*G5.*Z0^2 + G2.*G4.*G5.*Z0^2 + G3.*G4.*G5.*Z0^2).*sin(x5
    .*theta))) + 1i.*sin(x3.*theta).*(cos(x4.*theta).*((1 + G1.*(G3 + G4 +
     G5).*Z0^2 + G2.*(G3 + G4 + G5).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G1 +
     G2 + G5 + G1.*G3.*G5.*Z0^2 + G2.*G3.*G5.*Z0^2 + G1.*G4.*G5.*Z0^2 + G2
    .*G4.*G5.*Z0^2).*sin(x5.*theta)) - sin(x4.*theta).*((-1i).*Z0.*(G2 +
    G4 + G5 + G2.*G3.*G4.*Z0^2 + G2.*G3.*G5.*Z0^2 + G1.*(1 + G3.*(G4 + G5)
    .*Z0^2)).*cos(x5.*theta) + (1 + G4.*G5.*Z0^2 + G1.*Z0^2.*(G3 + G5 + G3
    .*G4.*G5.*Z0^2) + G2.*Z0^2.*(G3 + G5 + G3.*G4.*G5.*Z0^2)).*sin(x5.*
    theta)))) - sin(x2.*theta).*(cos(x3.*theta).*((-1i).*cos(x4.*theta)
    .*((1 + G1.*(G2 + G3 + G4 + G5).*Z0^2).*cos(x5.*theta) + 1i.*Z0.*(G5 +
     G1.*(1 + G2.*G5.*Z0^2 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2)).*sin(x5.*theta)
    ) + sin(x4.*theta).*(Z0.*(G4 + G5 + G1.*(1 + G2.*(G4 + G5).*Z0^2 +
G3.*(G4 + G5).*Z0^2)).*cos(x5.*theta) + 1i.*(1 + G4.*G5.*Z0^2 + G1.*Z0
    ^2.*(G2 + G3 + G5 + G2.*G4.*G5.*Z0^2 + G3.*G4.*G5.*Z0^2)).*sin(x5.*
    theta))) + sin(x3.*theta).*(cos(x4.*theta).*(Z0.*(G3 + G4 + G5 + G1
    .*(1 + G2.*(G3 + G4 + G5).*Z0^2)).*cos(x5.*theta) + 1i.*(1 + G3.*G5.*
    Z0^2 + G4.*G5.*Z0^2 + G1.*Z0^2.*(G2 + G5 + G2.*G3.*G5.*Z0^2 + G2.*G4.*
    G5.*Z0^2)).*sin(x5.*theta)) + 1i.*sin(x4.*theta).*((1 + G3.*(G4 + G5)
    .*Z0^2 + G1.*Z0^2.*(G4 + G5 + G2.*(1 + G3.*(G4 + G5).*Z0^2))).*cos(x5
    .*theta) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2 + G1.*(1 + G4.*G5.*Z0^2
     + G2.*Z0^2.*(G3 + G5 + G3.*G4.*G5.*Z0^2))).*sin(x5.*theta))))))));
   case 6
      G1 = individual(1);
      G2 = individual(2);
      G3 = individual(3);
      G4 = individual(4);
      G5 = individual(5);
      G6 = individual(6);
```

```
        x1 = individual(7);
        x2 = individual(8);
        x3 = individual(9);
        x4 = individual(10);
        x5 = individual(11);
        x6 = individual(12);
        gamma = mag2db(abs((-ZS + (1i.*Z0.*(cos(x1.*(theta)).*(cos(x2.*(
            theta)).*(cos(x3.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta))
            .*sin(x4.*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*sin(
            x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4.*(
            theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin
            (x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*
            Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(theta))
            .*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0
            .*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*
            cos(x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(
            theta)))) + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4
            .*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2)
            .*sin(x6.*(theta))) -
cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5).*
    Z0^2).*sin(x6.*(theta)))))) + sin(x2.*(theta)).*((-sin(x3.*(theta)))
    .*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(
    theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) +
    sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*
    sin(x6.*(theta))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta))
     + (1 + G4.*G5.*Z0^2).*sin(x6.*(theta))))) + cos(x3.*(theta)).*(cos(x4
    .*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5)
    .*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos
    (x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta))))
    + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6
    .*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) -
     cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 +
    G5).*Z0^2).*sin(x6.*(theta))))) + 1i.*G2.*Z0.*(cos(x3.*(theta)).*(cos(
    x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta))
    + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) +
sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(
    x6.*(theta))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) +
    (1 + G4.*G5.*Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(
    theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0
    .*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6
    .*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) +
    sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(
    theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) -
    cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5
    ).*Z0^2).*sin(x6.*(theta)))))))) + sin(x1.*(theta)).*((-sin(x2.*(theta
```

```
    ))).*(cos(x3.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4
    .*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta)))
    .*sin(x6.*(theta))) +
sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(
    x6.*(theta))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) +
    (1 + G4.*G5.*Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(
    theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0
    .*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6
    .*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) +
    sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(
    theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) -
    cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5
    ).*Z0^2).*sin(x6.*(theta)))))) + cos(x2.*(theta)).*((-sin(x3.*(theta))
    ).*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(
    theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) +
    sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*
    sin(x6.*(theta))) -
sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*
    sin(x6.*(theta))))) + cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(
    theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0.*sin(x6.*(theta)))
     - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6.*(theta)) + (1 + G3
    .*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-
    sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3
     + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i
    ).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(
    theta))))) + 1i.*G2.*Z0.*(cos(x3.*(theta)).*(cos(x5.*(theta)).*(cos(x6
    .*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*
    sin(x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4.*(
    theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin(x4.*(
    theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*sin(x6
    .*(theta))))) +
sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta))
     + 1i.*(G3 + G4 + G5).*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i
    ).*(G3 + G4).*Z0.*cos(x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2)
    .*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 +
    G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2)
    .*sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta
    )) + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(theta))))))) + 1i.*G1.*Z0.*(
    cos(x2.*(theta)).*(cos(x3.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(
    theta)).*sin(x4.*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*sin
    (x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4.*(theta)
    ).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin(x4.*(theta)
    ).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*sin(x6.*(
    theta))))) + sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*(
    cos(x6.*(theta)) + 1i.*(G3 + G4 +
```

```
G5).*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos
    (x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta))))
    + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6
    .*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) -
     cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 +
    G5).*Z0^2).*sin(x6.*(theta)))))) + sin(x2.*(theta)).*((-sin(x3.*(theta
    ))).*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4
    .*(theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) +
     sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*
    sin(x6.*(theta))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta))
     + (1 + G4.*G5.*Z0^2).*sin(x6.*(theta))))) + cos(x3.*(theta)).*(cos(x4
    .*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5)
    .*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos
    (x6.*(theta)) + (1 +
G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-
    sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3
     + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i
    ).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(
    theta))))) + 1i.*G2.*Z0.*(cos(x3.*(theta)).*(cos(x5.*(theta)).*(cos(x6
    .*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*
    sin(x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4.*(
    theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin(x4.*(
    theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*sin(x6
    .*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta))
    .*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0.*sin(x6.*(theta))) - sin(
    x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6.*(theta)) + (1 + G3.*G5.*
    Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-sin(x5
    .*(theta))).*((1 +
G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*
    sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta))
     + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(theta)))))))))./(cos(x1.*(
    theta)).*(cos(x2.*(theta)).*(cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos
    (x5.*(theta)).*(cos(x6.*(theta)) + 1i.*G6.*Z0.*sin(x6.*(theta))) - sin
    (x5.*(theta)).*((-1i).*(G5 + G6).*Z0.*cos(x6.*(theta)) + (1 + G5.*G6.*
    Z0^2).*sin(x6.*(theta)))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta))
    .*((G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*
    G6.*Z0^2).*sin(x6.*(theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*(G5 +
    G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2).*
    sin(x6.*(theta))))) + 1i.*sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5
    .*(theta)).*((G3 + G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G3.*
    G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) + 1i.*sin(
    x5.*(theta)).*((1 +
G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(
    G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta)))
    ) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((1 + G3.*(G4 + G5 + G6)
```

```
    .*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*
    G5.*G6.*Z0^2).*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*Z0.*(G5 +
     G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2)).*cos(x6.*(theta)) + (1 + G5.*G6.*
    Z0^2 + G3.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta))))) +
     1i.*sin(x2.*(theta)).*(cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5
    .*(theta)).*((G2 + G3 + G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 +
     G2.*G6.*Z0^2 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(
    theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*G5.*Z0^2 + G4.*G6.*Z0^2 +
    G2.*(G5 + G6).*Z0^2 + G3.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0
    .*(G2 + G3 + G4 + G6 +
G2.*G5.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta))
    )) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((1 + G2.*(G4 + G5 + G6)
    .*Z0^2 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 +
    G3 + G6 + G2.*G4.*G6.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*G5.*G6.*Z0^2 + G3
    .*G5.*G6.*Z0^2).*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*Z0.*(G3
     + G5 + G6 + G3.*G4.*G5.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*(1 + G4.*(G5 +
    G6).*Z0^2)).*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G4 + G6
     + G4.*G5.*G6.*Z0^2) + G3.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2)).*sin(x6
    .*(theta))))) - sin(x3.*(theta)).*(cos(x4.*(theta)).*((-1i).*cos(x5.*(
    theta)).*((1 + G2.*(G3 + G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*
    Z0.*(G6 + G2.*(1 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2)).*sin(
    x6.*(theta))) + sin(x5.*(theta)).*(Z0.*(G5 + G6 + G2.*(1 + G3.*(G5 +
    G6).*Z0^2 +
G4.*(G5 + G6).*Z0^2)).*cos(x6.*(theta)) + 1i.*(1 + G5.*G6.*Z0^2 + G2.*Z0
    ^2.*(G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(
    theta)))) + sin(x4.*(theta)).*(cos(x5.*(theta)).*(Z0.*(G4 + G5 + G6 +
    G2.*(1 + G3.*(G4 + G5 + G6).*Z0^2)).*cos(x6.*(theta)) + 1i.*(1 + G4.*
    G6.*Z0^2 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*
    G5.*G6.*Z0^2)).*sin(x6.*(theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*(
    G5 + G6).*Z0^2 + G2.*Z0^2.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2)))
    .*cos(x6.*(theta)) + 1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2 + G2.*(1 + G5
    .*G6.*Z0^2 + G3.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2))).*sin(x6.*(theta)
    )))))) + 1i.*Z0.*sin(x1.*(theta)).*((1./Z0).*(sin(x2.*(theta)).*(cos(
    x3.*(theta)).*(1i.*cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(
    theta)) + 1i.*G6.*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(
    G5 + G6).*Z0.*cos(x6.*(theta)) + (1 +
G5.*G6.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-1i).*cos(x5.*(
    theta)).*((-1i).*(G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + (1 + G4.*G6.*
    Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) + sin(x5.*(theta)).*((-1i).*(1
     + G4.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + Z0.*(G4 + G6 + G4.*G5.*G6
    .*Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*((-1i).*cos(x4.*(
    theta)).*(cos(x5.*(theta)).*((-1i).*(G3 + G4 + G5 + G6).*Z0.*cos(x6.*(
    theta)) + (1 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(
    theta))) + sin(x5.*(theta)).*((1 + G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6)
    .*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 +
```

```
 G4.*G5.*G6.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(theta)).*(cos(x5.*(
theta)).*((-1i).*(1 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + Z0
.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2).*sin(x6.*(theta))) +
 sin(x5.*(theta)).*(Z0.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2)).*cos
(x6.*(theta)) + 1i.*(1 + G5.*G6.*Z0^2 + G3.*Z0^2.*(G4 + G6 +
G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta)))))) + (1./Z0).*(cos(x2.*(theta)).*(
cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*((G2 + G3 + G4
 + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G2.*G6.*Z0^2 + G3.*G6.*Z0
^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) + 1i.*sin(x5.*(
theta)).*((1 + G4.*G5.*Z0^2 + G4.*G6.*Z0^2 + G2.*(G5 + G6).*Z0^2 + G3
.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 + G3 + G4 + G6 + G2
.*G5.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta)
))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((1 + G2.*(G4 + G5 + G6
).*Z0^2 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 +
G3 + G6 + G2.*G4.*G6.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*G5.*G6.*Z0^2 + G3
.*G5.*G6.*Z0^2).*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*Z0.*(G3
 + G5 + G6 + G3.*G4.*G5.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*(1 + G4.*(G5 +
G6).*Z0^2)).*cos(x6.*(theta)) + (1 +
G5.*G6.*Z0^2 + G2.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2) + G3.*Z0^2.*(G4 +
G6 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta))))) - sin(x3.*(theta)).*(cos(
x4.*(theta)).*((-1i).*cos(x5.*(theta)).*((1 + G2.*(G3 + G4 + G5 + G6)
.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G6 + G2.*(1 + G3.*G6.*Z0^2 + G4.*
G6.*Z0^2 + G5.*G6.*Z0^2)).*sin(x6.*(theta))) + sin(x5.*(theta)).*(Z0
.*(G5 + G6 + G2.*(1 + G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0^2)).*cos
(x6.*(theta)) + 1i.*(1 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G3 + G4 + G6 + G3.*
G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta)))) + sin(x4.*(theta
)).*(cos(x5.*(theta)).*(Z0.*(G4 + G5 + G6 + G2.*(1 + G3.*(G4 + G5 + G6
).*Z0^2)).*cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2 +
G2.*Z0^2.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2)).*sin(x6.*(
theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*(G5 +
G6).*Z0^2 + G2.*Z0^2.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2))).*cos(x6
.*(theta)) + 1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2 + G2.*(1 + G5.*G6.*Z0
^2 + G3.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2))).*sin(x6.*(theta)))))) +
 G1.*(cos(x2.*(theta)).*(cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5
.*(theta)).*(cos(x6.*(theta)) + 1i.*G6.*Z0.*sin(x6.*(theta))) - sin(x5
.*(theta)).*((-1i).*(G5 + G6).*Z0.*cos(x6.*(theta)) + (1 + G5.*G6.*Z0
^2).*sin(x6.*(theta)))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((
G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*G6.*
Z0^2).*sin(x6.*(theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*(G5 + G6).*
Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2).*sin(x6
.*(theta))))) + 1i.*sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(
theta)).*((G3 + G4 + G5 + G6).*Z0.*cos(x6.*(theta)) +
1i.*(1 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) +
 1i.*sin(x5.*(theta)).*((1 + G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0
^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 + G4
```

```
    .*G5.*G6.*Z0^2).*sin(x6.*(theta)))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(
    theta)).*((1 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(
    G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2).*sin(x6.*(theta))) -
    sin(x5.*(theta)).*((-1i).*Z0.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2)
    ).*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2 + G3.*Z0^2.*(G4 + G6 + G4.*G5
    .*G6.*Z0^2)).*sin(x6.*(theta)))))) + 1i.*sin(x2.*(theta)).*(cos(x3.*(
    theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*((G2 + G3 + G4 + G5 +
    G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G2.*G6.*Z0^2 + G3.*G6.*Z0^2 + G4
    .*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) + 1i.*sin(x5.*(theta))
    .*((1 + G4.*G5.*Z0^2 + G4.*G6.*Z0^2 + G2.*(G5 + G6).*Z0^2 + G3.*(G5 +
    G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 + G3 + G4 + G6 +
G2.*G5.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta))
    )) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((1 + G2.*(G4 + G5 + G6)
    .*Z0^2 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 +
    G3 + G6 + G2.*G4.*G6.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*G5.*G6.*Z0^2 + G3
    .*G5.*G6.*Z0^2).*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*Z0.*(G3
     + G5 + G6 + G3.*G4.*G5.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*(1 + G4.*(G5 +
    G6).*Z0^2)).*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G4 + G6
     + G4.*G5.*G6.*Z0^2) + G3.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2)).*sin(x6
    .*(theta))))) - sin(x3.*(theta)).*(cos(x4.*(theta)).*((-1i).*cos(x5.*(
    theta)).*((1 + G2.*(G3 + G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*
    Z0.*(G6 + G2.*(1 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2)).* sin(
    x6.*(theta))) + sin(x5.*(theta)).*(Z0.*(G5 + G6 + G2.*(1 + G3.*(G5 +
    G6).*Z0^2 +
G4.*(G5 + G6).*Z0^2)).*cos(x6.*(theta)) + 1i.*(1 + G5.*G6.*Z0^2 + G2.*Z0
    ^2.*(G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(
    theta)))) + sin(x4.*(theta)).*(cos(x5.*(theta)).*(Z0.*(G4 + G5 + G6 +
    G2.*(1 + G3.*(G4 + G5 + G6).*Z0^2)).* cos(x6.*(theta)) + 1i.*(1 + G4.*
    G6.*Z0^2 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*
    G5.*G6.*Z0^2)).*sin(x6.*(theta))) + 1i.*sin(x5.*(theta)).* ((1 + G4.*(
    G5 + G6).*Z0^2 + G2.*Z0^2.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2)))
    .*cos(x6.*(theta)) + 1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2 + G2.*(1 + G5
    .*G6.*Z0^2 + G3.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2))).*sin(x6.*(theta)
    ))))))))./ (ZS + (1i.*Z0.*(cos(x1.*(theta)).*(cos(x2.*(theta)).*(cos(
    x3.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) +
     (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(
    theta))) +
sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(
    x6.*(theta))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) +
    (1 + G4.*G5.*Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(
    theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0
    .*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6
    .*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).* sin(x6.*(theta)))) +
    sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(
    theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) -
```

```
    cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5
    ).*Z0^2).*sin(x6.*(theta)))))) + sin(x2.*(theta)).*((-sin(x3.*(theta))
    ).*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(
    theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) +
    sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*
    sin(x6.*(theta))) -
sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*
    sin(x6.*(theta))))) + cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(
    theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0.*sin(x6.*(theta)))
     - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6.*(theta)) + (1 + G3
    .*G5.*Z0^2 + G4.*G5.*Z0^2).* sin(x6.*(theta)))) + sin(x4.*(theta))
    .*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0
    .*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) - cos(x5.*(theta))
    .*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6
    .*(theta))))) + 1i.*G2.*Z0.*(cos(x3.*(theta)).*(cos(x5.*(theta)).*(cos
    (x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*
    Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4
    .*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin(x4
    .*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*sin
    (x6.*(theta))))) +
sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta))
     + 1i.*(G3 + G4 + G5).*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i
    ).*(G3 + G4).*Z0.*cos(x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2)
    .*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 +
    G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2)
    .*sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta
    )) + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(theta))))))) + sin(x1.*(
    theta)).*((-sin(x2.*(theta))).*(cos(x3.*(theta)).*(cos(x5.*(theta)).*(
    cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5)
    .*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4
    .*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin(x4
    .*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*sin
    (x6.*(theta))))) +
sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta))
     + 1i.*(G3 + G4 + G5).*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i
    ).*(G3 + G4).*Z0.*cos(x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2)
    .* sin(x6.*(theta)))) + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 +
    G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2)
    .*sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta
    )) + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(theta)))))) + cos(x2.*(theta)
    ).*((-sin(x3.*(theta))).*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4
    .*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta)))
    .*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(
    theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin(x4.*(theta)).*((-1i).*G4
    .*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*sin(x6.*(theta))))) + cos
```

```
(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta))
 + 1i.*(G3 + G4 +
G5).*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos
(x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).* sin(x6.*(theta))))
 + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6
.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) -
 cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 +
G5).*Z0^2).*sin(x6.*(theta))))) + 1i.*G2.*Z0.*(cos(x3.*(theta)).*(cos(
x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta))
+ 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(
theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(
theta))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 +
G4.*G5.*Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(theta
)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0.*sin
(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6.*(
theta)) + (1 +
G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-
sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3
 + G5 + G3.*G4.*G5.*Z0^2).*sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i
).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(
theta)))))) + 1i.*G1.*Z0.*(cos(x2.*(theta)).*(cos(x3.*(theta)).*(cos(
x5.*(theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta))
+ 1i.*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(
theta)).*(cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(
theta))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 +
G4.*G5.*Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(theta
)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0.*sin
(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6.*(
theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) + sin(
x4.*(theta)).*((-sin(x5.*(theta))).*((1 +
G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).*
sin(x6.*(theta))) - cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta))
 + (1 + G3.*(G4 + G5).*Z0^2).*sin(x6.*(theta)))))) + sin(x2.*(theta))
.*((-sin(x3.*(theta))).*(cos(x5.*(theta)).*(cos(x6.*(theta)).*sin(x4
.*(theta)) + (cos(x4.*(theta)) + 1i.*(G4 + G5).*Z0.*sin(x4.*(theta)))
.*sin(x6.*(theta))) + sin(x5.*(theta)).*(cos(x4.*(theta)).*(cos(x6.*(
theta)) + 1i.*G5.*Z0.*sin(x6.*(theta))) - sin(x4.*(theta)).*((-1i).*G4
.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5.*Z0^2).*sin(x6.*(theta))))) + cos
(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta))
 + 1i.*(G3 + G4 + G5).*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i
).*(G3 + G4).*Z0.*cos(x6.*(theta)) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2)
.*sin(x6.*(theta)))) + sin(x4.*(theta)).*((-sin(x5.*(theta))).*((1 +
G3.*G4.*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2)
.*sin(x6.*(theta))) -
```

```
cos(x5.*(theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 + G5).*
    Z0^2).*sin(x6.*(theta))))) + 1i.*G2.*Z0.*(cos(x3.*(theta)).*(cos(x5.*(
    theta)).*(cos(x6.*(theta)).*sin(x4.*(theta)) + (cos(x4.*(theta)) + 1i
    .*(G4 + G5).*Z0.*sin(x4.*(theta))).*sin(x6.*(theta))) + sin(x5.*(theta
    )).* (cos(x4.*(theta)).*(cos(x6.*(theta)) + 1i.*G5.*Z0.*sin(x6.*(theta
    ))) - sin(x4.*(theta)).*((-1i).*G4.*Z0.*cos(x6.*(theta)) + (1 + G4.*G5
    .*Z0^2).*sin(x6.*(theta))))) + sin(x3.*(theta)).*(cos(x4.*(theta)).*(
    cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*(G3 + G4 + G5).*Z0.*sin(x6
    .*(theta))) - sin(x5.*(theta)).*((-1i).*(G3 + G4).*Z0.*cos(x6.*(theta)
    ) + (1 + G3.*G5.*Z0^2 + G4.*G5.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(
    theta)).*((-sin(x5.*(theta))).*((1 + G3.*G4.*Z0^2).*cos(x6.*(theta)) +
     1i.*Z0.*(G3 + G5 + G3.*G4.*G5.*Z0^2).* sin(x6.*(theta))) - cos(x5.*(
    theta)).*((-1i).*G3.*Z0.*cos(x6.*(theta)) + (1 + G3.*(G4 +
G5).*Z0^2).*sin(x6.*(theta))))))))))./ (cos(x1.*(theta)).*(cos(x2.*(
    theta)).*(cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos
    (x6.*(theta)) + 1i.*G6.*Z0.*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1
    i).*(G5 + G6).*Z0.*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2).* sin(x6.*(
    theta)))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((G4 + G5 + G6).*
    Z0.*cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6
    .*(theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*(G5 + G6).*Z0^2).*cos(x6
    .*(theta)) + 1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta)))))
     + 1i.*sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*((G3 +
    G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G3.*G6.*Z0^2 + G4.*G6.*
    Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) + 1i.*sin(x5.*(theta)).*((1 +
    G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0
    .*(G3 + G4 + G6 +
G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta)))) + 1i.*sin(x4.*(
    theta)).*(cos(x5.*(theta)).*((1 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(
    theta)) + 1i.*Z0.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2).*sin
    (x6.*(theta))) - sin(x5.*(theta)).*((-1i).*Z0.*(G5 + G6 + G3.*(1 + G4
    .*(G5 + G6).*Z0^2)).*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2 + G3.*Z0
    ^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta)))))) + 1i.*sin(x2
    .*(theta)).*(cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta))
    .*((G2 + G3 + G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G2.*G6.*
    Z0^2 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).* sin(x6.*(theta)))
     + 1i.*sin(x5.*(theta)).*((1 + G4.*G5.*Z0^2 + G4.*G6.*Z0^2 + G2.*(G5 +
     G6).*Z0^2 + G3.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 + G3
     + G4 + G6 + G2.*G5.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*
    sin(x6.*(theta)))) +
1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((1 + G2.*(G4 + G5 + G6).*Z0^2 +
     G3.*(G4 + G5 + G6).*Z0^2).* cos(x6.*(theta)) + 1i.*Z0.*(G2 + G3 + G6
    + G2.*G4.*G6.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*G5.*G6.*Z0^2 + G3.*G5.*G6
    .*Z0^2).*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*Z0.*(G3 + G5 +
    G6 + G3.*G4.*G5.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*(1 + G4.*(G5 + G6).*Z0
    ^2)).*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G4 + G6 + G4.*
```

```
    G5.*G6.*Z0^2) + G3.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(
    theta))))) - sin(x3.*(theta)).*(cos(x4.*(theta)).*((-1i).*cos(x5.*(
    theta)).*((1 + G2.*(G3 + G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*
    Z0.*(G6 + G2.*(1 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2)).* sin(
    x6.*(theta))) + sin(x5.*(theta)).*(Z0.*(G5 + G6 + G2.*(1 + G3.*(G5 +
    G6).*Z0^2 + G4.*(G5 + G6).*Z0^2)).*cos(x6.*(theta)) + 1i.*(1 + G5.*G6
    .*Z0^2 + G2.*Z0^2.*(G3 + G4 + G6 +
G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta)))) + sin(x4.*(
    theta)).*(cos(x5.*(theta)).*(Z0.*(G4 + G5 + G6 + G2.*(1 + G3.*(G4 + G5
     + G6).*Z0^2)).* cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*G6.*Z0
    ^2 + G2.*Z0^2.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2)).*sin(
    x6.*(theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*(G5 + G6).*Z0^2 + G2.*
    Z0^2.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2))).*cos(x6.*(theta)) + 1
    i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2 + G2.*(1 + G5.*G6.*Z0^2 + G3.*Z0
    ^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2))).*sin(x6.*(theta)))))) + 1i.*Z0.*
    sin(x1.*(theta)).*((1./Z0).*(sin(x2.*(theta)).*(cos(x3.*(theta)).*(1i
    .*cos(x4.*(theta)).*(cos(x5.*(theta)).*(cos(x6.*(theta)) + 1i.*G6.*Z0
    .*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*(G5 + G6).*Z0.*cos(x6
    .*(theta)) + (1 + G5.*G6.*Z0^2).*sin(x6.*(theta)))) + sin(x4.*(theta))
    .*((-1i).*cos(x5.*(theta)).*((-1i).*(G4 + G5 +
G6).*Z0.*cos(x6.*(theta)) + (1 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(
    theta))) + sin(x5.*(theta)).*((-1i).*(1 + G4.*(G5 + G6).*Z0^2).*cos(x6
    .*(theta)) + Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta))))) +
    sin(x3.*(theta)).*((-1i).*cos(x4.*(theta)).*(cos(x5.*(theta)).*((-1i)
    .*(G3 + G4 + G5 + G6).*Z0.*cos(x6.*(theta)) + (1 + G3.*G6.*Z0^2 + G4.*
    G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) + sin(x5.*(theta)).*((1 +
    G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0
    .*(G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta
    )))) + sin(x4.*(theta)).*(cos(x5.*(theta)).*((-1i).*(1 + G3.*(G4 + G5
    + G6).*Z0^2).*cos(x6.*(theta)) + Z0.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3
    .*G5.*G6.*Z0^2).*sin(x6.*(theta))) + sin(x5.*(theta)).*(Z0.*(G5 + G6 +
     G3.*(1 + G4.*(G5 + G6).*Z0^2)).*cos(x6.*(theta)) + 1i.*(1 + G5.*G6.*
    Z0^2 + G3.*Z0^2.*(G4 + G6 +
G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta)))))) + (1./Z0).*(cos(x2.*(theta)).*(
    cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*((G2 + G3 + G4
     + G5 + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G2.*G6.*Z0^2 + G3.*G6.*Z0
    ^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).* sin(x6.*(theta))) + 1i.*sin(x5.*(
    theta)).*((1 + G4.*G5.*Z0^2 + G4.*G6.*Z0^2 + G2.*(G5 + G6).*Z0^2 + G3
    .*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 + G3 + G4 + G6 + G2
    .*G5.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2).*sin(x6.*(theta)
    ))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((1 + G2.*(G4 + G5 + G6
    ).*Z0^2 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G2 +
    G3 + G6 + G2.*G4.*G6.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*G5.*G6.*Z0^2 + G3
    .*G5.*G6.*Z0^2).*sin(x6.*(theta))) - sin(x5.*(theta)).*((-1i).*Z0.*(G3
     + G5 + G6 + G3.*G4.*G5.*Z0^2 + G3.*G4.*G6.*Z0^2 + G2.*(1 + G4.*(G5 +
```

```
    G6).*Z0^2)).* cos(x6.*(theta)) + (1 +
G5.*G6.*Z0^2 + G2.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2) + G3.*Z0^2.*(G4 +
    G6 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta))))) - sin(x3.*(theta)).*(cos(
    x4.*(theta)).*((-1i).*cos(x5.*(theta)).*((1 + G2.*(G3 + G4 + G5 + G6)
    .*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G6 + G2.*(1 + G3.*G6.*Z0^2 + G4.*
    G6.*Z0^2 + G5.*G6.*Z0^2)).* sin(x6.*(theta))) + sin(x5.*(theta)).*(Z0
    .*(G5 + G6 + G2.*(1 + G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0^2)).*cos
    (x6.*(theta)) + 1i.*(1 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G3 + G4 + G6 + G3.*
    G5.*G6.*Z0^2 + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta)))) + sin(x4.*(theta
    )).*(cos(x5.*(theta)).*(Z0.*(G4 + G5 + G6 + G2.*(1 + G3.*(G4 + G5 + G6
    ).*Z0^2)).* cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2 +
    G2.*Z0^2.*(G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2)).*sin(x6.*(
    theta))) + 1i.*sin(x5.*(theta)).*((1 + G4.*(G5 + G6).*Z0^2 +
G2.*Z0^2.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0^2))).*cos(x6.*(theta)) +
    1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2 + G2.*(1 + G5.*G6.*Z0^2 + G3.*Z0
    ^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2))).*sin(x6.*(theta)))))) + G1.*(cos(
    x2.*(theta)).*(cos(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta))
    .*(cos(x6.*(theta)) + 1i.*G6.*Z0.*sin(x6.*(theta))) - sin(x5.*(theta))
    .*((-1i).*(G5 + G6).*Z0.*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2).* sin(
    x6.*(theta)))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta)).*((G4 + G5 +
    G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin
    (x6.*(theta))) + 1i.*sin(x5.*(theta)).* ((1 + G4.*(G5 + G6).*Z0^2).*
    cos(x6.*(theta)) + 1i.*Z0.*(G4 + G6 + G4.*G5.*G6.*Z0^2).*sin(x6.*(
    theta))))) + 1i.*sin(x3.*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)
    ).*((G3 + G4 + G5 + G6).*Z0.*cos(x6.*(theta)) +
1i.*(1 + G3.*G6.*Z0^2 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2).*sin(x6.*(theta))) +
     1i.*sin(x5.*(theta)).* ((1 + G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0
    ^2).*cos(x6.*(theta)) + 1i.*Z0.*(G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 + G4
    .*G5.*G6.*Z0^2).*sin(x6.*(theta)))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(
    theta)).*((1 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(
    G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2).*sin(x6.*(theta))) -
    sin(x5.*(theta)).* ((-1i).*Z0.*(G5 + G6 + G3.*(1 + G4.*(G5 + G6).*Z0
    ^2)).*cos(x6.*(theta)) + (1 + G5.*G6.*Z0^2 + G3.*Z0^2.*(G4 + G6 + G4.*
    G5.*G6.*Z0^2)).*sin(x6.*(theta)))))) + 1i.*sin(x2.*(theta)).*(cos(x3
    .*(theta)).*(cos(x4.*(theta)).*(cos(x5.*(theta)).*((G2 + G3 + G4 + G5
    + G6).*Z0.*cos(x6.*(theta)) + 1i.*(1 + G2.*G6.*Z0^2 + G3.*G6.*Z0^2 +
    G4.*G6.*Z0^2 + G5.*G6.*Z0^2).* sin(x6.*(theta))) + 1i.*sin(x5.*(theta)
    ).*((1 + G4.*G5.*Z0^2 + G4.*G6.*Z0^2 +
G2.*(G5 + G6).*Z0^2 + G3.*(G5 + G6).*Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(
    G2 + G3 + G4 + G6 + G2.*G5.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2 + G4.*G5.*G6.*
    Z0^2).*sin(x6.*(theta)))) + 1i.*sin(x4.*(theta)).*(cos(x5.*(theta))
    .*((1 + G2.*(G4 + G5 + G6).*Z0^2 + G3.*(G4 + G5 + G6).*Z0^2).*cos(x6
    .*(theta)) + 1i.*Z0.*(G2 + G3 + G6 + G2.*G4.*G6.*Z0^2 + G3.*G4.*G6.*Z0
    ^2 + G2.*G5.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2).*sin(x6.*(theta))) - sin(x5
    .*(theta)).*((-1i).*Z0.*(G3 + G5 + G6 + G3.*G4.*G5.*Z0^2 + G3.*G4.*G6
```

```
    .*Z0^2 + G2.*(1 + G4.*(G5 + G6).*Z0^2)).* cos(x6.*(theta)) + (1 + G5.*
    G6.*Z0^2 + G2.*Z0^2.*(G4 + G6 + G4.*G5.*G6.*Z0^2) + G3.*Z0^2.*(G4 + G6
     + G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta))))) - sin(x3.*(theta)).*(cos(x4
    .*(theta)).*((-1i).*cos(x5.*(theta)).*((1 + G2.*(G3 + G4 + G5 + G6).*
    Z0^2).*cos(x6.*(theta)) + 1i.*Z0.*(G6 + G2.*(1 + G3.*G6.*Z0^2 + G4.*G6
    .*Z0^2 +
 G5.*G6.*Z0^2)).* sin(x6.*(theta))) + sin(x5.*(theta)).*(Z0.*(G5 + G6 + G2
    .*(1 + G3.*(G5 + G6).*Z0^2 + G4.*(G5 + G6).*Z0^2)).*cos(x6.*(theta)) +
     1i.*(1 + G5.*G6.*Z0^2 + G2.*Z0^2.*(G3 + G4 + G6 + G3.*G5.*G6.*Z0^2 +
    G4.*G5.*G6.*Z0^2)).*sin(x6.*(theta)))) + sin(x4.*(theta)).*(cos(x5.*(
    theta)).*(Z0.*(G4 + G5 + G6 + G2.*(1 + G3.*(G4 + G5 + G6).*Z0^2)).*
    cos(x6.*(theta)) + 1i.*(1 + G4.*G6.*Z0^2 + G5.*G6.*Z0^2 + G2.*Z0^2.*(
    G3 + G6 + G3.*G4.*G6.*Z0^2 + G3.*G5.*G6.*Z0^2)).*sin(x6.*(theta))) + 1
    i.*sin(x5.*(theta)).* ((1 + G4.*(G5 + G6).*Z0^2 + G2.*Z0^2.*(G5 + G6 +
     G3.*(1 + G4.*(G5 + G6).*Z0^2))).*cos(x6.*(theta)) + 1i.*Z0.*(G4 + G6
    + G4.*G5.*G6.*Z0^2 + G2.*(1 + G5.*G6.*Z0^2 + G3.*Z0^2.*(G4 + G6 + G4.*
    G5.*G6.*Z0^2))).*sin(x6.*(theta)))))))))));
end

record_theta = zeros(1,length(theta)); % Initialization for record_theta
record_theta(1,:) = ((gamma(1,:)) < dB_level).*theta_in_Hz(1,:); % Take
   values of theta when gamma < -10 dB
flag = 0; % Flag for helping calculate bandwidth
bandwidth_low_end = 0; % Initialization
bandwidth_high_end = 0; % Initialization

% Note: only records the first bandwidth that goes below -20 dB
for n = 1:length(theta)

   if ((record_theta(1,n) ~= 0) && (flag == 0))
      bandwidth_low_end = theta_in_Hz(1,n);
      flag = 1;
   end

   if ((record_theta(1,n) == 0) && (flag == 1))
      bandwidth_high_end = theta_in_Hz(1,n);
      flag = 2;
   end

end

if(flag == 1) % If the bandwidth high end isn't detected due to the
   crossing point too close to the end, then set the high end at the end
   of the bandwidth
   bandwidth_high_end = theta_in_Hz(1,n);
```

```
end

bandwidth = bandwidth_high_end - bandwidth_low_end;

fitness = bandwidth/f0;

end
```

A.6 MATLAB code for plotting convenience

```
function S11FitnessGraph_with_var_L( theta_in_Hz, gamma )
% An add-on function to S11Fitness_with_var_L
% Feed it theta_in_Hz and gamma and it outputs a graph of S11

dB_level = -20;

record_theta = zeros(1,length(theta_in_Hz)); % Initialization for
    record_theta
record_theta(1,:) = ((gamma(1,:)) < dB_level).*theta_in_Hz(1,:); % Take
    values of theta when gamma < dB_level
flag = 0; % Flag for helping calculate bandwidth
bandwidth_low_end = 0; % Initialization
bandwidth_high_end = 0; % Initialization

% Note: only records the first bandwidth that goes below the dB_level
for n = 1:length(theta_in_Hz)

    if ((record_theta(1,n) ~= 0) && (flag == 0))
        bandwidth_low_end = theta_in_Hz(1,n);
        flag = 1;
    end

    if ((record_theta(1,n) == 0) && (flag == 1))
        bandwidth_high_end = theta_in_Hz(1,n);
        flag = 2;
    end

end

if(flag == 1) % If the bandwidth high end isn't detected due to the
    crossing point too close to the end, then set the high end at the end
    of the bandwidth
    bandwidth_high_end = theta_in_Hz(1,n);
end

bandwidth = bandwidth_high_end - bandwidth_low_end;
% display_text = sprintf('Bandwidth: %f - %f',bandwidth_low_end,
    bandwidth_high_end);
% disp(display_text);

f0 = 2.50e2;
k_low = bandwidth_low_end/f0;
k_high = bandwidth_high_end/f0;
```

```
display_text = sprintf('Relative bandwidth: %f f0 - %f f0',k_low,k_high);
disp(display_text);
disp('Factors:');

display_text = sprintf('%f \t %f',k_low,k_high);
disp(display_text)

% Line markers
lineMarkers = ['-k ';':k ';'--k ';'-.k ';'-sk ';':sk ';'--sk';'-.sk'];
lineMarkers = cellstr(lineMarkers);

figure
hold on
plot(theta_in_Hz,gamma,char(lineMarkers(1)));
axis([min(theta_in_Hz) max(theta_in_Hz) -50 0])
grid on
title('S_{11} vs Frequency')
xlabel('Frequency (MHz)')
ylabel('S_{11} (dB)')
end
```

A.7 General form of reflection coefficients with variable transmission line length for 1, 2, 3, 4, 5, and 6 layers case

The following reflection coefficients are obtained through the same method described in chapter 2 except that each transmission line sections are treated as a unique variable. In other words, the transmission line length in the first, second, third, fourth, fifth, and sixth section are θ_1, θ_2, θ_3, θ_4, θ_5, and θ_6 respectively.

$$\Gamma_1 = -\frac{(Z_S \cos(x_1\theta) + jZ_0(-1 + G_1 Z_S)\sin(x_1\theta))}{(Z_S \cos(x_1\theta) + jZ_0(1 + G_1 Z_S)\sin(x_1\theta))} \tag{A.17}$$

$$\Gamma_2 = \frac{(\cos(x_1\theta)(Z_S\cos(x_2\theta)+jZ_0(-1+G_2Z_S)\sin(x_2\theta))+\sin(x_1\theta)(jZ_0(-1+G_1Z_S+G_2Z_S)\cos(x_2\theta)-(Z_S+G_1Z_0^2(-1+G_2Z_S))\sin(x_2\theta)))}{((-\cos(x_1\theta))(Z_S\cos(x_2\theta)+jZ_0(1+G_2Z_S)\sin(x_2\theta))+\sin(x_1\theta)(jZ_0(1+G_1Z_S+G_2Z_S)\cos(x_2\theta)+(Z_S+G_1Z_0^2(1+G_2Z_S))\sin(x_2\theta)))} \tag{A.18}$$

$$\Gamma_3 = \frac{(\cos(x_1\theta)(\cos(x_2\theta)(jZ_S\cos(x_3\theta)+Z_0(-1+G_3Z_S)\sin(x_3\theta))+\sin(x_2\theta)(Z_0(-1+G_2Z_S+G_3Z_S)\cos(x_3\theta)+j(Z_S+G_2Z_0^2(-1+G_3Z_S))\sin(x_3\theta)))+\sin(x_1\theta)(\cos(x_2\theta)(Z_0(-1+G_1Z_S+G_2Z_S+G_3Z_S)\cos(x_3\theta)+j(Z_S+G_1Z_0^2(-1+G_3Z_S)+G_2Z_0^2(-1+G_3Z_S))\sin(x_3\theta))+\sin(x_2\theta)(j(Z_S+G_1Z_0^2(-1+G_2Z_S+G_3Z_S))\cos(x_3\theta)-Z_0(-1+G_3Z_S+G_1(Z_S+G_2Z_0^2(-1+G_3Z_S)))\sin(x_3\theta))))}{(j\cos(x_1\theta)(\cos(x_2\theta)(Z_S\cos(x_3\theta)+jZ_0(1+G_3Z_S)\sin(x_3\theta))-\sin(x_2\theta)(jZ_0(1+G_2Z_S+G_3Z_S)\cos(x_3\theta)+(Z_S+G_2Z_0^2(1+G_3Z_S))\sin(x_3\theta)))+\sin(x_1\theta)(j\cos(x_2\theta)(jZ_0(1+G_1Z_S+G_2Z_S+G_3Z_S)\cos(x_3\theta)+(Z_S+G_1Z_0^2(1+G_3Z_S)+G_2Z_0^2(1+G_3Z_S))\sin(x_3\theta))+\sin(x_2\theta)(j(Z_S+G_1Z_0^2(1+G_2Z_S+G_3Z_S))\cos(x_3\theta)+Z_0(1+G_3Z_S+G_1(Z_S+G_2Z_0^2(1+G_3Z_S)))\sin(x_3\theta))))} \tag{A.19}$$

$$\Gamma_4 = \text{[illegible]} \tag{A.20}$$

$$\Gamma_5 = \text{[illegible]} \tag{A.21}$$

(Place holder for six layers case. Error because the expression is too long.)

A.8 Cut-and-try Algorithm Results

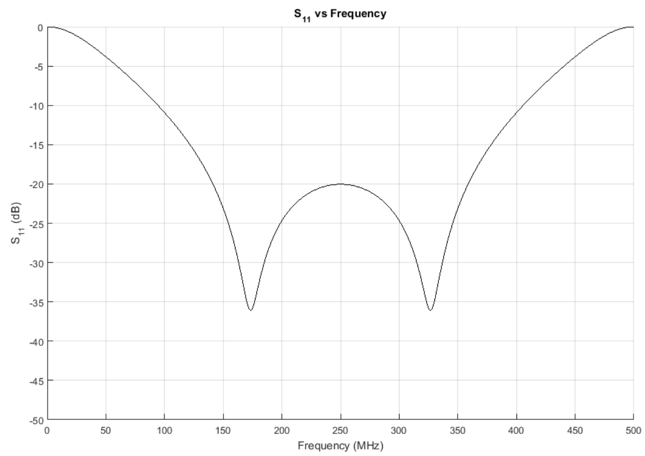

FIGURE A.1: Cut-and-try result for 2 sections case

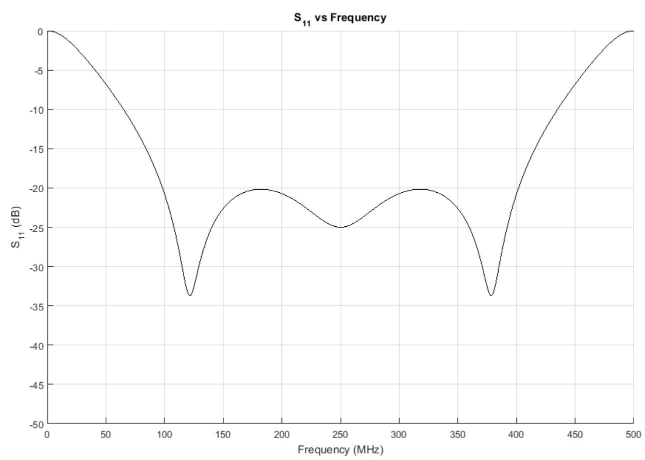

FIGURE A.2: Cut-and-try result for 3 sections case

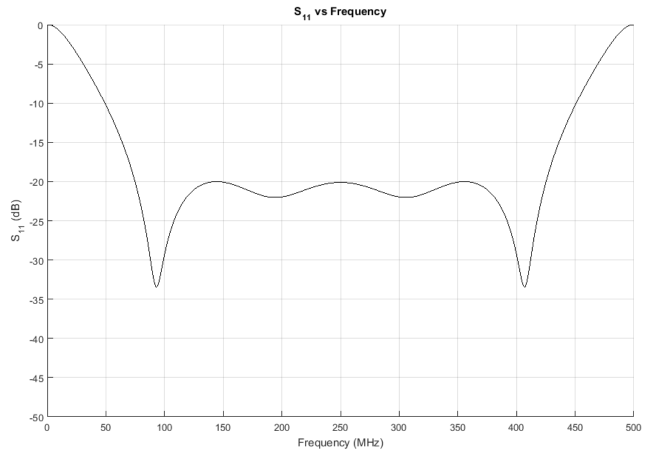

FIGURE A.3: Cut-and-try result for 4 sections case

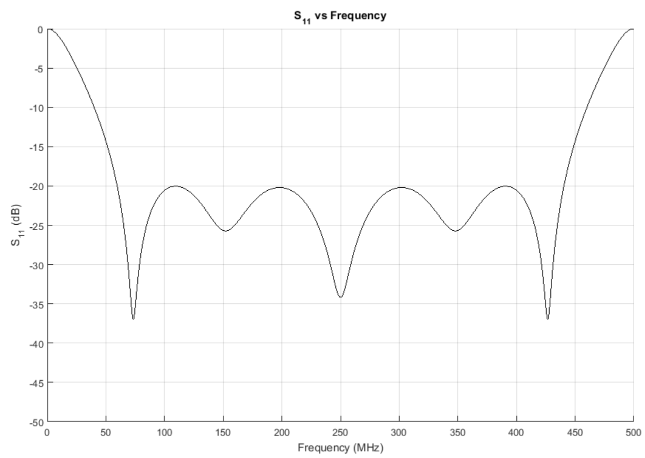

FIGURE A.4: Cut-and-try result for 5 sections case

A.9 Genetic Algorithm Expanded Results Tables

Note: the standard result is equivalent to expanded result with theta limit set to 1

This chapter lists all the GA expanded result. Column "G_n" is the normalized conductance and "X_n" is the separation between the element n and $(n-1)$ expressed in the multiples of $\lambda/4$ (the full detail is available in Section 2.3). The $0th$ element is always a short circuit. The "RBW", "Start", and "End" columns indicate the relative bandwidth, the start, and the end of the bandwidth in the multiples of f_0

respectively. For example, if the bandwidth spans from $0.5f_0$ to $1.7f_0$, then the entry is listed as "1.2" in the "RBW" column, "0.5" in the "Start" column, and "1.7" in the "End" column. Finally, the "T" columns stands for "Theta limit". The theta limit defines the search boundary in frequency in terms of f_0 between 0 to $2Tf_0$. For example, if the theta limit is set to "7", then the algorithm searches for the best solution—considering other limitations as described in Section 3.4)—between 0 to 14 f_0.

A.9.1 Threshold level: -10 dB

One section

Table A.1: GA expanded result for 1 section with Γ threshold -10 dB.

G_1	X_1	RBW	Start	End	T
1.210	0.695	1.122	0.878	2.000	1
1.220	0.347	2.244	1.756	4.000	2
1.232	0.232	3.364	2.632	5.995	3
1.222	0.174	4.486	3.509	7.995	4
1.224	0.139	5.608	4.386	9.994	5
1.224	0.116	6.730	5.271	12.000	6
1.221	0.099	7.853	6.147	14.000	7
1.216	0.087	8.975	7.025	16.000	8
1.220	0.077	10.097	7.903	18.000	9
1.220	0.070	11.220	8.780	20.000	10
1.225	0.063	12.342	9.654	21.996	11
1.223	0.058	13.465	10.536	24.000	12
1.223	0.054	14.586	11.409	25.995	13
1.225	0.050	15.707	12.293	28.000	14
1.226	0.046	16.811	13.189	30.000	15
1.222	0.044	17.939	14.031	31.970	16
1.225	0.041	19.072	14.929	34.000	17
1.219	0.039	20.180	15.821	36.000	18
1.223	0.037	21.312	16.688	38.000	19
1.219	0.035	22.400	17.521	39.922	20

Two sections

Table A.2: GA expanded result for 2 sections with Γ threshold -10 dB.

G_1	G_2	X_1	X_2	RBW	Start	End	T
1.274	0.360	0.843	1.654	1.646	0.354	2.000	1
0.979	0.901	0.419	0.419	3.229	0.771	4.000	2
0.943	0.860	0.282	0.278	4.844	1.154	5.997	3
1.674	0.657	0.426	0.216	6.478	1.400	7.878	4
1.543	0.657	0.354	0.175	8.251	1.560	9.812	5
1.569	0.599	0.256	0.130	9.886	2.114	12.000	6
1.589	0.577	0.211	0.106	11.400	2.600	14.000	7
1.364	0.389	0.105	0.220	13.337	2.643	15.980	8
1.370	0.465	0.093	0.186	15.180	2.820	18.000	9
0.960	0.879	0.082	0.084	16.134	3.866	20.000	10
0.941	0.825	0.078	0.075	17.749	4.249	21.997	11
1.573	0.639	0.145	0.073	19.902	3.827	23.728	12
1.541	0.676	0.121	0.060	21.415	4.585	26.000	13
1.656	0.664	0.114	0.056	22.666	5.334	28.000	14
0.984	0.860	0.058	0.054	24.180	5.820	30.000	15
1.028	0.929	0.052	0.053	25.789	6.212	32.000	16
1.270	0.392	0.051	0.101	28.153	5.577	33.730	17
0.939	0.851	0.046	0.047	29.058	6.942	36.000	18
0.969	0.861	0.045	0.044	30.648	7.325	37.973	19
1.308	0.396	0.041	0.079	32.998	7.003	40.000	20

Three sections

Table A.3: GA expanded result for 3 sections with Γ threshold -10 dB.

G_1	G_2	G_3	X_1	X_2	X_3	RBW	Start	End	T
1.600	0.524	0.769	0.928	0.916	1.796	1.817	0.183	2.000	1
1.487	0.497	0.699	0.431	0.445	0.933	3.628	0.373	4.000	2
1.411	0.568	0.644	0.286	0.277	0.533	5.373	0.627	6.000	3
1.159	0.628	0.537	0.208	0.198	0.618	7.281	0.719	8.000	4
1.256	0.782	0.611	0.336	0.387	0.166	9.175	0.825	10.000	5
1.191	0.611	0.470	0.142	0.159	0.468	10.985	1.016	12.000	6
1.184	0.656	0.570	0.122	0.122	0.374	12.834	1.167	14.000	7
1.442	0.660	0.490	0.106	0.108	0.328	14.590	1.410	16.000	8
1.314	0.609	0.522	0.097	0.096	0.291	16.460	1.541	18.000	9
1.236	0.628	0.553	0.089	0.096	0.280	18.388	1.568	19.956	10
1.243	0.707	0.543	0.077	0.083	0.237	20.160	1.840	22.000	11
1.088	0.616	0.521	0.068	0.080	0.225	21.994	2.006	24.000	12
1.188	0.776	0.507	0.067	0.062	0.197	23.696	2.304	26.000	13
1.283	0.606	0.381	0.057	0.179	0.121	25.947	2.053	28.000	14
1.169	0.604	0.423	0.051	0.144	0.100	27.574	2.427	30.000	15
1.194	0.709	0.533	0.055	0.056	0.163	29.322	2.679	32.000	16
1.233	0.668	0.580	0.055	0.052	0.160	31.317	2.684	34.000	17
1.225	0.683	0.545	0.046	0.043	0.142	32.839	3.162	36.000	18
1.230	0.681	0.566	0.044	0.042	0.128	34.632	3.369	38.000	19
1.363	0.425	0.619	0.044	0.038	0.099	36.147	3.853	40.000	20

Four sections

Table A.4: GA expanded result for 4 sections with Γ threshold -10 dB.

G_1	G_2	G_3	G_4	X_1	X_2	X_3	X_4	RBW	Start	End	T
1.042	0.572	0.281	0.579	0.962	1.811	0.965	0.860	1.855	0.145	2.000	1
1.337	0.681	0.618	0.567	0.619	0.826	0.994	0.367	3.747	0.253	4.000	2
1.614	0.703	0.357	0.383	0.305	0.270	0.534	0.897	5.599	0.402	6.000	3
1.393	0.732	0.427	0.297	0.304	0.456	0.490	0.168	7.484	0.501	7.985	4
2.932	5.932	1.027	0.485	0.569	0.125	0.174	0.354	8.582	1.389	9.972	5
0.656	0.796	0.468	0.370	0.116	0.492	0.157	0.288	11.313	0.687	12.000	6
2.098	2.952	0.876	0.548	1.140	0.129	0.114	0.230	12.272	1.728	14.000	7
1.515	0.675	0.374	0.187	0.088	0.364	0.220	0.286	15.126	0.875	16.000	8
1.022	0.705	0.528	0.407	0.081	0.301	0.090	0.190	16.917	1.083	18.000	9
1.282	0.876	0.287	0.337	0.246	0.313	0.079	0.090	18.918	1.083	20.000	10
1.578	1.110	0.456	0.379	0.087	0.306	0.075	0.155	20.699	1.301	22.000	11
1.459	0.921	0.347	0.312	0.070	0.314	0.070	0.168	22.795	1.206	24.000	12
3.667	6.517	1.210	0.514	0.049	0.058	0.132	0.065	22.229	3.771	26.000	13
1.972	1.025	0.426	0.297	0.075	0.186	0.060	0.060	25.866	2.134	28.000	14
1.129	0.843	0.469	0.253	0.059	0.068	0.250	0.196	28.527	1.473	30.000	15
1.691	7.033	0.862	0.845	0.329	0.009	0.049	0.053	25.977	6.024	32.000	16
4.403	1.220	0.457	0.414	0.050	0.046	0.050	0.143	30.945	3.055	34.000	17
0.712	6.286	1.090	0.418	0.009	0.301	0.044	0.085	29.541	6.459	36.000	18
2.834	3.214	0.835	0.544	0.116	0.034	0.040	0.083	32.627	5.373	38.000	19
3.054	0.996	0.713	0.441	0.029	0.037	0.032	0.092	35.739	4.261	40.000	20

Five sections

Table A.5: GA expanded result for 5 sections with Γ threshold -10 dB.

G_1	G_2	G_3	G_4	G_5	X_1	X_2	X_3	X_4	X_5	RBW	Start	End	T
1.265	0.669	0.445	0.361	0.469	0.899	0.977	0.816	1.004	2.507	1.882	0.119	2.000	1
2.588	0.969	0.685	0.318	0.225	4.531	0.453	0.872	0.769	0.317	3.640	0.360	4.000	2
0.677	2.713	1.610	0.579	0.478	0.167	1.972	2.406	0.285	0.234	5.061	0.939	6.000	3
2.066	1.086	0.669	0.370	0.368	1.501	0.175	0.439	0.452	0.224	7.411	0.590	8.000	4
0.943	0.674	0.684	0.306	0.268	0.354	0.098	0.596	0.190	0.127	9.472	0.528	10.000	5
1.687	1.352	0.568	0.291	0.278	0.237	0.120	0.527	0.134	0.268	11.390	0.610	12.000	6
1.872	0.741	0.437	0.463	0.330	0.123	0.221	0.255	0.129	0.516	13.361	0.640	14.000	7
2.507	3.903	0.720	0.735	0.414	0.308	0.135	0.106	0.113	0.313	14.588	1.412	16.000	8
0.826	1.853	2.710	0.818	0.516	0.257	0.099	0.057	0.096	0.196	15.700	2.300	18.000	9
1.668	0.889	0.459	0.523	0.390	0.153	0.061	0.168	0.265	0.081	19.016	0.984	20.000	10
1.661	3.515	1.045	0.321	0.423	0.019	0.042	0.229	0.086	0.071	20.276	1.724	22.000	11
2.237	1.139	2.496	0.802	0.624	0.481	0.013	0.037	0.074	0.064	19.939	4.061	24.000	12
1.526	3.731	1.200	0.676	0.406	0.168	0.945	0.058	0.067	0.209	23.634	2.367	26.000	13
1.295	1.894	0.967	0.332	0.381	0.096	0.064	0.208	0.061	0.057	25.861	2.139	28.000	14
2.453	2.270	0.721	0.625	0.255	0.024	0.062	0.057	0.055	0.177	27.358	2.642	30.000	15
1.308	1.668	0.744	0.450	0.424	0.034	0.126	0.172	0.056	0.048	29.784	2.216	32.000	16
1.766	4.296	1.117	0.582	0.431	0.567	0.131	0.058	0.048	0.165	31.148	2.852	34.000	17
0.244	2.179	3.745	0.823	0.445	0.722	0.152	0.031	0.047	0.099	31.259	4.741	36.000	18
3.917	2.462	1.542	0.629	0.423	0.347	0.217	0.042	0.090	0.046	33.657	4.343	38.000	19
0.549	2.394	2.471	0.762	0.446	0.419	0.676	0.055	0.088	0.040	35.004	4.996	40.000	20

Six sections

Table A.6: GA expanded result for 6 sections with Γ threshold -10 dB.

G_1	G_2	G_3	G_4	G_5	G_6	X_1	X_2	X_3	X_4	X_5	X_6	RBW	Start	End	T
1.998	0.663	0.413	0.262	0.433	0.346	1.255	1.738	1.718	2.194	2.090	0.704	1.929	0.071	2.000	1
0.363	1.956	1.211	0.544	0.519	0.324	0.328	1.274	0.767	1.009	0.374	0.334	3.625	0.375	4.000	2
1.052	0.721	1.138	0.516	0.616	0.379	0.506	0.395	2.366	0.680	0.523	0.250	5.423	0.577	6.000	3
1.543	1.846	2.344	1.019	0.558	0.451	0.120	0.165	1.537	0.458	0.457	0.220	7.256	0.744	8.000	4
1.016	0.893	0.370	0.515	0.327	0.434	0.081	0.291	0.488	0.233	0.391	0.138	9.578	0.423	10.000	5
2.865	2.909	1.555	0.845	0.466	0.291	1.297	1.684	0.906	0.265	0.372	0.128	11.097	0.903	12.000	6
2.514	1.403	2.577	0.720	0.548	0.439	0.301	0.114	0.869	0.257	0.248	0.128	12.788	1.212	14.000	7
0.906	1.605	2.982	2.932	0.928	0.511	1.575	1.257	0.997	0.082	0.106	0.219	13.967	2.033	16.000	8
2.091	1.710	2.304	0.695	0.522	0.234	1.413	0.092	0.557	0.099	0.312	0.214	16.896	1.104	18.000	9
1.049	1.846	2.737	0.936	0.448	0.302	1.264	1.047	0.099	0.185	0.292	0.084	18.676	1.249	19.925	10
0.103	0.547	4.109	0.955	0.522	0.392	0.393	2.510	0.070	0.066	0.079	0.223	19.937	2.063	22.000	11
0.706	1.256	1.636	2.796	0.893	0.444	0.486	0.939	0.215	0.286	0.134	0.065	20.320	3.680	24.000	12
0.980	1.204	1.697	0.840	0.404	0.356	0.063	0.166	0.070	0.271	0.065	0.138	24.512	1.488	26.000	13
2.440	1.008	3.055	2.133	0.567	0.448	1.253	1.805	0.081	0.056	0.058	0.129	24.795	3.205	28.000	14
0.325	2.671	2.492	1.072	0.467	0.306	1.214	0.329	0.062	0.108	0.194	0.050	28.022	1.978	30.000	15
1.763	0.891	3.364	3.164	0.888	0.511	0.662	1.575	0.457	0.059	0.053	0.109	27.956	4.045	32.000	16
1.606	1.993	1.472	1.621	0.777	0.540	0.014	0.486	0.016	0.068	0.093	0.044	30.209	3.791	34.000	17
1.521	0.943	2.281	0.723	0.513	0.352	0.072	0.029	0.081	0.149	0.046	0.050	33.053	2.947	36.000	18
1.694	1.521	3.474	1.968	0.639	0.504	0.579	1.494	0.426	0.083	0.092	0.042	33.963	4.037	38.000	19
2.109	2.164	0.229	3.664	0.988	0.405	0.045	0.324	0.233	0.146	0.039	0.080	33.857	6.143	40.000	20

A.9.2 Threshold level: -20 dB

One section

Table A.7: GA expanded result for 1 section with Γ threshold -20 dB.

G_1	X_1	RBW	Start	End	T
1.028	0.564	0.450	1.549	1.999	1
1.014	0.282	0.901	3.100	4.000	2
1.026	0.188	1.349	4.644	5.993	3
1.009	0.141	1.798	6.196	7.994	4
1.017	0.113	2.250	7.742	9.992	5
1.021	0.094	2.694	9.268	11.961	6
1.022	0.081	3.135	10.784	13.919	7
1.032	0.071	3.579	12.337	15.915	8
1.026	0.063	4.029	13.864	17.893	9
1.037	0.056	4.488	15.496	19.984	10
1.022	0.051	4.950	17.029	21.978	11
1.000	0.047	5.379	18.606	23.985	12
1.022	0.043	5.852	20.132	25.984	13
1.007	0.040	6.295	21.703	27.998	14
1.018	0.038	6.734	23.167	29.901	15
1.017	0.035	7.201	24.775	31.976	16
1.005	0.034	7.555	26.071	33.626	17
1.040	0.031	8.074	27.920	35.993	18
1.016	0.030	8.439	29.036	37.475	19
1.022	0.028	8.968	30.851	39.819	20

Two sections

Table A.8: GA expanded result for 2 sections with Γ threshold -20 dB.

G_1	G_2	X_1	X_2	RBW	Start	End	T
1.274	0.149	0.604	1.768	0.499	1.439	1.937	1
2.120	0.566	0.324	0.318	1.409	2.455	3.863	2
1.286	0.404	0.251	0.261	3.286	2.246	5.531	3
1.699	0.480	0.178	0.183	3.485	3.691	7.176	4
1.543	0.499	0.172	0.175	4.215	3.571	7.787	5
2.017	0.718	0.340	0.111	4.547	6.597	11.144	6
1.630	0.405	0.114	0.103	6.024	6.581	12.605	7
1.364	0.389	0.105	0.114	7.101	5.419	12.521	8
1.588	0.465	0.093	0.095	7.509	6.763	14.272	9
1.635	0.495	0.075	0.077	8.629	8.426	17.054	10
1.720	0.498	0.063	0.062	10.107	11.013	21.120	11
1.573	0.438	0.058	0.063	10.876	10.304	21.180	12
1.421	0.433	0.058	0.060	13.435	10.038	23.473	13
1.337	0.381	0.049	0.056	14.480	11.375	25.855	14
1.368	0.439	0.052	0.053	15.923	11.153	27.076	15
1.542	0.489	0.040	0.042	17.033	14.830	31.863	16
1.427	0.372	0.043	0.040	18.071	15.166	33.237	17
1.612	0.457	0.036	0.037	18.454	17.498	35.952	18
1.456	0.521	0.041	0.040	19.902	15.111	35.012	19
1.424	0.489	0.041	0.039	20.802	15.250	36.052	20

Three sections

Table A.9: GA expanded result for 3 sections with Γ threshold -20 dB.

G_1	G_2	G_3	X_1	X_2	X_3	RBW	Start	End	T
1.377	0.533	0.321	0.784	0.791	0.813	1.506	0.495	2.000	1
1.365	0.508	0.313	0.402	0.405	0.400	3.015	0.976	3.991	2
1.410	0.576	0.317	0.253	0.265	0.262	4.467	1.533	6.000	3
1.230	0.541	0.273	0.201	0.198	0.192	5.995	2.005	8.000	4
1.376	0.573	0.273	0.155	0.151	0.164	7.359	2.641	10.000	5
1.385	0.480	0.300	0.131	0.129	0.125	8.907	3.093	12.000	6
1.522	0.489	0.261	0.107	0.114	0.109	10.199	3.802	14.000	7
1.462	0.507	0.309	0.102	0.100	0.095	11.935	4.065	16.000	8
1.181	0.518	0.252	0.088	0.087	0.087	13.444	4.556	18.000	9
1.263	0.518	0.284	0.077	0.076	0.076	14.836	5.164	20.000	10
1.280	0.496	0.273	0.065	0.066	0.064	15.906	6.095	22.000	11
1.270	0.499	0.274	0.068	0.067	0.064	18.030	5.970	24.000	12
1.281	0.523	0.273	0.061	0.062	0.063	19.485	6.476	25.961	13
1.428	0.561	0.334	0.057	0.057	0.057	20.942	7.059	28.000	14
1.313	0.513	0.261	0.051	0.052	0.057	22.272	7.690	29.962	15
1.445	0.497	0.266	0.049	0.047	0.053	23.475	8.525	32.000	16
1.357	0.487	0.270	0.045	0.047	0.043	25.058	8.942	34.000	17
1.410	0.548	0.341	0.042	0.042	0.043	26.718	9.283	36.000	18
3.654	0.964	0.380	0.014	0.034	0.038	23.548	14.452	38.000	19
0.541	1.229	0.417	0.007	0.028	0.036	23.780	16.006	39.786	20

Table A.10: GA expanded result for 4 sections with Γ threshold -20 dB.

G_1	G_2	G_3	G_4	X_1	X_2	X_3	X_4	RBW	Start	End	T
2.397	0.899	0.437	0.277	0.400	0.733	0.765	0.765	1.530	0.470	2.000	1
1.562	0.553	0.322	0.200	0.420	0.358	0.437	0.349	3.180	0.820	4.000	2
2.350	1.023	0.471	0.307	0.105	0.220	0.245	0.254	4.507	1.494	6.000	3
0.861	1.925	1.435	0.688	0.012	0.472	0.476	0.162	3.518	4.482	8.000	4
1.660	5.952	1.068	0.387	0.387	0.604	0.143	0.146	6.045	3.955	10.000	5
0.899	1.632	0.624	0.254	0.008	0.126	0.126	0.106	7.972	4.028	12.000	6
1.327	3.805	0.995	0.394	0.313	0.081	0.102	0.101	8.596	5.404	14.000	7
1.084	0.390	0.496	0.311	0.088	0.028	0.081	0.089	11.884	4.117	16.000	8
2.769	0.947	1.201	0.450	0.008	0.009	0.068	0.077	10.862	7.122	17.984	9
2.261	2.790	0.990	0.400	0.248	0.042	0.065	0.071	12.326	7.674	20.000	10
1.452	1.406	0.617	0.256	0.007	0.060	0.071	0.063	15.680	6.320	22.000	11
1.885	1.786	0.662	0.305	0.010	0.032	0.056	0.061	15.979	7.916	23.895	12
3.799	6.517	1.091	0.396	0.581	0.015	0.054	0.055	15.819	10.144	25.963	13
1.870	3.722	0.952	0.370	0.170	0.040	0.051	0.051	17.227	10.708	27.936	14
1.564	0.470	0.308	0.146	0.046	0.049	0.041	0.039	22.585	7.416	30.000	15
1.373	7.392	1.186	0.389	0.006	0.452	0.042	0.044	18.849	13.151	32.000	16
3.520	3.187	0.969	0.385	0.131	0.017	0.035	0.036	19.304	14.696	34.000	17
1.821	6.212	1.159	0.454	0.121	0.442	0.040	0.039	21.670	14.330	36.000	18
2.251	3.347	1.104	0.372	0.130	0.006	0.034	0.037	22.659	15.342	38.000	19
0.880	0.874	1.368	0.669	0.091	0.005	0.094	0.031	17.452	22.548	40.000	20

Five sections

Table A.11: GA expanded result for 5 sections with Γ threshold -20 dB.

G_1	G_2	G_3	G_4	G_5	X_1	X_2	X_3	X_4	X_5	RBW	Start	End	T
1.315	1.872	1.219	0.628	0.250	1.014	0.433	0.890	0.792	0.910	1.162	0.607	1.769	1
0.690	0.885	2.024	1.358	0.484	1.387	0.424	0.161	0.361	0.396	1.501	1.757	3.257	2
0.321	1.708	1.680	1.193	0.408	1.056	1.463	1.246	0.265	0.251	2.404	2.760	5.164	3
0.357	1.170	2.329	1.261	0.358	1.292	0.166	0.655	0.166	0.181	3.327	3.750	7.077	4
0.902	1.787	1.725	1.627	0.598	0.059	0.151	0.286	0.132	0.126	3.568	6.096	9.664	5
0.139	1.869	1.451	1.688	0.632	0.142	0.341	1.741	0.329	0.111	3.910	6.969	10.879	6
2.222	1.263	1.583	1.062	0.420	0.159	0.094	0.448	0.136	0.135	5.507	4.406	9.913	7
1.687	2.067	1.980	0.829	0.387	0.850	1.225	0.110	0.110	0.101	8.439	5.408	13.848	8
0.271	1.189	2.018	2.936	0.759	1.055	0.413	0.964	0.203	0.068	5.292	12.067	17.359	9
0.437	1.494	1.664	1.721	0.704	0.663	0.728	0.481	0.190	0.062	7.310	12.400	19.710	10
3.745	2.268	1.319	2.025	0.673	0.538	0.058	0.289	0.170	0.059	7.436	13.491	20.927	11
2.694	1.145	2.376	1.112	0.414	1.791	0.854	0.762	0.081	0.086	8.778	7.408	16.185	12
1.312	2.052	1.729	2.429	0.713	0.395	0.166	0.250	0.152	0.050	8.544	15.687	24.230	13
1.793	2.363	2.197	1.300	0.382	0.503	0.475	0.003	0.057	0.051	13.006	12.099	25.105	14
2.657	1.157	1.437	1.680	0.658	0.797	0.133	0.548	0.129	0.045	9.569	17.629	27.198	15
1.367	1.030	1.575	1.854	0.688	0.782	0.065	0.095	0.126	0.044	9.546	18.375	27.921	16
1.806	3.019	0.708	1.634	0.574	0.317	0.770	0.273	0.026	0.035	11.677	21.153	32.829	17
1.246	1.976	1.790	2.547	0.772	0.607	0.284	0.210	0.127	0.042	10.083	18.981	29.065	18
0.595	1.231	3.031	1.441	0.543	0.329	0.223	0.409	0.110	0.037	12.475	21.021	33.496	19
1.170	0.991	1.626	2.625	0.691	0.242	0.052	0.161	0.242	0.030	13.057	26.261	39.318	20

Six sections

Table A.12: GA expanded result for 6 sections with Γ threshold -20 dB.

G_1	G_2	G_3	G_4	G_5	G_6	X_1	X_2	X_3	X_4	X_5	X_6	RBW	Start	End	T
3.242	0.904	0.885	1.113	0.652	0.339	1.464	1.027	2.039	0.499	0.699	0.820	1.091	0.676	1.767	1
0.245	1.446	1.211	1.705	1.145	0.324	0.328	0.045	1.450	0.266	0.373	0.333	1.589	2.073	3.663	2
0.869	0.641	0.364	1.400	1.747	0.662	0.643	0.138	1.492	0.244	0.642	0.200	1.957	3.928	5.884	3
0.222	0.597	1.305	1.896	1.791	0.663	0.136	0.103	1.148	0.860	0.457	0.154	2.929	4.994	7.923	4
0.628	0.704	1.558	1.455	1.762	0.690	0.670	0.191	0.768	0.261	0.363	0.121	3.468	6.431	9.899	5
1.385	2.562	1.521	1.703	1.430	0.562	1.296	1.196	0.578	0.344	0.337	0.107	4.315	6.781	11.097	6
2.142	1.332	1.973	2.395	1.859	0.623	1.279	1.274	0.382	0.658	0.092	0.091	4.618	8.621	13.238	7
0.686	2.499	1.496	2.566	2.363	0.656	0.058	0.455	2.150	0.819	0.227	0.078	5.251	10.107	15.358	8
1.506	1.833	1.960	1.526	1.443	0.564	1.640	0.255	0.201	0.026	0.079	0.087	6.086	8.209	14.294	9
0.030	1.257	1.825	1.893	1.733	0.626	1.776	0.745	0.257	0.347	0.187	0.063	7.122	12.183	19.306	10
0.974	0.745	2.175	2.267	2.250	0.653	0.631	2.667	1.637	0.009	0.179	0.058	7.237	13.290	20.527	11
1.016	1.240	2.259	2.600	2.289	0.701	0.281	1.545	0.347	0.874	0.159	0.050	7.823	15.640	23.463	12
1.453	1.747	1.992	1.813	2.119	0.634	1.041	0.370	0.209	0.061	0.373	0.047	8.433	17.372	25.804	13
1.142	0.813	2.553	1.940	1.390	0.483	0.044	0.756	1.729	0.475	0.059	0.059	10.089	11.888	21.976	14
1.137	1.562	2.401	2.281	1.967	0.675	2.250	0.285	1.312	0.897	0.132	0.043	10.329	17.755	28.084	15
1.534	1.306	1.245	2.639	1.536	0.628	0.922	1.710	0.502	0.063	0.120	0.037	11.513	20.116	31.628	16
1.637	1.156	2.005	2.354	1.548	0.522	0.871	0.263	0.233	0.148	0.033	0.036	12.570	20.607	33.176	17
1.468	0.203	1.858	2.015	1.422	0.387	0.126	0.976	1.206	0.069	0.036	0.035	14.097	21.392	35.489	18
1.628	1.267	1.939	2.337	1.138	0.423	0.227	0.726	0.272	0.230	0.040	0.037	19.540	16.232	35.772	19
1.877	2.275	0.229	3.722	1.118	0.431	0.045	0.448	0.300	0.106	0.039	0.037	21.235	15.392	36.626	20

A.10 Transmission Line Design and Calculation

In this work, the first step in transmission line design is to determine the width-to-height ($\frac{w}{h}$) ratio for parallel plate line on an FR-4 board dielectric. Here we use subscript *"ms"* to denote "microstrip" and *"pp"* to denote parallel plate. We use the following formula for microstrip [40, 41]:

If $\left(\frac{w}{h}\right)_{ms} < 2$:

$$\left(\frac{w}{h}\right)_{ms} = \frac{8e^{A}}{e^{2A}-2} \tag{A.22}$$

where:

$$A = \frac{Z_0}{60}\sqrt{\frac{\epsilon_r+1}{2}} + \frac{\epsilon_r-1}{\epsilon_r+1}\left(0.23+\frac{0.11}{\epsilon_r}\right) \tag{A.23}$$

If $\left(\frac{w}{h}\right)_{ms} \geq 2$:

$$\left(\frac{w}{h}\right)_{ms} = \frac{2}{\pi}\left[B-1-\ln(2B-1)+\frac{\epsilon_r-1}{2\epsilon_r}\left(\ln(B-1)+0.39-\frac{0.61}{\epsilon_r}\right)\right] \tag{A.24}$$

where:

$$B = \frac{377\pi}{2Z_0\sqrt{\epsilon_r}} \tag{A.25}$$

Where $\epsilon_r = \epsilon_d = 4.8$ for FR-4 board and $Z_0 = 25\ \Omega$ (for microstrip). The height of the board is fixed at $\frac{1}{16}$ inch for parallel plate ($\frac{1}{32}$ inch for microstrip). Plugging in the numbers, we get:

$$\left(\frac{w}{h}\right)_{msA} = 5.1086 \text{ (not used)}$$

$$\left(\frac{w}{h}\right)_{msB} = 4.9614$$

$$h_{ms} = \frac{1}{32} \text{ inch} = 0.0794 \text{ cm} \rightarrow w = 0.3938 \text{ cm}$$

Then we stack two microstrips together in series like from Fig. A.5 into A.6 to form 50 Ω parallel plate line. The $\frac{w}{h}$ ratio for parallel plate case is: $\left(\frac{w}{h}\right)_{pp} = 2.4807$.

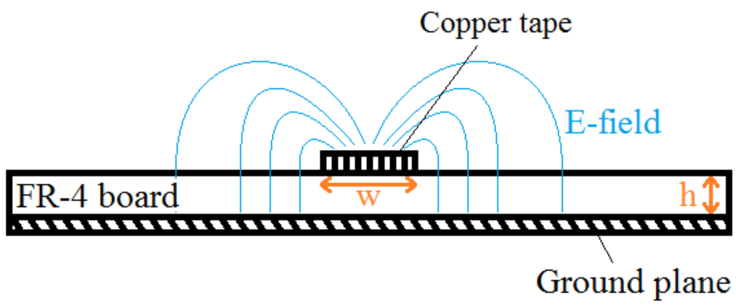

FIGURE A.5: Microstrip line

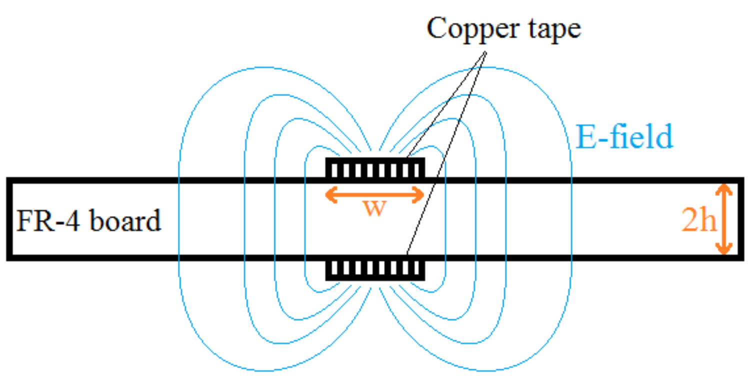

FIGURE A.6: Parallel plate line

Finally, we can figure out the effective permittivity of the board using Eq. A.26 below:

$$\epsilon = \epsilon_0 \left(\frac{\epsilon_d + 1}{2} + \frac{\epsilon_d - 1}{2} \frac{1}{\sqrt{1 + \frac{12h}{w}}} \right) \tag{A.26}$$

$$\epsilon_{pp} = 3.9276\ \epsilon_0 \tag{A.27}$$

In conclusion, we know the width, height, and effective permittivity of the transmission line. The length of the board can be easily calculated after we decide on operating frequency (which gives us the wavelength and separation between resistor elements).

In section 5.1, the dimensions are:

$$\begin{aligned}
W_B &= 10.16\ cm\ \text{(Width of board)} \\
L_B &= 15.24\ cm\ \text{(Length of board)} \\
w_C &= 0.3937\ cm\ \text{(Width of copper tape)} \\
t_B &= 0.15875\ cm\ \text{(Thickness of FR-4 Board)} \\
\epsilon_d &= 4.8\quad \text{(Relative permittivity of FR-4 Board/Epoxy-glass)}
\end{aligned} \tag{A.28}$$

A.11 Details of Location Measurements

A.11.1 CIEMAS

Table A.13: Details of measurement locations in CIEMAS.

Location	Category
Outside 1411	near router
Outside DiVE virtual reality lab	corner
Outside 3467	hall, near router
Outside 3478A	room
3rd floor S wing near stairs NE corner	near router, corner
3rd floor S wing near stairs S corner	near router, corner, near window
3rd floor skyway	hall, near window
2nd floor conference room above Twinnies	room

A.11.2 Teer

Table A.14: Details of measurement locations in Teer.

Location	Category
2nd floor near information board	corner
near 213D	corner, near window
outside basement lab	near router
106	room
114	room
outside 118	near window
outside 423A	near router
breakroom 4rd floor	room, near router
outside P04	near router

A.11.3 Physics

Table A.15: Details of measurement locations in Physics.

Location	Category
Outside 151	hall
Outside 161	near router, corner
Outside 179	near router, hall
101 commons	room
Outside 029	near router, corner
Outside 014	near router, hall
Outside 06	hall, near router, near window
Outside 035	near router
NW basement exit	near router, corner
Outside 094	near router, corner
Outside 285	hall, near router
Outside 0010	near router, hall

A.11.4 French

Table A.16: Details of measurement locations in French.

Location	Category
Outside 2225	hall
Outside 2336	hall, near window
Outside 2313	near router, corner
Conf room near 2313	room, near router
Outside 2303	near router, hall
Outside 2117	hall
Outside 2135	corner, near router, near window
Outside 1124	corner
Outside 1125	corner
Biosci hall in front of 1207	hall, near router
Outside 0069	near router, hall
1243	room
Basement	corner, near router
Outside 0012	hall
Outside 3135B	near window, corner
Outside 3232	hall
Outside 4223	corner
5316	room

Bibliography

[1] P. J. Nahin, "Maxwell's grand unification," *IEEE Spectrum*, vol. 29, no. 3, pp. 45–, March 1992.

[2] G. Rao and S. Mahulikar, "Integrated review of stealth technology and its role in airpower," *The Aeronautical Journal (1968)*, vol. 106, no. 1066, pp. 629–642, 2002.

[3] D. Schurig, J. Mock, B. Justice, S. A. Cummer, J. B. Pendry, A. Starr, and D. Smith, "Metamaterial electromagnetic cloak at microwave frequencies," *Science*, vol. 314, no. 5801, pp. 977–980, 2006.

[4] J. B. Pendry, D. Schurig, and D. R. Smith, "Controlling electromagnetic fields," *science*, vol. 312, no. 5781, pp. 1780–1782, 2006.

[5] T. Liu, X. Cao, J. Gao, Q. Zheng, W. Li, and H. Yang, "Rcs reduction of waveguide slot antenna with metamaterial absorber," *IEEE Transactions on Antennas and Propagation*, vol. 61, no. 3, pp. 1479–1484, March 2013.

[6] X. Sheng and X. Yuanming, "Simulation analysis of an active cancellation stealth system," *Optik-International Journal for Light and Electron Optics*, vol. 125, no. 18, pp. 5273–5277, 2014.

[7] S. Xu and Y. Xu, "Research on active cancelation stealth technique," *Optik-International Journal for Light and Electron Optics*, vol. 125, no. 20, pp. 6219–6222, 2014.

[8] G. G. Macfarlane, "The schornsteinfeger project," http://www.cdvandt.org/cios_xxvi-24.htm, May 1945.

[9] G. T. Ruck, D. E. Barrick, W. D. Stuart, and C. K. Krichbaum, *Radar cross section handbook*. Plenum press New York, 1970, vol. 1.

[10] W. W. Salisbury, "Absorbent body for electromagnetic waves," Jun. 10 1952, uS Patent 2,599,944.

[11] W. Emerson, "Electromagnetic wave absorbers and anechoic chambers through the years," *Antennas and Propagation, IEEE Transactions on*, vol. 21, no. 4, pp. 484–490, Jul 1973.

[12] E. Knott, J. Shaeffer, and M. Tuley, *Radar Cross Section: Its Prediction, Measurements and Reduction.* Massachusetts: Artech House, 1985, p. 235.

[13] J. Nortier, C. A. V. d. Neut, and D. E. Baker, "Table for the design of Jaumann microwave absorber," *Microwave Journal*, 1987.

[14] R. L. Fante and M. T. Mccormack, "Reflection properties of the Salisbury screen," *Antennas and Propagation, IEEE Transactions on*, vol. 36, no. 10, pp. 1443–1454, 1988.

[15] L. J. Du Toit, "The design of Jaumann absorbers," *Antennas and Propagation Magazine, IEEE*, vol. 36, no. 6, pp. 17–25, 1994.

[16] Y. Nikawa, M. Chino, S. Kitahata, and K. Kurata, "Wave absorber," Sep. 14 1999, uS Patent 5,952,953. [Online]. Available: http://www.google.com/patents/US5952953

[17] E. Knott and C. Lunden, "The two-sheet capacitive Jaumann absorber," *Antennas and Propagation, IEEE Transactions on*, vol. 43, no. 11, pp. 1339–1343, 1995.

[18] D. Whitley, "A genetic algorithm tutorial," *Statistics and computing*, vol. 4, no. 2, pp. 65–85, 1994.

[19] G. R. Harik, F. G. Lobo, and D. E. Goldberg, "The compact genetic algorithm," *IEEE transactions on evolutionary computation*, vol. 3, no. 4, pp. 287–297, 1999.

[20] J. D. Kraus, D. A. Fleisch, and S. H. Russ, *Electromagnetics: with applications.* WCB/McGraw-Hill, 1999, p. 199.

[21] B. Chambers and A. Tennant, "Design of wideband jaumann radar absorbers with optimum oblique incidence performance," *Electronics Letters*, vol. 30, no. 18, pp. 1530–1532, 1994.

[22] E. Knott and K. Langseth, "Performance degradation of jaumann absorbers due to curvature," *IEEE Transactions on Antennas and Propagation*, vol. 28, no. 1, pp. 137–139, 1980.

[23] B. Chambers, "Frequency tuning characteristics of an adaptive jaumann radar absorber incorporating variable impedance layers," *Electronics Letters*, vol. 30, no. 22, pp. 1892–1893, 1994.

[24] B. Chambers and A. Tennant, "Optimised design of jaumann radar absorbing materials using a genetic algorithm," *IEE Proceedings-Radar, Sonar and Navigation*, vol. 143, no. 1, pp. 23–30, 1996.

[25] B. A. Munk, P. Munk, and J. Pryor, "On designing jaumann and circuit analog absorbers (ca absorbers) for oblique angle of incidence," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 1, pp. 186–193, Jan 2007.

[26] A. R. Foroozesh, A. Cheldavi, and F. Hodjat, "Design of jaumann absorbers using adaptive genetic algorithm," in *Antennas, Propagation and EM Theory, 2000. Proceedings. ISAPE 2000. 5th International Symposium on*. IEEE, 2000, pp. 227–230.

[27] B. Chambers and A. Tennant, "Optimum design of cylindrical jaumann radar absorbers," in *Antennas and Propagation, 1995., Ninth International Conference on (Conf. Publ. No. 407)*, vol. 1. IET, 1995, pp. 450–454.

[28] A. Charles, M. Towers, and A. McCowen, "Sensitivity analysis of jaumann absorbers," *IEE Proceedings-Microwaves, Antennas and Propagation*, vol. 146, no. 4, pp. 257–262, 1999.

[29] E. Michielssen, J.-M. Sajer, S. Ranjithan, and R. Mittra, "Design of lightweight, broad-band microwave absorbers using genetic algorithms," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, no. 6, pp. 1024–1031, 1993.

[30] A. Cheldavi, "Optimum design of n sheet capacitive jaumann absorber using genetic algorithm," *IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 82, no. 4, pp. 704–706, 1999.

[31] S. Han, Y. Park, and D. Neikirk, "Broadband infrared detection using jaumann absorbers with genetic algorithm," *Electronics Letters*, vol. 41, no. 24, pp. 1307–1308, 2005.

[32] D. S. Weile and E. Michielssen, "Genetic algorithm optimization applied to electromagnetics: A review," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 3, pp. 343–353, 1997.

[33] A. Bajwa, T. Williams, and M. Stuchly, "Design of broadband radar absorbers with genetic algorithm," in *IEEE ANTENNAS AND PROPAGATION SOCIETY INTERNATIONAL SYMPOSIUM*, vol. 4. IEEE; 1999, 2001, pp. 672–675.

[34] P. Saville, "Genetic algorithm optimization of multilayer jaumann absorbers. oblique incidence," DEFENCE RESEARCH AND DEVELOPMENT ATLANTIC DARTMOUTH (CANADA), Tech. Rep., 2005.

[35] R. Vullers, R. van Schaijk, I. Doms, C. Van Hoof, and R. Mertens, "Micropower energy harvesting," *Solid-State Electronics*, vol. 53, no. 7, pp. 684–693, 2009.

[36] M. Piuela, P. D. Mitcheson, and S. Lucyszyn, "Ambient rf energy harvesting in urban and semi-urban environments," *IEEE Transactions on Microwave Theory and Techniques*, vol. 61, no. 7, pp. 2715–2726, July 2013.

[37] Y. Ding, T. Arslan, and A. Hamilton, "Broadband antenna for RF energy scavenging system," in *Antennas and Propagation Conference (LAPC), 2012 Loughborough*. IEEE, 2012, pp. 1–4.

[38] H. J. Visser, A. C. Reniers, and J. A. Theeuwes, "Ambient RF energy scavenging: GSM and WLAN power density measurements," in *Microwave Conference, 2008. EuMC 2008. 38th European*. IEEE, 2008, pp. 721–724.

[39] T. Umeda, H. Yoshida, S. Sekine, Y. Fujita, T. Suzuki, and S. Otaka, "A 950-MHz rectifier circuit for sensor network tags with 10-m distance," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 1, pp. 35–41, 2006.

[40] D. M. Pozar, *Microwave engineering*. John Wiley & Sons, 2009.

[41] I. Bahl and D. Trivedi, "Designers guide to microstrip line," *Microwaves*, vol. 16, no. 5, p. 174, 1977.

Biography

Wiwi Samsul was born March 25, 1988 in Jakarta, Indonesia. He attended SMUK 1 Penabur in Jakarta during which time he won 2nd place in individual physics competition held by Bogor Institute of Agriculture, 2nd place in group science competition held by SMUK 1 Penabur, and 1st place in group physics competition held by Daan Mogot Mall.

Then he attended undergraduate program in Electrical Engineering at University of Pelita Harapan in Karawaci, Banten, Indonesia. He received his Bachelor of Engineering magna cum laude with awards in Student Paper of the Year 2010 and best graduate of his program.

He then attended graduate program in Electrical and Computer Engineering at Duke University in Durham, North Carolina. He received his Master of Science in December 2013 and continued onto doctoral program in the same university. He is currently looking for a position in electrical engineering specifically in RF technology.