FORENSIC CHARACTERIZATION OF RF CIRCUITS

A Dissertation
Submitted to the Faculty
of
Purdue University
by
Anthony Frank Martone

In Partial Fulfillment of the
Requirements for the Degree
of
Doctor of Philosophy

December 2007
Purdue University
West Lafayette, Indiana
To my mother, Susanne Martone. Without her guidance and steadfast belief in me, this document would not have been written.
ACKNOWLEDGMENTS

During the past five years at Purdue University, graduate school has been an exciting and challenging experience. I would like to take this opportunity to thank those who have provided support, encouragement, and friendship.

My advisor Professor Edward J. Delp has been a mentor and motivator during my time at Purdue University. I attribute my success to his ongoing encouragement and challenging approach to learning. Professor Delp’s repeated appeals for me to “think like a scholar” and not like an “undergraduate” has finally come to realization, though the journey has seemed long at times. Special thanks for his patience, guidance, and assistance with writing this dissertation. To him, I owe my sincere gratitude.

I would like to thank my committee members, Professor Charles Bouman, Professor Michael Zoltowski, and Professor Zygmunt Pizlo for their guidance and advice. I would also like to thank Professor DeCarlo for his detailed teaching of “Lumped System Parameters” and jovial humor, Professor Mireille (Mimi) Boutin for her excellent coverage of “Pattern Recognition,” and Professor Mark Bell for his insightful lessons on “Radar Systems.”

I would like to thank the research group of the SIAMES project. Special thanks to Professor Steer, Professor Gard, Professor Chappell, Greg Mazzaro, Andrew Christianson, and Justin Henrie for their research collaboration. Special thanks to Dr. Dev Palmer and ARO for supporting this research.

I would like to thank the microwave branch at ARL for the summer internship opportunity. I learned much during my internship and have applied that knowledge to this research. Special thanks to Karl Kappra, Ken Ranney, Marc Ressler, and Brian Stanton.

I would like to thank my office-mates and friends at Purdue University: Dr. Eugene Lin, Dr. Cuneyt Taskiran, Dr. Yuxin (Zoe) Liu, Dr. Hyung Cook Kim, Dr.
Jinwha Yang, Dr. Sahng-Gyu Park, Dr. Zhen Li, Dr. Yajie Sun, Dr. Greg Cook, Dr. Hwayoung Um, Dr. Josep Predes, Dr. Thomas Talavage, Aravind Mikkilineni, Jennifer Talavage, Limin Liu, Ashok Mariappan, Hakeem Ogunleye, Deen King-Smith, Golnaz Abdollahian, Fengging (Maggie) Zhu, Carlos Wang, Ying Chen, Ka Ki Ng, Liang Liang, Nitin Khanna, Anand Mariappan, Marc Bosch, Oriol Guitart Pla, Rafael Villoria, Pau Sabria, Alex Lipka, Simone Caroti, and Gioia Massa. Special thanks to Nitin Khanna, Ying Chen, Maggie Zhu, and Golnaz Abdollahian for their help in depositing the thesis. Special thanks to Dr. Eugene Lin, Dr. Cuneyt Taskiran, and Aravind Mikkilineni for their camaraderie in the trenches of EE-32 and the VIPER lab. Special thanks to Alex Lipka, Simone Caroti, and Gioia Massa for their friendship and support.

I would especially like to thank all my family for their love and support. Thanks to Héctor, Tara, Niklas, and René Polanco for their support and philosophy of why engineers are inspired. Thanks to Ginger and Terry Monson for their ongoing encouragement and support. Special thanks to my parents, Susanne and Frank Martone. My father has always provided friendship, love, and support during these challenging years at Purdue and for which I am eternally grateful. My mother always encouraged me to “follow my dreams” and “never give up.” This dream has taken a long time to follow, but, WE DID IT! Most importantly, I would like to thank my wife Lindsey (Pepper) Martone. Throughout these five years at Purdue University, Pepper has provided me with unconditional encouragement and love, and without this, I could not have finished this dissertation. Her love, support, and guidance has made me a better researcher and, most importantly, a better person.
This research was supported by a grant from the U.S. Army Research Office as a Multi-disciplinary University Research Initiative on Standoff Inverse Analysis and Manipulation of Electronic Systems under grant number W911NF-05-1-0337.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF TABLES</td>
<td>ix</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>xi</td>
</tr>
<tr>
<td>ABBREVIATIONS</td>
<td>xv</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>xvi</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Our Approach to RF Device Characterization</td>
<td>4</td>
</tr>
<tr>
<td>1.2 Sources of Nonlinearities</td>
<td>5</td>
</tr>
<tr>
<td>1.3 RF Device Characterization Using Two-Tone Probe Signals</td>
<td>6</td>
</tr>
<tr>
<td>1.4 Linear Chirp Signals</td>
<td>11</td>
</tr>
<tr>
<td>1.5 Proposed Forensic Characterization System</td>
<td>15</td>
</tr>
<tr>
<td>1.6 Other Applications of Forensic Fingerprint</td>
<td>17</td>
</tr>
<tr>
<td>1.6.1 RF Circuit Forensics</td>
<td>19</td>
</tr>
<tr>
<td>1.7 Outline of Dissertation</td>
<td>21</td>
</tr>
<tr>
<td>1.8 Contributions</td>
<td>21</td>
</tr>
<tr>
<td>2 PROBE SIGNAL DESIGN</td>
<td>23</td>
</tr>
<tr>
<td>2.1 Two-Tone Probe Signals</td>
<td>23</td>
</tr>
<tr>
<td>2.1.1 Third Order Analysis</td>
<td>26</td>
</tr>
<tr>
<td>2.2 Windowed Linear Chirp Signal</td>
<td>29</td>
</tr>
<tr>
<td>2.2.1 Estimation of the Filter Response Using the Gaussian Chirp Probe Signal</td>
<td>40</td>
</tr>
<tr>
<td>2.3 Linear Chirp Probe Signal</td>
<td>40</td>
</tr>
<tr>
<td>3 FEATURES FOR DEVICE CHARACTERIZATION</td>
<td>44</td>
</tr>
<tr>
<td>3.1 Features For The Two-Tone Probe Signal</td>
<td>44</td>
</tr>
<tr>
<td>3.2 Features for the Windowed Linear Chirp Probe Signal</td>
<td>45</td>
</tr>
</tbody>
</table>
3.3 Features for the Linear Chirp Probe Signal ................................ 51

4 REVIEW OF CLASSIFIERS ......................................................... 53
   4.1 Bayesian Classifiers ......................................................... 57
      4.1.1 Discriminate Functions ............................................. 58
      4.1.2 Bayesian Classifier .................................................... 59
      4.1.3 Bayesian Classifier .................................................... 59
   4.2 Parzen Window Classifier .................................................... 60
   4.3 K Nearest Neighbor Classifier ............................................ 63
   4.4 Binary Tree Classifier ....................................................... 64
      4.4.1 Splitting Rule and Terminal Nodes ............................... 67
      4.4.2 Pruning ................................................................. 68
      4.4.3 Binary Tree Classifier ................................................ 68
   4.5 Support Vector Machines ................................................... 70
      4.5.1 Nonlinear Support Vector Machine ................................. 72
      4.5.2 Support Vector Machine Classifier ............................... 72

5 SIMULATIONS AND EXPERIMENTS ........................................... 74
   5.1 RF Circuit Models .......................................................... 74
      5.1.1 Signal to Noise Ratio ............................................... 78
   5.2 Two-Tone Signal Simulations ............................................. 79
      5.2.1 Probe Signal Design ................................................... 80
      5.2.2 RF Circuit Model Design ............................................ 80
      5.2.3 Return Signals ......................................................... 83
      5.2.4 Training and Testing .................................................. 85
      5.2.5 Classification Results ................................................. 87
      5.2.6 Conclusions ............................................................ 90
   5.3 Gaussian Chirp Signal Simulations ..................................... 90
      5.3.1 Probe Signal Design ................................................. 91
      5.3.2 Circuit Model Design ................................................. 93
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.3.3</td>
<td>Generation of the Return Signals</td>
<td>96</td>
</tr>
<tr>
<td>5.3.4</td>
<td>Features Selected at Regular Intervals</td>
<td>97</td>
</tr>
<tr>
<td>5.3.5</td>
<td>Features Selected at Precise Locations</td>
<td>103</td>
</tr>
<tr>
<td>5.3.6</td>
<td>Conclusions</td>
<td>110</td>
</tr>
<tr>
<td>5.4</td>
<td>Linear Chirp Probe Signal Simulations and Experiments</td>
<td>110</td>
</tr>
<tr>
<td>5.4.1</td>
<td>Linear Chirp Experiments</td>
<td>111</td>
</tr>
<tr>
<td>5.4.2</td>
<td>Linear Chirp Simulations</td>
<td>120</td>
</tr>
<tr>
<td>5.4.3</td>
<td>Conclusions</td>
<td>129</td>
</tr>
<tr>
<td>5.5</td>
<td>Summary of Simulations</td>
<td>130</td>
</tr>
<tr>
<td>6</td>
<td>CONCLUSIONS AND FUTURE WORK</td>
<td>132</td>
</tr>
<tr>
<td>6.1</td>
<td>Contributions</td>
<td>132</td>
</tr>
<tr>
<td>6.2</td>
<td>Future Work</td>
<td>133</td>
</tr>
<tr>
<td>6.3</td>
<td>Publications</td>
<td>134</td>
</tr>
<tr>
<td>LIST OF REFERENCES</td>
<td></td>
<td>137</td>
</tr>
<tr>
<td>APPENDICES</td>
<td></td>
<td>146</td>
</tr>
<tr>
<td>A</td>
<td>Two-Tone Signal in the Time Domain</td>
<td>146</td>
</tr>
<tr>
<td>B</td>
<td>Two-Tone Signal in the Frequency Domain</td>
<td>150</td>
</tr>
<tr>
<td>C</td>
<td>Fourier Transform of the Gaussian Chirp Signal</td>
<td>154</td>
</tr>
<tr>
<td>D</td>
<td>Power Spectrum of the Gaussian Chirp Signal</td>
<td>163</td>
</tr>
<tr>
<td>VITA</td>
<td></td>
<td>170</td>
</tr>
</tbody>
</table>
LIST OF TABLES

Table | Page
---|---
2.1 Two-Tone Probe Signal Parameters. | 24
2.2 Parameter Definitions for Constants in $y(t)$. | 28
2.3 Linear Chirp Probe Signal Parameters. | 30
2.4 Windowed Chirp Signal Parameters. | 32
3.1 Four Features Used for the Two-Tone Probe Signal. | 45
5.1 Power Series Coefficients Used by the Circuit Models in the Two-Tone Simulations. | 82
5.2 Average SNR and IMR for the RF Circuit Models Used in the Two-Tone Simulations. | 85
5.3 Number of Feature Vectors Per Category for the Training Set for the Two-Tone Simulations. | 87
5.4 Number of Feature Vectors Per Category for the Testing Set for the Two-Tone Simulations. | 87
5.5 Average Classification Accuracies for the Two-Tone Simulations. | 89
5.6 Gaussian Chirp Signal Parameters. | 92
5.7 Circuit Model Parameters Used in the Gaussian Chirp Simulations. | 94
5.8 Average SNR for Each Circuit Model and Each Noise Power. SNR is Defined in dB. | 97
5.9 Number of Feature Vectors Per Category for the Training Set. Features are Sampled at Regular Intervals. | 100
5.10 Number of Feature Vectors Per Category for the Testing Set. Features are Sampled at Regular Intervals. | 100
5.11 Average Classification Accuracies for the Gaussian Chirp Simulations Using Features Sampled at Regular Intervals. | 102
5.12 Number of Feature Vectors Per Category for the Training Set. Features are Sampled at Precise Locations. | 108
Table | Page
--- | ---
5.13 Number of Feature Vectors Per Category for the Testing Set. Features are Sampled at Precise Locations. | 108
5.14 Average Classification Accuracies for the Gaussian Chirp Simulations Using Features Sampled at Precise Locations. | 110
5.15 Number of Feature Vectors Per Category for the Training Set. Features are Used in the Linear Chirp Experiments. | 117
5.16 Number of Feature Vectors Per Category for the Testing Set. Features are Used in the Linear Chirp Experiments. | 118
5.17 Classification Results for the Linear Chirp Experiment. | 120
5.18 Number of Feature Vectors Per Category for the Training Set. Features Vectors are Used in the Linear Chirp Simulations. | 126
5.19 Number of Feature Vectors Per Category for the Testing Set. Features Vectors are Used in the Linear Chirp Simulations. | 126
5.20 Classification Results for the Linear Chirp Simulation. | 129
A.1 Parameter Definitions for the Constants in \( y(t) \). | 149
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Example Filter Response of a Chebyshev Bandpass Filter. A Distinct Oscillation Pattern is Shown in the Passband of the Filter Response.</td>
<td>4</td>
</tr>
<tr>
<td>1.2 RF Device Measurement System for the Walkie-Talkie.</td>
<td>7</td>
</tr>
<tr>
<td>1.3 Power Spectrum of the Walkie-Talkie.</td>
<td>8</td>
</tr>
<tr>
<td>1.4 Intermodulation Distortion Amplitude for the Power Spectrums in $P_r$.</td>
<td>9</td>
</tr>
<tr>
<td>1.5 Example of a Linear Chirp Probe Signal. As is Shown, the Frequency of Oscillations Increases With Respect to Time.</td>
<td>12</td>
</tr>
<tr>
<td>1.6 Example of Instantaneous Frequency of the Linear Chirp Signal.</td>
<td>13</td>
</tr>
<tr>
<td>1.7 Proposed RF Device Characterization System.</td>
<td>16</td>
</tr>
<tr>
<td>1.8 Example of Probe Signal Input Into RF Device.</td>
<td>16</td>
</tr>
<tr>
<td>2.1 Example of a Two-Tone Signal in the Time Domain and Frequency Domain. $f_1 = 40$MHz, $f_2 = 40.1$MHz, and $t$ ranges from 0 to $2 \times 10^{-6}$ seconds.</td>
<td>25</td>
</tr>
<tr>
<td>2.2 Example Output from a Nonlinearity of Order M=3 When the Input is a Two-Tone Signal. $a_1 = a_2 = a_3 = 1$, $f_1 = 40$MHz, $f_2 = 40.1$MHz, and $t$ ranges from 0 to $2 \times 10^{-6}$ seconds.</td>
<td>27</td>
</tr>
<tr>
<td>2.3 Example of a Gaussian Chirp Signal in the Time Domain, Where Parameters are Defined in Table 2.4.</td>
<td>31</td>
</tr>
<tr>
<td>2.4 Output from the Nonlinearity of Order $M = 3$, Where the Input is the Gaussian Chirp Signal.</td>
<td>33</td>
</tr>
<tr>
<td>2.5 Comparison of the Power Spectrum of the Gaussian Chirp Probe Signal $P_x(t)$ with the Output of the Nonlinearity $P_y(f)$. Parameters are Defined in Table 2.4.</td>
<td>36</td>
</tr>
<tr>
<td>2.6 Example Output of a Third Ordered Nonlinear System. Each GHD is Generated by a Square-Term.</td>
<td>37</td>
</tr>
<tr>
<td>2.7 Multiplication of Two Gaussian Functions. The Overlap is Dependent on the Intercept Point. As is Illustrated, $Y_1 \ast Y_2$ is the Multiplication of $Y_1$ and $Y_2$.</td>
<td>39</td>
</tr>
<tr>
<td>Figure</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>2.8</td>
<td>41</td>
</tr>
<tr>
<td>2.9</td>
<td>42</td>
</tr>
<tr>
<td>2.10</td>
<td>43</td>
</tr>
<tr>
<td>2.11</td>
<td>43</td>
</tr>
<tr>
<td>3.1</td>
<td>47</td>
</tr>
<tr>
<td>3.2</td>
<td>48</td>
</tr>
<tr>
<td>3.3</td>
<td>49</td>
</tr>
<tr>
<td>3.4</td>
<td>50</td>
</tr>
<tr>
<td>3.5</td>
<td>52</td>
</tr>
<tr>
<td>4.1</td>
<td>56</td>
</tr>
<tr>
<td>4.2</td>
<td>59</td>
</tr>
<tr>
<td>4.3</td>
<td>60</td>
</tr>
<tr>
<td>4.4</td>
<td>62</td>
</tr>
<tr>
<td>4.5</td>
<td>65</td>
</tr>
<tr>
<td>4.6</td>
<td>66</td>
</tr>
<tr>
<td>4.7</td>
<td>69</td>
</tr>
<tr>
<td>4.8</td>
<td>70</td>
</tr>
</tbody>
</table>

Figure 2.8 Example Fourier Transform of the Gaussian Chirp Probe Signal.
Figure 2.9 Example Response of a Filter Using a Passband Between 400MHz and 450MHz. This Filter is Probed Using the Gaussian Chirp Probe Signal.
Figure 2.10 Linear Chirp Signal is Input into a Filter. The Output Signal Contains an Estimate of the Filter Response.
Figure 2.11 Example Power Spectrum of the Filter Response for a K&L Microwave Filter.

3.1 Sampling Locations of the Power Spectrum at Regular Intervals (No Filter).
3.2 Sampling Locations of the Power Spectrum at Regular Intervals (Filter Present).
3.3 Sampling Locations of the Power Spectrum at Precise Locations (No Filter).
3.4 Sampling Locations of the Power Spectrum at Precise Locations (Filter Present).
3.5 Filter Response Sampled at Regular Intervals.

4.1 Block Diagram of Performance Evaluation.
4.2 Block Diagram of the Bayesian\(_\mu\) Classifier and the Performance Evaluation Process.
4.3 Block Diagram of the Bayesian\(_\Sigma\) Classifier and the Performance Evaluation Process.
4.4 Block Diagram of the Parzen Window Classifier and the Performance Evaluation Process.
4.5 Block Diagram of the K Nearest Neighbor Classifier and the Performance Evaluation Process.
4.6 Example of a Binary Tree. The Test Vector is Input into the Root Node and the Features are Compared to Thresholds. A Category is Decided at the Terminal Nodes.
4.7 Block Diagram of the Binary Tree Classifier and the Performance Evaluation Process.
4.8 Example of the Feature Space \(\mathbb{R}^2\) Used by the Support Vector Machine. The X’s Denote the position of the Training Vectors in Region \(R_1\) and the O’s Denote the Position of the Training Vectors in Region \(R_2\).
Figure Page
4.9 Block Diagram of the Support Vector Machine Classifier and the Performance Evaluation Process. .......................... 73
5.1 RF Circuit with Filter Present. .................................................. 75
5.2 RF Circuit Model - Filter Present. ........................................... 76
5.3 RF Scenario - No Filter Present. .............................................. 77
5.4 RF Circuit Model - No Filter Present. ...................................... 77
5.5 Noise Model. ........................................................................ 78
5.6 Power Spectrums Generated by the 6 Circuit Models in Response to a Two-Tone Probe Signal. ................................. 81
5.7 Example of the Gaussian Chirp Probe Signal. Parameters of the Probe Signal are Defined in Table 5.6. ......................... 92
5.8 Example Power Spectrums of the Return Signal From Each Circuit Model. .......................................................... 95
5.9 Example of Power Spectrums Generated by Circuit Models \( \Upsilon_1, \Upsilon_3, \Upsilon_4, \) and \( \Upsilon_5 \) in Response to a Gaussian Chirp Signal Using the Noise Power Spectrum \( \sigma^2_c \). .......................................................... 103
5.10 Estimated Filter Response of the Power Spectrum of by \( \Upsilon_1 \) in response to a Gaussian Chirp Signal. Features are Precisely Chosen in the Passband of the Filter Response. .......................................................... 105
5.11 Gaussian Harmonic Distortion of \( \Upsilon_2 \) in response to a Gaussian Chirp Signal. Features are Precisely Chosen at Peaks of Each Harmonic. .......... 106
5.12 Estimated Filter Response of a Power Spectrum of \( \Upsilon_3 \) in response to a Gaussian Chirp Signal. Features are Precisely Chosen in the Passband of the Filter Response. .......................................................... 107
5.13 Block Diagram of the Two RF Circuits Used in the Linear Chirp Experiments. ......................................................... 111
5.14 RF Circuit Measurement System Used in the Linear Chirp Experiments. ................................................................. 112
5.15 Example of the Probe Signal Used in the Linear Chirp Experiments. .......................................................................... 113
5.16 Power Spectrums for the Linear Chirp Experiments. As is Illustrated, the Passband Ripple is Distinct. ......................... 121
5.17 Circuit Models Used in the Simulations. .................................... 122
5.18 Example Power Spectrum from the Linear Chirp Experiments. The noise is between -55dB and -45dB. ....................... 123
<table>
<thead>
<tr>
<th>Figure</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.19</td>
<td>128</td>
</tr>
<tr>
<td>C.1</td>
<td>157</td>
</tr>
</tbody>
</table>

5.19 Power Spectrums for the Linear Chirp Simulations. The Passband Ripple is Not Distinct.

C.1 Right Triangle Where $\gamma$ is the Length of the Hypothenuse.
ABBREVIATIONS

RF         Radio Frequency
FOD        First Order Distortion
IMD        Intermodulation Distortion
dB         Decibels
PIMD       Passive Intermodulation Distortion
IMR        Intermodulation Distortion Ratio
UWB        Ultra Wideband
FCC        Federal Communication Commission
SVM        Support Vector Machine
BT         Binary Tree
PW         Parzen Window
K-NN       K-Nearest Neighbor
EMIS       Electromagnetic Induction Spectroscopy
CBCD       Content-Based Copy Detection
EP         Electrophotographic
PRNU       Photoresponse Nonuniformity
ISM        Industrial Scientific Medical
GHD        Gaussian Harmonic Distortion
CART       Classification and Regression Tree
SNR        Signal to Noise Ratio
MHz        Megahertz
ns         Nanoseconds
ABSTRACT


Given the wide use of Radio Frequency (RF) devices for applications ranging from data networks to wireless sensors, it is of interest to identify the types of RF devices that are located in an environment. For example, one could be on campus at a university and would want to prevent their RF device from interfering with other devices. There are currently two techniques that are used to identify RF devices in the environment, which are known as passive and active fingerprinting. These fingerprinting techniques require that the RF device comply to some transmission standard (IEEE 802.11) and can therefore only be used to identify a subset of RF devices.

We propose an approach that can theoretically be used to identify any RF device. The proposed approach is used to determine sources of nonlinearities and identify filters that are present in the RF device. In our approach, a probe signal is transmitted to the RF device. The RF device retransmits or reradiates a return signal in response to the probe signal. The properties of the return signal are measured and these properties can be shown to contain distinct information that is inherent to sources of the nonlinearities and the filters of the RF device. This information constitutes features that can be used to identify the RF device.

In an effort to identify sources of nonlinearities, we will analyze the output signal obtained from a RF circuit model when the input is either a two-tone probe signal or a windowed linear chirp probe signal. It will be shown that the power spectral density of the output signal contains distinct features caused by nonlinearities in the RF circuit. These features are used to identify the nonlinearity and consequently the
RF device. In addition, we will analyze the output signal from the bandpass filter in the RF circuit when the input is either a windowed linear chirp probe signal or a linear chirp probe signal.

In this dissertation, we will use simulations to verify our methods. For each simulation, several RF circuit models are considered, where each circuit model contains a unique nonlinearity and bandpass filter. The features are classified using statistical pattern recognition techniques. Some of our results indicate that our simulation approach needs further investigation to validate it against actual measurement data.
1. INTRODUCTION

Given the wide use of Radio Frequency (RF) devices for applications ranging from data networks to wireless sensors, it is of interest to identify the types of RF devices that are located in an environment. For example, one could be on campus at a university and would want to prevent their RF device from interfering with other devices. This becomes the problem of determining the properties of an RF device/circuit by sending it a carefully designed signal, known as the probe signal $p(t)$. It can be shown that this probe will encounter the RF circuit and will be reradiated by the circuit, whether it is powered or not [1]. By examining the reradiated signal, which will be referred to here as the return signal $r(t)$, one can determine the type of RF device that is in the environment. Forensic characterization, or characterization, of the RF device refers to determining the make, model, configuration, and other characteristics of the device based on observations of the return signal that the device produces. The characteristics of the return signal that identify the RF device are known as a device signature.

To forensically characterize an RF device, we consider the distortions that are generated in the circuitry of the device in response to the probe signal, where the distortion will be defined as a modification of the original probe signal characteristics [2]. The distortion considered in this research is generated by nonlinear components, such as mixers, transistors, diodes, and passive connectors. An overview of the nonlinear sources that are commonly found in RF devices is presented in Section 1.2. When stimulated by the probe signal, the nonlinearity generates distortions at a priori frequency locations in the return signal spectrum. The amplitude/power at these frequency locations are distinct to the nonlinearity, and the device, and can therefore be used to characterize the RF device. For example, consider a two-tone probe signal, which is defined as
\[ p(t) = A_1 \cos(2\pi f_1 t + \alpha_1) + A_2 \cos(2\pi f_2 t + \alpha_2) \quad (1.1) \]

where \( A_1 \) and \( A_2 \) are constant amplitudes, \( f_1 \) and \( f_2 \) are the probe frequencies of the two-tone signal, and \( \alpha_1 \) and \( \alpha_2 \) are phase angles. Let the probe frequency \( f_2 = f_1 + \Delta \), where \( \Delta \) is the frequency offset. When the two-tone signal is input into a nonlinearity, the nonlinearity generates distortion at the following discrete frequencies \([3]\)

\[ f_D(n_1, n_2) = n_1 f_1 + n_2 f_2 \quad (1.2) \]

where \( n_1 \) and \( n_2 \) are positive integers. The order of the distortion is defined as

\[ O = |n_1| + |n_2| \quad (1.3) \]

For the case when \( \{n_1 = 1, n_2 = 0\} \) or \( \{n_1 = 0, n_2 = 1\} \), distortion occurs at the probe frequencies and is known as first order distortion (FOD). For the case when \( n_1 = 0 \) or \( n_2 = 0 \) in Equation 1.2 and \( O > 1 \) in Equation 1.3, the distortion is known as harmonic distortion \([4]\) of the two-tone signal and is located at \([5]\)

\[ f_{har}(n_1) = n_1 f_1 \quad (1.4) \]

\[ n_1 > 1 \text{ and } n_2 = 0, \]

\[ f_{har}(n_2) = n_2 f_2 \quad (1.5) \]

when \( n_2 > 1 \) and \( n_1 = 0 \). The harmonic distortion that occurs at \( O = n \) is known as the \( n^{th} \) order harmonic, where \( n \) is a positive integer and \( n > 1 \). For the case when \( n_1 \neq 0 \) and \( n_2 \neq 0 \) in Equation 1.2, the distortion is known as intermodulation distortion (IMD) \([6]\) and is located at \([7]\)

\[ f_{imd}(n_1, n_2) = n_1 f_1 + n_2 f_2 \quad (1.6) \]
when \( n_1 \neq 0 \) and \( n_2 \neq 0 \). The IMD that occurs at \( O = n \) will be referred to as the \( n^{th} \) order IMD, where \( n \) is a positive integer and \( n > 1 \).

Another way to forensically characterize an RF device is to analyze the response from an individual device component. As will be shown later, a linear chirp probe signal, or linear chirp signal, is used to characterize the frequency response [8], or filter response, of a bandpass filter in the RF device. The linear chirp signal is defined as

\[
p(t) = A \cos(2\pi(f_0 + \frac{(f_1 - f_0)}{2t_1}t)t) \quad (1.7)
\]

where \( A \) is the amplitude, \( f_0 \) is the initial frequency at time \( t = 0 \), \( f_1 \) is the final frequency at time \( t_1 \), and \( t_1 \) is the offset time between \( f_0 \) and \( f_1 \). The linear chirp signal probes a continuum of frequencies, or probe frequencies, from \( f_0 \) to \( f_1 \). As will be discussed in Section 1.1, the filter response is assumed to be known a priori. For example, consider the return signal \( r(t) \) that is generated in response to a Chebyshev \( 4^{th} \) order bandpass filter. This type of filter is used in the experiments of Section 5.4.1 of Chapter 5. The power spectrum of \( r(t) \) is shown in Figure 1.1 and defined as [9]

\[
P_r(f) = R(f)R^*(f) \quad (1.8)
\]

where \( R^*(f) \) is the complex conjugate of \( R(f) \), and \( R(f) \) is the Fourier transform [9]

\[
R(f) = F[r(t)] = \int_{-\infty}^{\infty} r(t)e^{-i2\pi ft}dt \quad (1.9)
\]

As shown in Figure 1.1, distinct oscillation patterns are present in the filter passband. This pattern can be used to characterize the RF device. The amplitudes in Figure 1.1 are measured in units of decibels (dB), where for an arbitrary power value \( P_a \)

\[
P_{dB} = 10\log_{10}\left(\frac{P_a}{P_{ref}}\right) \quad (1.10)
\]
where $P_{ref}$ is the reference power level. It is also possible to express power in dBm, where $P_{ref} = 0.001$ watts, which is decibels referenced at 1 milliwatt of power.

1.1 Our Approach to RF Device Characterization

The approach presented in this dissertation is used to characterize/identify a finite number of RF devices. Each RF device type is assumed to be known a priori. We therefore have knowledge of each source of nonlinearity and knowledge of the filter properties in each RF device. This a priori information is used in the design of probe signals. The probe signal is designed such that, when input into the RF device, a

![Example Filter Response of a Chebyshev Bandpass Filter](image)

Fig. 1.1. Example Filter Response of a Chebyshev Bandpass Filter. A Distinct Oscillation Pattern is Shown in the Passband of the Filter Response.
unique signature is generated at specified frequency locations in the return signal. For example, if a two-tone probe signal is input into a nonlinearity, IMD/power is produced at known frequency locations in the return signal and are used as the RF forensic device signature. As another approach to forensic analysis developed here, consider the filter shown in Figure 1.1 (presented above). Since the filter is known a priori, the frequency range of the passband is known a priori. Therefore, the probe signal can be designed to estimate the response of the filter and use this as a forensic signature.

1.2 Sources of Nonlinearities

A variety of nonlinearities exists in an RF device. Common nonlinearities include mixers, transistors, diodes, and passive components such as connectors and cables. Mixers are commonly used circuit components in superheterodyne RF receivers and constitute the primary sources of nonlinear distortions [10] in RF devices. The mixer operation is designed to move or translate the input signal in the frequency domain [11]. This operation is known as down conversion. A mixer is essentially a multiplier where the received signal (or probe signal) is multiplied by a sinusoid known as the local oscillator signal. For example, consider $X_1 = \cos(2\pi f_1 t)$ as the probe signal and $X_2 = \cos(2\pi f_2 t)$ as the local oscillator signal. The mixer operation then becomes $Y = X_1 X_2 = \cos(2\pi f_1 t) \cos(2\pi f_2 t) = 0.5(\cos(2\pi(f_1 - f_2)t) + \cos(2\pi(f_1 + f_2)t))$. The result of the multiplication is a sum of two cosine functions with “mixed” frequencies at $f_1 - f_2$ and $f_1 + f_2$. By low pass filtering $Y$, one can obtain the downconverted signal with a frequency of $(f_1 - f_2)$. When $X_1$ and $X_2$ are multiplied in a circuit, mixing currents are created in the mixer at harmonic multiples [12], thereby leading to the nonlinear operation of the mixer. This nonlinear behavior has been used to characterize and model the mixer [13] [14].

In addition to mixers, diodes are also sources of nonlinearities. Diodes are fabricated as the union of two doped semiconductors [2]. The junction of these two
semiconductors is a distinct nonlinearity [2]. Transistors are also sources of non-linearities. For example, bipolar transistors are a union of multiple semiconductors and have a distinct nonlinearity [2]. The nonlinear behavior of transistors is used to characterize the transistors performance. For example, the IMD can be used to characterize, model, and measure the performance of transistors [15] [16] [17].

When a two-tone signal is input into passive components, passive intermodulation distortion (PIMD) is generated [3] [18]. The actual mechanisms that cause nonlinear effects are not well understood [19]. However, the following have been identified to generate PIMD [20]: electron tunnelling at metallic contacts [21], planar transmission lines [22], coaxial connectors [23], micro discharge in metallic structure, dirt on metal surfaces, resistivity of carbon fiber, hysteresis effects of ferromagnetic materials, loose connections, cracks in metals, and oxidation at joints. PIMD generation is therefore highly dependent upon the quality and structure of the material. Any defect in the material can lead to the unwanted generation of PIMD. The PIMD has been used to measure the defects in the material and also used to characterize sources of the passive components [24].

1.3 RF Device Characterization Using Two-Tone Probe Signals

As discussed in Section 1.2, RF devices contain a wide variety of nonlinearities. The nonlinearities generate information in response to probe signals that can be used to characterize the RF device. Several probe signals have been used for RF device characterization purposes [25] [26]. One commonly used probe signal is the two-tone signal [27]. The two-tone signal was defined in Equation 1.1 and can be used to probe the nonlinearity. As discussed, the two-tone probe signal, \( p(t) \), is input into an RF device and the response signal, \( r(t) \), is returned. If nonlinearities are present in the RF device, then distortion is generated at the frequency locations specified by Equation 1.2. This distortion consists of first order distortion (FOD), harmonic
Fig. 1.2. RF Device Measurement System for the Walkie-Talkie.

distortion, and intermodulation distortion (IMD). The power spectrum of the return signal is used to analyze this distortion.

To study the distortion, consider the set of measured power spectrums of return signals $P_r = \{P_{r_1}(f), ... P_{r_{517}}(f)\}$ [28]. These power spectrums were generated by a “walkie-talkie”\(^1\) in response to the following set of two-tone probe signals $\{p_1(t), ... p_{517}(t)\}$, where $p_j(t) = \cos(2\pi\alpha_j t) + \cos(2\pi\beta_j t)$, and $\alpha_j = 240.5 \times 10^6 + \frac{i}{2}$, $\beta_j = \alpha_j + 0.1 \times 10^6$, and $j \in \{1, 2, ..., 517\}$. $\alpha_j$ and $\beta_j$ are the probe frequencies of $p_j(t)$.

The system used to measure $P_r$ is shown in Figure 1.2 \(^2\). For iteration $i$, the waveform $p_i(t)$ is generated and sent to the RF device through the coupler. The return signal is sent to a spectrum analyzer through the coupler. The spectrum analyzer measures 216840 samples between 240MHz and 500MHz.

Example power spectrums for $j = 19$, $\alpha_{19} = 250$MHz, and $\beta_{19} = 250.1$MHz are shown in Figure 1.3. As shown in the Figure, the amplitudes of the FOD, located at $\alpha_{19} = 250$MHz and $\beta_{19} = 250.1$MHz, are -31.5 dBm and -30.33dBm.

\(^1\)A walkie-talkie is a hand-held RF device (receiver and transmitter) that transmits and receives information to and from other walkie-talkies.

\(^2\)The set of power spectrums were generated at North Carolina State University.
The amplitudes of the second order harmonics, located at $2\alpha_{19} = 500\text{MHz}$ and $2\beta_{19} = 500.2\text{MHz}$, are -58.64dBm and -58.26dBm. The third order IMD’s, located at $(2\alpha_{19} - \beta_{19}) = 249.1\text{MHz}$ and $(2\beta_{19} - \alpha_{19}) = 250.2\text{MHz}$, are not visible in the Figure, since they are close to the noise floor, where the noise floor is the minimum signal level that can be detected [12]. The noise floor is used to describe the power level of the noise in the experiment and is shown in Figure 1.3(d). The third order IMD is determined as $P_{r19}(2\alpha_{19} - \beta_{19}) = -81.6\text{dBm}$ and $P_{r19}(2\beta_{19} - \alpha_{19}) = -84.8\text{dBm}$.

To determine if the third order IMD is detectable, we shall examine Figure 1.4(a) for the following amplitudes $\{P_{r1}(2\alpha_{1} - \beta_{1}), ... P_{r517}(2\alpha_{517} - \beta_{517})\}$. As shown in
the Figure, a pattern is indicated, which suggests that the third order IMD is detectable for this example. In addition, we will examine the fifth order IMD using \( \{P_r(3\alpha_1 - 2\beta_1), ... P_r\alpha_{517}(3\alpha_{517} - 2\beta_{517})\} \), which is shown in Figure 1.4(b). As is shown in the Figure, no pattern is indicated, which suggests that the fifth order IMD is not detectable for this example.

(a) Third Order. A Unique Pattern is Indicated.

(b) Fifth Order. No Pattern is Indicated.

Fig. 1.4. Intermodulation Distortion Amplitude for the Power Spectrums in \( P_r \).
This experiment shows that the FOD, the second order harmonic, and the third order IMD is present in the power spectrum presented in Figure 1.3(a). It was shown that the fifth order IMD, and consequently the higher order IMD’s, were not detectable. This information will be used for the design of several circuit models that will be used for the simulations in Section 5.2 of Chapter 5. Although the second order harmonic is present in this experiment, typically it is discarded by the bandpass of the circuit [2].

A variety of methods can be used to measure the nonlinear performance of an RF device. Some performance measures include adjacent channel power ratio, co-channel power ratio, noise-to-power ratio [29], the third-order intercept point [6], and the intermodulation distortion ratio (IMR) [30]. The performance of the nonlinearities, and consequently the performance rating of the RF device, are dependent on these testing schemes [2] [7]. In addition to the performance measures, models of the nonlinear source can be designed. By modeling the sources of the nonlinearity, accurate prediction and control of the IMD is achieved [31]. Modeling is also a way to lower the computational complexity of direct analysis [32].

We will measure nonlinear performance using the IMR. The IMR is used since it measures the contribution of all detectable IMD. Since the third order IMD was the only detectable IMD in the experiment of this Section, it will be used to measure the IMR. The IMR is defined as [2] [7]

\[ IMR_L = \frac{P_r(2f_1 - f_2)}{P_r(f_1)} \]  

or

\[ IMR_U = \frac{P_r(2f_2 - f_1)}{P_r(f_2)} \]  

where \( P_r(f) \) is the power spectrum of the return signal \( r(t) \), \( f_1 \) and \( f_2 \) are the probe frequencies of the two-tone signal.
1.4 Linear Chirp Signals

The linear chirp probe signal was described by Equation 1.7. The linear chirp signal is used in a variety of applications including pulse compression [33], radar [34], sonar [35], and spread spectrum communications [36]. The linear chirp signal is

\[ p(t) = A \cos(2\pi(f_0 + \frac{(f_1 - f_0)}{2t_1})t) \] (1.13)

where \( A \) is the amplitude, \( f_0 \) is the initial frequency at time \( t = 0 \), \( f_1 \) is the final frequency at time \( t_1 \), and \( t_1 \) is the offset time between \( f_0 \) and \( f_1 \). The instantaneous frequency of the signal is defined by [37]:

\[ f_i = f_0 + k t \] (1.14)

where

\[ k = \frac{(f_1 - f_0)}{t_1} \] (1.15)

One unique property of the linear chirp signal is that its instantaneous frequency increases over time. The increase in frequency is noticeable in the time domain for the linear chirp signal as illustrated in Figure 1.5, where \( f_0 = 1 \text{MHz}, f_1 = 100 \text{MHz}, t_1 = 2 \text{ microseconds (\mu s)} \). An example of the instantaneous frequency is shown in Figure 1.6.

The increase in instantaneous frequency allows the linear chirp signal to have a relatively large bandwidth. For example, it can be used to transmit a low peak power signal over a long time duration while maintaining a large bandwidth. The bandwidth of the chirp signal \( p(t) \) is defined as [38]

\[ B^2 = \frac{1}{E} \int_{-\infty}^{\infty} (2\pi f)^2 |P(f)|^2 df \] (1.16)

\(^3\)It should be noted that there are many definitions of “bandwidth” of a signal. We are using this definition for chirp signals to illustrate the bandwidth properties of the signal.
Fig. 1.5. Example of a Linear Chirp Probe Signal. As is Shown, the Frequency of Oscillations Increases With Respect to Time.
Fig. 1.6. Example of Instantaneous Frequency of the Linear Chirp Signal.
where \(|P(f)|\) is the magnitude of the Fourier transform of \(p(t)\), and \(E\) is the energy of \(p(t)\) defined by [9]

\[
E = \int_{-\infty}^{\infty} |p(t)|^2 dt
\]  

(1.17)

Pulse compression was first introduced in 1951 by Bernard Oliver at Bell Labs [39] [40]. Pulse compression is useful for applications that have peak signal limitations. To increase the power in a signal, one can either increase its amplitude or increase its time duration. However, increasing its time duration will decrease its bandwidth. The linear chirp signal can therefore replace short duration pulses that require high peak power and need to have the same bandwidth. Pulse compression is used by radar systems where high range resolution [34] [41] and maximum SNR is needed [39] [38] [42]. Pulse compression has also been considered for use in medical imaging, specifically ultrasound systems [43]. In the ultrasound system, a high power probe signal can be used for high resolution images. However, for safety reasons, a high power probe signal cannot be transmitted. Instead, a linear chirp signal can replace the high power signal using pulse compression.

In other applications, linear chirp signals have been used in spread spectrum communication as a way of transmitting information utilizing a wide band of frequencies [44]. “Spread spectrum is a means of transmission in which the signal occupies a bandwidth in excess of the minimum necessary to send the information; the band spread is accomplished by means of a code which is independent of the data, and a synchronized reception with the code at the receiver is used for despreading and subsequent data recovery [45].” Applications of spread spectrum communication include transmission of ultra wideband (UWB) signals [46], wireless local area networks [34], personal area networks [47], and noise jamming [41]. One way to “spread” the spectrum of a signal is by using the linear chirp signal [44].

In [48], a linear chirp signal is designed to have spectral content within a specified frequency band of 3.1GHz - 10.6GHz. The need for such a system is due to
Federal Communication Commission (FCC) regulations for the transmission of UWB signals. The shape of the linear chirp signal is designed using an electromagnetic pulse shaper \cite{49}. The linear chirp signal was used to increase waveform energy and the RF power spectral density given peak voltage limitations in the electromagnetic pulse shaper. As mentioned above for pulse compression, the linear chirp signal is used since it can transmit a great deal of power by sending a low peak-power long-duration signal. The two waveforms studied in \cite{48} are a linear chirp signal using a rectangular window and a linear chirp signal using a Gaussian window. It was shown that the Gaussian window function had a wider main lobe and more rapidly decaying side lobes as compared with the rectangular window.

For the characterization of RF devices, the linear chirp probe signal is attractive because it allows for the use of a probe signal that has a large spectrum that can be transmitted with a reasonable amount of power. A large probe bandwidth can be used to analyze multiple device components that operate at different frequencies simultaneously. For example, consider an RF device that contains two filters $\beta_1$ and $\beta_2$. Let the passband of $\beta_1$ be 200MHz-250MHz and the passband of $\beta_2$ be 500MHz-550MHz. A linear chirp signal can be designed to probe frequencies between $f_0 = 200$MHz and $f_1 = 550$MHz, which would identify the passband of both filters.

1.5 Proposed Forensic Characterization System

Our proposed RF device characterization system is illustrated in Figure 1.7 \cite{50}, where $p(t)$ is the probe signal and $r(t)$ is the return signal. $p(t)$ is transmitted into the environment and received by the RF device. As described in Section 1.1, the probe signal is designed to generate a specific RF signature at an a priori frequency location. Once received by the antenna of the RF device, the probe signal will encounter a nonlinearity and be reflected and reradiated from the device. An example is shown in Figure 1.8 for when the probe is filtered before the nonlinearity.
Fig. 1.7. Proposed RF Device Characterization System.

Fig. 1.8. Example of Probe Signal Input Into RF Device.
Once reradiated from the RF device, the proposed RF device characterization system obtains the return signal \( r(t) \) and the power spectrum \( P_r(f) \) of the return signal is estimated. Once the power spectrum of the signal is determined, the amplitude of frequency locations are sampled. These amplitudes are known as the features, where the features constitute the device signature and will be used to characterize the device. Each feature is modeled as a random variable \( e_i \). The set of features are formed into a random vector, known as a feature vector, and is defined as

\[
\eta = [e_1 \ldots e_G]
\]

(1.18)

The feature vector is processed using statistical pattern recognition techniques [51]. Statistical pattern recognition is the process of mapping the feature vector into a category, or label, where the category corresponds to the RF device that generated the features. Six pattern recognition systems, or classifiers, are studied in this dissertation. These classifiers include a support vector machine (SVM), a binary tree (BT) classifier, two Bayesian classifiers, a Parzen window (PW) classifier, and a K-nearest neighbor (K-NN) classifier.

### 1.6 Other Applications of Forensic Fingerprinting

Forensic fingerprinting is the process of characterizing an object or electronic device based on unique characteristics. The fingerprint is the unique information used to characterize the object or device. This section provides an overview of forensic fingerprinting methods. Several fingerprinting applications are discussed which include fingerprinting humans, objects, multimedia content, printers, scanners, cameras, and RF devices.

In the most basic example, fingerprints are used in criminal investigations to identify people. A human fingerprint leaves a residue from the surface ridges of the finger [52]. The pattern of a fingerprint for any individual is unique, meaning that no two fingerprints are the same. The features of the human fingerprint are unique and
correspond to the spacial location of the surface ridges, which can be analyzed in the frequency domain. Human fingerprinting is an early form of forensic analysis and is the basis for many forensic applications.

Fingerprints are also used to characterize metallic objects buried in the ground. One application of this fingerprinting technique is landmine detection [53]. Similar to RF devices, the metallic object returns information (a signature) in response to a probe signal. The information is used to characterize the object. The metallic object has a distinct permeability, which can be identified using Electromagnetic Induction Spectroscopy (EMIS) [54]. The object is first probed by a time-varying low frequency electromagnetic field. Currents are induced in the object, which generates a magnetic field that can be used to characterize the object. This characterization approach is very similar to the proposed RF device characterization system.

In more recent applications, video forensic fingerprinting systems, which identify copyrighted video material, are being developed for internet websites. Video sharing websites, such as YouTube, allows users to upload videos and other content, which can then be downloaded by others. A growing problem with such websites is that copyrighted material, such as movies, are uploaded without the consent of the content owners. A particular video fingerprinting technique is content-based copy detection (CBCD) [55]. In CBCD, a fingerprint is extracted from the video and matched with known candidates in a database [56]. The fingerprint contains features that are inherent to the video. Typically, few features are needed for the fingerprint, which allows for a low computational complexity when matching the fingerprint with a candidate in the database [57].

In general, a device can be authenticated by developing a fingerprint based on information generated by the device [50]. For example, it may be of interest to identify a printer that was used in illegal activity or identify a source camera to authenticate digital photographs. Several approaches have been developed to characterize printers [58–61] [62–64] [65–67]. Two types of printers that can be characterized are electrophotographic (EP) and inkjet printers. The fingerprinting techniques are based
on the electromechanical properties of the printer. “In EP printing, artifacts are created in the printed output due to electromechanical imperfections in the printer such as fluctuations in the angular velocity of the OPC drum, gear eccentricity, gear backlash, and polygon mirror wobble [50].” The inkjet printer generates similar effects. These “imperfections” can be used to characterize the printer.

A digital camera can be authenticated by the noise characteristics in the digital image [68]. The two types of noise that can be used are fixed pattern noise and “photoresponse nonuniformity (PRNU) which is caused by variations in pixel responsivity [50] [69].” The fixed pattern noise is created by dark currents, which are defined as stray currents that leak into the sensor substrate [68]. The noise in image $X$ can be identified by subtracting it from a noiseless version, $Y$, of itself. $Y$ is estimated using a wavelet denoising filter, and the noise can be found $X - Y$. The method used to characterize a digital camera can be extended to scanners, where the scanned image contains noise unique to the sensors that generated the image [70] [71].

1.6.1 RF Circuit Forensics

There are currently two existing techniques that are used to detect RF devices in the environment. These methods require that the RF device comply to a standard, for example the IEEE 802.11 standard. The RF device is therefore designed to transmit information independently, or when prompted by a probe signal. The RF devices that comply to the IEEE 802.11 standard use methods, described by the standard, for processing incoming and outgoing signals for the RF device. The manufacture is allowed, under the guidelines of the standard, to specify certain settings of the device. Therefore certain bytes of information are set by the manufacturer that are used to develop a fingerprint.

One approach used for RF device characterization is known as passive fingerprinting. In the passive fingerprinting, the RF device is independently transmitting information into the environment. Mobile telephone companies used a technique similar
to passive fingerprinting, known as Radio Frequency Fingerprinting [72], to prevent mobile phone cloning. In more recent applications, passive fingerprinting techniques are used to detect RF devices that comply to the IEEE 802.11 standard. In addition to fingerprinting hardware, a fingerprint can also be developed for the device driver. The device driver specifies the amount of information that is transmitted into the environment. The information is transmitted in packets or frames. The number of transmitted frames is different for each device driver. A fingerprint for the device driver can be developed based on the rate at which these frames are transmitted [73].

Another approach used for RF device characterization is known as active fingerprinting. The fingerprinting system transmits and receives information to and from the RF device that complies to a transmission protocol. The probe signal must have the ability to initiate connections with the RF device [74], meaning that the RF device can recognize and process the probe signal. Once a connection is established, the RF device transmits and receives information to and from the fingerprinting system. A fingerprint is then developed based on this information [72].

It is important to note that the approach used for the passive and active fingerprinting techniques described in this Section is very different from the approach used by our proposed RF device characterization system outlined in Figure 1.7. The fingerprinting techniques described in this Section require that the RF device comply to some transmission standard (IEEE 802.11) and can therefore only be used to characterize a subset of RF devices. Our proposed RF device characterization system can theoretically be used to characterize any device and, in particular, FCC Part 15 devices. Part 15 is a FCC mandate used for unlicensed radiators [75]. The general conditions of operation for Part 15 devices state that “an intentional, unintentional, or incidental radiator is subject to the conditions that no harmful interference is caused and that interference must be accepted that may be caused by the operation of an authorized radio station, by another intentional or unintentional radiator, by industrial, scientific and medical (ISM) equipment, or by an incidental radiator [76].” The term harmful interference is any transmission that seriously degrades, obstructs,
or repeatedly interrupts a radio communications service [77]. Some examples of Part 15 devices include wireless local area networks, cordless phones, walkie-talkies, garage door openers, wireless fences, and surveillance systems.

1.7 Outline of Dissertation

One of the first steps for RF device characterization is probe signal design. Chapter 2 presents the design of three probe signals. Time and frequency domain analysis is presented for each design. These three probe signals are a two-tone probe signal, a windowed linear chirp probe signal, and a linear chirp probe signal. Chapter 3 describes three feature types, where each feature is designed with respect to one of the three probe signals. All features considered in this Chapter are amplitudes that are sampled at deterministic frequency locations of the power spectrum of the return signal. Chapter 4 presents a review of 6 classification systems. This review is used to provide an overview of each classifier. Three simulations are presented in Chapter 5. The first simulation is designed to classify FOD and IMD features using 6 circuit models in response to a two-tone signal. The second simulation is used to classify harmonic distortion features using 5 circuit models in response to a windowed linear chirp signal. The third simulation is used to classify features that are sampled from the passband of two filter models. In addition, an experiment is presented which classifies the output of two RF device filters. The experiments are designed to follow the procedure of the third simulation.

1.8 Contributions

The contributions of this dissertation are as follows:

- Mathematical analysis of the output of a nonlinear system when a windowed linear chirp signal is its input. The power spectral density is derived in closed form. This analysis is used in the simulations.
• Developed methods for forensically characterizing circuit models using intermodulation distortion generated by nonlinearities in response to a two-tone probe signal. Classification errors are examined when the return signal is close to the noise floor. These simulations are presented in Section 5.2 of Chapter 5.

• Developed methods for forensically characterizing circuit models using harmonic distortion generated by nonlinearities in response to a windowed linear chirp signal. Classification errors are examined when the return signal is close to the noise floor. These simulations are presented in Section 5.3 of Chapter 5.

• Developed methods for forensically characterizing circuit models using the filter response of two Chebyshev filters. The filter response is estimated by the linear chirp probe signal. Both experimental and simulation results are compared. These results indicate that the distinct oscillating patterns in the passband of the “real” RF circuits may be caused by other sources.

• Comparison of classification accuracy of the pattern recognition methods.
2. PROBE SIGNAL DESIGN

In this Chapter, three probe signals are considered. These are a two-tone probe signal, a windowed linear chirp probe signal, and a linear chirp probe signal. From these three probe signals we will study various methods to characterize the return signal after they are used as inputs to an RF circuit model. The two-tone probe signal is presented in Section 2.1. The two-tone signal is input into a third order nonlinearity modeled by a power series. The power spectrum is derived for the output of this nonlinearity and contains intermodulation distortion (IMD). The windowed linear chirp probe signal is presented in Section 2.2. This signal is also input into a third order nonlinearity and the power spectrum is derived for the output of the nonlinearity. It will be shown that this power spectrum contains harmonic distortions. The linear chirp probe is presented in Section 2.3 and is used to estimate the filter response of the RF circuit. In this Chapter, we will denote the probe signal as \( x(t) \) and the return signal as \( y(t) \) since we are not analyzing the entire RF system.

2.1 Two-Tone Probe Signals

The two-tone signal is

\[
x(t) = A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)
\]  
(2.1)

The parameters of the signal are defined in Table 2.1. An example of \( x(t) \) is shown in Figure 2.1(a), where \( f_1 = 40\text{MHz}, \ f_2 = 40.1\text{MHz}, \) and \( t \) ranges from 0 to \( 2 \times 10^{-6} \) seconds. The Fourier transform of \( x(t) \) is [9]
Table 2.1
Two-Tone Probe Signal Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A_1$</td>
<td>Amplitude of first sinusoid</td>
<td>$A_1 \in \mathbb{R}$</td>
</tr>
<tr>
<td>$A_2$</td>
<td>Amplitude of second sinusoid</td>
<td>$A_2 \in \mathbb{R}$</td>
</tr>
<tr>
<td>$t$</td>
<td>Time (seconds)</td>
<td>$t \in \mathbb{R}$</td>
</tr>
<tr>
<td>$f_1$</td>
<td>Probe frequency of first sinusoid</td>
<td>$f_1 \in \mathbb{R}$</td>
</tr>
<tr>
<td>$f_2$</td>
<td>Probe frequency of second sinusoid</td>
<td>$f_2 = f_1 + \Delta$</td>
</tr>
<tr>
<td>$\Delta$</td>
<td>Offset between probe frequencies</td>
<td>$\Delta \in \mathbb{R}$</td>
</tr>
</tbody>
</table>

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft} dt = \frac{A_1 \delta(f - f_1) + A_1 \delta(f + f_1)}{2} + \frac{A_2 \delta(f - f_2) + A_2 \delta(f + f_2)}{2}$$

(2.2)

This derivation is shown in Appendix B. The power spectrum of $x(t)$ is defined as $P_x(f) = X(f)X^*(f)$, where $X^*(f)$ is the complex conjugate of $X(f)$. The energy of a two-tone signal is concentrated at the locations of the discrete probe frequencies $f_1$ and $f_2$ as shown in Figure 2.1(b).

When input into a nonlinearity, the two-tone signal generates IMD. The nonlinearity is modeled as a power series of order $M$ as shown in [78] [79]

$$y(t) = \sum_{j=1}^{M} a_j [x(t)]^j$$

(2.3)

The IMD is located at discrete positions in the power spectrum of $y(t)$. The IMD locations are defined as

$$f_{imd}(n_1, n_2) = n_1 f_1 + n_2 f_2$$

(2.4)

where $n_1 \neq 0$ and $n_2 \neq 0$. The order of the IMD is defined by [3]

$$O = |n_1| + |n_2|$$

(2.5)
Fig. 2.1. Example of a Two-Tone Signal in the Time Domain and Frequency Domain. $f_1 = 40$MHz, $f_2 = 40.1$MHz, and $t$ ranges from 0 to $2 \times 10^{-6}$ seconds.
2.1.1 Third Order Analysis

In this Section, the two-tone signal defined in Equation 2.1 is input into a power series of order \( M = 3 \) \[78\] \[79\]

\[
y(t) = \sum_{j=1}^{3} a_j[A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)]^j
\]  (2.6)

Expanding Equation 2.6

\[
y(t) = b_0^1 + b_1^1 \cos(2\pi f_1 t) + b_2^1 \cos(2\pi f_2 t) + b_1^2 \cos(2\pi [2f_1] t) + b_2^2 \cos(2\pi [2f_2] t) + b_1^3 \cos(2\pi [3f_1] t) + b_2^3 \cos(2\pi [3f_2] t) + b_1^4 \cos(2\pi [f_1 + f_2] t) + b_2^4 \cos(2\pi [f_2 - f_1] t)
\]  (2.7)

The constant coefficients of Equation 2.7 are defined in Table 2.2. The complete derivation of \( y(t) \) is shown in Appendix A. An example of \( y(t) \) is shown in Figure 2.2.

The Fourier transform of Equation 2.7 is

\[
Y(f) = b_0^1 \delta(f) + \frac{b_1^1}{2} \delta(f - f_1) + \frac{b_1^1}{2} \delta(f + f_1) + \frac{b_2^2}{2} \delta(f - f_2) + \frac{b_2^2}{2} \delta(f + f_2) + \frac{b_1^3}{2} \delta(f - [2f_1]) + \frac{b_2^3}{2} \delta(f - [2f_2]) + \frac{b_1^3}{2} \delta(f + [2f_1]) + \frac{b_2^3}{2} \delta(f + [2f_2]) + \frac{b_1^4}{2} \delta(f - [3f_1]) + \frac{b_2^4}{2} \delta(f - [3f_2]) + \frac{b_1^4}{2} \delta(f + [3f_1]) + \frac{b_2^4}{2} \delta(f + [3f_2]) + \frac{b_1^4}{2} \delta(f - [f_1 + f_2]) + \frac{b_2^4}{2} \delta(f + [f_1 + f_2]) + \frac{b_2^4}{2} \delta(f - [f_2 - f_1]) + \frac{b_2^4}{2} \delta(f + [f_2 - f_1]) + \frac{b_2^4}{2} \delta(f - [2f_1 + f_2]) + \frac{b_2^4}{2} \delta(f + [2f_1 + f_2]) + \frac{b_2^4}{2} \delta(f - [f_2 + f_1]) + \frac{b_2^4}{2} \delta(f + [f_2 + f_1]) + \frac{b_2^4}{2} \delta(f - [2f_2 - f_1]) + \frac{b_2^4}{2} \delta(f + [2f_2 - f_1])
\]  (2.8)

where \( \delta(f) \) is a Dirac Delta Function. The derivation of this Equation is shown in Appendix B. The power spectrum of \( Y(f) \) is \( P_y(f) = Y(f)Y^*(f) \). The power spectrum \( P_y(f) \) of \( Y(f) \) is
Fig. 2.2. Example Output from a Nonlinearity of Order M=3 When the Input is a Two-Tone Signal. $a_1 = a_2 = a_3 = 1$, $f_1 = 40\text{MHz}$, $f_2 = 40.1\text{MHz}$, and $t$ ranges from 0 to $2\times10^{-6}$ seconds.
Table 2.2
Parameter Definitions for Constants in $y(t)$.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_0^1$</td>
<td>Constant</td>
<td>$\frac{a_2(A_1)^2}{2} + \frac{a_2(A_2)^2}{2}$</td>
</tr>
<tr>
<td>$b_1^1$</td>
<td>1st order harmonic</td>
<td>$a_1A_1 + \frac{a_3(A_1)^3}{2} + \frac{a_3(A_1)^3}{4} + \frac{3a_3A_1(A_2)^2}{2}$</td>
</tr>
<tr>
<td>$b_2^1$</td>
<td>1st order harmonic</td>
<td>$a_1A_2 + \frac{3a_3(A_1)^2A_2}{2} + \frac{a_3(A_2)^3}{2} + \frac{a_3(A_2)^3}{4}$</td>
</tr>
<tr>
<td>$b_1^2$</td>
<td>2nd order harmonic</td>
<td>$\frac{a_2(A_1)^2}{2}$</td>
</tr>
<tr>
<td>$b_2^2$</td>
<td>2nd order harmonic</td>
<td>$\frac{a_2(A_2)^2}{2}$</td>
</tr>
<tr>
<td>$b_3^1$</td>
<td>3rd order harmonic</td>
<td>$\frac{a_3(A_1)^3}{4}$</td>
</tr>
<tr>
<td>$b_3^2$</td>
<td>3rd order harmonic</td>
<td>$\frac{a_3(A_2)^3}{4}$</td>
</tr>
<tr>
<td>$b_4^1$</td>
<td>2nd order intermodulation</td>
<td>$a_2A_1A_2$</td>
</tr>
<tr>
<td>$b_4^2$</td>
<td>2nd order intermodulation</td>
<td>$a_2A_1A_2$</td>
</tr>
<tr>
<td>$b_5^1$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3(A_1)^2A_2}{4}$</td>
</tr>
<tr>
<td>$b_5^2$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3(A_1)^2A_2}{4}$</td>
</tr>
<tr>
<td>$b_5^3$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3A_1(A_2)^2}{4}$</td>
</tr>
<tr>
<td>$b_5^4$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3A_1(A_2)^2}{4}$</td>
</tr>
</tbody>
</table>

$$P_y(f) = [b_0^1]^2\delta(f)^2 + \left[\frac{b_1^1}{2}\right]^2\delta(f - f_1)^2$$

$$+ \left[\frac{b_2^1}{2}\right]^2\delta(f + f_1)^2 + \left[\frac{b_3^1}{2}\right]^2\delta(f - f_2)^2 + \left[\frac{b_4^1}{2}\right]^2\delta(f + f_2)^2$$

$$+ \left[\frac{b_5^1}{2}\right]^2\delta(f - [2f_1])^2 + \left[\frac{b_1^2}{2}\right]^2\delta(f + [2f_1])^2 + \left[\frac{b_2^2}{2}\right]^2\delta(f - [2f_2])^2 + \left[\frac{b_3^2}{2}\right]^2\delta(f + [2f_2])^2$$

$$+ \left[\frac{b_4^2}{2}\right]^2\delta(f - [3f_1])^2 + \left[\frac{b_1^3}{2}\right]^2\delta(f + [3f_1])^2 + \left[\frac{b_2^3}{2}\right]^2\delta(f - [3f_2])^2 + \left[\frac{b_3^3}{2}\right]^2\delta(f + [3f_2])^2$$

$$+ \left[\frac{b_4^3}{2}\right]^2\delta(f - [f_1 + f_2])^2 + \left[\frac{b_1^4}{2}\right]^2\delta(f + [f_1 + f_2])^2 + \left[\frac{b_2^4}{2}\right]^2\delta(f - [f_2 - f_1])^2$$

$$+ \left[\frac{b_3^4}{2}\right]^2\delta(f + [f_2 - f_1])^2 + \left[\frac{b_4^4}{2}\right]^2\delta(f - [2f_1 + f_2])^2 + \left[\frac{b_1^5}{2}\right]^2\delta(f + [2f_1 + f_2])^2$$

$$+ \left[\frac{b_2^5}{2}\right]^2\delta(f - [2f_1 - f_2])^2 + \left[\frac{b_3^5}{2}\right]^2\delta(f + [2f_1 - f_2])^2 + \left[\frac{b_4^5}{2}\right]^2\delta(f - [2f_2 + f_1])^2$$

$$+ \left[\frac{b_5^5}{2}\right]^2\delta(f + [2f_2 + f_1])^2 + \left[\frac{b_1^6}{2}\right]^2\delta(f - [2f_2 - f_1])^2 + \left[\frac{b_2^6}{2}\right]^2\delta(f + [2f_2 - f_1])^2$$

(2.9)
The derivation of this Equation is shown in Appendix B. Note that all the energy occurs at discrete frequency locations.

### 2.2 Windowed Linear Chirp Signal

We are interested in studying other types of probe signals to see whether they can perform better when used to forensically characterize a RF circuit. In this Section, we describe linear chirp signals. As discussed in Section 1.4 of Chapter 1, chirp signals are attractive because they allow the probe to have a large bandwidth and higher transmission power. The frequency content of the probe may then generate more harmonic components that can be used to characterize the circuit. Following the approach of the two-tone signal, a windowed linear chirp signal is input into a nonlinearity and the response is analyzed in the frequency domain. A linear chirp signal is defined as

\[ l(t) = C_1 e^{j\phi(t)} \]  
\[ \phi(t) = 2\pi f_0 t + \pi kt^2 \]  

Where \( \phi(t) \) is known as the instantaneous phase of the chirp signal. The parameters in Equations 2.10 and 2.11 are defined in Table 2.3. The frequency at any given time \( t \) is known as the instantaneous frequency and is defined as [80]

\[ f_i = \frac{\dot{\phi}(t)}{2\pi} = f_0 + kt \]  

A Gaussian window function is multiplied by the linear chirp signal in the time domain [81] [82]. As mentioned in Section 1.4 of Chapter 1, the Gaussian window function is used since it generates a wide main lobe and produces rapidly decaying side lobes in the frequency domain [48]. The Gaussian window function will be described in the form of a Gaussian envelop [37]
Table 2.3
Linear Chirp Probe Signal Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_1$</td>
<td>Amplitude of signal</td>
<td>$C_1 \in \mathbb{R}$</td>
</tr>
<tr>
<td>$i$</td>
<td>Imaginary number</td>
<td>$\sqrt{-1}$</td>
</tr>
<tr>
<td>$t$</td>
<td>Time</td>
<td>$t \in \mathbb{R}$</td>
</tr>
<tr>
<td>$t_1$</td>
<td>Time at frequency $f_1$</td>
<td>$t_1 \in \mathbb{R}, t_1 &gt; 0$</td>
</tr>
<tr>
<td>$f_0$</td>
<td>Frequency at $t = 0$</td>
<td>$f_0 \in \mathbb{R}$</td>
</tr>
<tr>
<td>$f_1$</td>
<td>Frequency at time $t_1$</td>
<td>$f_1 \in \mathbb{R}, f_1 &gt; f_0$</td>
</tr>
<tr>
<td>$k$</td>
<td>Chirp rate</td>
<td>$\frac{(f_1 - f_0)}{t_1}$</td>
</tr>
</tbody>
</table>

$$w(t) = e^{-\frac{(t-\mu)^2}{2\sigma^2}}$$ (2.13)

where

$$\mu = \int_{-\infty}^{\infty} tw(t)dt$$ (2.14)

and

$$\sigma^2 = \int_{-\infty}^{\infty} (t - \mu)^2 w(t)dt > 0$$ (2.15)

The windowed linear chirp signal, which is known as a Gaussian chirp signal, is then formed, $x(t) = w(t)l(t)$. $x(t)$ is

$$x(t) = C_1 e^{i\phi(t)} e^{-\frac{(t-\mu)^2}{2\sigma^2}}$$ (2.16)

An example of a Gaussian chirp signal is shown in Figure 2.3. The parameters of this signal are shown in Table 2.4. These parameters were chosen for illustration purposes.
Fig. 2.3. Example of a Gaussian Chirp Signal in the Time Domain, Where Parameters are Defined in Table 2.4.
Table 2.4  
Windowed Chirp Signal Parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_1$</td>
<td>Amplitude of signal</td>
<td>1</td>
</tr>
<tr>
<td>$i$</td>
<td>Imaginary number</td>
<td>$\sqrt{-1}$</td>
</tr>
<tr>
<td>$f_0$</td>
<td>Initial frequency at $t = 0$</td>
<td>400MHz</td>
</tr>
<tr>
<td>$f_1$</td>
<td>Final frequency at time $t_1$</td>
<td>450MHz</td>
</tr>
<tr>
<td>$t_1$</td>
<td>Final time</td>
<td>$0.1 \times 10^{-6}$</td>
</tr>
<tr>
<td>$k$</td>
<td>Chirp rate</td>
<td>$5 \times 10^{14}$</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Mean</td>
<td>$5 \times 10^{-8}$</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>Standard deviation</td>
<td>$1 \times 10^{-8}$</td>
</tr>
</tbody>
</table>

Similar to the approach used for the two-tone probe, $x(t)$ is input into a non-linearity modeled as the power series described in Equation 2.6. The output then becomes

$$y(t) = \sum_{j=1}^{M} a_j(C_1)^j e^{i j \phi(t)} e^{-j(t-\mu)^2}$$  \hspace{1cm} (2.17)

$y(t)$ is shown in Figure 2.4. The parameters of this Figure are defined in Table 2.4, and $a_1 = a_2 = a_3 = 1$.

The signal $y(t)$ contains distortion at specific frequencies. Substituting $y(t)$ into the Fourier transform produces

$$Y(f) = \int_{-\infty}^{\infty} \left[ \sum_{j=1}^{M} a_j(C_1)^j e^{i j \phi(t)} e^{-j(t-\mu)^2} \right] e^{-i2\pi ft} dt$$  \hspace{1cm} (2.18)

where the closed form solution is

$$Y(f) = \sum_{j=1}^{M} \left[ \sqrt{\pi} a_j(C_1)^j e^{-\frac{\mu^2}{2\sigma^2}} \right] e^{\frac{-(2\pi f - j2\pi f_0 + i(\mu)^2)}{\pi \sigma^2}} e^{\frac{-2j(\frac{f}{\sigma^2} - i2\pi k)}{2 \sqrt{\sigma^2 - i j \pi k}}}$$  \hspace{1cm} (2.19)
Fig. 2.4. Output from the Nonlinearity of Order $M = 3$, Where the Input is the Gaussian Chirp Signal.
The derivation of Equation 2.19 is shown in Appendix C. $Y(f)$ is formed into its magnitude and phase components as follows

$$Y(f) = C_2 \sum_{j=1}^{M} [C_3(j)e^{iC_4(j)}] \xi(j, f) e^{i\psi(j, f)}$$  (2.20)

where

$$\xi(j, f) = e^{-\frac{2\pi^2 \sigma^2}{\gamma^2} (f - j(f_0 + \mu k))^2}$$  (2.21)

$$\psi(j, f) = \frac{-4\pi^3 \sigma^4 k}{j \gamma^2} (f + \frac{j \mu}{4\pi^2 \sigma^4 k} - j f_0)^2$$  (2.22)

$$C_3(j) = \frac{a_j(C_1)^j}{\sqrt{j}} e^{-\frac{j \mu}{2\sigma^2} + \frac{j \mu^2}{2\sigma^4} - \frac{4\pi^2 \sigma^2 \mu \mu f_0 j}{\gamma^2} + \frac{2\pi^2 \sigma^2 \mu \mu f_0 + j \mu}{\gamma^2} - \frac{2\pi^2 \sigma^2 j f_0^2}{\gamma^2}}$$  (2.23)

$$C_4(j) = \frac{j k \pi \mu^2}{\gamma^2} - \frac{j k \pi^3 \sigma^4 f_0^2}{\gamma^2} + \frac{2\pi j f_0 \mu}{\gamma^2} + \frac{4\pi^3 \sigma^4 k}{j \gamma^2} \left( \frac{\mu j}{4\pi^2 \sigma^4 k} - j f_0 \right)^2$$  (2.24)

$$C_2 = \frac{\sqrt{2\pi \sigma^2} e^{-\arctan(2\pi \sigma^2 k)}}{\sqrt{\gamma}}$$  (2.25)

$$\gamma = \sqrt{1 + 4\pi^2 \sigma^4 k^2}$$  (2.26)

The derivation of Equation 2.20 is shown in Appendix C. Note that when $j=1$ in Equation 2.20

$$X(f) = C_2[C_3(1)e^{iC_4(1)}] \xi(1, f) e^{i\psi(1, f)}$$  (2.27)

which is the Fourier transform of $x(t)$. The power spectrum of $Y(f)$ is

$$P_y(f) = Y(f)Y(f)^* = C_2 C_2^* \left\{ \sum_{j=1}^{M} [C_3(j)\xi(j, f)]^2 + \sum_{j=1}^{M-1} \sum_{l=j+1}^{M} 2C_3(j)C_3(l)\xi(j, f)\xi(l, f) \cos(C_4(j) - C_4(l) + \psi(j, f) - \psi(l, f)) \right\}$$  (2.28)
The derivation of $P_y(f)$ is shown in Appendix D. Note that, when $j = 1$ and $l = 0$ in Equation 2.28

$$P_x(f) = [C_3(1)\xi(1, f)]^2 \quad (2.29)$$

which is the power spectrum of $x(t)$. A graph of $P_x(f)$ and $P_y(f)$ are illustrated in Figure 2.5, where the Gaussian chirp signal parameters are shown in Table 2.4, where $a_1 = a_2 = a_3 = 1$. As is shown, harmonic distortion is located at

$$f_{har}(n_1) = n_1[f_0 + \mu k] \quad (2.30)$$

where $n_1 \in \{1, \ldots, M\}$, $[f_0 + \mu k]$ is defined as the mean of $\xi(1, f)$, and, for these Figures, $[f_0 + \mu k] = 425 MHz$. The harmonic distortion defined by Equation 2.30 is known as Gaussian harmonic distortion (GHD). GHD is present in the power spectrum at each harmonic.

Equation 2.28 is a combination of square-terms and cross-terms. The square-terms are defined as

$$\text{square-terms} = \left\{ \sum_{j=1}^{M} [C_3(j)\xi(j, f)]^2 \right\} \quad (2.31)$$

The cross-terms are defined as

$$\text{cross-terms} = \left\{ \sum_{j=1}^{M} \sum_{l=j+1}^{M} 2C_3(j)C_3(l)\xi(j, f)\xi(l, f)\cos(C_4(j) - C_4(l) + \psi(j, f) - \psi(l, f)) \right\} \quad (2.32)$$

Closer examination of the square-terms illustrates that the square-terms are a sum of GHD. The mean of $\xi(j, f)^2$ is

$$\mu_\xi(j) = j[f_0 + \mu k] \quad (2.33)$$

and the variance of $\xi(j, f)^2$ is
Fig. 2.5. Comparison of the Power Spectrum of the Gaussian Chirp Probe Signal $P_x(t)$ with the Output of the Nonlinearity $P_y(f)$. Parameters are Defined in Table 2.4.
Fig. 2.6. Example Output of a Third Ordered Nonlinear System. Each GHD is Generated by a Square-Term.

\[ \sigma^2(j) = \frac{j\gamma^2}{8\pi^2\sigma^2} \]  

(2.34)

The peak of each GHD is located using Equation 2.33. An example response of a third order system using a Gaussian chirp signal is illustrated in Figure 2.6. Each GHD illustrated in this Figure is an example of a square-term. These square-terms will be used as features and will be discussed in Chapter 3.
The cross-terms \( \sum_{j=1}^{M-1} \sum_{l=j+1}^{M} 2C_3(j)C_3(l)\xi(j, f)\xi(l, f) \cos(C_4(j) - C_4(l) + \psi(j, f) - \psi(l, f)) \) are a multiple of two GHD’s of different order. The magnitude of the cross-terms is of the form \( \xi(j, f)\xi(l, f) = e^{-\frac{2\pi^2}{\sigma^2} (f - j)^2} e^{-\frac{2\pi^2}{\sigma^2} (f - l)^2} \). If minimal overlap exists between the GHD, then the product is approximately zero. For example, closer examination of Figure 2.6 illustrates that each GHD is well separated and has no overlap. A minimal overlap condition is defined as

\[
\xi(j, f)\xi(l, f) = \epsilon \tag{2.35}
\]

where \( \epsilon \simeq 0 \), and \( l > j \). Note that since \( \xi(j, f) \) and \( \xi(l, f) \) are both Gaussian functions, \( \epsilon \neq 0 \). An example of an overlap between two GHD’s is shown in Figure 2.7, where \( Y_1 = e^{-\frac{(f - j)^2}{2\sigma^2}} \) and \( Y_2 = e^{-\frac{(f - l)^2}{2\sigma^2}} \). As is shown, the amount of overlap is small relative to the amplitude of \( Y_1 \) and \( Y_2 \) and has a maximum at the intercept point \( f_{\text{int}} \). Equation 2.35 is considered at the intercept point \( f = f_{\text{int}} \). The intercept occurs when \( \xi(j, f_{\text{int}}) = \xi(l, f_{\text{int}}) \). Solving for \( f_{\text{int}} \) yields

\[
f_{\text{int}} = \frac{\mu_\xi(l)\sigma_\xi(j) + \mu_\xi(j)\sigma_\xi(l)}{\sigma_\xi(j) + \sigma_\xi(l)} \tag{2.36}
\]

where \( \mu_\xi() \) is defined by Equation 2.33 and \( \sigma_\xi() \) is defined by Equation 2.34. The derivation of \( f_{\text{int}} \) is shown in Appendix D. If \( f = f_{\text{int}} \), then Equation 2.35 reduces to

\[
\epsilon = \frac{(\sigma_\xi(j) + \sigma_\xi(l))^2}{(\mu_\xi(l) - \mu_\xi(j))^2} \tag{2.37}
\]

where \( \mu_\xi(l) > \mu_\xi(j), \sigma_\xi(j) > 0 \), and \( \sigma_\xi(l) > 0 \). The derivation of Equation 2.37 is shown in Appendix D. This condition states that as \( \epsilon \) decreases, \( (\mu_\xi(l) - \mu_\xi(j)) \) increases (i.e., the GHD’s become more separated causing less overlap). Therefore as \( \epsilon \) decreases, \( (\sigma_\xi(j) + \sigma_\xi(l)) \) decreases (i.e., the variance of the GHD’s decreases causing less overlap).
Fig. 2.7. Multiplication of Two Gaussian Functions. The Overlap is Dependent on the Intercept Point. As is Illustrated, \( Y_1 \times Y_2 \) is the Multiplication of \( Y_1 \) and \( Y_2 \).
2.2.1 Estimation of the Filter Response Using the Gaussian Chirp Probe Signal.

In addition to the GHD generated by the nonlinearity, the Gaussian chirp signal can also be used to estimate the filter response of a circuit. Estimation of the filter response is useful if the probe signal is filtered before input into a nonlinearity.

As discussed in Section 1.5 of Chapter 1, the filter is assumed to be known a priori. The Gaussian chirp signal can be designed to estimate the filter response. The design requires that the passband of the filter is within the frequency band (i.e., \( f_0 \) to \( f_1 \)) of the Gaussian chirp signal. For example, considered the Gaussian chirp signal used in the simulations of Chapter 5 and shown in Figure 2.8, where \( f_0 = 1 \text{Hz}, f_1 = 2 \text{GHz}, \mu = 1.5 \times 10^{-5}, \sigma = 3.5 \times 10^{-6} \), and \( t_1 = 50 \) microseconds. As demonstrated in the simulations, this probe signal can be used to estimate a filter with a passband of 400MHz - 450MHz. An example of an estimated filter response is shown in Figure 2.9.

2.3 Linear Chirp Probe Signal

As mentioned above in Section 2.2.1, the windowed linear chirp probe signal can be used to estimate the filter response. In the same way, a linear chirp probe signal (not windowed) can also be used to estimate the filter response. The linear chirp signal will be used in Section 5.4 of Chapter 5 to estimate the filter response of two Chebyshev filters. The nonlinearity of the RF device is not considered. The linear chirp signal is

\[
x(t) = A \cos(2\pi (f_0 + \frac{(f_1 - f_0)}{2t_1}) t)
\]

(2.38)

where \( A \) is the amplitude, \( f_0 \) is the initial frequency at time \( t = 0 \), \( f_1 \) is the final frequency at time \( t_1 \). The output signal \( y(t) \) is generated when \( x(t) \) is input into a filter as shown in Figure 2.10. An example filter response of a actual filter is shown in Figure 2.11. This filter is a K&L Microwave 5MC10-500/T25-O/O 5th-
Fig. 2.8. Example Fourier Transform of the Gaussian Chirp Probe Signal.
Fig. 2.9. Example Response of a Filter Using a Passband Between 400MHz and 450MHz. This Filter is Probed Using the Gaussian Chirp Probe Signal.
order Chebyshev filter with a center frequency of 500MHz, a passband of 25MHz, and a passband ripple of 0.05dB. The probe signal used to generate this filter response is a linear chirp probe signal with \( f_0 = 495\text{MHz} \), \( f_1 = 505\text{MHz} \), and \( t_1 = 1000 \) nanoseconds. As is illustrated, the passband ripple of this filter has a distinct pattern in the frequency range 495MHz - 505MHz. The passband presented in this Figure indicates that it may be possible to characterize the output signal of the filter based on these ripple patterns.

![Diagram of filter](image)

Fig. 2.10. Linear Chirp Signal is Input into a Filter. The Output Signal Contains an Estimate of the Filter Response.

![Graph of filter response](image)

Fig. 2.11. Example Power Spectrum of the Filter Response for a K&L Microwave Filter.
3. FEATURES FOR DEVICE CHARACTERIZATION

After the RF device is probed, a response is reradiated by the device back to a receiver. Information, or features, are extracted from the returned signal and used to determine which type of RF device has been probed. The features should be distinct information inherent to the device. These features will then be used by classification techniques developed in the areas of pattern recognition and machine learning to determine which RF circuit was probed from a class of potential devices. Classification systems as discussed in more detail in Chapter 4.

In general the features used will be of two types, features corresponding to power at discrete frequencies caused by distortion or features that are related in the filter response in the RF circuit. For each of the three probe signals introduced in Chapter 2, different features will be used. For example, the features for the two-tone probe signal are power/energy of the intermodulation distortion (IMD) at discrete frequencies in the power spectrum of the return signal. The Gaussian chirp signal is used to estimate the filter response of the RF device and is also used to generate harmonic distortion. The amplitudes in the passband of the filter response and the amplitudes of the harmonic distortion will be used as features in this case. Finally, the features for the linear chirp signal are also amplitudes of the passband of the filter response.

3.1 Features For The Two-Tone Probe Signal

It was shown in Section 2.1 that, when a two-tone probe signal is input into a nonlinearity, first order distortion (FOD) and IMD occurs in the return signal $r(t)$ and is located at discrete (and deterministic) frequencies. The power in the FOD and IMD are inherent to the nonlinearities and will be used as features to characterize the RF device. The power in the FOD and IMD are obtained by sampling the power
spectrum of \( r(t) \), \( P_r(f) \), at the FOD and IMD frequency locations and using this power as features. For a given \( P_r(f) \), 2 FOD’s and 2 third order IMD’s are considered. The discrete frequency locations are \( \rho_1 = f_1 \), \( \rho_2 = f_2 \), \( \rho_3 = (2f_1 - f_2) \), and \( \rho_4 = (2f_2 - f_1) \). Where \( f_1 \) and \( f_2 \) are the two probe frequencies as described in Chapter 2. Hence the 4 features are

\[
e_g = P_r(\rho_g)
\]  

(3.1)

where \( 1 \leq g \leq 4 \). Each feature of Equation 3.1 is modeled as a random variable, and will be denoted as \( e_g \) where \( e_g \in \mathbb{R} \). Feature details are shown in Table 3.1. These features are used to form the random feature vector

\[
\eta = [e_1 \ldots e_4]
\]  

(3.2)

3.2 Features for the Windowed Linear Chirp Probe Signal

Similar to the two-tone probe discussed above, the harmonic distortion generated by the windowed linear chirp probe signal, or the Gaussian chirp signal, is also located at deterministic frequencies. The distortion is dependent on the parameters of the window function. If an input filter is present in the RF circuit before the nonlinearity,

<table>
<thead>
<tr>
<th>Feature</th>
<th>Feature Location</th>
<th>Feature Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( e_1 )</td>
<td>( \rho_1 = f_1 )</td>
<td>( e_1 = P_r(\rho_1) )</td>
</tr>
<tr>
<td>( e_2 )</td>
<td>( \rho_2 = f_2 )</td>
<td>( e_2 = P_r(\rho_2) )</td>
</tr>
<tr>
<td>( e_3 )</td>
<td>( \rho_3 = 2f_1 - f_2 )</td>
<td>( e_3 = P_r(\rho_3) )</td>
</tr>
<tr>
<td>( e_4 )</td>
<td>( \rho_4 = 2f_2 - f_1 )</td>
<td>( e_4 = P_r(\rho_4) )</td>
</tr>
</tbody>
</table>
the shape of the filter response can also be used for features (assuming the filter response 
is known a priori).

Two techniques are considered for sampling the power spectrum of the return 
signal to create features. Both techniques are used in the simulations described in 
Section 5.3. In the first technique, the power spectrum is sampled using a fixed 
pattern in the frequency domain. An example is shown in Figure 3.1 for the case 
when no filters are present in the circuit. If the filters are present, the same approach 
is used. An example is shown in Figure 3.2. In the second technique, the power 
spectrum is sampled at precise fixed locations. An example is shown in Figure 3.3 for 
the case when no filters are present in the circuit. The peak location of the distortion 
is sampled and used as features. The peak locations of the distortion are

\[ \mu_{\xi}(j) = j[f_0 + \mu k] \]  

\[ \sigma^2_{\xi}(j) = \frac{j\gamma^2}{8\pi^2\sigma^2} \]  

These are known a priori using the analysis of the return signal described in 
Section 2.2.1 of Chapter 2. If the filters are present, the same sampling technique 
is used. An example is shown in Figure 3.4. The filter response can be sampled at 
precise locations since the filter response is assumed to be known a priori.

Based on the above, consider a total of \( G \) features, sampled from the power 
spectrum \( P_r(f) \) of the return signal. Define \( \rho \) to be the set of \( G \) frequencies \( \rho = \{\rho_1, \ldots, \rho_G\} \) 
corresponding to the frequency location of each feature. If the samples are determined 
at regular intervals, then \( \rho_g = \rho_{g-1} + \Delta \) for \( 2 \leq g \leq G \), where \( \Delta \) is a constant offset 
value. If samples are at precise locations, the set \( \rho \) must be determined based on 
Equations 3.3 and 3.4. Once \( \rho \) is determined, samples from the power spectrum (and 
the features) are

\[ e_g = P_r(\rho_g) \]
Fig. 3.1. Sampling Locations of the Power Spectrum at Regular Intervals (No Filter).
Fig. 3.2. Sampling Locations of the Power Spectrum at Regular Intervals (Filter Present).
Fig. 3.3. Sampling Locations of the Power Spectrum at Precise Locations (No Filter).
Fig. 3.4. Sampling Locations of the Power Spectrum at Precise Locations (Filter Present).
where $1 \leq g \leq G$. As mentioned in the above Section, the features are modeled as random variables that form the set $\{e_1,...,e_G\}$, where $e_g \in \mathbb{R}$. These features are used to form the random feature vector

$$\eta = [e_1 \ldots e_G]$$

(3.6)

### 3.3 Features for the Linear Chirp Probe Signal

It was demonstrated in Section 2.3 how the linear chirp probe signal can be used to estimate the filter response of the circuit. The features introduced in this Section are obtained by sampling the filter response and scaling the samples at regular intervals as shown in Figure 3.5. Consider a total of $G$ features, sampled from the power spectrum $P_r(f)$ of the return signal. Define the set of $G$ frequency locations, $\rho = \{\rho_1,...,\rho_G\}$, where $\rho_g = \rho_{g-1} + \Delta$ for $2 \leq g \leq G$. $\Delta$ is a constant offset value. The features are

$$e_g = P_r(\rho_g)$$

(3.7)

where $1 \leq g \leq G$. The samples form the set $\varrho = \{\varrho_1,...,\varrho_G\}$. The samples are then normalized from 0 to 1. Scaling the features maintains the “pattern” of passband ripple in the filter response. Each feature $e_g$ is scaled as follows

$$e_g = \frac{e_g - \min(\varrho)}{\max(\varrho) - \min(\varrho)}$$

(3.8)

where the function $\min(\varrho)$ is the minimum value of $\varrho$ and the function $\max(\varrho)$ is the maximum value of $\varrho$. As mentioned above, the features are modeled as random variables that form the set $\{e_1,...,e_G\}$, where $e_g \in \mathbb{R}$. These features are used to form the following random feature vector

$$\eta = [e_1 \ldots e_G]$$

(3.9)
Fig. 3.5. Filter Response Sampled at Regular Intervals.
4. REVIEW OF CLASSIFIERS

This Chapter provides a review of classifiers that are used in a variety of pattern recognition and machine learning applications ranging from automatic speech recognition to the characterization of printers. This chapter provides a theoretical overview and an operational description of each classifier. Six classifiers are considered and are used in the simulations described in Chapter 5. The features discussed in Chapter 3 represent information that can be used to identify the RF device that was probed. The features presented in Chapter 3 were of two types, features corresponding to power at discrete frequencies in the return signal caused by distortion or features that are related in the filter response in the RF circuit. These features are considered as continuous random variables that form the set

\[ e = \{e_1, \ldots, e_G\} \tag{4.1} \]

where \( G \) is the total number of features, \( e_g \in \mathbb{R} \), where \( g \in \{1, \ldots, G\} \). The bold notation is used in this dissertation to denote random variables and random vectors. The features form the feature vector

\[ \eta = [e_1 \ldots e_G] \tag{4.2} \]

The features are used by classifier, which maps observed features into a label or category, representing an RF device from a set of \( D \) potential devices. For example \( D = 2 \) would correspond to a situation where one was interested in identifying an RF circuit from one of two potential circuits. The set of RF devices are defined as

\[ \Upsilon = \{\Upsilon_1, \ldots, \Upsilon_D\} \tag{4.3} \]
where $\mathbf{Y}_d$ would correspond to a particular RF device or circuit, and $d \in \{1, \ldots, D\}$. We assume the devices have equal probability of occurring, $P(\mathbf{Y}_d) = \frac{1}{D}$.

All the classifiers considered in this dissertation must first be trained. By training, the classifier “learns” how to map the features into the category or label. The training is not perfect and the classifier will make mistakes in actual operation by assigning the wrong label to an observed feature vector. Training the classifier has two goals. One goal is to determine how it will assign labels to observed feature vectors and the other is to estimate its error performance or its classification accuracy. This is accomplished by using a set of feature vectors where the label or category is known a priori. This is sometimes referred to as “ground truth” information. This ground truth information is divided into two sets, a training set and a testing set. The training set or data is used to train the classifier and the testing set or data is used to test the error performance of the classifier.

For training, a set of $J$ observations is considered. Each observation consists of a feature vector, known as the training vector, and a label or category from the ground truth data. The training vector is defined as

$$
\mathbf{\theta}_j = [\mathbf{e}_{(\text{tr},j,1)} \ldots \mathbf{e}_{(\text{tr},j,G)}]
$$

(4.4)

where $\mathbf{\theta}_j \in \mathbb{R}^G$. The feature $\mathbf{e}_{(\text{tr},j,g)}$ in Equation 4.4 uses the notation from Equation 4.2 to denote the $j^{th}$ vector in the set of training vectors. The label or category, $\tau_j \in \{1, \ldots, D\}$ (ground truth), of the training vector, $\mathbf{\theta}_j$, denotes a particular RF device defined in Equation 4.3, where $\tau_j = d$ is the label for the RF device $\mathbf{Y}_d$. The training set is defined as

$$
\Theta = \{[\mathbf{\theta}_1, \tau_1], \ldots, [\mathbf{\theta}_J, \tau_J]\}
$$

(4.5)

where $[\mathbf{\theta}_j, \tau_j]$ is the $j^{th}$ observation in the training set. When dividing the ground truth information into the training set, the number of vectors per category will vary. Define the set $Q = \{Q_1, \ldots, Q_D\}$ to be the number of vectors per category in the training set. Define $Q_d \in Q$ as the number of vectors with labels equal to $\mathbf{Y}_d$. The
classifier estimates the mapping of the features vectors to the labels, \( \varphi : \theta_j \Rightarrow \tau_j \) for each observation [83].

To test the accuracy of the classifier, the test data is used from the ground truth data. A test vector is defined as

\[
\omega_i = [e_{(te,i,1)} \ldots e_{(te,i,G)}]
\]  \hspace{1cm} (4.6)

The feature \( e_{(te,i,g)} \) in Equation 4.6 uses the notation from Equation 4.2. Each test vector has a ground truth label \( \lambda_i \), where \( \lambda_i \in \{1,...D\} \) and \( \lambda_i = d \) denotes the RF device \( \Upsilon_d \). We will consider a test set of \( I \) test vectors

\[
\Omega = \{[\omega_1, \lambda_1], \ldots [\omega_I, \lambda_I]\}
\]  \hspace{1cm} (4.7)

where \([\omega_i, \lambda_i]\) is the \( i^{th} \) observation in the test set. Similar to the training set, the number of vectors per category varies in the test set. Define the set \( R = \{R_1, \ldots R_D\} \) to be the number of vectors per category in the test set. Define \( R_d \in R \) as the number of vectors with labels equal to \( \Upsilon_d \).

The trained classifier is then used to map the test vector into a label or category \( \varphi : \omega_i \Rightarrow \chi_i \), where \( \chi_i \in \{1, \ldots D\} \) and \( \chi_i = d \) denotes the RF device \( \Upsilon_d \). The mapping function \( \varphi \) was determined using the training set. This label may be incorrect. Let \( \chi_i \) be known as the predicted label for \( \omega_i \) determined by \( \varphi \).

Once a classification decision is made, the classification accuracy of the classifier can be measured for performance evaluation. The performance evaluation process is illustrated in Figure 4.1. To measure the accuracy of the classifier, \( \chi_i \) is compared with the ground truth label \( \lambda_i \) using

\[
I_f(i) = \begin{cases} 
1, & \lambda_i = \chi_i \\
0, & \text{else}
\end{cases}
\]  \hspace{1cm} (4.8)

Hence the test set is used to measure classification accuracy. Accuracy is estimated for each RF device \( \Upsilon_d \). The accuracy of \( \Upsilon_d \) is measured by using observations from
Fig. 4.1. Block Diagram of Performance Evaluation.
the test set that have the label $\lambda_i = d$. The average classification accuracy for $\mathbf{Y}_d$ is defined as

$$
P_d = \frac{\sum_{i : \lambda_i = d} I_f(i)}{R_d}
$$

(4.9)

$P_d$ is used in Chapter 5 to evaluate the performance of the classifier.

### 4.1 Bayesian Classifiers

A Bayesian classifier is based on Bayes decision theory [51] where a posterior distribution $P(\mathbf{Y}_d | \omega_i)$ is estimated. The Bayesian classifier determines the most probable category by comparing the posterior distribution of each category in $\mathbf{Y}$. For example, consider two RF devices, a garage door opener, $\mathbf{Y}_1$, and a walkie-talkie, $\mathbf{Y}_2$. For a given test vector $\omega_i$, the Bayesian classifier would decide $\mathbf{Y}_1$ (garage door opener) if $P(\mathbf{Y}_1 | \omega_i) > P(\mathbf{Y}_2 | \omega_i)$. Otherwise, it would decide $\mathbf{Y}_2$.

In this Section, two Bayesian classifiers are introduced, which are known as the Bayesian$_\mu$ and Bayesian$_\Sigma$ classifiers. Both classifiers use the same decision procedure as described above. Bayes decision theory uses Bayes rule [84]

$$
P(\mathbf{Y}_d | \omega_i) = \frac{p(\omega_i | \mathbf{Y}_d)P(\mathbf{Y}_d)}{p(\omega_i)}
$$

(4.10)

where

$$
P(\mathbf{Y}_d) = \frac{1}{D}
$$

(4.11)

has an equal probability distribution for each label, and

$$
p(\omega_i) = \sum_{d=1}^{D} p(\omega_i | \mathbf{Y}_d)P(\mathbf{Y}_d)
$$

(4.12)

is the probability of $\omega_i$, and
\[ p(\omega_i | \mathbf{Y}_d) = \frac{e^{-\frac{(\omega_i - \hat{\mu}_d)^T (\hat{\Sigma}_d)^{-1} (\omega_i - \hat{\mu}_d)}{2}}}{(2\pi)^{G/2} |\hat{\Sigma}_d|^{1/2}} \]  

(4.13)

is the conditional density function modeled as a the Gaussian distribution, where

\[
\hat{\mu}_d = \frac{\sum_{j=1}^{J} \theta_j}{Q_d} \quad \text{(4.14)}
\]

and

\[
\hat{\Sigma}_d = \frac{\sum_{j=1}^{J} (\theta_j - \hat{\mu}_d)(\theta_j - \hat{\mu}_d)^T}{Q_d - 1} \quad \text{(4.15)}
\]

are parameters defined using the maximum likelihood (ML) estimate [85]. The parameter \( \hat{\mu}_d \) is the mean vector of the Gaussian density function of size \( G \). The parameter \( \hat{\Sigma}_d \) is the covariance matrix of the Gaussian density function of size \( G \times G \). Note the conditions \( \tau_j = d \) in Equations 4.14 and 4.15, which requires that only the training vectors from category \( \mathbf{Y}_d \) are used to estimate \( \hat{\mu}_d \) and \( \hat{\Sigma}_d \).

### 4.1.1 Discriminate Functions

The decision rule used to classify the test vector \( \omega_i \) maximizes the posterior probability \( P(\mathbf{Y}_d | \omega_i) \). If two RF devices are considered, \( D = 2 \), then the decision rule uses a binary hypothesis test [51]. For the multi-category case, the decision rule uses a discriminate function [86]. The discriminate function is [84]

\[
g_d(\omega_i) = ln(p(\omega_i | \mathbf{Y}_d)) + ln(P(\mathbf{Y}_d)) \quad \text{(4.16)}
\]

where \( g_d(\omega_i) \) is a monotonically increasing function. \( \omega_i \) is then determined as

\[
\chi_i = \arg \max_d g_d(\omega_i) \quad \text{(4.17)}
\]
Fig. 4.2. Block Diagram of the Bayesian_μ Classifier and the Performance Evaluation Process.

4.1.2 Bayesian_μ Classifier

The first Bayesian classifier presented will be referred to as the Bayesian_μ classifier and is sometimes known as a distance classifier [51]. Bayesian_μ estimates the parameter \( \tilde{\mu}_d \) and defines \( \tilde{\Sigma}_d = \Sigma_I \) in Equation 4.13, where \( \Sigma_I \) is the identity matrix. The discriminate function of the Bayesian_μ classifier for a given test vector \( \omega_i \) is [84]

\[
g_d(\omega_i) = \frac{-(\omega_i - \tilde{\mu}_d)^T(\omega_i - \tilde{\mu}_d)}{2} - \frac{Gln(2\pi)}{2} + ln\left(\frac{1}{D}\right) \quad (4.18)
\]

A block diagram of the Bayesian_μ classifier is shown in Figure 4.2. This block diagram also shows how the average classification accuracy \( P_d \) is measured. The parameter \( \tilde{\mu}_d \) is estimated using Equation 4.14. A test vector is input into the classifier and the discriminate function \( g_d(\omega_i) \) is determined using Equation 4.18. The predicted label \( \chi_i \) is then determined using Equation 4.17. To determine the average classification accuracy measurement, the indicator function of Equation 4.8 compares \( \chi_i \) to \( \lambda_i \), and the average probability \( P_d \) of Equation 4.9 is measured for \( \Upsilon_d \).

4.1.3 Bayesian_Σ Classifier

The second Bayesian classifier is known to as the Bayesian_Σ classifier. The Bayesian_Σ classifier estimates the parameters \( \tilde{\mu}_d \) and \( \tilde{\Sigma}_d \). The discriminate function of the Bayesian_Σ classifier for a given test vector \( \omega_i \) is [84]
The Parzen window (PW) classifier [51] estimates a density function in a local region around the test vector $\omega_i$. The PW classifier discussed in this dissertation estimates a density function, $p_d(\omega_i|Y_d)$, for each category $Y_d$ [87]. The subscript $d$ denotes $Q_d$, which is the number of training vectors that have the label $\tau_i = d$ [88]. The volume, $V_d$, of the local region is determined by a window function, $\varphi()$, with a window width, $h_d$. $p_d(\omega_i|Y_d)$ is estimated using the training vectors with a label $\tau_j = d$ that are within the window defined by the window function. For example,
consider the case when two training vectors, $\bm{\theta}_1$ and $\bm{\theta}_2$, fall within the window. Let $\bm{\theta}_1$ have the ground truth label $\tau_1 = 1$ and let $\bm{\theta}_2$ have the ground truth label $\tau_2 = 2$. For a given test vector $\omega_i$, a density estimate $p_d(\omega_i|Y_1)$ is found using $\bm{\theta}_1$ and a density estimate $p_d(\omega_i|Y_2)$ is found using $\bm{\theta}_2$. The PW classifier decides $Y_1$ if $p_d(\omega_i|Y_1) > p_d(\omega_i|Y_2)$. Otherwise, $Y_2$ is chosen. The PW density estimate of $Y_d$ is defined as [88] [84]

$$p_d(\omega_i|Y_d) = \sum_{j=1}^{J} \frac{1}{Q_d V_d} \varphi\left(\frac{\omega_i - \theta_j}{h_d}\right)$$

(4.20)

The density estimate $p_d(\omega_i|Y_d)$ converges to $p(\omega_i|Y_d)$ in quadratic mean if the window function $\varphi(U)$ and $h_d$ are chosen under certain conditions [88] [89], where $U$ is defined as $U = [u_1\ldots u_G]$ and $U \in \mathbb{R}^G$. For convergence, the window function must be nonnegative

$$\varphi(U) \geq 0$$

(4.21)

and [90]

$$\int_{-\infty}^{\infty} \varphi(U)dU = 1$$

(4.22)

Other convergence conditions are [91] [86]

$$\sup_{U} \varphi(U) < \infty$$

(4.23)

$$\lim_{||U||\to\infty} \varphi(U) \prod_{i=1}^{G} u_i = 0$$

(4.24)

$$\lim_{Q_d \to \infty} V_d = 0$$

(4.25)

$$\lim_{Q_d \to \infty} Q_d V_d = \infty$$

(4.26)
We will use a zero-mean unit-variance Gaussian window function, which satisfies the convergence conditions [84] [88]. The PW density function using a Gaussian window is [87]

$$p_d(\omega_i|\mathbf{Y}_d) = \sum_{j=1}^{T} e^{-\frac{(\omega_i-\theta_j)^T(\omega_i-\theta_j)}{2(h)^2}}$$

where $h$ is a constant window width used for all categories. The predicted label $\chi_i$ is determined by

$$\chi_i = \arg \max_{d} p_d(\omega_i|\mathbf{Y}_d)$$

A block diagram of the PW classifier is shown in Figure 4.4. This block diagram also shows how the classification accuracy $P_d$ is measured. The window size $h$ must be determined a priori and input into this algorithm. For the simulations in Chapter 5, several window sizes will be tested. $p_d(\omega_i|\mathbf{Y}_d)$ is estimated by Equation 4.27. The predicted label $\chi_i$ is then determined using Equation 4.28. To determine the average classification accuracy, the indicator function of Equation 4.8 compares $\chi_i$ to $\lambda_i$, and the average probability $P_d$ of Equation 4.9 is measured for $\mathbf{Y}_d$.
4.3  K Nearest Neighbor Classifier

Similar to the PW classifier, the K nearest neighbor (K-NN) classifier also estimates a density function in a local region around the test vector \( \mathbf{w}_i \). However, instead of examining the number of training vectors that fall within a fixed window, the K-NN classifier examines the number of training vectors “nearest\(^1\)” to the test vector. For example, consider classifying using two nearest neighbors. Define \( \theta_1 \) and \( \theta_2 \) as two training vectors nearest to the test vector \( \mathbf{w}_i \). The first training vector has the ground truth label \( \tau_1 = 1 \) denoting \( \Upsilon_1 \) and the other has ground truth label \( \tau_2 = 2 \) denoting \( \Upsilon_2 \). A distance is found between each training vector and the test vector. Let the distance between \( \theta_1 \) and \( \mathbf{w}_i \) to be \( d_1 \). Let the distance between \( \theta_2 \) and \( \mathbf{w}_i \) to be \( d_2 \). The K-NN classifier decides \( \Upsilon_1 \) if \( d_1 < d_2 \). Otherwise, \( \Upsilon_2 \) is chosen. The density function estimate for the K-NN classifier is [92]

\[
p_J(\mathbf{w}_i) = \frac{K_J}{V_J} \tag{4.29}
\]

where \( V_J \) is the volume of the local region, \( K_J \) is the number of nearest training vectors in the volume \( V_J \), and \( J \) is the total number of training vectors. The subscript \( J \) is used, instead of \( d \) from the PW classifier, to indicate that all training vectors are used to estimate a posterior distribution [84]. Define \( K_d \) as the number of nearest training vectors with a label \( \tau_i = d \). A posterior distribution is estimated as follows. First define the joint probability of \( \Upsilon_d \) and \( \mathbf{w}_i \) as [93]

\[
p_J(\Upsilon_d, \mathbf{w}_i) = \frac{K_d}{J V_J} \tag{4.30}
\]

Using Bayes rule [94]

\[
p_J(\Upsilon_d | \mathbf{w}_i) = \frac{p_J(\Upsilon_d, \mathbf{w}_i)}{p_J(\mathbf{w}_i)} \tag{4.31}
\]

The posterior distribution becomes [84]

\(^1\)The nearest training vectors are determined using a distance function.
\[ p_J(\mathbf{Y}_d | \omega_i) = \frac{K_d}{K_J} \] (4.32)

and is determined by substituting Equations 4.29 and 4.30 into Equation 4.31. The \( K_J \) training vectors nearest to the test vector are determined using the distance function: [95]

\[ d(j; i) = q(e(tr;j;1) - e(te;i;1))^2 + \ldots + (e(tr;j;G) - e(te;i;G))^2 \] (4.33)

The training vectors that belong to the first \( K_J \) smallest distances are defined as the nearest neighbors.

A block diagram of the K-NN classifier is shown in Figure 4.5. This block diagram also shows how the classification accuracy \( P_d \) is measured. The number of nearest neighbors \( K_J \) must be known a priori and input into the algorithm. For the simulations in Chapter 5, several values of \( K_J \) are examined. A distance \( d(\theta_j, \omega_i) \) is estimated by Equation 4.33 between all training vectors and the test vector. The \( K_J \) training vectors with the smallest distances are used as the nearest neighbors. The posterior density \( p_J(\mathbf{Y}_d | \omega_i) \) is estimated using Equation 4.32. The predicted label \( \chi_i \) is estimated using

\[ \chi_i = \text{arg max}_d p_J(\mathbf{Y}_d | \omega_i) \] (4.34)

To determine the average classification accuracy, the indicator function of Equation 4.8 compares \( \chi_i \) to \( \lambda_i \), and the average probability \( P_d \) of Equation 4.9 is measured for \( \mathbf{Y}_d \).

### 4.4 Binary Tree Classifier

The binary tree (BT) classifier splits a single classification decision into a set of simpler binary decisions. Let the binary tree be \( T \), where \( T \) is made up of several elements known as nodes, where a node is denoted as \( t \). The starting node in the tree is the root node. The ending nodes in the tree are the terminal nodes. A classification
Fig. 4.5. Block Diagram of the K Nearest Neighbor Classifier and the Performance Evaluation Process.
A category is chosen at the terminal node. CART outlines a technique to grow a tree from the training data, and the threshold values \( T_1, T_2, \) and \( T_3 \) are estimated during training. As shown, the test vector \( \omega_i \) is input into the root node. If the feature \( e_{(te,i,1)} > T_1 \), then \( \omega_i \) is sent to the left child, otherwise it is sent to the right child. This process continues until a terminal node is reached, which is represented by the square block in the Figure. A category is chosen at the terminal node.

A common tree growing method is known as the classification and regression tree (CART) procedure [96]. CART outlines a technique to grow a tree from the
training data. The CART framework consists of splitting nodes, labelling nodes, and determining terminal nodes. A binary decision rule is used to split a node. The decision rule is determined based on an impurity measure \[96\]. Determining the terminal nodes is a critical procedure, which greatly affects the classification accuracy. Typically, instead of using a stopping criteria, a tree is grown to a very large size and then pruned \[96\]. The terminal nodes are labelled based on a majority vote.

### 4.4.1 Splitting Rule and Terminal Nodes

A binary decision is made at each node based on an impurity measure. One impurity measure is known as the Gini impurity measure \[97\]. The Gini impurity measure is defined as \[84\]

\[
i(t) = \sum_{d \neq f} P(\mathbf{y}_d|t)P(\mathbf{y}_f|t)
\]

where

\[
P(\mathbf{y}_d|t) = \frac{N_{(t,d)}}{N_t}
\]

and \(d \in \{1, \ldots, D\}\), \(f \in \{1, \ldots, D\}\). \(N_t\) is the total number of training vectors in node \(t\) and \(N_{(t,d)}\) is the total number of training vectors in node \(t\) that have the label \(\tau_j = d\). \(i(t)\) is the impurity at node \(t\). The tree is grown so that the children are purer then the parent node. A splitting rule is defined as in \[84\] \[98\]

\[
\Delta i(s_t, t) = i(t) - P_L i(t_L) - P_R i(t_R)
\]

where \(s_t\) is defined as a split at node \(t\), \(t_L\) and \(t_R\) are the left and right descendent nodes, \(i(t_L)\) is the impurity measure of \(t_L\), \(i(t_R)\) is the impurity measure of \(t_R\), \(P_L\) is the probability that the training vectors in node \(t\) are sent to \(t_L\), and \(P_R\) is the probability that the training vectors in \(t\) are sent to \(t_R\). The splitting rule requires that the impurity of a node is as large as possible. Define \(s_t \in S\), where \(S\) is the set
of all possible splits. Then, define \( s_{max} = \arg \max_{s_t} \Delta i(s_t, t) \) \cite{98}. \( s_{max} \) is the split that generates the largest \( \Delta i(s_t, t) \) and is used as the splitting rule for node \( t \).

### 4.4.2 Pruning

The binary tree described in Section 4.4.1 is grown to full size, meaning a zero impurity measure is reached at all terminal nodes. The full size binary tree may be larger than necessary. Pruning \cite{98} \cite{96} is a technique used to reduce, or “trim,” the binary tree. Pruning is started at the terminal nodes. The impurity is measured at the parent of a given terminal node. The impurity is again measured after removing the terminal node from the parent. If a small increase in impurity exists (close to 0 increase in impurity), then the terminal node is removed from the parent and the parent node becomes a terminal node \cite{84}. This process is continued until all terminal nodes, including all newly transformed parent nodes, are examined.

### 4.4.3 Binary Tree Classifier

A block diagram of the BT classifier is shown in Figure 4.7. This block diagram also shows how the classification accuracy \( P_d \) is measured. The BT classifier generates the tree \( T \) using the algorithm described in \cite{98–100}. “This algorithm randomly divides the training data into two sets, one to build the pure” terminal node tree “and the other to prune; then the roles of two sets are changed and the procedure iterates until converging to a smallest pruned subtree \cite{99}.” Once the tree is built using this procedure, a test vector \( \omega_i \) is input into the tree and the predicted label \( \chi_i \) is determined. To determine the average classification accuracy, the indicator function of Equation 4.8 compares \( \chi_i \) to \( \lambda_i \), and the average probability \( P_d \) of Equation 4.9 is measured for \( \Upsilon_d \).
Fig. 4.7. Block Diagram of the Binary Tree Classifier and the Performance Evaluation Process.
A support vector machine (SVM) generates a hyperplane or separation boundary that separates the training data into two regions in the feature space $\mathbb{R}^D$. For example, consider the two category case using training vectors of size $G = 2$ that are positioned in a feature space of size $\mathbb{R}^2$ as illustrated in Figure 4.8. The hyperplane is positioned to separate training vectors with different labels into the regions $R_1$ and $R_2$. $R_1$ contains training vectors with the label $\tau_j = 1$ and the positions of these training vectors are denoted using $X$’s. $R_2$ contains training vectors with the label $\tau_j = 2$ and the positions of these training vectors are denoted using O’s. The test vector $\omega_j$ is located in $R_1$, and is therefore assigned the label $\lambda_i = 1$.

For the theoretical development of the SVM, $D = 2$ classes are considered. The labels for the training vectors are defined as $\tau_j \in \{-1, 1\}$. The linear SVM defines a hyperplane [101]

$$ (\mathbf{v} \cdot \mathbf{\theta}_j) + b = 0 $$

where $(\mathbf{v} \cdot \mathbf{\theta}_j)$ is the dot product between $\mathbf{v}$ and $\mathbf{\theta}_j$, $\mathbf{v} \in \mathbb{R}^G$ is a random vector that is perpendicular to the hyperplane, $b \in \mathbb{R}$ is defined as an offset value [83].
The training vectors are assumed to be linearly separable meaning that the training vectors satisfy the boundary conditions of the following Equations [102]

\[
(v \cdot \theta_j) + b \geq +1 \quad \tau_j = 1 \tag{4.39}
\]

\[
(v \cdot \theta_j) + b \leq -1 \quad \tau_j = -1 \tag{4.40}
\]

The training vectors on the boundaries, specified by \((v \cdot \theta_j) + b = +1\) for \(\tau_j = 1\) and \((v \cdot \theta_j) + b = -1\) for \(\tau_j = -1\), are defined as support vectors [84]. The margin between the two boundaries in Equations 4.39 and 4.40 must be maximized. A unique solution exists for maximizing the boundaries using Lagrange’s theorem [103]:

\[
L_D = \sum_{j=1}^{J} \alpha_j - \frac{1}{2} \sum_{k=1}^{J} \sum_{j=1}^{J} \alpha_k \alpha_j \tau_k \tau_j (\theta_k \cdot \theta_j) \tag{4.41}
\]

where the constraints are defined by [102] [83]

\[
0 \leq \alpha_j \leq C \tag{4.42}
\]

\[
\sum_{j=1}^{J} \alpha_j \tau_j = 0 \tag{4.43}
\]

where \(\alpha_1, \ldots, \alpha_J\) are the Lagrange multipliers. The resulting decision function of the Lagrangian maximization procedure is [102]

\[
f(\omega_i) = \text{sign}\left[\sum_{j=1}^{J} \alpha_j \tau_j (\theta_j \cdot \omega_i)\right] \tag{4.44}
\]

where, for \(x \in \mathbb{R}\),

\[
\text{sign}[x] = \begin{cases} 
1 & , \ x \geq 0 \\
0 & , \ x < 0 
\end{cases} \tag{4.45}
\]
4.5.1 Nonlinear Support Vector Machine

A nonlinear SVM is used in place of the linear SVM if the training vectors are not linearly separable. The training vectors are separated by a nonlinear boundary. The nonlinear decision boundaries are determined by mapping the training vectors to some other Euclidian space $H$, $\Phi : \mathbb{R}^G \Rightarrow H$, where the mapping function is denoted as $\Phi$ [83]. A hyperplane is then used in $H$ to separate the mapped training vectors. The training vectors $\theta_k$ and $\theta_j$ of Equation 4.41 are mapped as $\Phi(\theta_k)$ and $\Phi(\theta_j)$, thereby replacing $(\theta_k \cdot \theta_j)$ as $\Phi(\theta_k) \cdot \Phi(\theta_j)$.

Mapping the training vectors to a higher dimension increases the complexity of the problem [104]. To avoid complexity, replace $\Phi(\theta_k) \cdot \Phi(\theta_j)$ by the kernel function $K(\theta_k, \theta_j) = \Phi(\theta_k) \cdot \Phi(\theta_j)$ [105], where the kernel function must satisfy Mercer’s condition [106].

The Lagrangian of Equation 4.41 is redefined for the nonlinear case by substituting $K(\theta_k, \theta_j)$ in the place of $(\theta_k \cdot \theta_j)$, which produces the same maximization problem. The Lagrangian for this problem is formulated as

$$L_D = \sum_{j=1}^{J} \alpha_j - \frac{1}{2} \sum_{k=1}^{J} \sum_{j=1}^{J} \alpha_k \alpha_j \tau_k \tau_j K(\theta_k, \theta_j)$$

(4.46)

under the same constraints of Equations 4.42 and 4.43. The Lagrangian is maximized under its constraints, which results in the decision function

$$f(\omega) = \text{sign} \left[ \sum_{j=1}^{J} \alpha_j \tau_j K(\theta_j, \omega) \right]$$

(4.47)

where $\text{sign}[]$ is defined by Equation 4.45.

4.5.2 Support Vector Machine Classifier

The SVM classifier generates its training model using the LIBSVM library, which supports multi-class classification [107]. A multi-class SVM is easily extended from
the binary SVM discussed above [102]. The kernel function used in the SVM algorithm is the radial basis function (RBF)

\[ K(\theta_j, \omega_i) = e^{-\gamma||\theta_j - \omega_i||^2} \]  \hspace{1cm} (4.48)

which satisfies Mercer's condition.

A block diagram of the SVM classifier is shown in Figure 4.5. This block diagram also shows how the classification accuracy \( P_d \) is measured. The LIBSVM library generates a training model using the parameters \( C \) and \( \gamma \). \( C \) is the upper bound of the Lagrange multipliers and \( \gamma \) is a parameter in the RBF of Equation 4.48. \( C \) and \( \gamma \) must be input into the classifier. For the simulations in Chapter 5, several combinations of \( C \) and \( \gamma \) are tested. Once the LIBSVM training model is generated, a test vector \( \omega_i \) is classified by LIBSVM based on the generated training model. To determine the average classification accuracy, the indicator function of Equation 4.8 compares \( \chi_i \) to \( \lambda_i \), and the average probability \( P_d \) of Equation 4.9 is measured for \( \mathbf{Y}_d \).
5. SIMULATIONS AND EXPERIMENTS

Three simulations and one experiment are presented in this Chapter. In the first simulation presented in Section 5.2, we develop a method to forensically characterize RF circuit models using first order distortion (FOD) and intermodulation distortion (IMD) generated by a nonlinearity in response to a two-tone probe signal. In the second simulation presented in Section 5.3, we develop methods to forensically characterize RF circuit models using harmonic distortion generated by nonlinearities in response to a Gaussian chirp signal. In this Section, we also develop methods to forensically characterize RF circuit models using the estimated filter response. In Section 5.4, we develop a method to forensically characterize actual physical RF circuits and circuit models using the filter response of two Chebyshev filters. The experimental results are compared with the simulations. Some of our results indicate that our simulation approach needs further investigation to validate it against actual measurement data.

5.1 RF Circuit Models

Three RF circuit models are introduced in this Section. Each circuit model is designed to simulate an RF device. A popular architecture that is widely used in RF devices is the superheterodyne architecture [108]. The superheterodyne architecture provides a high level of selectivity which is the ability to cancel out unwanted signals, and a high level of sensitivity which is the ability to detect small signals [12]. In some superheterodyne designs, the antenna is followed by a filter and then an amplifier [12]. When a probe signal is input into this superheterodyne architecture, it will be reflected by the amplifier since the amplifier is a source of a nonlinearity. A block diagram of this scenario is shown in Figure 5.1. This is simulated using the circuit model shown in
Figure 5.2, where $p(t)$ is the probe signal and $x(t)$ is the input into the nonlinearity. The nonlinearity is modeled empirically [2] and is represented by a power series of order $M$ [109]

$$y(t) = \sum_{j=1}^{M} a_j [x(t)]^j$$ \hspace{1cm} (5.1)

where $y(t)$ is the output of the nonlinearity. $d(t)$ is the output from the second filter that is added to the noise $n_c(t)$, and $r(t)$ is the return signal. The noise $n_c(t)$ is modeled as a white noise random process with a power spectrum, $\sigma_c^2$ [85]. The circuit models in the simulations of this Chapter will use random noise having multiple noise power spectrums. Multiple noise power spectrum are considered since it is possible that the noise in an RF device varies, this is typically dependent on the thermal level of the device [12] [108].

The two filters in Figure 5.2 are identical and assumed to be Chebyshev bandpass filters [8], which are common in RF circuits. The Chebyshev filter allows for flexible design of the filter response at the expense of complexity [110]. The Chebyshev filter has two main design features. The first is the passband ripple, which controls the quality of the passband, and the second is the rejection, which controls the quality of the stopband [110]. It is therefore possible to tradeoff passband quality with stopband quality.

This circuit model was suggested by Professor Larry Carin of Duke University.
Fig. 5.2. RF Circuit Model - Filter Present.
In other superheterodyne architectures, the antenna is followed directly by a amplifier [108], which is a source of a nonlinearity. When the probe signal is transmitted to this receiver, it is input into a nonlinearity and reflected. A block diagram of this scenario is shown in Figure 5.3. This scenario is simulated using the circuit model shown in Figure 5.4. Where \( p(t) \) (\( x(t) = p(t) \)) is the probe signal that is input into the nonlinearity. The nonlinearity is modeled empirically [2] and is represented by the power series in Equation 5.1. \( d(t) \) (where \( d(t) = y(t) \)) is the output of the nonlinearity that is added to the noise. The noise \( n_c(t) \) is again modeled as a white noise random process with a power spectrum \( \sigma_c^2 \) and \( r(t) \) is the return signal.
Let the average power of the output signal (return signal) of the circuit models in Figure 5.2 and 5.4 be \( \sigma^2 \). \( \sigma^2 \) is again modeled as a white noise random process with a power spectrum \( \sigma_c^2 \). \( r(t) \) is the return signal. Note that \( p(t) = 0 \), and therefore \( r(t) = n_c(t) \).

The final circuit model is the noise only model. This simulates the scenario when a probe signal is input into the RF circuit and only noise returns. This simulates the case when the probe does not incur any RF circuits. This model is shown in Figure 5.5. \( p(t) \) is the probe signal that is added to the noise \( n_c(t) \), where the noise is again modeled as a white noise random process with a power spectrum \( \sigma_c^2 \). \( r(t) \) is the return signal. Note that \( p(t) = 0 \), and therefore \( r(t) = n_c(t) \).

### 5.1.1 Signal to Noise Ratio

The definition of signal to noise ratio (SNR) for the circuit models is shown in this Section. The output signal (return signal) of the circuit models in Figure 5.2 and 5.4 is \( r(t) = d(t) + n_c(t) \), where \( d(t) \) is a deterministic signal and \( n_c(t) \) is a white noise random process with a power spectrum \( \sigma_c^2 \). We will define SNR in terms of average signal power to average noise power [108]. Let the average power of \( d(t) \) (signal) be [41] [9]

\[
P_d = \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} [d(t)]^2 dt
\]

Fig. 5.5. Noise Model.

\[
p(t) = 0 \quad \rightarrow \quad r(t)
\]

\[
\begin{align*}
\begin{array}{c}
\text{n(t)} \\
\end{array}
\end{align*}
\]
where $T = (t_2 - t_1)$ is the time interval from $t_1$ to $t_2$. The “noise power” will be assumed to be $\sigma_c^2$. The SNR of $r(t)$ is defined as \[108\]

$$\text{SNR} = \frac{P_d}{\sigma_c^2}$$ \hfill (5.3)

The SNR can also be expressed in dB or dBm:

$$\text{SNR}_{dB} = 10\log_{10}\left(\frac{P_d}{\sigma_c^2}\right)$$ \hfill (5.4)

$$\text{SNR}_{dBm} = 10\log_{10}\left(\frac{1000P_d}{\sigma_c^2}\right)$$ \hfill (5.5)

### 5.2 Two-Tone Signal Simulations

In this Section, we present a method to forensically characterize circuit models using first order distortion (FOD) and intermodulation distortion (IMD) generated by a nonlinearity in response to a two-tone probe signal. We are specifically interested in knowing how classification accuracy is effected by the power level of the FOD and IMD. For example, how is classification accuracy effected when the amplitude of the FOD and IMD is close to the noise floor?

The circuit models used in this simulation are based on the walkie-talkie experiment in Section 1.3 of Chapter 1. We will consider 6 circuit models, where one circuit model is the noise model. The noise model is shown in Figure 5.5. The other circuit models have a distinct nonlinearity and do not use filters as shown in Figure 5.4. No filters are considered in this simulation since we are only interested in the distortion caused by the nonlinearities. We will use 2 different noise power spectrums, $\{\sigma_{c1}^2, \sigma_{c2}^2\}$, in each circuit model. A set of two-tone probe signals are input into the circuit models. A set of return signals are generated by each circuit model in response to the $\sigma_c^2$ is not the true noise power since the power in a white noise process is undefined. We will use the spectral height of the white noise process as a measure of the power in the noise.
probe signals. The power spectrums of the return signals are obtained and features are sampled from the power spectrum and then classified.

5.2.1 Probe Signal Design

A set of two-tone probe signals were used in the walkie-talkie experiments in Section 1.3 of Chapter 1. Each two-tone signal of this experiment used different probe frequencies. The probe frequencies were chosen to be within the 240MHz - 500MHz frequency band. Following this approach, we will design a set of two-tone probe signals, \( P = \{p_1(t), \ldots, p_L(t)\} \), where

\[
p_l(t) = \cos(2\pi \alpha_l t) + \cos(2\pi \beta_l t)
\]

and \( \alpha_l = 100 \times 10^6 + (l - 1) \times 10^6 \), \( \beta_l = \alpha_l + 0.1 \times 10^6 \), and \( L = 901 \). Note that the probe signals in \( P \) are in the 100MHz-1000MHz frequency band, which is a different band than that used in the two-tone experiment presented in Section 1.3 of Chapter 1. A wider band of frequencies are considered since we are using multiple circuit models instead of only one (i.e., the walkie-talkie). In a practical application, multiple RF devices will be present in the environment. Each RF device will operate within various frequency bands.

5.2.2 RF Circuit Model Design

As discussed at the start of this Section, six RF circuit models \( \mathcal{Y} = \{\gamma_1, \ldots, \gamma_6\} \) are used to generate synthetic data. The circuit models \( \{\gamma_1, \ldots, \gamma_5\} \) have a distinct nonlinearity and do not use filters. \( \gamma_6 \) is the noise only model. An example power spectrum of the return signal generated by each circuit model in response to the two-tone probe signal is shown in Figure 5.6, where the noise power spectrum \( \sigma_c^2 = 5 \times 10^{-13} \), \( \alpha_{l=100} = 199 \times 10^6 \), and \( \beta_{l=100} = 199.1 \times 10^6 \).

The circuit models \( \{\gamma_1, \ldots, \gamma_5\} \) are designed with nonlinearities that create distortion (FOD, harmonic distortion, IMD) at specified amplitudes based on the walkie-
Fig. 5.6. Power Spectrums Generated by the 6 Circuit Models in Response to a Two-Tone Probe Signal.
talkie experiments. The coefficients used in the power series of Equation 5.1 are shown in Table 5.1 for each circuit model ($M = 3$). $\Upsilon_1$ is designed to simulate the FOD and second order harmonic distortion of the walkie-talkie experiments. It was shown in Section 1.3 of Chapter 1 that the FOD was -31.5dBm and -30.33dBm and the second order harmonic distortion was -58.64dBm and -58.26dBm. It was also shown that the third order IMD was close to the noise floor. For these simulations, the third order IMD is used as features and will therefore be designed to have detectable amplitudes at -50 dBm for $\Upsilon_1$. $\Upsilon_2$ is designed to generate distortion amplitudes, in response to the two-tone signal, near $\Upsilon_1$. This is done to determine if the classifiers can distinguish between circuit models that have similar features. $\Upsilon_3$ is designed to produce unique distortion amplitudes, in response to the two-tone signal, that are well separated from the distortion amplitudes of the other circuit models. $\Upsilon_4$ is also designed to produce unique distortion amplitudes, in response to the two-tone signal, that are well separated from the distortion amplitudes of the other circuit models. $\Upsilon_5$ is designed to generate distortion, in response to the two-tone signal, close to the noise floor.

Multiple noise power spectrums are used in each circuit model. We will consider the following noise power spectrums: $\sigma_c^2 = \{\sigma_{c_1}^2 = 5 \times 10^{-13}, \sigma_{c_2}^2 = 3 \times 10^{-12}\}$. The notation $\sigma_{c_n}^2 \in \sigma_c^2$ is used to denote the $n^{th}$ noise power spectrum, where $n \in \{1,..,N\}$ and $N = 2$.

<table>
<thead>
<tr>
<th></th>
<th>$\Upsilon_1$</th>
<th>$\Upsilon_2$</th>
<th>$\Upsilon_3$</th>
<th>$\Upsilon_4$</th>
<th>$\Upsilon_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_1$</td>
<td>2x10^{-6}</td>
<td>1.5x10^{-6}</td>
<td>0.15x10^{-6}</td>
<td>0.05x10^{-6}</td>
<td>0.0028x10^{-6}</td>
</tr>
<tr>
<td>$a_2$</td>
<td>0.1x10^{-6}</td>
<td>0.1x10^{-6}</td>
<td>0.1x10^{-6}</td>
<td>0.1x10^{-6}</td>
<td>0.1x10^{-6}</td>
</tr>
<tr>
<td>$a_3$</td>
<td>0.5x10^{-6}</td>
<td>0.4x10^{-6}</td>
<td>0.04x10^{-6}</td>
<td>0.02x10^{-6}</td>
<td>0.0028x10^{-6}</td>
</tr>
</tbody>
</table>
5.2.3 Return Signals

The circuit models are used to generate simulated ground truth data in response to the set of two-tone probes signals in \( P \). The set of probe signals is input into each of the six models using a noise power in \( \sigma^2_c \). Each circuit model outputs \( L, L = 901 \), return signals. Each return signal will have a label, which is used to denote the model that produced the return signal. The return signal and label are the observations or ground truth data. Circuit model \( \Upsilon_d \), using the noise power spectrum \( \sigma^2_{c_n} \), generates the follow set of ground truth data in response to the set \( P \):

\[
O_{(n,d)} = \{[r_{(n,d,1)}(t), \gamma_{(n,d,1)} = d], ...[r_{(n,d,L)}(t), \gamma_{(n,d,L)} = d]\} \tag{5.7}
\]

where the notation \( d \) signifies the circuit model \( \Upsilon_d \) and \( n \) signifies the noise power spectrum \( \sigma^2_{c_n} \), \( \gamma_{(n,d,l)} \) is the ground truth label, and \( r_{(n,d,l)}(t) \) is the return signal generated in response to the probe \( p_l(t) \). The set of observations produced by all circuit models utilizing both noise power spectrums is

\[
O = \{O_{(1,1)}, ...O_{(1,6)}, O_{(2,1)}, ...O_{(2,6)}\} \tag{5.8}
\]

Note that the total number of return signals is 10,812. For all the return signals in \( O \) from Equation 5.8, power spectrums are obtained. The power spectrums in \( O_{(n,d)} \) are

\[
P_{O_{(n,d)}} = \{[P_{r_{(n,d,1)}(f)}, \gamma_{(n,d,1)} = d], ...[P_{r_{(n,d,L)}(f)}, \gamma_{(n,d,L)} = d]\} \tag{5.9}
\]

where

\[
P_{r_{(n,d,l)}(f)} = R_{(n,d,l)}(f)[R_{(n,d,l)}(f)]^* \tag{5.10}
\]

and

\[
R_{(n,d,l)}(f) = \int_{-\infty}^{\infty} r_{(n,d,l)}(t)e^{-i2\pi ft} dt \tag{5.11}
\]
The power spectrums form the set

\[ P_O = \{ P_{O(1,1)}, \ldots, P_{O(2,6)} \} \] (5.12)

We will next determine the average SNR for each set \( P_{O(n,d)} \) of power spectrums. First, by using Equation 5.5, we can define the SNR for the given power spectrum \( P_{r(n,d,l)}(f) \) as \( SNR_{(n,d,l)} \), where \( P_{r(n,d,l)}(f) \in P_{O(n,d)} \). Then, we will let the average SNR for the power spectrums in \( P_{O(n,d)} \) be

\[ ASNR_{(n,d)} = \frac{\sum_{l=1}^{L} SNR_{(n,d,l)}}{L} \] (5.13)

Note \( d \neq 6 \) in Equation 5.13 since the SNR is not determined for the noise model. In addition to the average SNR, we will also determine the average intermodulation distortion ratio (IMR) for each set \( P_{O(n,d)} \) of power spectrums. The IMR for the power spectrum \( P_{r(n,d,l)}(f) \in P_{O(n,d)} \) is

\[ IMR_{(n,d,l)} = \frac{P_{r(n,d,l)}(2f_1 - f_2)}{P_{r(n,d,l)}(f_1)} \] (5.14)

Then, we will let the average IMR for the power spectrums in \( P_{O(n,d)} \) as

\[ AIMR_{(n,d)} = \frac{\sum_{l=1}^{L} IMR_{(n,d,l)}}{L} \] (5.15)

where \( d \neq 6 \) in Equation 5.15 since the SNR is not determined for the noise models. The average SNR and IMR for the circuit models is shown in Table 5.2. Note that \( ASNR_{(1,5)} \) and \( ASNR_{(2,5)} \) are low relative to the other average SNR values. Also note that \( AIMR_{(1,5)} \) and \( AIMR_{(2,5)} \) have large IMR’s. This is caused by the high noise power used in circuit model \( \Upsilon_5 \). As shown in Figure 5.6(e), the amplitude of the FOD is near the noise floor and the IMD is noise. Therefore, the ratio of the IMD to FOD increases.
Table 5.2
Average SNR and IMR for the RF Circuit Models Used in the Two-Tone Simulations.

<table>
<thead>
<tr>
<th></th>
<th>d=1</th>
<th>d=2</th>
<th>d=3</th>
<th>d=4</th>
<th>d=5</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASNR(_{(1,d)}) (dBm)</td>
<td>44.5</td>
<td>42.2</td>
<td>22.5</td>
<td>15.3</td>
<td>2.4</td>
</tr>
<tr>
<td>ASNR(_{(2,d)}) (dBm)</td>
<td>36.7</td>
<td>34.4</td>
<td>14.7</td>
<td>7.5</td>
<td>-5.4</td>
</tr>
<tr>
<td>AIMR(_{(1,d)})</td>
<td>0.020</td>
<td>0.021</td>
<td>0.021</td>
<td>0.032</td>
<td>0.142</td>
</tr>
<tr>
<td>AIMR(_{(2,d)})</td>
<td>0.020</td>
<td>0.021</td>
<td>0.022</td>
<td>0.035</td>
<td>0.510</td>
</tr>
</tbody>
</table>

5.2.4 Training and Testing

Features are extracted from each power spectrum in \(P_O\) of Equation 5.12. The method used for feature extraction was discussed in Section 3.1 of Chapter 3. Four amplitudes were sampled from a power spectrum and formed into a feature vector. For the set of power spectrums in \(P_O\), the four features are extracted as follows:

\[
e_{(k,1)} = P_{r(n,d,l)}(\alpha_l) \quad (5.16)
\]

\[
e_{(k,2)} = P_{r(n,d,l)}(\beta_l) \quad (5.17)
\]

\[
e_{(k,3)} = P_{r(n,d,l)}(2\alpha_l - \beta_l) \quad (5.18)
\]

\[
e_{(k,4)} = P_{r(n,d,l)}(2\beta_l - \alpha_l) \quad (5.19)
\]

where, given any values of \(n\), \(d\), and \(l\), \(k = (n - 1)DL + (d - 1)L + l\), \(k \in \{1, \ldots K\}\), and \(K = 10,812\). The features form the feature vector

\[
\eta_k = [e_{(k,1)} \ e_{(k,2)} \ e_{(k,3)} \ e_{(k,4)}]
\]

(5.20)
where \( \eta_k \in \mathbb{R}^d \). Also, let \( \Gamma_k = \gamma(n,d,l) \). We can then order the feature vectors and labels as

\[
\{[\eta_1, \Gamma_1], \ldots, [\eta_K, \Gamma_K]\}
\]

where \([\eta_k, \Gamma_k]\) is the \( k \)th observation. Each observation or ground truth set in Equation 5.21 is randomly divided into the training and testing ground truth sets. The training and testing sets were introduced in Chapter 4. Before dividing the observations in Equation 5.21, first let \( \nu(k) \) be a random variable with a uniform distribution on the interval \([0, 1]\) indexed by the integer \( k \). For the \( k \)th observation in Equation 5.21, we let

\[
\begin{align*}
\{\theta_j = \eta_k, \tau_j = \Gamma_k\} & , \quad \nu(k) \geq 0.5 \\
\{\omega_i = \eta_k, \lambda_i = \Gamma_k\} & , \quad \nu(k) < 0.5
\end{align*}
\]

where \( \theta_j \) is the \( j \)th training vector with label \( \tau_j \), and \( \omega_i \) is the \( i \)th test vector with a label \( \lambda_i \). The training vectors form the set

\[
\Theta = \{[\theta_1, \tau_1], \ldots, [\theta_J, \tau_J]\}
\]

Note that since we are randomly dividing the observations in Equation 5.21, \( J \) is a discrete random variable where \( J \in \{0, 1, \ldots, K\} \). The test vectors form the set

\[
\Omega = \{[\omega_1, \lambda_1], \ldots, [\omega_I, \lambda_I]\}
\]

Also note \( I \) is a discrete random variable where \( I \in \{0, 1, \ldots, K\} \) and \( K = J + I \).

For the training set described by Equation 5.23, the number of total observations and the number observations per category are shown in Table 5.3, where \( Q_d \in J \) was described in Chapter 4 as the number of observations that have the label \( \tau_j = d \). For the test set defined in Equation 5.24, the total number of observations and the

\(^3\)These simulations used the \textit{MATLAB} function \texttt{rand()} to generate a pseudorandom number drawn from the uniform distribution.
Table 5.3
Number of Feature Vectors Per Category for the Training Set for the Two-Tone Simulations.

<table>
<thead>
<tr>
<th>J</th>
<th>Q_1</th>
<th>Q_2</th>
<th>Q_3</th>
<th>Q_4</th>
<th>Q_5</th>
<th>Q_6</th>
</tr>
</thead>
<tbody>
<tr>
<td>5437</td>
<td>901</td>
<td>906</td>
<td>889</td>
<td>908</td>
<td>927</td>
<td>906</td>
</tr>
</tbody>
</table>

Table 5.4
Number of Feature Vectors Per Category for the Testing Set for the Two-Tone Simulations.

<table>
<thead>
<tr>
<th>I</th>
<th>R_1</th>
<th>R_2</th>
<th>R_3</th>
<th>R_4</th>
<th>R_5</th>
<th>R_6</th>
</tr>
</thead>
<tbody>
<tr>
<td>5375</td>
<td>901</td>
<td>896</td>
<td>913</td>
<td>894</td>
<td>875</td>
<td>896</td>
</tr>
</tbody>
</table>

number observations per category are shown in Table 5.4, where $R_d \in I$ was defined in Chapter 4 as the number of observations that have the label $\lambda_i = d$.

5.2.5 Classification Results

We will measure the average classification accuracy for each circuit model in $\mathcal{Y}$. As discussed in Chapter 4, this accuracy is

$$P_d = \frac{\sum_{i=1}^{I} I_f(i)}{R_d}$$

where

$$I_f(i) = \begin{cases} 1 & , & \lambda_i = \chi_i \\ 0 & , & \text{else} \end{cases}$$

is the indicator function and $\chi_i$ is the predicted label decided by the classifier. The K nearest neighbor (K-NN) classifier, Parzen window (PW) classifier, and support vector machine (SVM) require the selection of parameters prior to classification. For example, the K-NN classifier requires the number of nearest neighbors $K_J$. An
interesting question is: what value of $K_J$ has the highest classification accuracy? To answer this question, we define $\pi$ as a classifier parameter. For the K-NN classifier, $\pi = K_J$. To determine the value of the parameter that produces the highest classification accuracy, we will need to test several values of $\pi$. Define a set of parameter values as $\Pi = \{\pi_1, ..., \pi_V\}$. The total number of parameter values in the set is denoted as $V$. For the K-NN classifier, assume we would like to test the following number of nearest neighbors: $K_J = 1$, $K_J = 2$, $K_J = 3$. The parameter set $\Pi$ would therefore be defined as $\Pi = \{\pi_1 = 1, \pi_2 = 2, \pi_3 = 3\}$.

An average classification accuracy is determined for each parameter $\pi_v \in \Pi$. Define

$$P_{(\text{ave}, v)} = \frac{1}{I} \sum_{i=1}^{I} I_f(i)$$

(5.27)

as the average classification accuracy of all categories using the parameter $\pi_v$. Next, let

$$v_{\text{max}} = \arg \max_v P_{(\text{ave}, v)}$$

(5.28)

The parameter $\pi_{v_{\text{max}}}$ is the parameter that yields the highest average classification accuracy of the circuit models, and is used by the classifier for the results that will be presented in this Section.

The K-NN classifier requires the number of nearest neighbors, $K_J$, prior to classification. Define the parameter set for the K-NN classifier as $\Pi = \{\pi_1 = 1, ..., \pi_{50} = 50\}$, where the parameters correspond to the number of nearest neighbors used by the classifier. It was determined that $v_{\text{max}} = 24$, therefore $K_J = \pi_{24} = 24$. The PW classifier requires the window width, $h$, prior to classification. Define the parameter set for the PW classifier as $\Pi = \{\pi_1 = 0.1, \pi_2 = 0.3, ..., \pi_{25} = 4.9\}$, where these parameters correspond to the window width used by the classifier. It was determined that $v_{\text{max}} = 10$, therefore $h = \pi_{10} = 1.9$. The SVM classifier requires two inputs $C$ and $\gamma$ prior to classification. Note that $C \in \{c_1 = 1, c_2 = 3, ..., c_7 = 13\}$ and
Table 5.5
Average Classification Accuracies for the Two-Tone Simulations.

<table>
<thead>
<tr>
<th></th>
<th>$P_1$</th>
<th>$P_2$</th>
<th>$P_3$</th>
<th>$P_4$</th>
<th>$P_5$</th>
<th>$P_6$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>92.1%</td>
<td>94.3%</td>
</tr>
<tr>
<td>BTC</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>91.4%</td>
<td>92.7%</td>
</tr>
<tr>
<td>Bayesian$_\mu$</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>99.3%</td>
<td>68.9%</td>
</tr>
<tr>
<td>Bayesian$_\Sigma$</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>93.6%</td>
<td>90.8%</td>
</tr>
<tr>
<td>PW</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>93.1%</td>
<td>93.6%</td>
</tr>
<tr>
<td>K-NN</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>93%</td>
<td>94%</td>
</tr>
</tbody>
</table>

$\gamma \in \{\gamma_1 = 0.01, \gamma_2 = 0.03, ..., \gamma_{250} = 4.99\}$. Let the parameter set for the SVM classifier be $\Pi = \{\pi_1 = \{c_1, \gamma_1\}, ..., \pi_{1750} = \{c_7, \gamma_{250}\}\}$. It was determined that $v_{\text{max}} = 251$, therefore $C = 3$ and $\gamma = 0.01$.

The average classification accuracies for the two-tone simulations are shown in Table 5.5. By examining $P_5$ and $P_6$, this Table indicates that all classifiers misclassified test vectors that have labels $\lambda_i = \Upsilon_5$ and $\lambda_i = \Upsilon_6$ (noise category). This is caused by the fact that the feature vectors with the label $\lambda_i = \Upsilon_5$ are extracted from power spectrums that have FOD and IMD near the noise floor, which is evident by the average SNR values $\text{ASNR}_{(1,5)}$ and $\text{ASNR}_{(2,5)}$ in Table 5.2. This is also evident by examining Figure 5.6(e), which is a power spectrum generated by $\Upsilon_5$, in response to a two-tone signal, using $\sigma^2_{c_1}$. As illustrated in this Figure, the FOD is near the noise floor and the IMD is not detectable. Therefore, the feature vectors with label $\lambda_i = \Upsilon_5$ are being classified as category $\Upsilon_6$ (noise) and the feature vectors with label $\lambda_i = \Upsilon_6$ are being classified as category $\Upsilon_5$. In addition, the Bayesian$_\mu$ classifier has a large number of classification errors, which is indicated by examining $P_6$ in Table 5.5.
5.2.6 Conclusions

For the Two-Tone simulations, we presented a method to forensically characterize circuit models using features of the FOD and IMD that were generated in response to a two-tone signal. It was shown that classification errors occur for features sampled from power spectrums with low SNR. All classifiers, except the Bayesian $\mu$, had high average classification accuracies, which is evident by examining Table 5.5.

5.3 Gaussian Chirp Signal Simulations

In this Section, we present a method to forensically characterize circuit models using harmonic distortion generated by nonlinearities in response to a Gaussian chirp signal. This method was discussed in Section 2.2 of Chapter 2. We will also present a method to forensically characterize circuit models using the filter response to a Gaussian chirp signal. This method was discussed in Section 2.2.1 of Chapter 2. We will implement two feature selection methods and compare the classification results. Both feature selection techniques were discussed in Section 3.2 of Chapter 3. The first feature selection technique samples the power spectrum of the return signal at regular intervals within a set band of frequencies. The second feature selection method samples the power spectrum of the return signal at precise locations. For each feature selection method, five circuit models are used, where one of the circuit models is a noise only model. We will use five noise power spectrums, \( \{\sigma_{c_1}^2, \sigma_{c_2}^2, \sigma_{c_3}^2, \sigma_{c_4}^2, \sigma_{c_5}^2\} \), for each circuit model. A set of Gaussian chirp probe signals are input into the circuit models. A set of return signals are generated by each circuit model in response to the set of probe signals. Power spectrums of the return signals are obtained and features are extracted based on one of the two feature selection techniques. The features are classified and the results are compared.
5.3.1 Probe Signal Design

A set of Gaussian chirp signals will be used as input to the five circuit models. The set of signals are \( P = \{p_1(t), \ldots p_L(t)\} \), where

\[
p_t(t) = p(t) = C_1 e^{i(2\pi f_0 t + \pi kt^2)} e^{-\frac{(t-\mu)^2}{2\sigma^2}}
\] (5.29)

where each parameter is described in Table 5.6 and \( L = 500 \). An example of \( p(t) \) is shown in Figure 5.7. The bandwidth of \( p(t) \) was defined in Section 1.4 of Chapter 1 as \[38\]

\[
B^2 = \frac{1}{E} \int_{-\infty}^{\infty} (2\pi f)^2 |P(f)|^2 df
\] (5.30)

where \(|P(f)|\) is the magnitude of the Fourier transform of \( p(t) \), and \( E \) is the energy of \( p(t) \) defined by \[9\]

\[
E = \int_{-\infty}^{\infty} |p(t)|^2 dt
\] (5.31)

The bandwidth of \( p(t) \) is \( B = 3.82 \text{GHz} \).

As discussed in Section 2.2 of Chapter 2, harmonic distortion occurs in the return signal. It is possible that an overlap of the harmonic distortion components will occur in the frequency domain. The concept of the overlapping components in the frequency domain of the return signal was discussed in Section 2.2 of Chapter 2. The overlap is measured by

\[
\epsilon = \frac{(\sigma_\xi(j) + \sigma_\xi(l))^2}{(\mu_\xi(l) - \mu_\xi(j))^2}
\] (5.32)

where

\[
\mu_\xi(j) = j[f_0 + \mu k]
\] (5.33)

where \( j \) denotes the \( j^{th} \) order harmonic, and
Table 5.6
Gaussian Chirp Signal Parameters.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Description</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_1$</td>
<td>Amplitude of signal</td>
<td>1</td>
</tr>
<tr>
<td>$i$</td>
<td>Imaginary number</td>
<td>$\sqrt{-1}$</td>
</tr>
<tr>
<td>$f_0$</td>
<td>Frequency at $t = 0$</td>
<td>1Hz</td>
</tr>
<tr>
<td>$f_1$</td>
<td>Frequency at $t_1$</td>
<td>2GHz</td>
</tr>
<tr>
<td>$t_1$</td>
<td>Final time</td>
<td>50$\mu$s</td>
</tr>
<tr>
<td>$t$</td>
<td>Time</td>
<td>$0 \leq t \leq 50\mu$s</td>
</tr>
<tr>
<td>$k$</td>
<td>Chirp rate</td>
<td>$4 \times 10^{13}$</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Mean</td>
<td>$1.5 \times 10^{-5}$</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>Standard deviation</td>
<td>$3.5 \times 10^{-6}$</td>
</tr>
</tbody>
</table>

Fig. 5.7. Example of the Gaussian Chirp Probe Signal. Parameters of the Probe Signal are Defined in Table 5.6.
\[ \sigma^2_\xi(j) = \frac{j\gamma^2}{8\pi^2\sigma^2} \]  
(5.34)

and

\[ \gamma = \sqrt{1 + 4\pi^2\sigma^4k^2} \]  
(5.35)

It was also shown in Section 2.2 of Chapter 2 that \( \mu_\xi(l) > \mu_\xi(j) \), \( \sigma_\xi(j) > 0 \), and \( \sigma_\xi(l) > 0 \). Note that \( l \) can replace \( j \) in Equations 5.33 and 5.34.

Two values of \( \epsilon \) are determined, where the first is for the overlap between the first and second order harmonics. The second value of \( \epsilon \) is for the overlap between the second and third order harmonics. Before finding \( \epsilon \), we will determine \( \mu_\xi(j) \) and \( \sigma^2_\xi(j) \) for \( j = 1 \), \( j = 2 \), and \( j = 3 \). Using Equations 5.33 and 5.34, it can be shown that \( \mu_\xi(1) \approx 600 \times 10^6 \), \( \mu_\xi(2) \approx 1200 \times 10^6 \), \( \mu_\xi(3) \approx 1800 \times 10^6 \), \( \sigma_\xi(1) \approx 99 \times 10^6 \), \( \sigma_\xi(1) \approx 140 \times 10^6 \), and \( \sigma_\xi(1) \approx 171 \times 10^6 \). Substituting these values into Equation 5.32 for \( j = 1 \) and \( l = 2 \) yields \( \epsilon_{12} = 0.159 \), which is the overlap between the first and second order harmonics. Substituting these values into Equation 5.32 for \( j = 2 \) and \( l = 3 \) yields \( \epsilon_{23} = 0.269 \), which is the overlap between the second and third order harmonics. Note that \( \epsilon_{23} > \epsilon_{12} \), indicates that more overlap occurs for the second and third order harmonics. We assume that the overlap, determined by \( \epsilon_{12} \) and \( \epsilon_{23} \), is small enough for accurate device characterization.

### 5.3.2 Circuit Model Design

\( D = 5 \) circuit models are considered in these simulations and are referred to as \( \Upsilon = \{ \Upsilon_1, \ldots, \Upsilon_D \} \). \( \Upsilon_d \) is known as the \( d^{th} \) circuit model, where \( d \in \{ 1, \ldots, D \} \). Circuit model \( \Upsilon_5 \) is the noise only model. The circuit models \( \{ \Upsilon_1, \ldots, \Upsilon_4 \} \) have a distinct nonlinearity of order \( M = 3 \). The circuit models for \( \Upsilon_1 \) and \( \Upsilon_3 \) use filters and the circuit models for \( \Upsilon_2 \) and \( \Upsilon_4 \) do not use filters. As discussed in Section 2.2.1 of Chapter 2, the Gaussian chirp signal can be designed to estimate the filter response of the circuits with filters if the filter response is known a priori. This requires that the
Table 5.7
Circuit Model Parameters Used in the Gaussian Chirp Simulations.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Y₁</th>
<th>Y₂</th>
<th>Y₃</th>
<th>Y₄</th>
</tr>
</thead>
<tbody>
<tr>
<td>a₁</td>
<td>0.01</td>
<td>0.001</td>
<td>0.003</td>
<td>0.007</td>
</tr>
<tr>
<td>a₂</td>
<td>0.008</td>
<td>0.0009</td>
<td>0.001</td>
<td>0.0009</td>
</tr>
<tr>
<td>a₃</td>
<td>0.009</td>
<td>0.00007</td>
<td>0.0007</td>
<td>0.00007</td>
</tr>
<tr>
<td>Filter Passband (MHz)</td>
<td>400-450</td>
<td>N/A</td>
<td>670-710</td>
<td>N/A</td>
</tr>
<tr>
<td>Filter Order</td>
<td>2</td>
<td>N/A</td>
<td>4</td>
<td>N/A</td>
</tr>
</tbody>
</table>

The passband of the filters in Y₁ and Y₃ are within the frequency band (i.e., f₀ to f₁) of the Gaussian chirp signal. As will be shown, the passband of both filters in Y₁ and Y₃ are within the frequency band of the Gaussian chirp signal and can therefore be estimated.

Figure 5.8 shows an example of a power spectrum of a return signal generated by each circuit model in response to the probe signal defined in Equation 5.29, where the noise power spectrum is \( \sigma_c^2 = 5 \times 10^{-9} \). As shown in Figures 5.8(a) and 5.8(c), the Gaussian chirp signal is used to estimate the filter response. In addition, the Gaussian chirp signal is used to produce harmonic distortion as shown in Figures 5.8(b) and 5.8(d).

The nonlinearities and filter parameters for the circuit models are shown in Table 5.7. The nonlinearities are chosen such that the passband of the filters in Y₁ generates, in response to the Gaussian chirp signal, a filter response with higher energy than that of Y₃. Also, the nonlinearities are chosen such that Y₂ generates, in response to the Gaussian chirp signal, harmonic distortions with higher energy compared with that of Y₄, where the harmonic distortion for Y₄ is near the noise floor.

As mentioned at the beginning of this Section, multiple noise power spectrums are used in each circuit model. We will consider the following noise power spectrums:

\( \sigma_C = \{ \sigma_{c_1}^2 = 5 \times 10^{-10}, \sigma_{c_2}^2 = 5 \times 10^{-9}, \sigma_{c_3}^2 = 1 \times 10^{-8}, \sigma_{c_4}^2 = 5 \times 10^{-8}, \sigma_{c_5}^2 = 5 \times 10^{-7} \} \).
Fig. 5.8. Example Power Spectrums of the Return Signal From Each Circuit Model.

notation $\sigma_{cn} \in \sigma_C$ is used to denote the $n^{th}$ noise power spectrum, where $n \in \{1, \ldots N\}$ and $N = 5$. 
5.3.3 Generation of the Return Signals

The circuit models are used to generate simulated data in response to the set of Gaussian chirp signals in $P$. Each circuit model, using a particular noise power spectrum, outputs $L$ return signals, where $L = 500$. Each return signal will have a label (ground truth) associated with it, which is used to denote the circuit model that generated the return signal. Each return signal and label pair are known as an observation. Circuit model $\gamma_d$, using the noise power spectrum $\sigma_{cn}$, generates the follow set of observations in response to the set $P$:

$$O_{(n,d)} = \{[r_{(n,d,1)}(t), \gamma_{(n,d,1)} = d], \ldots [r_{(n,d,L)}(t), \gamma_{(n,d,L)} = d]\} \quad (5.36)$$

where the notation $d$ signifies the circuit model $\gamma_d$ and $n$ signifies the noise power spectrum $\sigma_{cn}^2$, $\gamma_{(n,d,l)}$ is the ground truth label, and $r_{(n,d,l)}(t)$ is the return signal generated in response to the probe $p_l(t)$. The set of observations produced by all circuit models and all noise power spectrums is

$$O = \{O_{(1,1)}, \ldots O_{(1,5)}, O_{(2,1)}, \ldots O_{(2,5)}, \ldots, \ldots O_{(5,1)}, \ldots O_{(5,5)}\} \quad (5.37)$$

Note that the total number of return signals is 12,500. The power spectrums are obtained for all return signals in the set $O$ from Equation 5.37. The power spectrums of the return signals in the set $O_{(n,d)}$ are

$$P_{O_{(n,d)}} = \{[P_{r_{(n,d,1)}(f)}, \gamma_{(n,d,1)} = d], \ldots [P_{r_{(n,d,L)}(f)}, \gamma_{(n,d,L)} = d]\} \quad (5.38)$$

where

$$P_{r_{(n,d,l)}(f)} = R_{(n,d,l)}(f)[R_{(n,d,l)}(f)]^* \quad (5.39)$$

and

$$R_{(n,d,l)}(f) = \int_{-\infty}^{\infty} r_{(n,d,l)}(t)e^{-i2\pi ft} dt \quad (5.40)$$
Table 5.8
Average SNR for Each Circuit Model and Each Noise Power. SNR is Defined in dB.

<table>
<thead>
<tr>
<th></th>
<th>$\gamma_1$</th>
<th>$\gamma_2$</th>
<th>$\gamma_3$</th>
<th>$\gamma_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\sigma^2_{c_1}$</td>
<td>28.1</td>
<td>37.9</td>
<td>20.6</td>
<td>22.9</td>
</tr>
<tr>
<td>$\sigma^2_{c_2}$</td>
<td>18.1</td>
<td>27.9</td>
<td>10.6</td>
<td>12.9</td>
</tr>
<tr>
<td>$\sigma^2_{c_3}$</td>
<td>15.04</td>
<td>24.9</td>
<td>7.54</td>
<td>9.9</td>
</tr>
<tr>
<td>$\sigma^2_{c_4}$</td>
<td>8.05</td>
<td>17.9</td>
<td>0.55</td>
<td>2.91</td>
</tr>
<tr>
<td>$\sigma^2_{c_5}$</td>
<td>-1.94</td>
<td>7.93</td>
<td>-9.45</td>
<td>-7.09</td>
</tr>
</tbody>
</table>

The power spectrums of the return signals form the set

$$P_O = \{P_{O(1,1)}, \ldots, P_{O(5,5)}\} \quad (5.41)$$

We will next determine the average SNR for each set $P_{O(n,d)}$ of power spectrums. First, by using Equation 5.4, we can define the SNR for the given power spectrum $P_{r(n,d,l)}(f)$ as $SNR_{(n,d,l)}$, where $P_{r(n,d,l)}(f) \in P_{O(n,d)}$. Then, we will define the average SNR for the power spectrums in $P_{O(n,d)}$ as

$$ASNR_{(n,d)} = \frac{\sum_{l=1}^{L} SNR_{(n,d,l)}}{L} \quad (5.42)$$

where $d \neq 5$ in Equation 5.42 since the SNR is not determined for the noise model. The average SNR is shown in Table 5.8. Note that the average SNR for row $\sigma^2_{c_5}$ and columns $\gamma_1$, $\gamma_3$, and $\gamma_4$ are low relative to the other average SNR values.

5.3.4 Features Selected at Regular Intervals

Features are extracted from each power spectrum in $P_O$ of Equation 5.41. The methods used for feature extraction were discussed in Section 3.2 of Chapter 3. As discussed, amplitudes (features) were sampled from a power spectrum and formed
into a feature vector. The locations of the samples are determined based on one of two methods: features selected at regular intervals and features selected at precise locations. Each method will be implemented separately and will constitute a different simulation. The process of selecting features at regular intervals is discussed in this Section.

When features are selected at regular intervals, \( G = 34 \) features are used. The frequency range sampled is 200MHz - 1900MHz in increments of \( \Delta G \), where \( \Delta G = 50MHz \). This frequency range is chosen so that the first, second, and third order harmonic distortions are sampled. As discussed in Section 3.2 of Chapter 3, the set of frequencies corresponding to the features are \( \rho_g = \rho_{g-1} + \Delta \), for \( 2 \leq g \leq G \). \( \Delta \) is a constant offset value and \( \rho_1 = 200MHz \). The amplitude is sampled at:

\[
e_{(k,g)} = P_{r(n,d,l)}(\rho_g)
\]

where, given any values of \( n, d, \) and \( l, k = (n-1)DL + (d-1)L + l, k \in \{1, \ldots K\}, \) and \( K = 12,500 \). The features form the feature vector

\[
\eta_k = [e_{(k,1)} \ldots e_{(k,G)}]
\]

where \( \eta_k \in \mathbb{R}^G \). Also, we let

\[
\Gamma_k = \gamma(n,d,l)
\]

We can then order the feature vectors and labels as

\[
\{[\eta_1, \Gamma_1], \ldots [\eta_K, \Gamma_K]\}
\]

where \( [\eta_k, \Gamma_k] \) is known as the \( k^{th} \) observation (ground truth). Each observation in Equation 5.46 is randomly divided into the training and testing sets. The concept training and testing sets were introduced in Chapter 4. Before dividing the observations in Equation 5.46, first let \( \nu(k) \) be a random variable with a uniform
For the $k^{th}$ observation in Equation 5.46, we let

\[\{\theta_j = \eta_k, \tau_j = \Gamma_k\}, \quad \nu(k) \geq 0.5\]
\[\{\omega_i = \eta_k, \lambda_i = \Gamma_k\}, \quad \nu(k) < 0.5\]  

(5.47)

where $\theta_j$ is the $j^{th}$ training vector with a label $\tau_j$, and $\omega_i$ is the $i^{th}$ test vector with a label $\lambda_i$. The training vectors form the set

\[\Theta = \{[\theta_1, \tau_1], \ldots, [\theta_J, \tau_J]\}\]  

(5.48)

Note that since we are randomly dividing the observations in Equation 5.46, $J$ is a discrete random variable where $J \in \{0, 1, \ldots, K\}$.

The test vectors form the set

\[\Omega = \{[\omega_1, \lambda_1], \ldots, [\omega_I, \lambda_I]\}\]  

(5.49)

Also note $I$ is a discrete random variable where $I \in \{0, 1, \ldots, K\}$ and $K = J + I$.

We will introduce the notation $\Omega_{(n,d)}$, which signifies the set of test vectors with features extracted from the power spectrums in $P_{O_{(n,d)}}$. This notation is necessary when discussing the classification results presented in the next Section.

For the training set described by Equation 5.48, the number of total observations and the number observations per category are shown in Table 5.9, where $Q_d \in J$ was defined in Chapter 4 as the number of observations that have the label $\tau_j = d$.

For the test set in Equation 5.49, the total number of observations and the number observations per category are shown in Table 5.10, where $R_d \in I$ was defined in Chapter 4 as the number of observations that have the label $\lambda_i = d$.

---

4 These simulations used the MATLAB function `rand()` to generate a pseudorandom number drawn from the uniform distribution.
Table 5.9
Number of Feature Vectors Per Category for the Training Set. Features are Sampled at Regular Intervals.

<table>
<thead>
<tr>
<th>J</th>
<th>Q_1</th>
<th>Q_2</th>
<th>Q_3</th>
<th>Q_4</th>
<th>Q_5</th>
</tr>
</thead>
<tbody>
<tr>
<td>6243</td>
<td>1261</td>
<td>1217</td>
<td>1294</td>
<td>1249</td>
<td>1222</td>
</tr>
</tbody>
</table>

Table 5.10
Number of Feature Vectors Per Category for the Testing Set. Features are Sampled at Regular Intervals.

<table>
<thead>
<tr>
<th>I</th>
<th>R_1</th>
<th>R_2</th>
<th>R_3</th>
<th>R_4</th>
<th>R_5</th>
</tr>
</thead>
<tbody>
<tr>
<td>6257</td>
<td>1239</td>
<td>1283</td>
<td>1206</td>
<td>1251</td>
<td>1278</td>
</tr>
</tbody>
</table>
Simulation Results

We will measure the average classification accuracy for each circuit model in \( \Upsilon \). As discussed in Chapter 4, this accuracy is

\[
P_d = \frac{\sum_{i=1}^{I} I_f(i)}{R_d} \tag{5.50}
\]

where

\[
I_f(i) = \begin{cases} 
1 & , \quad \lambda_i = \chi_i \\
0 & , \quad \text{else} 
\end{cases} \tag{5.51}
\]

is the indicator function and \( \chi_i \) is the predicted label decided by the classifier. The K nearest neighbor (K-NN) classifier, Parzen window (PW) classifier, and support vector machine (SVM) require parameters prior to classification. As discussed in Section 5.2.5, we will define a set of parameters \( \Pi = \{\pi_1, ..., \pi_V\} \) for each classifier to determine the highest classification accuracy. The K-NN classifier requires the number of nearest neighbors, \( K_J \), prior to classification. Let the parameter set for the K-NN classifier be \( \Pi = \{\pi_1 = 1, ..., \pi_4 = 4\} \), where the parameters correspond to the number of nearest neighbors used by the classifier. This set of nearest neighbors was chosen initially to determine the accuracy of the K-NN. As will be shown, the classification accuracies for this set of parameters is high, therefore no other parameters are tested. It was determined that \( v_{max} = 3 \), therefore \( K_J = \pi_3 = 3 \). The PW classifier requires the window width, \( h \), prior to classification. Define the parameter set for the PW classifier as \( \Pi = \{\pi_1 = 0.1, \pi_2 = 0.3, ..., \pi_{25} = 4.9\} \), where these parameters correspond to the window width used by the classifier. It was determined that \( v_{max} = 10 \), therefore \( h = \pi_{10} = 2.9 \). The SVM classifier requires two inputs \( C \) and \( \gamma \) prior to classification. Note that \( C \in \{c_1 = 1, c_2 = 3, ..., c_{10} = 19\} \) and \( \gamma \in \{\gamma_1 = 0.01, \gamma_2 = 0.03, ..., \gamma_{100} = 1.99\} \). Let the parameter set for the SVM classifier be \( \Pi = \{\pi_1 = \{c_1, \gamma_1\}, ..., \pi_{1000} = \{c_{10}, \gamma_{100}\}\} \). It was determined that \( v_{max} = 201 \), therefore \( C = 5 \) and \( \gamma = 0.01 \).
Table 5.11
Average Classification Accuracies for the Gaussian Chirp Simulations
Using Features Sampled at Regular Intervals.

<table>
<thead>
<tr>
<th></th>
<th>$P_1$</th>
<th>$P_2$</th>
<th>$P_3$</th>
<th>$P_4$</th>
<th>$P_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>100%</td>
<td>99.9%</td>
<td>100%</td>
<td>99.9%</td>
<td>100%</td>
</tr>
<tr>
<td>BTC</td>
<td>99.9%</td>
<td>100%</td>
<td>99.8%</td>
<td>100%</td>
<td>99.7%</td>
</tr>
<tr>
<td>Bayesian$_\mu$</td>
<td>79.5%</td>
<td>100%</td>
<td>79.5%</td>
<td>78.4%</td>
<td>79.3%</td>
</tr>
<tr>
<td>Bayesian$_\Sigma$</td>
<td>100%</td>
<td>100%</td>
<td>99.9%</td>
<td>87.1%</td>
<td>100%</td>
</tr>
<tr>
<td>PW</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>K-NN</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

The average classification accuracy of each classifier for each circuit model is shown in Table 5.11. Recall that $\Omega_{(n,d)}$ denotes the set of test vectors with features that are extracted from the power spectrums in $P_{0_{(n,d)}}$. All classifiers produce high classification accuracies except for the Bayesian$_\mu$ classifier and the Bayesian$_\Sigma$ classifier. Classification errors occur for the Bayesian$_\mu$ classifier using test vectors in $\Omega_{(n=5,1)}$, $\Omega_{(n=5,3)}$, $\Omega_{(n=5,4)}$, and $\Omega_{(n=5,5)}$. Note that $n = 5$, signifies the noise power spectrum $\sigma^2_{c_5}$. The reason for these classification errors is due to the increase in noise power, which is evident by the average SNR values for $\sigma^2_{c_5}$ of Table 5.8. To better clarify these classification errors, consider the power spectrums in Figure 5.9. These power spectrums are examples generated by the circuit models in response to the Gaussian chirp signal using the noise power spectrum $\sigma^2_{c_5}$. As indicated in the Figures, all power spectrums are noisy, which is the cause of the classification errors.

Classification errors also occur for the Bayesian$_\Sigma$ classifier for test vectors in $\Omega_{(5,4)}$. The reason for these errors is due to the increase in noise power, which is evident by the average SNR value for row $\sigma^2_{c_5}$ and column $\Upsilon_4$ of Table 5.8. This is also evident in Figure 5.9(c), which is a power spectrum from $P_{0_{(5,4)}}$. As shown in this Figure, the harmonic distortion is near the noise floor, therefore the test vectors in $\Omega_{(5,4)}$ are being classified as noise.
Fig. 5.9. Example of Power Spectrums Generated by Circuit Models \( \Upsilon_1 \), \( \Upsilon_3 \), \( \Upsilon_4 \), and \( \Upsilon_5 \) in Response to a Gaussian Chirp Signal Using the Noise Power Spectrum \( \sigma^2_{\text{noise}} \).

### 5.3.5 Features Selected at Precise Locations

In this Section, features are selected at precise locations as described in Section 3.2 of Chapter 3. This feature selection technique uses the a priori information of the harmonic distortion and filter response to sample the highest amplitudes available in a given power spectrum. This is done since the highest amplitudes are most resilient to noise. For the given set of power spectrums in Equation 5.41, we will sample the locations of the estimated filter responses generated by \( \Upsilon_1 \) and \( \Upsilon_3 \) in response to the Gaussian chirp signal. We will also sample the peak locations of the
harmonic distortion that is generated by \( \Upsilon_2 \) and \( \Upsilon_4 \) in response to the Gaussian chirp signal. The features are selected as follows. First, examine the filter response of \( \Upsilon_1 \) in Figure 5.10. We will sample three frequencies in the passband of this filter. As is illustrated, the frequencies \( \rho_1 = 410\text{MHz} \), \( \rho_2 = 430\text{MHz} \), and \( \rho_3 = 445\text{MHz} \) are sampled. Next, consider the first harmonic in the power spectrum generated by \( \Upsilon_2 \) as is shown in Figure 5.11. This Figure illustrates that the power spectrum is sampled at the three frequencies corresponding to the peak of the first harmonic. These frequency locations are \( \rho_4 = 598\text{MHz} \), \( \rho_5 = 600\text{MHz} \), and \( \rho_6 = 602\text{MHz} \). The filter response of \( \Upsilon_3 \) is sampled using the same procedure as described for \( \Upsilon_1 \). As shown in Figure 5.12, the filter response of \( \Upsilon_3 \) is sampled at the frequencies \( \rho_7 = 682\text{MHz} \), \( \rho_8 = 690\text{MHz} \), and \( \rho_9 = 698\text{MHz} \). Finally, a close examination of the second and third harmonics in Figure 5.11 illustrates that amplitudes are selected as frequencies \( \rho_{10} = 1198\text{MHz} \), \( \rho_{11} = 1200\text{MHz} \), \( \rho_{12} = 1202\text{MHz} \), \( \rho_{13} = 1798\text{MHz} \), \( \rho_{14} = 1800\text{MHz} \), and \( \rho_{15} = 1802\text{MHz} \). The set of frequencies form the set \( \rho = \{ \rho_1, \ldots, \rho_{15} \} \).

As discussed in Section 3.2 of Chapter 3, the set of frequencies are \( \rho = \{ \rho_1, \ldots, \rho_G \} \). Let \( \rho_g \) denote the \( g \)'th frequency, where \( g \in \{ 1, \ldots, G \} \). The amplitude at each frequency in \( \rho \) is determined by

\[
e_{(k,g)} = P_{r(n,d,l)}(\rho_g)
\]

where, given any values of \( n, d, \) and \( l \), \( k = (n-1)DL + (d-1)L + l, k \in \{ 1, \ldots, K \}, K = 12500 \). The features form the feature vector

\[
\eta_k = [e_{(k,1)} \ldots e_{(k,G)}]
\]

where \( \eta_k \in \mathbb{R}^G \). We can then order the feature vectors and labels as

\[
\{ [\eta_1, \Gamma_1], \ldots, [\eta_K, \Gamma_K] \}
\]

where \( \Gamma_k \) was defined by Equation 5.45. The training and testing sets are then determined using the exact same procedure as described in Section 5.3.4 for features
Fig. 5.10. Estimated Filter Response of the Power Spectrum of by $\gamma_1$ in response to a Gaussian Chirp Signal. Features are Precisely Chosen in the Passband of the Filter Response.
Fig. 5.11. Gaussian Harmonic Distortion of $\Gamma_2$ in response to a Gaussian Chirp Signal. Features are Precisely Chosen at Peaks of Each Harmonic.
Fig. 5.12. Estimated Filter Response of a Power Spectrum of $\Upsilon_3$ in response to a Gaussian Chirp Signal. Features are Precisely Chosen in the Passband of the Filter Response.
Table 5.12
Number of Feature Vectors Per Category for the Training Set. Features are Sampled at Precise Locations.

<table>
<thead>
<tr>
<th>J</th>
<th>Q_1</th>
<th>Q_2</th>
<th>Q_3</th>
<th>Q_4</th>
<th>Q_5</th>
</tr>
</thead>
<tbody>
<tr>
<td>6279</td>
<td>1264</td>
<td>1241</td>
<td>1249</td>
<td>1276</td>
<td>1249</td>
</tr>
</tbody>
</table>

Table 5.13
Number of Feature Vectors Per Category for the Testing Set. Features are Sampled at Precise Locations.

<table>
<thead>
<tr>
<th>I</th>
<th>R_1</th>
<th>R_2</th>
<th>R_3</th>
<th>R_4</th>
<th>R_5</th>
</tr>
</thead>
<tbody>
<tr>
<td>6221</td>
<td>1236</td>
<td>1259</td>
<td>1251</td>
<td>1224</td>
<td>1251</td>
</tr>
</tbody>
</table>

selected at regular intervals. For the simulation discussed in this Section, the training vectors form the set

\[ \Theta = \{ [\theta_1, \tau_1], \ldots, [\theta_J, \tau_J] \} \] (5.55)

and the testing vectors form the set

\[ \Omega = \{ [\omega_1, \lambda_1], \ldots, [\omega_I, \lambda_I] \} \] (5.56)

For the training set defined by Equation 5.55, the number of total observations and the number observations per category are shown in Table 5.12. For the testing set in Equation 5.56, the total number of observations and the number observations per category are shown in Table 5.13.

**Classification Results**

An average classification accuracy is measured for each category and each classifier using the same procedure discussed in Section 5.3.4. Once again, several parameters must be tested for the K-NN classifier, PW classifier, and SVM classifier priori to
classification. The K-NN classifier requires the number of nearest neighbors, $K_J$, prior to classification. Let the parameter set for the K-NN classifier be $\Pi = \{\pi_1 = 1, \ldots, \pi_4 = 4\}$, where the parameters correspond to the number of nearest neighbors used by the classifier. This set of nearest neighbors was chosen initially to determine the accuracy of the K-NN. As will be shown, the classification accuracies for this set of parameters is high, therefore no other parameters are tested. It was determined that $v_{\text{max}} = 1$, therefore $K_J = \pi_1 = 1$. The PW classifier requires the window width, $h$, prior to classification. Define the parameter set for the PW classifier as $\Pi = \{\pi_1 = 0.1, \pi_2 = 0.3, \ldots, \pi_{25} = 4.9\}$, where these parameters correspond to the window width used by the classifier. It was determined that $v_{\text{max}} = 10$, therefore $h = \pi_{10} = 2.9$, which is the same window width used above for features selected at regular intervals described in Section 5.3.4. The SVM classifier requires two inputs $C$ and $\gamma$ prior to classification. Note that $C \in \{c_1 = 1, c_2 = 3, \ldots, c_{10} = 19\}$ and $\gamma \in \{\gamma_1 = 0.01, \gamma_2 = 0.03, \ldots, \gamma_{100} = 1.99\}$. Let the parameter set for the SVM classifier be $\Pi = \{\pi_1 = \{c_1, \gamma_1\}, \ldots, \pi_{1000} = \{c_{10}, \gamma_{100}\}\}$. It was determined that $v_{\text{max}} = 1$, therefore $C = 1$ and $\gamma = 0.01$.

The average classification accuracy of each classifier for each circuit model is shown in Table 5.14. The results are very similar to the results for features selected at regular intervals, as described in Section 5.3.4. The classifications errors for the Bayesian$_\mu$ and Bayesian$_\Sigma$ classifiers in Table 5.14 are caused by the increase in noise, as discussed in Section 5.3.4. However, an average classification accuracy increase is indicated by $P_4$ in Table 5.14 for both the Bayesian$_\mu$ and Bayesian$_\Sigma$ classifiers. The average classification accuracy for the Bayesian$_\mu$ classifier increased from 78.4% to 100%. The average classification accuracy for the Bayesian$_\Sigma$ classifier increased from 87.1% to 92.6%. These results demonstrate that the sampling technique affects the classification accuracy and that features sampled at precise locations increases the classification accuracy.
Table 5.14
Average Classification Accuracies for the Gaussian Chirp Simulations Using Features Sampled at Precise Locations.

<table>
<thead>
<tr>
<th></th>
<th>$P_1$</th>
<th>$P_2$</th>
<th>$P_3$</th>
<th>$P_4$</th>
<th>$P_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>BTC</td>
<td>100%</td>
<td>99.4%</td>
<td>100%</td>
<td>99.5%</td>
<td>99.6%</td>
</tr>
<tr>
<td>Bayesian$_\mu$</td>
<td>79.4%</td>
<td>100%</td>
<td>79.5%</td>
<td>100%</td>
<td>79.9%</td>
</tr>
<tr>
<td>Bayesian$_\Sigma$</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>92.6%</td>
<td>100%</td>
</tr>
<tr>
<td>PW</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>K-NN</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

5.3.6 Conclusions

For the Gaussian chirp simulations, we presented a method to forensically characterize circuit models using features of the harmonic distortion and features of an estimated filter response. It was shown that classification errors occur for features sampled from a power spectrum with low SNR. In addition, we demonstrated that features sampled at precise locations increased the average classification accuracy. The SVM, BT, PW, and K-NN classifiers had near perfect classification accuracies.

5.4 Linear Chirp Probe Signal Simulations and Experiments

In this Section, we present a method to forensically characterize actual RF circuits and circuit models using the filter response of two Chebyshev filters. The filter response is estimated using a linear chirp signal. In Section 5.4.1, two actual RF circuits are considered. A set of linear chirp probe signals are input into the two RF circuits and return signals are measured. Once power spectrums of the return signals are obtained, features are extracted and then classified. In Section 5.4.2, two circuit models are considered to simulate the experiment in Section 5.4.1. For both
the experiment and simulation, we will characterize the circuits (real and simulated) by the estimated filter response.

5.4.1 Linear Chirp Experiments

Two RF circuits are used in the experiments presented in this Section. The RF circuits are known as $\Upsilon = \{\Upsilon_1, \Upsilon_2\}$. A block diagram of these RF circuits are shown in Figure 5.13. The input probe signal $p(t)$ is a linear chirp signal (not windowed) and $r(t)$ is the return signal. Each circuit has a bandpass filter and an amplifier. The same amplifier is used in both RF circuits, however, the filters are different. For both RF circuits, the probe signal is filtered and sent to the amplifier. The amplifier reflects the incident signal, where the reflected signal is passed back through filter and is measured.

![Block Diagram of Two RF Circuits](image)

Fig. 5.13. Block Diagram of the Two RF Circuits Used in the Linear Chirp Experiments.

Three circuit components are considered for the RF circuits shown in Figure 5.13. These components are a Trilithic 4BC500/20-3-KK bandpass filter, a K&L Microwave 5MC10-500/T25-O/O bandpass filter, and a Hittite HMC479ST89 MMIC amplifier. The Trilithic 4BC500/20-3-KK bandpass filter is a 4\textsuperscript{th} order Chebyshev bandpass filter with a center frequency of 500MHz, a passband of 20MHz, and a passband ripple of 0.05dB and is labelled as Filter 1 in the block diagram of Figure 5.13. The K&L Microwave 5MC10-500/T25-O/O bandpass filter is a 5\textsuperscript{th} order Chebyshev filter...
with a center frequency of 500MHz, a passband of 25MHz, and a passband ripple of 0.05dB and is labelled as Filter 2 in the block diagram of Figure 5.13.

The return signal $r(t)$ in Figure 5.13 is measured using the system illustrated in Figure 5.14. This system uses a waveform generator to produce a linear chirp signal. The RF circuit is connected to the waveform generator by a circulator. Note that the probe signal is not transmitted through the environment to the circuit. The circulator is used to connect the probe signal to the RF circuit and connect the return signal to a spectrum analyzer. The spectrum analyzer measures the response from the RF circuit.

**Probe Signal Design**

The probe signal used by the waveform generator of Figure 5.14 is a combination of multiple signals

$$
p(t) = \begin{cases} 
0 & , & 0 \leq t < 50\text{ns} \\
A \cos(2\pi f_0 t) & , & 50\text{ns} \leq t < 550\text{ns} \\
A \cos(2\pi (f_1 + \frac{(f_2-f_0)}{2t_1}) t) & , & 550\text{ns} \leq t < 1550\text{ns} \\
A \cos(2\pi f_0 t) & , & 1550\text{ns} \leq t < 2050\text{ns} 
\end{cases} \quad (5.57)
$$

\(^5\)All measurements were taken at North Carolina State University.
where $A$ is the amplitude, $f_0 = 495\text{MHz}$, $f_1 = 505\text{MHz}$, and $t_1 = 1550\text{ nanoseconds (ns)}$. The first signal is a cosine function and is used before the linear chirp signal to eliminate any transients. The cosine function is active for 500ns. At 550ns, the cosine function is deactivated and the linear chirp signal is activated. The linear chirp signal is designed to probe the frequency band 495MHz - 505MHz, which is within the passband of both filters. After 1000ns seconds, the linear chirp signal is deactivated and the cosine function is activated for 500ns. An example of this response is shown in Figure 5.15.

The waveform generator produces three sets of probe signals at different amplitudes $A$. The first set of probe signals is $P_1 = \{p_{(1,1)}(t), \ldots p_{(1,L)}(t)\}$, where $p_{(1,l)}(t) = p(t)$, $A=0.316$ volts, $l \in \{1, \ldots L\}$, and $L = 100$. The second set of probe signals is $P_2 = \{p_{(2,1)}(t), \ldots p_{(2,L)}(t)\}$, where $p_{(2,l)}(t) = p(t)$, and $A=0.562$ volts. The third set of probe signals is $P_3 = \{p_{(3,1)}(t), \ldots p_{(3,L)}(t)\}$, where $p_{(3,l)}(t) = p(t)$, and $A=1.0$ volts.
Generation of the Return Signals

The probe signals are input into each RF circuit and the responses are measured by the spectrum analyzer of Figure 5.14. The set of probe signals $P_1$ is input into both RF circuits. The set of probe signals $P_2$ is input into circuit $\Upsilon_1$. The set of probe signals $P_3$ is input into circuit $\Upsilon_2$. The return signal is $r_{(n,d,l)}(t)$, where $n \in \{1, 2, 3\}$ denotes the set $P_n$, $d \in \{1, 2\}$ denotes the RF circuit $\Upsilon_d$, and $l \in \{1, \ldots, L\}$ denotes the probe signal $p_l(t) \in P_n$. Each return signal will have a label (ground truth) associated with it, which is used to denote the RF circuit that generated the return signal. Each return signal and label pair are known as an observation. The set of observations $^6$ is

$$\begin{align*}
\{[r_{(n,d,1)}(t), \gamma_{(n,d,1)} = d], \ldots,[r_{(n,d,L)}(t), \gamma_{(n,d,L)} = d]\} 
\end{align*}$$

(5.58)

where $\gamma_{(n,d,l)} = d$ is the ground truth label.

The linear chirp signal is used to estimate the filter response. The linear chirp signal is active in the probe $p(t)$ of Equation 5.57 between 550ns - 1550ns. We will therefore extract a section of the return signal $r_{(n,d,l)}(t)$ between 550ns - 1550ns, which is done to study the filter response. Let the offset time be $T = 550$ns. Each return signal $r_{(n,d,l)}(t)$ is redefined as

$$
s_{(n,d,l)}(t) = \begin{cases} 
  r_{(n,d,l)}(t + T), & 0ns \leq t \leq 1000ns \\
  0, & \text{else}
\end{cases}
$$

(5.59)

The observations from Equation 5.58 are then

$$O_{(n,d)} = \{[s_{(n,d,1)}(t), \gamma_{(n,d,1)} = d], \ldots,[s_{(n,d,L)}(t), \gamma_{(n,d,L)} = d]\}$$

(5.60)

The set of observations produced by all RF circuits is

$$O = \{O_{(1,1)}, O_{(1,2)}, O_{(2,1)}, O_{(3,2)}\}$$

(5.61)

$^6$This set of observations were measured at North Carolina State University.
Note that the subscript notation in Equation 5.61 is not ordered. The notation is only used to indicate the RF circuit \( \Upsilon_d \), and the set of probe signals \( P_n \) that are input into \( \Upsilon_d \). The total number of signals in Equation 5.61 is 400. The power spectrums of the signals in the set \( O_{(n,d)} \) are

\[
P_{O_{(n,d)}} = \{ [P_{s_{(n,d,1)}}(f), \gamma(n,d,1) = d], \ldots [P_{s_{(n,d,L)}}(f), \gamma(n,d,L) = d] \} \tag{5.62}
\]

where

\[
P_{s_{(n,d,l)}}(f) = S_{(n,d,l)}(f)[S_{(n,d,l)}(f)]^* \tag{5.63}
\]

and

\[
S_{(n,d,l)}(f) = \int_{-\infty}^{\infty} s_{(n,d,l)}(t)e^{-i2\pi ft} dt \tag{5.64}
\]

The power spectrums form the set

\[
P_O = \{ P_{O_{(1,1)}}, P_{O_{(1,2)}}, P_{O_{(2,1)}}, P_{O_{(3,2)}} \} \tag{5.65}
\]

**Features Selection**

Features are extracted from each power spectrum in \( P_O \) of Equation 5.65. The method used for feature extraction was discussed in Section 3.3 of Chapter 3. As discussed, the set of frequencies corresponding to the features are \( \rho_g = \rho_{g-1} + \Delta \), for \( 2 \leq g \leq G \). \( \Delta \) is a constant offset value. Let \( \rho_1 = 495\text{MHz} \), \( G = 10 \), and \( \Delta = 1\text{MHz} \). Features are therefore selected between 495MHz-505MHz in increments of 1MHz. We will describe the feature extraction process for each set \( P_{O_{(n,d)}} \) of power spectrums separately. The features extracted from the set of power spectrums \( P_{O_{(1,1)}} \) are

\[
\varrho(k,g) = P_{s_{(1,1,l)}}(\rho_g) \tag{5.66}
\]
where \( k = l, k \in \{1, \ldots, 100\} \), and let the ground truth label be \( \Gamma_k = \gamma_{(1,1,l)} \). The features extracted from the set of power spectrums \( P_{O(1,2)} \) are

\[
\varrho(k,g) = P_{s(1,2,l)}(\rho_g)
\]

(5.67)

where \( k = 100 + l, k \in \{101, \ldots, 200\} \), and let \( \Gamma_k = \gamma_{(1,2,l)} \). The features extracted from the set of power spectrums \( P_{O(2,1)} \) are

\[
\varrho(k,g) = P_{s(2,1,l)}(\rho_g)
\]

(5.68)

where \( k = 200 + l, k \in \{201, \ldots, 300\} \), and let \( \Gamma_k = \gamma_{(2,1,l)} \). The features extracted from the set of power spectrums \( P_{O(3,2)} \) are

\[
\varrho(k,g) = P_{s(3,2,l)}(\rho_g)
\]

(5.69)

where \( k = 300 + l, k \in \{301, \ldots, 400\} \), and let \( \Gamma_k = \gamma_{(3,2,l)} \). The features for a given \( k \) form the set \( \varrho_k = \{\varrho_{(k,1)}, \ldots, \varrho_{(k,G)}\} \). The samples are then normalized from 0 to 1. Scaling the features maintains the “pattern” of the passband ripple in the filter response. Each feature \( \varrho_{(k,g)} \) is scaled as follows

\[
e_{(k,g)} = \frac{\varrho_{(k,g)} - \min(\varrho_k)}{\max(\varrho_k) - \min(\varrho_k)}
\]

(5.70)

where the function \( \min(\varrho_k) \) is the minimum value of \( \varrho_k \) and the function \( \max(\varrho_k) \) is the maximum value of \( \varrho_k \). The features form the feature vector

\[
\eta_k = [e_{(k,1)} \ldots e_{(k,G)}]
\]

(5.71)

where \( \eta_k \in \mathbb{R}^G \). We can then order the feature vectors and labels as

\[
\{[\eta_1, \Gamma_1], \ldots, [\eta_K, \Gamma_K]\}
\]

(5.72)

where \([\eta_k, \Gamma_k]\) is known as the \( k^{th} \) observation and \( K = 400 \). Each observation in Equation 5.72 is randomly divided into the training and testing sets. The training and testing sets were introduced in Chapter 4. Before dividing the observations in
Equation 5.72, first let $\nu(k)$ be a uniform random variable on the interval $[0, 1]$ indexed by the integer $k$. For the $k^{th}$ observation in Equation 5.72, we let

\[
\begin{align*}
\{\theta_j = \eta_k, \tau_j = \Gamma_k\} & , \quad \nu(k) \geq 0.5 \\
\{\omega_i = \eta_k, \lambda_i = \Gamma_k\} & , \quad \nu(k) < 0.5
\end{align*}
\] (5.73)

where $\theta_j$ is the $j^{th}$ training vector with a label $\tau_j$, and $\omega_i$ is the $i^{th}$ test vector with a label $\lambda_i$. The training vectors form the set

\[
\Theta = \{[\theta_1, \tau_1], \ldots, [\theta_J, \tau_J]\}
\] (5.74)

Note that since we are randomly dividing the observations in Equation 5.72, $J$ is a discrete random variable where $J \in \{0, 1, \ldots, K\}$. The test vectors form the set

\[
\Omega = \{[\omega_1, \lambda_1], \ldots, [\omega_I, \lambda_I]\}
\] (5.75)

Also note that $I$ is a discrete random variable where $I \in \{0, 1, \ldots, K\}$ and $K = J + I$.

For the training set described by Equation 5.74, the number of total observations and the number observations per category are shown in Table 5.15, where $Q_d \in J$ was defined in Chapter 4 as the number of observations that have the label $\tau_j = d$. For the test set in Equation 5.75, the total number of observations and the number observations per category are shown in Table 5.16, where $R_d \in I$ was described in Chapter 4 as the number of observations that have the label $\lambda_i = d$.

\footnote{These simulations used the \texttt{MATLAB} function \texttt{rand()} to generate a pseudorandom number drawn from the uniform distribution.}

Table 5.15
Number of Feature Vectors Per Category for the Training Set. Features are Used in the Linear Chirp Experiments.

<table>
<thead>
<tr>
<th>$J$</th>
<th>$Q_1$</th>
<th>$Q_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>211</td>
<td>103</td>
<td>108</td>
</tr>
</tbody>
</table>
Table 5.16
Number of Feature Vectors Per Category for the Testing Set. Features are Used in the Linear Chirp Experiments.

<table>
<thead>
<tr>
<th></th>
<th>R_1</th>
<th>R_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>189</td>
<td>97</td>
</tr>
<tr>
<td>2</td>
<td>92</td>
<td></td>
</tr>
</tbody>
</table>
Experimental Results

We will find the average classification accuracy for each RF circuit in $\Upsilon$. As discussed in Chapter 4, this accuracy is

$$P_d = \frac{\sum_{i=1}^{I} I_f(i)}{R_d}$$

(5.76)

where

$$I_f(i) = \begin{cases} 1, & \lambda_i = \chi_i \\ 0, & \text{else} \end{cases}$$

(5.77)

is the indicator function and $\chi_i$ is the predicted label decided by the classifier. The K nearest neighbor (K-NN) classifier, Parzen window (PW) classifier, and support vector machine (SVM) require input parameters priori to classification. As discussed in Section 5.2.5, we will define a set of parameters $\Pi = \{\pi_1, \ldots, \pi_V\}$ for each classifier to determine the highest classification accuracy. The K-NN classifier requires the number of nearest neighbors, $K_J$, prior to classification. Let the parameter set for the K-NN classifier be $\Pi = \{\pi_1 = 1, \ldots, \pi_{99} = 99\}$, where the parameters correspond to the number of nearest neighbors used by the classifier. It was determined that all parameters obtained perfect classification accuracy. We will therefore use $v_{max} = 1$ which yields $K_J = \pi_1 = 1$. The PW classifier requires the window width, $h$, prior to classification. Define the parameter set for the PW classifier as $\Pi = \{\pi_1 = 0.1, \pi_2 = 0.2, \ldots, \pi_{50} = 5.0\}$, where these parameters correspond to the window width used by the classifier. It was determined that all parameters obtained perfect classification accuracy. We will therefore use $v_{max} = 1$ which yields $h = \pi_1 = 0.1$. The SVM classifier requires two inputs $C$ and $\gamma$ prior to classification. Note that $C \in \{c_1 = 1, c_2 = 2, \ldots, c_{19} = 19\}$ and $\gamma \in \{\gamma_1 = 0.1, \gamma_2 = 0.2, \ldots, \gamma_{19} = 1.9\}$. Let the parameter set for the SVM classifier be $\Pi = \{\pi_1 = \{c_1, \gamma_1\}, \ldots, \pi_{361} = \{c_{19}, \gamma_{19}\}\}$. It was determined that all parameters obtained perfect classification accuracy. We will therefore use $v_{max} = 1$ which yields $C = 1$ and $\gamma = 0.1$. 
Table 5.17
Classification Results for the Linear Chirp Experiment.

<table>
<thead>
<tr>
<th></th>
<th>$P_1$</th>
<th>$P_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>BT</td>
<td>100%</td>
<td>98%</td>
</tr>
<tr>
<td>Bayesian$_\mu$</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Bayesian$_\Sigma$</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>PW</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>K-NN</td>
<td>100%</td>
<td>100%</td>
</tr>
</tbody>
</table>

The average classification accuracy of each classifier for each RF circuit is shown in Table 5.17. All classifiers obtained near perfect classification accuracies. These results indicate that the ripple in the passband of the filter response is distinct. This fact is illustrated in Figure 5.16 where an example power spectrum is shown for $\Upsilon_1$ and $\Upsilon_2$. A distinct ripple pattern is present in the passband. The simulations presented in the next Section are used to verify these results.

5.4.2 Linear Chirp Simulations

Our goal is to use our simulation techniques to attempt to verify the experimental results we described above. For the simulations, two circuit models are considered. Each circuit model is shown in Figure 5.17. $p(t)$ is the input probe signal and $d(t)$ is the output from the second filter that is added to the noise $n_c(t)$. $r(t)$ is the return signal. The noise $n_c(t)$ is modeled as a white noise random process with a power spectrum, $\sigma_c^2$. The nonlinearity in both circuit models is modeled by a power series of order $M = 3$ as defined by Equation 5.1. The power series coefficients are adjusted to obtain a filter response at the same power level as the experiments as shown in Figure 5.16(a). The coefficients are defined as $a_1 = a_2 = a_3 = 0.6$. The filters in each circuit model, which are identical, are designed to simulate the filters of
Fig. 5.16. Power Spectrums for the Linear Chirp Experiments. As is Illustrated, the Passband Ripple is Distinct.
the experiments. Therefore $\Upsilon_1$ uses two $4^{th}$ order Chebyshev bandpass filters with a center frequency of 500MHz, a passband of 20MHz, and a passband ripple of 0.05dB. $\Upsilon_2$ uses two $5^{th}$ order Chebyshev bandpass filters with a center frequency of 500MHz, a passband of 25MHz, and a passband ripple of 0.05dB. The noise power spectrum $\sigma_n^2$ is chosen to generate noise similar to the noise from the experiments. An example power spectrum from $P_{O(n,d)}$, as described by Equation 5.62, is shown in Figure 5.18 with the noise floor at -45dB. To simulate this noise, the power spectrum of $n_c(t)$ is $\sigma_n^2 = 3 \times 10^{-5}$.

The probe signal used to probe $\Upsilon_1$ and $\Upsilon_2$ for the simulations is a linear chirp probe signal

$$p(t) = Acos(2\pi(f_0 + \frac{(f_1 - f_0)}{2t_1}t)t) \quad (5.78)$$

where $A$ is the amplitude, $f_0=495$MHz is the start frequency at $t = 0$, $f_1=505$MHz is the end frequency at $t_1 = 1000ns$. Since transients are not a concern here, only the linear chirp signal is used to probe each circuit model.

Fig. 5.17. Circuit Models Used in the Simulations.
Fig. 5.18. Example Power Spectrum from the Linear Chirp Experiments. The noise is between -55dB and -45dB.
Three sets of probe signals are designed using \( p(t) \) in Equation 5.78. Each set will use a different amplitude \( A \). The first set of probe signals is \( P_1 = \{ p_{(1,1)}(t), \ldots, p_{(1,L)}(t) \} \), where \( p_{(1,l)}(t) = p(t) \), \( A = 0.316 \) volts, \( l \in \{1, \ldots, L\} \), and \( L = 100 \). The second set of probe signals is \( P_2 = \{ p_{(2,1)}(t), \ldots, p_{(2,L)}(t) \} \), where \( p_{(2,l)}(t) = p(t) \), and \( A = 0.562 \) volts. The third set of probe signals is \( P_3 = \{ p_{(3,1)}(t), \ldots, p_{(3,L)}(t) \} \), where \( p_{(3,l)}(t) = p(t) \), and \( A = 1.0 \) volts.

**Generation of the Return Signals**

The return signals for the simulations are created by a similar procedure as used in the experiments, except that the probe signals are input into the circuit models instead of physical RF circuits. The probe signals in \( P_1 \) are input into both circuit models \( \Upsilon_1 \) and \( \Upsilon_2 \). The probe signals in \( P_2 \) are input into \( \Upsilon_1 \). The probe signals in \( P_3 \) are input into \( \Upsilon_2 \). A given return signal is known as \( r_{(n,d,l)}(t) \), where \( n \in \{1, 2, 3\} \) denotes the set \( P_n \), \( d \in \{1, 2\} \) denotes the RF circuit \( \Upsilon_d \), and \( l \in \{1, \ldots, L\} \) denotes the probe signal \( p_{l}(t) \in P_n \). Each return signal will have a label (ground truth) associated with it, which is used to denote the circuit model and the return signal. Each return signal and label pair are known as an observation. The set of observations is defined as

\[
O_{(n,d)} = \{ [r_{(n,d,1)}(t), \gamma_{(n,d,1)} = d], \ldots, [r_{(n,d,L)}(t), \gamma_{(n,d,L)} = d] \} \tag{5.79}
\]

where \( \gamma_{(n,d,l)} = d \) is the ground truth label. The set of observations produced by both circuit models is

\[
O = \{ O_{(1,1)}, O_{(1,2)}, O_{(2,1)}, O_{(3,2)} \} \tag{5.80}
\]

Note that the total number of signals in Equation 5.80 is 400. The power spectrums of the signals in the set \( O_{(n,d)} \) are

\[
P_{O_{(n,d)}} = \{ [P_{r_{(n,d,1)}}(f), \gamma_{(n,d,1)} = d], \ldots, [P_{r_{(n,d,L)}}(f), \gamma_{(n,d,L)} = d] \} \tag{5.81}
\]
where

\[ P_{r_{\{n, d, l\}}} (f) = R_{\{n, d, l\}} (f) [R_{\{n, d, l\}} (f)]^* \] (5.82)

and

\[ R_{\{n, d, l\}} (f) = \int_{-\infty}^{\infty} r_{\{n, d, l\}} (t) e^{-i2\pi ft} dt \] (5.83)

The power spectrums form the set

\[ P_O = \{ P_O_{(1, 1)}, P_O_{(1, 2)}, P_O_{(2, 1)}, P_O_{(3, 2)} \} \] (5.84)

**Features Selection**

Features are extracted from each power spectrum in Equation 5.84 using the exact same procedure described in Section 5.4.1 for the experiments. A set of training vectors are formed by

\[ \Theta = \{ [\theta_1, \tau_1], ..., [\theta_J, \tau_J] \} \] (5.85)

and a set of test vectors are formed by

\[ \Omega = \{ [\omega_1, \lambda_1], ..., [\omega_I, \lambda_I] \} \] (5.86)

For the training set in Equation 5.85, the number of total observations and the number observations per category are shown in Table 5.18. For the test set in Equation 5.86, the total number of observations and the number observations per category are shown in Table 5.19.

**Simulation Results**

An average classification accuracy is measured for each category and each classifier using the exact same procedure discussed in Section 5.4.1. Once again, several
Table 5.18
Number of Feature Vectors Per Category for the Training Set. Features Vectors are Used in the Linear Chirp Simulations.

<table>
<thead>
<tr>
<th>J</th>
<th>Q₁</th>
<th>Q₂</th>
</tr>
</thead>
<tbody>
<tr>
<td>199</td>
<td>95</td>
<td>104</td>
</tr>
</tbody>
</table>

Table 5.19
Number of Feature Vectors Per Category for the Testing Set. Features Vectors are Used in the Linear Chirp Simulations.

<table>
<thead>
<tr>
<th>I</th>
<th>R₁</th>
<th>R₂</th>
</tr>
</thead>
<tbody>
<tr>
<td>201</td>
<td>105</td>
<td>96</td>
</tr>
</tbody>
</table>
parameters must be tested for the K-NN classifier, PW classifier, and SVM classifier priori to classification. The K-NN classifier requires the number of nearest neighbors, $K_j$, prior to classification. Let the parameter set for the K-NN classifier be $\Pi = \{\pi_1 = 1, \ldots, \pi_{99} = 99\}$, where the parameters correspond to the number of nearest neighbors used by the classifier. It was determined that $v_{\text{max}} = 50$, therefore $K_j = \pi_{50} = 50$. The PW classifier requires the window width, $h$, prior to classification. Let the parameter set for the PW classifier be $\Pi = \{\pi_1 = 0.1, \pi_2 = 0.2, \ldots, \pi_{50} = 5.0\}$, where these parameters correspond to the window width used by the classifier. It was determined that $v_{\text{max}} = 2$, therefore $h = \pi_2 = 0.2$. The SVM classifier requires two inputs $C$ and $\gamma$ prior to classification. Note that $C \in \{c_1 = 1, c_2 = 2, \ldots, c_{19} = 19\}$ and $\gamma \in \{\gamma_1 = 0.1, \gamma_2 = 0.2, \ldots, \gamma_{19} = 1.9\}$. Let the parameter set for the SVM classifier be $\Pi = \{\pi_1 = \{c_1, \gamma_1\}, \ldots, \pi_{361} = \{c_{19}, \gamma_{19}\}\}$. It was determined that $v_{\text{max}} = 109$, therefore $C = 6$ and $\gamma = 1.4$.

The average classification accuracy of each classifier for each circuit model is shown in Table 5.20. As shown, a large number of classification errors occur. These results indicate that the ripple in the passband of the filter response is not distinct. This fact is illustrated using Figure 5.19 where an example power spectrum is shown for $\Upsilon_1$ and $\Upsilon_2$, the passband ripple pattern is not distinct.

These results are disappointing in that they are different from those of the experiments discussed in Section 5.4.1. For the experiments, the ripple patterns were distinct thereby leading to high classification accuracies. This suggests that the distinct ripple patterns produced by the Chebyshev filters in the RF circuits from the experiments could be generated by another source. For example, the ripple patterns could be caused by nonlinearities of the circulator used in the measurement system of Figure 5.14. These results indicate that our simulation approach needs further investigation to validate it against actual measurement data.
Fig. 5.19. Power Spectrums for the Linear Chirp Simulations. The Passband Ripple is Not Distinct.
Table 5.20
Classification Results for the Linear Chirp Simulation.

<table>
<thead>
<tr>
<th></th>
<th>$P_1$</th>
<th>$P_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>54.3%</td>
<td>59.4%</td>
</tr>
<tr>
<td>BT</td>
<td>48.6%</td>
<td>47.9%</td>
</tr>
<tr>
<td>Bayesian$_\mu$</td>
<td>3.8%</td>
<td>100%</td>
</tr>
<tr>
<td>Bayesian$_\Sigma$</td>
<td>51.4%</td>
<td>57.3%</td>
</tr>
<tr>
<td>PW</td>
<td>56.2%</td>
<td>54.2%</td>
</tr>
<tr>
<td>K-NN</td>
<td>46%</td>
<td>73%</td>
</tr>
</tbody>
</table>

5.4.3 Conclusions

The results of the experiment indicate that the ripple patterns in the passband of the estimated filter response for each Chebyshev filter are distinct. This distinct pattern can be used to characterize each real RF circuit. However, these results were not verified by the simulations. The distinct ripple patterns produced by the Chebyshev filters in the RF circuits could be generated by another source. Further experimentation is required to determine if the ripple patterns, or other features of the estimated filter response, can be used to characterize the RF filter. These results indicate that our simulation approach needs further investigation to validate it against actual measurement data.

For example, consider the passband of the RF filters used in the experiment. The passband of Filter 1 of Figure 5.13 from Section 5.4.1 is defined to be 487.5MHz - 512.5MHz. Also, the passband of Filter 2 of Figure 5.13 is defined to be 490MHz - 510MHz. Remember that the linear chirp probe signal was defined using $f_0 = 495MHz$ - $f_1 = 505MHz$, which does not cover the entire passband of either filter. Therefore, it may be possible that if the linear chirp probe signal was defined over the entire passband of both filters, for example $f_0 = 485MHz$ - $f_1 = 515MHz$, then other distinct information may be available.
5.5 Summary of Simulations

Three simulations and one experiment were presented in this Chapter. In the first simulation presented in Section 5.2, we developed a method to forensically characterize RF circuit models using first order distortion (FOD) and intermodulation distortion (IMD) caused by a nonlinearity in response to a two-tone probe signal. Classification errors occurred when the return signal is close to the noise floor. In the second simulation presented in Section 5.3, we developed a method to forensically characterize RF circuit models using harmonic distortion and using the estimated filter response. Again, classification errors occurred when the return signal is close to the noise floor. In Section 5.4, we developed a method to forensically characterize physical RF circuits and circuit models using the filter response of two Chebyshev filters. The results indicated that further experimentation is needed to determine if the ripple patterns can be used to characterize the filter responses.

Although the linear chirp simulations/experiments of Section 5.4 demonstrated that the experimental results were inconclusive and further study is needed, there is no evidence to suggest that the results for the two-tone simulations of Section 5.2 and the Gaussian chirp simulations of Section 5.3 are also inconclusive. For the two-tone simulations, the circuit models were designed to simulate experimental data that was described in Section 1.3 of Chapter 1. The features used in the Gaussian chirp simulations are a mix of features taken from the harmonic distortion and the estimated filter responses. The results of the Gaussian chirp simulations demonstrated a high classification accuracy based on using these mixed features. It is therefore possible that these features can be used to accurately characterize physical RF circuits.

Of the three simulations presented in this Chapter, the Gaussian chirp simulations demonstrated that circuit models can be characterized based on features sampled from harmonic distortion produced by nonlinearities and based on features from the estimated filter response. We believe that the Gaussian chirp probe signal is the most effective probe signal of the three that were examined in this dissertation. Two
feature extraction techniques were implemented in the Gaussian chirp simulations. It was shown in Section 5.3.5 that features selected at precise locations produced higher classification accuracies than features selected at regular intervals. We therefore conclude that smaller sized feature vectors that contain feature patterns selected at precise locations produce a higher classification accuracy than larger sized feature vectors sampled at regular intervals.
6. CONCLUSIONS AND FUTURE WORK

In this dissertation we have proposed a forensic fingerprinting approach for RF device. This approach can theoretically be used to characterize any RF device. In this Chapter, we will briefly summarize the main contributions of this dissertation and discuss possible future developments.

6.1 Contributions

The contributions of this dissertation are as follows:

- Mathematical analysis of the output of a nonlinear system when a Gaussian chirp signal is its input. The power spectral density is derived in closed form. This analysis was used in the simulations.

- Developed methods for forensically characterizing circuit models using intermodulation distortion generated by nonlinearities in response to a two-tone probe signal. Classification errors were examined when the return signal is close to the noise floor. These simulations were presented in Section 5.2 of Chapter 5.

- Developed methods for forensically characterizing circuit models using harmonic distortion generated by nonlinearities in response to a Gaussian chirp signal. Classification errors were examined when the return signal is close to the noise floor. These simulations were presented in Section 5.3 of Chapter 5.

- Developed methods for forensically characterizing circuit models using the filter response of two Chebyshev filters. The filter response was estimated by the Gaussian probe signal. Both experimental and simulation results were com-
pared. These results indicated that the distinct oscillating patterns in the passband of the “real” RF circuits may be caused by other sources.

- Comparison of classification accuracy of the pattern recognition methods.

6.2 Future Work

The following issues need to be further studied:

- Our RF model is relatively simple and needs to be validated against real observations.

- The methods we developed are based on simulations. How these techniques can be used to characterize actual RF devices needs to be studied.

- The generation of arbitrary RF probe signals, in particular chirped waveforms, is a difficult problem and also needs further study.

- How well our approach scales relative to a large number of RF circuits has not been determined.

- We did not investigate the computational complexity of our methods. How these techniques would be deployed in an actual system needs further investigation.

- We did not investigate characterizing circuit models with filters using intermodulation distortion generated by nonlinearities in response to a two-tone probe signal. When the filters are used in the circuit models, a priori information of the filter response is needed. Assuming that we have a priori information of the filter response, we can design the probe frequencies of the two-tone signal to be within the passband of the filter. Therefore the return signal, generated by a circuit model using a bandpass filter, in response to a two-tone probe signal with probe frequencies within the passband of this filter, would contain first order distortion and intermodulation distortion that can be used to characterize the circuit model.
Additional chirp probe signals should be considered when characterizing RF devices. One example of an additional chirp signal is the exponential chirp signal [111], where its instantaneous frequency increases exponentially over time. Furthermore, different window functions, other than the Gaussian window, should be considered for the linear chirp signal. One popular window function is the rectangular window [80], which has been used with the linear chirp signal for several applications [48] [112]. Other window functions include Hanning and Hamming [8].

A possible alternative RF device identification technique is synthetic aperture radar (SAR) imaging [113] [114]. With SAR a “radar” image of a target area is obtained. To construct a SAR image, several probe signals are first transmitted to the target area. The probe signals are scattered from the target area and are received by the SAR system. The received signals are used to form an image of the target area. Providing that an RF device is in the target area, a signature would be present in the SAR image and features can therefore be extracted based on this signature. The features from the SAR image can then be combined with features generated from a feature extraction method discussed in Chapter 3 of this dissertation. The “fused” features can then be used to characterize the RF device.

6.3 Publications

Journal Papers:


Conference Papers:


LIST OF REFERENCES
LIST OF REFERENCES


APPENDICES
A: Two-Tone Signal in the Time Domain

The derivation of the two-tone signal in the time domain is presented in this Section. The two-tone signal is defined as

\[ x(t) = A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t) \]  \hspace{1cm} (A.1)

This signal is input into a power series of order M=3

\[ y(t) = \sum_{j=1}^{3} a_j [A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)]^j \]  \hspace{1cm} (A.2)

Equation A.2 is expanded as follows

\[
y(t) = \sum_{j=1}^{3} a_j [A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)]^j \\
= a_1 [A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)] + a_2 [A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)]^2 \\
+ a_3 [A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)]^3 \\
= a_1 A_1 \cos(2\pi f_1 t) + a_1 A_2 \cos(2\pi f_2 t) + a_2 (A_1)^2 \cos(2\pi f_1 t) \\
+ a_2 2A_1 A_2 \cos(2\pi f_1 t) \cos(2\pi f_2 t) + a_2 (A_2)^2 \cos(2\pi f_2 t) \\
+ a_3 (A_1)^3 \cos(2\pi f_1 t)^3 + 3a_3 (A_1)^2 A_2 \cos(2\pi f_1 t)^2 \cos(2\pi f_2 t) \\
+ 3a_3 A_1 (A_2)^2 \cos(2\pi f_1 t) \cos(2\pi f_2 t)^2 + a_3 (A_2)^3 \cos(2\pi f_2 t)^3
\]

Two trigonometry identities are used to expand this development further. These identities are \[115\]

\[ \cos^2(a) = \frac{1}{2} (1 + \cos(2a)) \]  \hspace{1cm} (A.3)

\[ \cos(a)\cos(b) = \frac{1}{2} \cos(a + b) + \frac{1}{2} \cos(a - b) \]  \hspace{1cm} (A.4)
Applying these trigonometry identities produces the following

\[ y(t) = a_1 A_1 \cos(2\pi f_1 t) + a_1 A_2 \cos(2\pi f_2 t) + \frac{a_2(A_1)^2}{2} [1 + \cos(2\pi [2f_1] t)] \\
+ a_2 A_1 A_2 \cos(2\pi [f_1 + f_2] t) + a_2 A_1 A_2 \cos(2\pi [f_2 - f_1] t) \\
+ \frac{a_2(A_2)^2}{2} [1 + \cos(2\pi [2f_2] t)] + a_3(A_1)^3 \cos(2\pi f_1 t)[1 + \cos(2\pi [2f_1] t)] \\
+ \frac{3a_3(A_1)^2 A_2}{2} \cos(2\pi f_2 t)[1 + \cos(2\pi [2f_1] t)] \\
+ \frac{3a_3 A_1 (A_2)^2}{2} \cos(2\pi f_1 t)[1 + \cos(2\pi [2f_2] t)] \\
+ \frac{a_3(A_2)^3}{2} \cos(2\pi f_2 t)[1 + \cos(2\pi [2f_2] t)] \\
= a_1 A_1 \cos(2\pi f_1 t) + a_1 A_2 \cos(2\pi f_2 t) + \frac{a_2(A_1)^2}{2} [1 + \cos(2\pi [2f_1] t)] \\
+ a_2 A_1 A_2 \cos(2\pi [f_1 + f_2] t) + a_2 A_1 A_2 \cos(2\pi [f_2 - f_1] t) \\
+ \frac{a_2(A_2)^2}{2} [1 + \cos(2\pi [2f_2] t)] + \frac{a_3(A_1)^3}{2} \cos(2\pi f_1 t) \\
+ \frac{3a_3(A_1)^2 A_2}{4} [\cos(2\pi [3f_1] t) + \cos(2\pi [f_1] t)] + \frac{3a_3(A_1)^2 A_2}{2} \cos(2\pi f_2 t) \\
+ \frac{3a_3 A_1 (A_2)^2}{4} [\cos(2\pi [2f_1 + f_2] t) + \cos(2\pi [2f_1 - f_2] t)] \\
+ \frac{3a_3 A_1 (A_2)^2}{2} \cos(2\pi f_1 t) + \frac{3a_3 A_1 (A_2)^2}{4} [\cos(2\pi [2f_2 + f_1] t) \\
+ \cos(2\pi [2f_2 - f_1] t)] + \frac{a_3(A_2)^3}{2} \cos(2\pi f_2 t) \\
+ \frac{a_3(A_2)^3}{4} [\cos(2\pi [3f_2] t) + \cos(2\pi [f_2] t)] \\
\]

Further reduction produces the following
\[
y(t) = a_1 A_1 \cos(2\pi f_1 t) + a_1 A_2 \cos(2\pi f_2 t) + \frac{a_2(A_1)^2}{2} + \frac{a_2(A_1)^2}{2} \cos(2\pi[2f_1]t)
\]
\[
+ a_2 A_1 A_2 \cos(2\pi[f_1 + f_2]t) + a_2 A_1 A_2 \cos(2\pi[f_2 - f_1]t) + \frac{a_2(A_2)^2}{2}
\]
\[
+ \frac{a_2(A_2)^2}{2} \cos(2\pi[2f_2]t) + \frac{a_3(A_1)^3}{2} \cos(2\pi f_1 t) + \frac{a_3(A_1)^3}{2} \cos(2\pi[3f_1]t)
\]
\[
+ \frac{a_3(A_1)^3}{4} \cos(2\pi f_1 t) + \frac{3a_3(A_1)^2 A_2}{2} \cos(2\pi f_2 t)
\]
\[
+ \frac{3a_3(A_1)^2 A_2}{4} \cos(2\pi[2f_1 + f_2]t) + \frac{3a_3(A_1)^2 A_2}{4} \cos(2\pi[2f_1 - f_2]t)
\]
\[
+ \frac{3a_3 A_1(A_2)^2}{2} \cos(2\pi f_1 t) + \frac{3a_3 A_1(A_2)^2}{2} \cos(2\pi[2f_2 + f_1]t)
\]
\[
+ \frac{3a_3 A_1(A_2)^2}{4} \cos(2\pi[2f_2 - f_1]t) + \frac{a_3(A_2)^3}{2} \cos(2\pi f_2 t)
\]
\[
+ \frac{a_3(A_2)^3}{4} \cos(2\pi[3f_2]t) + \frac{a_3(A_2)^3}{4} \cos(2\pi[3f_2]t)
\]
\[
= \left[ \frac{a_2(A_1)^2}{2} + \frac{a_2(A_2)^2}{2} \right]
\]
\[
+ [a_1 A_1 + \frac{a_3(A_1)^3}{2} + \frac{a_3(A_1)^3}{4} + \frac{3a_3 A_1(A_2)^2}{2}] \cos(2\pi f_1 t)
\]
\[
+ [a_1 A_2 + \frac{3a_3(A_1)^2 A_2}{2} + \frac{a_3(A_2)^3}{2} + \frac{a_3(A_2)^3}{4}] \cos(2\pi f_2 t)
\]
\[
+ \left[ \frac{a_2(A_1)^2}{2} \right] \cos(2\pi[2f_1]t) + \left[ \frac{a_2(A_2)^2}{2} \right] \cos(2\pi[2f_2]t) + \left[ \frac{a_3(A_1)^3}{4} \right] \cos(2\pi[3f_1]t)
\]
\[
+ \left[ \frac{a_3(A_2)^3}{4} \right] \cos(2\pi[3f_2]t) + [a_2 A_1 A_2] \cos(2\pi[f_1 + f_2]t)
\]
\[
+ [a_2 A_1 A_2] \cos(2\pi[f_2 - f_1]t) + \left[ \frac{3a_3 A_1(A_2)^2}{4} \right] \cos(2\pi[2f_1 + f_2]t)
\]
\[
+ \left[ \frac{3a_3(A_1)^2 A_2}{4} \right] \cos(2\pi[2f_1 - f_2]t) + \left[ \frac{3a_3 A_1(A_2)^2}{4} \right] \cos(2\pi[2f_2 + f_1]t)
\]
\[
+ \left[ \frac{3a_3 A_1(A_2)^2}{4} \right] \cos(2\pi[2f_2 - f_1]t)
\]

For simplicity, the constant coefficients will be defined by the parameters in Table A.1. \( y(t) \) then becomes
Table A.1
Parameter Definitions for the Constants in $y(t)$.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Parameter Description</th>
<th>Parameter Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$b_0^1$</td>
<td>Constant</td>
<td>$\frac{a_2(A_1)^2}{2} + \frac{a_2(A_2)^2}{2}$</td>
</tr>
<tr>
<td>$b_1^1$</td>
<td>1st order harmonic</td>
<td>$a_1 A_1 + \frac{a_3(A_1)^3}{2} + \frac{3a_2A_1(A_2)^2}{2}$</td>
</tr>
<tr>
<td>$b_2^1$</td>
<td>1st order harmonic</td>
<td>$a_1 A_2 + \frac{3a_3(A_1)^2A_2}{2} + \frac{a_3(A_2)^3}{2}$</td>
</tr>
<tr>
<td>$b_1^2$</td>
<td>2nd order harmonic</td>
<td>$\frac{a_2(A_1)^2}{2}$</td>
</tr>
<tr>
<td>$b_2^2$</td>
<td>2nd order harmonic</td>
<td>$\frac{a_2(A_2)^2}{2}$</td>
</tr>
<tr>
<td>$b_1^3$</td>
<td>3rd order harmonic</td>
<td>$\frac{a_3(A_1)^3}{4}$</td>
</tr>
<tr>
<td>$b_2^3$</td>
<td>3rd order harmonic</td>
<td>$\frac{a_3(A_2)^3}{4}$</td>
</tr>
<tr>
<td>$b_1^4$</td>
<td>2nd order intermodulation</td>
<td>$a_2 A_1 A_2$</td>
</tr>
<tr>
<td>$b_2^4$</td>
<td>2nd order intermodulation</td>
<td>$a_2 A_1 A_2$</td>
</tr>
<tr>
<td>$b_1^5$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3(A_1)^2A_2}{4}$</td>
</tr>
<tr>
<td>$b_2^5$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3(A_1)^2A_2}{4}$</td>
</tr>
<tr>
<td>$b_3^5$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3A_1(A_2)^2}{4}$</td>
</tr>
<tr>
<td>$b_4^5$</td>
<td>3rd order intermodulation</td>
<td>$\frac{3a_3A_1(A_2)^2}{4}$</td>
</tr>
</tbody>
</table>

$$y(t) = b_0^1 + b_1^1 \cos(2\pi f_1 t) + b_2^1 \cos(2\pi f_2 t) + b_1^2 \cos(2\pi [2f_1] t) + b_2^2 \cos(2\pi [2f_2] t) + b_3^1 \cos(2\pi [3f_1] t) + b_2^3 \cos(2\pi [3f_2] t) + b_1^4 \cos(2\pi [f_1 + f_2] t) + b_2^4 \cos(2\pi [f_1 + 2f_2] t) + b_1^5 \cos(2\pi [2f_1 + f_2] t) + b_2^5 \cos(2\pi [2f_1 + f_2] t) + b_3^5 \cos(2\pi [2f_2 + f_1] t) + b_4^5 \cos(2\pi [2f_2 - f_1] t)$$

(A.5)

Note that Equation A.5 is the same as Equation 2.7 from Chapter 2.
B: Two-Tone Signal in the Frequency Domain

$y(t)$, as defined by Equation A.5 in Appendix A, is transformed into the frequency domain by the Fourier transform:

$$Y(f) = F[y(t)] = \int_{-\infty}^{\infty} y(t)e^{-i2\pi ft}dt$$  \hspace{1cm} \text{(B.1)}

For comparison purposes, we will also derive the Fourier transform of $x(t)$ in Equation A.1. First, let the Fourier transform of the constant $b \in \mathbb{R}$ and the cosine function be $F[b] = b\delta(f)$ \hspace{1cm} \text{(B.2)}

$$F[\cos(bt)] = \frac{\delta(f - \frac{b}{2\pi}) + \delta(f + \frac{b}{2\pi})}{2}$$  \hspace{1cm} \text{(B.3)}

where $\delta(f)$ is the Dirac Delta Function at $f = 0$. The Dirac Delta Function is defined in terms of the sifting property

$$z(f_1) = \int_{-\infty}^{\infty} z(f)\delta(f - f_1)df$$  \hspace{1cm} \text{(B.4)}

where $f_1 \in \mathbb{R}$ and $z(f)$ is a continuous function at $f_1$ and $f_1$ is a constant. Another property of the Dirac Delta Function is $[41]$

$$\delta(f - f_1) = 0$$  \hspace{1cm} \text{(B.5)}

for $f \neq f_1$. The Fourier transform of $x(t)$ is $[9]$
\[ X(f) = F[x(t)] = \int_{-\infty}^{\infty} x(t) e^{-i2\pi ft} dt \]
\[ = \int_{-\infty}^{\infty} [A_1 \cos(2\pi f_1 t) + A_2 \cos(2\pi f_2 t)] e^{-i2\pi ft} dt \]
\[ = A_1 \int_{-\infty}^{\infty} \cos(2\pi f_1 t) e^{-i2\pi ft} dt + A_2 \int_{-\infty}^{\infty} \cos(2\pi f_2 t) e^{-i2\pi ft} dt \]
\[ = A_1 \delta(f - f_1) + A_1 \delta(f + f_1) + A_2 \delta(f - f_2) + A_2 \delta(f + f_2) \]

where \( f_1 \) and \( f_2 \) are constants. Note that Equation B.6 is the same as Equation 2.2 from Chapter 2. The Fourier transform of \( y(t) \) is derived as follows

\[ Y(f) = F[y(t)] \]
\[ = F[b_0^1 + b_1^1 \cos(2\pi f_1 t) + b_2^1 \cos(2\pi f_2 t) + b_3^1 \cos(2\pi [2f_1] t) + b_2^1 \cos(2\pi [2f_2] t) + b_3^1 \cos(2\pi [3f_1] t) + b_2^1 \cos(2\pi [3f_2] t) + b_1^1 \cos(2\pi [f_1 + f_2] t) \]
\[ + b_2^2 \cos(2\pi [f_2 - f_1] t) + b_3^2 \cos(2\pi [2f_1 + f_2] t) + b_2^2 \cos(2\pi [2f_1 - f_2] t) + b_3^2 \cos(2\pi [2f_2 + f_1] t) + b_2^3 \cos(2\pi [2f_2 - f_1] t)] \]

Using the linearity property [9] of the Fourier transform, \( F[y(t)] \) is rewritten as
\[ F[y(t)] = F[b_0^1] + b_1^1 F[\cos(2\pi f_1 t)] + b_2^1 F[\cos(2\pi f_2 t)] + b_2^2 F[\cos(2\pi [f_1] t)] \\
+ b_3^2 F[\cos(2\pi [f_2] t)] + b_3^3 F[\cos(2\pi [f_3] t)] + b_2^3 F[\cos(2\pi [3f_2] t)] \\
+ b_4^1 F[\cos(2\pi [f_1 + f_2] t)] + b_2^4 F[\cos(2\pi [f_2 - f_1] t)] \\
+ b_2^5 F[\cos(2\pi [2f_1 + f_2] t)] + b_3^5 F[\cos(2\pi [3f_1 + f_2] t)] \\
+ b_3^6 F[\cos(2\pi [2f_1 - f_2] t)] + b_2^7 F[\cos(2\pi [2f_2 - f_1] t)] \\
= b_4^1 \delta(f) + \frac{b_1^1}{2} [\delta(f - f_1) + \delta(f + f_1)] + \frac{b_2^1}{2} \delta(f - [2f_1]) \\
+ \frac{b_3^1}{2} [\delta(f - [2f_1]) + \delta(f + [2f_1])] + \frac{b_3^2}{2} \delta(f - [2f_1]) + \delta(f + [2f_1]) \\
+ \frac{b_4^1}{2} \delta(f - [f_1 + f_2]) + \delta(f + [f_1 + f_2]) \\
+ \frac{b_2^2}{2} \delta(f - [f_2 - f_1]) + \delta(f + [f_2 - f_1]) \\
+ \frac{b_3^1}{2} \delta(f - [2f_1 + f_2]) + \delta(f + [2f_1 + f_2]) \\
+ \frac{b_4^1}{2} \delta(f - [f_1 + f_2]) + \delta(f + [f_1 + f_2]) \\
+ \frac{b_5^1}{2} \delta(f - [2f_1 - f_2]) + \delta(f + [2f_1 - f_2]) + \frac{b_3^2}{2} \delta(f - [2f_2 + f_1]) \\
+ \delta(f + [2f_2 + f_1]) + \frac{b_5^2}{2} \delta(f - [2f_2 - f_1]) + \delta(f + [2f_2 - f_1]) \\
\]

The above can be rewritten as

\[ Y(f) = b_4^1 \delta(f) + \frac{b_1^1}{2} \delta(f - f_1) + \frac{b_1^1}{2} \delta(f + f_1) + \frac{b_2^1}{2} \delta(f - f_2) + \frac{b_2^1}{2} \delta(f + f_2) \\
+ \frac{b_3^1}{2} \delta(f - [2f_1]) + \frac{b_3^1}{2} \delta(f + [2f_1]) + \frac{b_3^2}{2} \delta(f - [2f_2]) + \frac{b_3^2}{2} \delta(f + [2f_2]) \\
+ \frac{b_4^1}{2} \delta(f - [3f_1]) + \frac{b_4^1}{2} \delta(f + [3f_1]) + \frac{b_4^1}{2} \delta(f - [3f_2]) + \frac{b_4^1}{2} \delta(f + [3f_2]) \\
+ \frac{b_5^1}{2} \delta(f - [f_1 + f_2]) + \frac{b_5^1}{2} \delta(f + [f_1 + f_2]) + \frac{b_5^1}{2} \delta(f - [f_2 - f_1]) + \frac{b_5^1}{2} \delta(f + [f_2 - f_1]) \\
+ \frac{b_5^2}{2} \delta(f + [f_2 - f_1]) + \frac{b_5^2}{2} \delta(f - [2f_1 + f_2]) + \frac{b_5^2}{2} \delta(f + [2f_1 + f_2]) \\
+ \frac{b_5^3}{2} \delta(f - [2f_1 - f_2]) + \frac{b_5^3}{2} \delta(f + [2f_1 - f_2]) + \frac{b_5^3}{2} \delta(f - [2f_2 - f_1]) + \frac{b_5^3}{2} \delta(f + [2f_2 - f_1]) \\
\]

Note that Equation B.7 is the same as Equation 2.8 from Chapter 2.
The power spectrum of $Y(f)$ is determined by multiplying $Y(f)$ by its complex conjugate $Y^*(f)$, $P_y(f) = Y(f)Y^*(f)$. For the purposes of this research, we assume $\delta^*(f) = \delta(f)$. $P_y(f)$ is obtained as follows:

$$
P_y(f) = Y(f)Y^*(f) = Y(f)^2
= [b_0^2 \delta(f)]^2 + [b_1^2 \delta(f) - \frac{b_1^2}{2} \delta(f - f_1)] + ... + [b_0^2 \delta(f) - \frac{b_1^2}{2} \delta(f + [2f_2 - f_1])]
+ \left[ \frac{b_1^2}{2} \delta(f - f_1) b_0^2 \delta(f) \right] + ... + \left[ \frac{b_1^2}{2} \delta(f - f_1) b_0^2 \delta(f + [2f_2 - f_1]) \right]
+ ... + ...
+ \left[ \frac{b_1^2}{2} \delta(f + [2f_2 - f_1]) b_0^2 \delta(f) \right] + ... + \left[ \frac{b_1^2}{2} \delta(f + [2f_2 - f_1]) \right]^2
$$

The above equation contains square-terms and cross-terms. The square-terms consist of $\{ \delta(f)^2, \delta(f - f_1)^2, ... \delta(f + [2f_2 - f_1])^2 \}$. Note that there exists 25 square-terms. The cross-terms consist of the set $\{ [b_0^2 \delta(f)][b_1^2 \delta(f - f_1)], ... [\frac{b_1^2}{2} \delta(f + [2f_2 - f_1])\frac{b_1^2}{2} \delta(f - [2f_2 - f_1])] \}$. Since $f_1 \neq f_2$, Equation B.5 can be used to show that for any value of $f$, $\delta(f - f_1) \delta(f - f_2) = 0$. $P_y(f)$ is then

$$
P_y(f) = [b_0^2] \delta(f)^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - f_1)^2
+ [\frac{b_1^2}{2}] \delta(f + f_1)^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - f_2)^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + f_2)^2
+ [\frac{b_1^2}{2}] \delta(f - [2f_1])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [2f_1])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - [2f_2])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [2f_2])^2
+ [\frac{b_1^2}{2}] \delta(f - [3f_1])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [3f_1])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - [3f_2])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [3f_2])^2
+ [\frac{b_1^2}{2}] \delta(f - [f_1 + f_2])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [f_1 + f_2])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - [f_2 - f_1])^2
+ [\frac{b_1^2}{2}] \delta(f + [f_2 - f_1])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - [2f_1 + f_2])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [2f_1 + f_2])^2
+ [\frac{b_1^2}{2}] \delta(f - [2f_1 - f_2])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [2f_1 - f_2])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - [2f_2 + f_1])^2
+ [\frac{b_1^2}{2}] \delta(f + [2f_2 + f_1])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f - [2f_2 - f_1])^2 + \left[ \frac{b_1^2}{2} \right] \delta(f + [2f_2 - f_1])^2

(B.8)

Note that Equation B.8 is the same as Equation 2.9 from Chapter 2.
C: Fourier Transform of the Gaussian Chirp Signal

This Appendix provides the derivation of the Fourier Transform of the Gaussian Chirp Signal. Recall that \( y(t) \) is the following

\[
y(t) = \sum_{j=1}^{M} a_j (C_1)^j e^{ij\phi(t)} e^{-\frac{j(t-\mu)^2}{2\sigma^2}} \quad (C.1)
\]

The Fourier transform of \( y(t) \) is

\[
Y(f) = \int_{-\infty}^{\infty} \left[ \sum_{j=1}^{M} a_j (C_1)^j e^{ij\phi(t)} e^{-\frac{j(t-\mu)^2}{2\sigma^2}} \right] e^{-i2\pi ft} dt
\]

\[
= \int_{-\infty}^{\infty} \left[ \sum_{j=1}^{M} a_j (C_1)^j e^{ij[2\pi f_0 t + \pi \kappa t]} e^{-\frac{j[t^2 - 2\mu t + \mu^2]}{2\sigma^2}} \right] e^{-i2\pi ft} dt
\]

\[
= \int_{-\infty}^{\infty} \left[ \sum_{j=1}^{M} a_j (C_1)^j e^{i j 2\pi f_0 t} e^{ij\pi \kappa t^2} e^{-\frac{j\mu^2}{2\sigma^2}} e^{\frac{j2\mu t}{2\sigma^2}} e^{-\frac{-j\mu^2}{2\sigma^2}} e^{-i2\pi ft} dt
\]

\[
= \sum_{j=1}^{M} [a_j (C_1)^j e^{\frac{-j\mu^2}{2\sigma^2}} \int_{-\infty}^{\infty} e^{-\left(\frac{j}{2\sigma^2} - j\pi \kappa\right)t^2 + (j2\pi f_0 + \frac{j\mu}{\sigma} - i2\pi f)t} dt]
\]

Define \( p_1 = \frac{j}{2\sigma^2} - j\pi k \), \( p_2 = ji2\pi f_0 + \frac{i\mu}{\sigma} - i2\pi f \), and \( (p_3)^2 = \frac{1}{2p_1} \). Substitute \( p_1 \) and \( p_2 \) into the above expression and reform the integral into a Gaussian density function. The integration of a Gaussian density function with full support is 1 [85].
\[ Y(f) = \sum_{j=1}^{M} [a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} \int_{-\infty}^{\infty} e^{-(p_1 t^2 + (p_2 t)^4)} dt] \]

\[ = \sum_{j=1}^{M} [a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} \int_{-\infty}^{\infty} e^{-p_1 [t^2 - (p_2 \over 2p_1 t)^2 - (p_2 \over 2p_1 t^2)]^2} dt] \]

\[ = \sum_{j=1}^{M} [a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} \int_{-\infty}^{\infty} e^{-p_1 [t^2 - (p_2 \over 2p_1 t)^2] + (p_2 \over 2p_1 t^2)]^2} dt] \]

\[ = \sum_{j=1}^{M} [a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} + (p_2 \over 2p_1)^2 \int_{-\infty}^{\infty} e^{-[t^2 - (p_2 \over 2p_1 t)^2] \over 2(\sigma^2)^2} \sqrt{2\pi(p_3)^2} dt] \]

\[ = \sum_{j=1}^{M} [a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} + (p_2 \over 2p_1)^2 \sqrt{2\pi(p_3)^2}] \]

Replace the definitions of \( p_1, p_2, \) and \( p_3 \) into the above produces

\[ Y(f) = \sum_{j=1}^{M} a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} + (p_2 \over 2p_1)^2 \sqrt{2\pi(p_3)^2} \]

\[ = \sum_{j=1}^{M} a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} e^{j[2\pi f_0 + \frac{j\mu}{\sigma^2} - 2\pi f]^2 \over 4[\frac{\mu}{\sigma^2} - j\pi k]} \sqrt{2\pi(\frac{1}{2p_1})} \]

\[ = \sum_{j=1}^{M} a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} e^{(\pi[2\pi f_0 + \frac{j\mu}{\sigma^2} - 2\pi f]^2 \over 2[\frac{\mu}{\sigma^2} - j\pi k]} \sqrt{2\pi(\frac{1}{2p_1})} \]

\[ = \sum_{j=1}^{M} a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} e^{-[2\pi f - j\pi f_0 - \frac{j\mu}{\sigma^2}]^2 \over 2j[\frac{j\mu}{\sigma^2} - 2\pi f]} \sqrt{\frac{\pi}{2\sigma^2} - j\pi k} \]

The resulting expression for \( Y(f) \) is

\[ Y(f) = \sum_{j=1}^{M} \sqrt{\pi} a_j(C_1)^j e^{-j\mu^2 \over 2\sigma^2} e^{-(2\pi f - j\pi f_0 - \frac{j\mu}{\sigma^2})^2 \over 2j[\frac{j\mu}{\sigma^2} - 2\pi f]} \] (C.2)

Equation C.2 is formed into its magnitude and phase components. The equation will be split into 4 parts and rewritten as \( Y(f) = \sum_{j=1}^{M} p_4 \over p_5 \over p_6 \), where the terms \( p_4, p_5, p_6, \) and \( p_7 \) are defined and expanded below.
\[ p_4 = \sqrt{\pi} a_j (C_1) j e^{-\frac{j \omega^2}{2 \sigma^2}} \]  
(C.3)

\[ p_5 = - (2\pi f - j 2\pi f_0 + \frac{i \mu}{\sigma \tau})^2 \]

\[ = - (4\pi^2 f^2 - 4\pi^2 j f_0 f + \frac{4\pi j \mu f}{\sigma^2} - 4\pi^2 j f_0 f + j^2 4\pi^2 f_0^2 \]
\[ - \frac{i 2\pi j f_0 \mu}{\sigma^2} + \frac{i 2\pi j \mu f}{\sigma^2} - \frac{i j^2 2\pi \mu f_0}{\sigma^2} - \frac{j^2 \mu^2}{\sigma^2} \]
\[ = - 4\pi^2 f^2 - 8\pi^2 j f_0 f - \frac{i 4\pi j \mu f}{\sigma^2} + \frac{i 4\pi j^2 f_0 \mu}{\sigma^2} - j^2 4\pi^2 f_0^2 + \frac{j^2 \mu^2}{\sigma^2} \]
\[ = [-4\pi^2 f^2 + 8\pi^2 j f_0 f - j^2 4\pi^2 f_0^2 + \frac{j^2 \mu^2}{\sigma^2}] + i \left[ \frac{4\pi j^2 f_0 \mu}{\sigma^2} - \frac{4\pi j \mu f}{\sigma^2} \right] \]
\[ = [p_8 + i p_9] \]

\[ p_6 = 2j (\frac{1}{\sigma^2} - i 2\pi k) \]
\[ = \frac{2j}{\sigma^2} \sqrt{1 + (2\pi \sigma^2 k)^2 e^{i \arctan(-2\pi^2 k)}} \]
\[ = \frac{2j}{\sigma^2} \sqrt{1 + 4\pi^2 \sigma^4 k^2} e^{-i \arctan(2\pi \sigma^2 k)} \]
\[ = \frac{2j}{\sigma^2} \gamma e^{-i \arctan(2\pi \sigma^2 k)} \]  
(C.5)

\[ p_7 = \sqrt{\frac{j}{2\pi^2} - i j \pi k} \]
\[ = \sqrt{\left(\frac{j}{2\pi^2}\right)(1 - i 2\pi \sigma^2 k)} \]
\[ = \sqrt{\frac{j}{2\pi^2}} \sqrt{1 - i 2\pi \sigma^2 k} \]
\[ = \sqrt{\frac{j}{2\pi^2}} \sqrt{1 + (2\pi \sigma^2 k)^2 e^{i \arctan(-2\pi^2 k)}} \]
\[ = \sqrt{\frac{j}{2\pi^2}} (1 + 4\pi^2 \sigma^4 k^2) \frac{i}{2} e^{-i \arctan(2\pi \sigma^2 k)} \]
\[ = \sqrt{\frac{j}{2\pi^2}} \gamma e^{-i \arctan(2\pi \sigma^2 k)} \]  
(C.6)

Three substitutions, \( p_8, p_9, \) and \( \gamma, \) are made in Equations C.3 - C.6. These substitutions are defined as

\[ p_8 = -4\pi^2 f^2 + 8\pi^2 j f_0 f - j^2 4\pi^2 f_0^2 + \frac{j^2 \mu^2}{\sigma^4} \]  
(C.7)

\[ p_9 = \frac{4\pi j^2 f_0 \mu}{\sigma^2} - \frac{4\pi j \mu f}{\sigma^2} \]  
(C.8)

\[ \gamma = \sqrt{1 + 4\pi^2 \sigma^4 k^2} \]  
(C.9)
\[ \gamma = \sqrt{1 + 4\pi^2 \sigma^4 k^2} \]

Fig. C.1. Right Triangle Where \( \gamma \) is the Length of the Hypotenuse.

Substitute back in \( p_4, p_5, p_6, \) and \( p_7 \) into \( Y(f) = \sum_{j=1}^{M} \frac{p_4}{p_7} e^{\frac{j\pi}{p_6}} \) to obtain

\[
Y(f) = \sum_{j=1}^{M} \frac{p_4}{p_7} e^{\frac{j\pi}{p_6}} \\
= \frac{1}{\sqrt{j}} \sum_{j=1}^{\frac{\gamma^2}{2}} \sqrt{\frac{\gamma^2}{2}} e^{-\frac{j\mu^2}{2\sigma^2}} e^{\frac{j\arctan(2\pi^2 k)}{2\gamma}} e^{\frac{\arctan(2\pi^2 k)}{2\gamma}} \frac{\sigma^2 p_4 + ij \arctan(2\pi^2 k)}{2\gamma}
\]

\[
Y(f) \text{ is now rewritten as}
\]

\[
Y(f) = \frac{\sqrt{2\pi\sigma^2}}{\sqrt{j}} \sum_{j=1}^{M} a_j(C_1) \frac{e^{-j\mu^2}{2\sigma^2}}{\sqrt{j}} e^{\frac{\arctan(2\pi^2 k)}{2\gamma}} e^{\frac{\arctan(2\pi^2 k)}{2\gamma}} \frac{\sigma^2 p_4 + ij \arctan(2\pi^2 k)}{2\gamma}
\]

(C.10)

The expression \( e^{\arctan(2\pi^2 k)} \) in Equation C.10 will be rewritten below using Euler’s formula [117]. Let \( \alpha = \arctan(2\pi^2 k) \) thereby \( e^{\arctan(2\pi^2 k)} = e^{i\alpha} \). \( \alpha \) can be conceptualized as an angle in a right triangle as shown by Figure C.1. The length of the hypotenuse in this triangle is \( \gamma \). Therefore \( \cos[\alpha] = \frac{1}{\gamma} \) and \( \sin[\alpha] = \frac{2\pi^2 k}{\gamma} \). Using these facts, \( e^{\arctan(2\pi^2 k)} \) is becomes

\[
e^{\arctan(2\pi^2 k)} = e^{i\alpha} = \cos[\alpha] + i\sin[\alpha] = \frac{1}{\gamma} + i \frac{2\pi^2 k}{\gamma}
\]
Substituting this into Equation C.10 produces

\[ Y(f) = \frac{\sqrt{2\pi} \sigma^2 e^{i \arctan(2\pi \sigma^2 k)}}{\sqrt{\gamma}} \sum_{j=1}^{M} \frac{a_j (C_1)^j e^{-\gamma^2 (p_j + i p_g)^2 \sigma^2}}{\sqrt{j}} e^{i \arctan(2\pi \sigma^2 k)} \]

Substituting the expressions of \( p_8 \) and \( p_9 \) into the magnitude of the above yields...
\[ Y(f) = \frac{\sqrt{2\pi\sigma^2} e^{i\arctan(2\sigma^2k)}}{\sqrt{\gamma}} \sum_{j=1}^{M} a_j(C_1) e^{-\frac{\mu^2}{2\sigma^2}} e^{\frac{\sigma^2(p_j - p_0k + 2\sigma^2k^2)}{2\gamma^2}} e^{i\frac{(2\sigma^2kp_k + p_0)\sigma^2}{2\gamma^2}} \]

\[ = \frac{\sqrt{2\pi\sigma^2} e^{i\arctan(2\sigma^2k)}}{\sqrt{\gamma}} \sum_{j=1}^{M} a_j(C_1) e^{-\frac{\mu^2}{2\sigma^2}} e^{\frac{4\pi^2 j^2 f_0 f_j}{2\gamma^2}} e^{\frac{4\pi^2 j^2 f_0 f_j}{2\gamma^2}} e^{\frac{2\pi^2 j^2 f_0 f_j}{2\gamma^2}} e^{i\frac{(2\sigma^2kp_k + p_0)\sigma^2}{2\gamma^2}} \]

For simplicity, two substitutions are made to the above expression.

\[ C_2 = \frac{\sqrt{2\pi\sigma^2} e^{i\arctan(2\sigma^2k)}}{\sqrt{\gamma}} \] (C.11)

\[ C_3(j) = \frac{a_j(C_1) e^{-\frac{\mu^2}{2\sigma^2}} e^{\frac{\mu^2}{2\sigma^2}} e^{\frac{-4\pi^2 j^2 f_0 f_j}{2\gamma^2}} e^{\frac{2\pi^2 j^2 f_0 f_j}{\gamma^2}} e^{\frac{-2\pi^2 j^2 f_0^2}{\gamma^2}}}{\sqrt{\gamma}} \] (C.12)
$C_2$ is a constant variable and $C_3(j)$ is a function dependent on the nonlinearity of order $M$. Substituting $C_2$ and $C_3(j)$ into the above expression produces

$$Y(f) = C_2 \sum_{j=1}^{M} C_3(j)e^{-\frac{2\sigma^2(x^2)}{j\gamma^2}}e^{\frac{i(2\pi\sigma^2p_k+p_9)\sigma^2}{2j\gamma^2}}$$

Substituting in the expressions for $p_8$ and $p_9$ into the phase component of $Y(f)$ produces
\[ Y(f) = \sum_{j=1}^{M} C_3(j) e^{\frac{-2\pi^2 \sigma^2 (f - j f_0 - j \mu k)^2}{j^2 \gamma^2}} e^{i \left( 2\pi \sigma^2 k p_k + \eta_j \right) \sigma^2} e^{i \left( \frac{16\pi^2 \sigma^4 f_0^2 \eta_j}{\sigma^2} - 4\pi \mu f \right) \sigma^2} \]

For simplicity, three substitutions are made to the above expression.
\[ C_4(j) = \frac{j k \pi^2}{\gamma^2} - \frac{j k 4 \pi^3 \sigma^4 f_0^2}{\gamma^2} + \frac{2 \pi j f_0 \mu}{\gamma^2} + \frac{4 \pi^3 \sigma^4 k}{j \gamma^2} \left( \frac{\mu j}{4 \pi^2 \sigma^4 k} - j f_0 \right)^2 \]  
(C.13)

\[ \psi(j, f) = \frac{-4 \pi^3 \sigma^4 k}{j \gamma^2} (f + \frac{j \mu}{4 \pi^2 \sigma^4 k} - j f_0)^2 \]  
(C.14)

\[ \xi(j, f) = e^{\frac{-2 \pi^2 \sigma^2 (f - j f_0 - j \mu)^2}{j \gamma^2}} \]  
(C.15)

\( C_4(j) \) is a function dependent on the nonlinearity of order \( M \). Equations C.14 and C.15 are dependent on frequency and the nonlinearity of order \( M \). It is of interest to note that \( \xi(j, f) \) has a Gaussian form. Substituting these equations into \( Y(f) \) produces

\[ Y(f) = C_2 \sum_{j=1}^{M} [C_3(j) e^{iC_4(j)}] \xi(j, f) e^{i\psi(j, f)} \]  
(C.16)

Note that Equation C.16 is equal to Equation 2.20 from Chapter 2.
D: Power Spectrum of the Gaussian Chirp Signal

The derivation of the power spectrum for a Gaussian chirp signal is shown here. The power spectrum of $Y(f)$ is realized by multiplying $Y(f)$ by its complex conjugate $Y^*(f)$. $Y(f)$ was derived in Appendix C and was defined by Equation C.16. $P_y(f)$ is

$$P_y(f) = \{Y(f)\}\{Y^*(f)\} = \{C_2 \sum_{j=1}^{M} C_3(j)\xi(j, f)e^{iC_4(j)}e^{i\psi(j, f)}\}$$

$$= C_2C_2^*\{C_3(1)\xi(1, f)e^{iC_4(1)}e^{i\psi(1, f)} + ... + C_3(M)\xi(M, f)e^{iC_4(M)}e^{i\psi(M, f)}\}$$

$$= C_2C_2^*[\{C_3(1)\xi(1, f)]^2 + [C_3(1)C_3(2)\xi(1, f)\xi(2, f)]$$

$$e^{i[C_4(1)-C_4(2)+\psi(1, f)\psi(2, f)]} + ... + [C_3(1)C_3(M)\xi(1, f)\xi(M, f)]$$

$$e^{i[C_4(1)-C_4(M)+\psi(1, f)-\psi(M, f)]}$$

$$+ ... +$$

$$+ ... +$$

$$+[C_3(M)C_3(1)\xi(M, f)\xi(1, f)]e^{i[C_4(M)-C_4(1)+\psi(M, f)-\psi(1, f)]}$$

$$+ ... + [C_3(M)\xi(M, f)]^2$$

There exists a total of $M^2$ terms that are summed together. Define a square-term as any term in $Y(f)$ of order $j$ that is multiplied by a term of the same order (i.e., $l = j$) in $Y^*(f)$. Note that there is no phase component for a square-term. For example, when $j = l$, a square-term would be $[C_3(j)C_3(l)\xi(j, f)\xi(l, f)]e^{i[C_4(j)-C_4(l)+\psi(j, f)-\psi(l, f)]}$

$$= [C_3(j)C_3(j)\xi(j, f)\xi(j, f)]e^{i[C_4(j)-C_4(j)+\psi(j, f)-\psi(j, f)]} = [C_3(j)\xi(j, f)]^2.$$
square-terms. The set of square-terms is \{[C_3(1)\xi(1, f)]^2, [C_3(2)\xi(2, f)]^2, \ldots [C_3(M)\xi(M, f)]^2\}. Define a cross-term as any term in \(Y(f)\) of order \(j\) multiplied by a term of a different order \((l \neq j)\) in \(Y^*(f)\). There exists a magnitude and phase component for a cross-term. An example of a cross-term is \([C_3(j)C_3(l)\xi(j, f)\xi(l, f)]\).

\[e^{i[C_4(j) - C_4(l) + \psi(j, f) - \psi(l, f)]}.\]

Since there exists \(M\) square-terms, the number of cross-terms is \(M^2 - M = M(M - 1)\). The set of cross-terms is \{\(C_3(1)C_3(2)\xi(1, f)\xi(2, f)\ e^{i[C_4(1) - C_4(2) + \psi(1, f) - \psi(2, f)]}, C_3(1)C_3(3)\xi(1, f)\xi(3, f)\ e^{i[C_4(1) - C_4(3) + \psi(1, f) - \psi(3, f)]}, \ldots C_3(M)C_3(M - 1)\xi(M, f)\xi(M - 1, f)\ e^{i[C_4(M) - C_4(M - 1) + \psi(M, f) - \psi(M - 1, f)]}\}.

To simplify the power spectrum, let the function \(W(j, l, f)\) be

\[W(j, l, f) = C_3(j)C_3(l)\xi(j, f)\xi(l, f)\] (D.1)

This Equation replaces the magnitude components in \(P_y(f)\). In addition, define the function \(Q(j, l, f)\) as

\[Q(j, l, f) = C_4(j) - C_4(l) + \psi(j, f) - \psi(l, f)\] (D.2)

This Equation replaces the phase components in \(P_y(f)\). The following Lemma’s are used to denote properties of \(W(j, l, f)\) and \(Q(j, l, f)\).

**Lemma 1 (Gaussian Power Spectrum Lemma 1)** Let \(W(j, l, f)\) be the function defined by Equation D.1. A property of this function is that \(W(j, l, f) = W(l, j, f)\).

**Proof** \(W(j, l, f) = C_3(j)C_3(l)\xi(j, f)\xi(l, f) = C_3(l)C_3(j)\xi(l, f)\xi(j, f) = W(l, j, f)\) ■

**Lemma 2 (Gaussian Power Spectrum Lemma 2)** Let \(Q(j, l, f)\) be the function defined by Equation D.2. A property of this function is that \(Q(j, l, f) = -Q(l, j, f)\).

**Proof** \(Q(j, l, f) = C_4(j) - C_4(l) + \psi(j, f) - \psi(l, f) = -[-C_4(j) + C_4(l) - \psi(j, f) + \psi(l, f)] = -[C_4(l) - C_4(j) + \psi(l, f) - \psi(j, f)] = -Q(l, j, f)\) ■

Note that \(W(j, j, f)\) and \(W(l, l, f)\) are square-terms and \(W(j, l, f)e^{iQ(j, l, f)}\) and \(W(l, j, f)e^{iQ(l, j, f)}\) are cross-terms. Substituting \(W(j, l, f)\) and \(Q(j, l, f)\) into \(P_y(f)\) produces
\[ P_y(f) = C_2 C_2^* \left\{ [C_3(1)\xi_1, f]^2 
+ [C_3(1)C_3(2)\xi_1, f, f_1, f]e^{i[C_4(1) - C_4(2) + \psi_1, f, \psi_2, f]} 
+ ... + [C_3(1)C_3(M)\xi_1, f, f_1, f]e^{i[C_4(1) - C_4(L) + \psi_1, f, \psi_L, f]} 
+ ... + 
... + 
+ [C_3(M)C_3(1)\xi_1, f, f_1, f]e^{i[C_4(M) - C_4(1) + \psi(M, f) - \psi_1, f]} 
+ ... + [C_3(M)\xi_1, f, f] \right\} 
= C_2 C_2^* \left\{ W(1, 1, f) + W(1, 2, f)e^{i\psi(1, 2, f)} + ... + W(1, M, f)e^{i\psi(1, M, f)} 
+ ... + 
... + 
+ W(M, 1, f)e^{i\psi(M, 1, f)} + ... + W(M, M, f) \right\} \]

Grouping the square-terms and cross-terms produces
\[ P_y(f) = C_2 C_n^*(W(1, 1, f) + \ldots + W(M, M, f)) + \]
\[ + [W(1, 2, f) e^{iQ(1,2,f)} + \ldots + W(1, M, f) e^{iQ(1,M,f)} + W(2, 1, f) e^{iQ(2,1,f)} + W(2, 3, f) e^{iQ(2,3,f)} + \ldots + W(2, M, f) e^{iQ(2,M,f)} + \ldots + \]
\[ + W(M, 1, f) e^{iQ(M,1,f)} + W(M, 2, f) e^{iQ(M,2,f)} + \ldots + W(M, M - 1, f) e^{iQ(M,M-1,f)]} \]
\[ = C_2 C_n^* \left\{ \sum_{j=1}^{M} W(j, j, f) \right\} + \]
\[ + [W(1, 2, f) e^{iQ(1,2,f)} + W(2, 1, f) e^{iQ(2,1,f)}] + [W(1, 3, f) e^{iQ(1,3,f)} + W(3, 1, f) e^{iQ(3,1,f)}] + \ldots + [W(1, M, f) e^{iQ(1,M,f)} + W(M, 1, f) e^{iQ(M,1,f)]} + [W(2, 3, f) e^{iQ(2,3,f)} + W(3, 2, f) e^{iQ(3,2,f)}] + [W(2, 4, f) e^{iQ(2,4,f)} + W(4, 2, f) e^{iQ(4,2,f)}] + \ldots + [W(2, M, f) e^{iQ(2,M,f)} + W(M, 2, f) e^{iQ(M,2,f)]} + \ldots + \]
\[ + [W(M - 1, M, f) e^{iQ(M-1,M,f)} + W(M, M - 1, f) e^{iQ(M,M-1,f)]} \]

Using Lemmas 1 and 2 produces
\[ P_y(f) = C_2C_2 \left\{ \left[ \sum_{j=1}^{M} W(j, j, f) \right] + \right. \\
W(1, 2, f)e^{iQ(1,2)} + W(1, 2, f)e^{-iQ(1,2)} + W(1, 3, f)e^{iQ(1,3)} + W(1, 3, f)e^{-iQ(1,3)} + \right. \\
+ W(1, M, f)e^{iQ(1,M)} + W(1, M, f)e^{-iQ(1,M)} + \right. \\
+ \left[ W(2, 3, f)e^{iQ(2,3)} + W(2, 3, f)e^{-iQ(2,3)} + W(2, 4, f)e^{iQ(2,4)} + W(2, 4, f)e^{-iQ(2,4)} \right] + \right. \\
+ \left[ W(2, M, f)e^{iQ(2,M)} + W(2, M, f)e^{-iQ(2,M)} \right] + \right. \\
+ \cdots + \right. \\
\left. \left. + \left[ W(M - 1, M, f)e^{iQ(M-1,M)} + W(M - 1, M, f)e^{-iQ(M-1,M)} \right] \right\} \\
= C_2C_2 \left\{ \left[ \sum_{j=1}^{M} W(j, j, f) \right] \right. \\
+ \left[ \sum_{j=1}^{M-1} \sum_{l=j+1}^{M} \left( W(j, l, f)e^{iQ(j,l)} + W(j, l, f)e^{-iQ(j,l)} \right) \right] \right\} \\
= C_2C_2 \left\{ \left[ \sum_{j=1}^{M} W(j, j, f) \right] + \left[ \sum_{j=1}^{M-1} \sum_{l=j+1}^{M} 2W(j, l, f) \cos(Q(j, l, f)) \right] \right\} \\
\]

Substituting the definitions of \( W(j, l, f) \) and \( Q(j, l, f) \) into the above

\[ P_y(f) = C_2C_2 \left\{ \sum_{j=1}^{M} [C_3(j)\xi(j, f)]^2 + \right. \\
\sum_{j=1}^{M-1} \sum_{l=j+1}^{M} 2C_3(j)C_3(l)\xi(j, f)\xi(l, f) \cos(C_4(j) - C_4(l) + \psi(j, f) - \psi(l, f)) \right\} \]  \hspace{1cm} (D.3)

Note that Equation D.3 is equal to Equation 2.28 from Chapter 2.

If minimal overlap exists between the Gaussian functions in the cross-terms, then a minimal amount of interference is generated. A minimal overlap condition is defined as
\[ \xi(j, f) \xi(l, f) = \epsilon \]  
\[ \text{(D.4)} \]

where \( \epsilon \approx 0 \), and \( l > j \). Equation D.4 is considered at the intercept point \( f = f_{\text{int}} \). At \( f = f_{\text{int}} \), \( \xi(j, f_{\text{int}}) = \xi(l, f_{\text{int}}) \). Then the following reduction can be made.

\[
\frac{-(f_{\text{int}} - \mu_\xi(j))^2}{2(\sigma_\xi(j))^2} = \frac{-(f_{\text{int}} - \mu_\xi(l))^2}{2(\sigma_\xi(l))^2}
\]

Note that \( \mu_\xi(j) < f_{\text{int}} < \mu_\xi(l) \).

\[
\frac{f_{\text{int}} - \mu_\xi(j)}{\sqrt{2}\sigma_\xi(j)} = \frac{f_{\text{int}} - \mu_\xi(l)}{\sqrt{2}\sigma_\xi(l)}
\]

\[
\frac{f_{\text{int}} - \mu_\xi(j)}{\sqrt{2}\sigma_\xi(j)} = \frac{f_{\text{int}} - \mu_\xi(l)}{\sqrt{2}\sigma_\xi(l)}
\]

\[
\frac{f_{\text{int}}}{\sigma_\xi(j)} - \frac{\mu_\xi(j)}{\sigma_\xi(j)} = \frac{\mu_\xi(l)}{\sigma_\xi(l)} - \frac{f_{\text{int}}}{\sigma_\xi(l)}
\]

\[
\frac{f_{\text{int}}}{\sigma_\xi(j)} + \frac{f_{\text{int}}}{\sigma_\xi(l)} = \frac{\mu_\xi(j)}{\sigma_\xi(j)} + \frac{\mu_\xi(l)}{\sigma_\xi(l)} = \frac{\mu_\xi(j)\sigma_\xi(l) + \mu_\xi(l)\sigma_\xi(j)}{\sigma_\xi(l)\sigma_\xi(j)}
\]

\[
\frac{f_{\text{int}}\sigma_\xi(j) + f_{\text{int}}\sigma_\xi(l)}{\sigma_\xi(l)\sigma_\xi(j)} = \frac{\mu_\xi(j)\sigma_\xi(l) + \mu_\xi(l)\sigma_\xi(j)}{\sigma_\xi(l)\sigma_\xi(j)}
\]

\( f_{\text{int}} \) therefore equals

\[
f_{\text{int}} = \frac{\mu_\xi(l)\sigma_\xi(j) + \mu_\xi(j)\sigma_\xi(l)}{\sigma_\xi(j) + \sigma_\xi(l)} \quad \text{(D.5)}
\]
If $f = f_{int}$, then Equation D.4 reduces to the following

$$
e^{-\frac{(f_{int} - \mu_{x}(j))^2}{2\sigma_{x}(j)^2}} e^{-\frac{(f_{int} - \mu_{x}(l))^2}{2\sigma_{x}(l)^2}} = \epsilon$$

\[
\frac{- (f_{int} - \mu_{x}(j))^2}{2(\sigma_{x}(j))^2} - \frac{(f_{int} - \mu_{x}(l))^2}{2(\sigma_{x}(l))^2} = \ln(\epsilon)
\]

where $\ln()$ is the natural log.

\[
\ln(\epsilon) = \frac{- (f_{int} - \mu_{x}(j))^2}{2\sigma_{x}(j)^2} - \frac{(f_{int} - \mu_{x}(l))^2}{2\sigma_{x}(l)^2}
\]

\[
= \frac{- (f_{int}^2 - 2\mu_{x}(j)f_{int} + \mu_{x}(j)^2)}{2\sigma_{x}(j)^2} - \frac{(f_{int}^2 - 2\mu_{x}(l)f_{int} + \mu_{x}(l)^2)}{2\sigma_{x}(l)^2}
\]

\[
= \frac{-2(\sigma_{x}(j)^2 + \sigma_{x}(l)^2)f_{int}^2 + 4(\mu_{x}(j)\sigma_{x}(l)^2 + \mu_{x}(l)\sigma_{x}(j)^2)f_{int} + 2(\mu_{x}(j)^2\sigma_{x}(l)^2 + \mu_{x}(l)^2\sigma_{x}(j)^2)}{4\sigma_{x}(l)^2\sigma_{x}(j)^2}
\]

Substituting $f_{int}$ from Equation D.5 into the above reduces to

\[
\frac{(\mu_{x}(l) - \mu_{x}(j))}{(\sigma_{x}(j) + \sigma_{x}(l))} = \sqrt{\frac{1}{\epsilon}} \quad (D.6)
\]

Note that Equation D.6 is equal to Equation 2.37 from Chapter 2.
VITA
VITA

Anthony Frank Martone was born in Denville, New Jersey. He received the AS and AAS in 1999 at the County College of Morris in Randolph, New Jersey where he won the Phi Theta Kappa Award. He received the BS degree at Rensselaer Polytechnic Institute where he graduated first in his class with a 4.0/4.0 grade point average. His research interests included automatic control systems and robotics. While at Rensselaer, he received the Joseph H. Smith Scholarship, the 4.0 Award, and was a member of the National Dean’s List eight times.

Since 2002, Anthony has studied in the direct Ph.D. program at Purdue University in West Lafayette, Indiana, where he received a one year graduate fellowship. He has worked as a research assistant for Professor Edward Delp, the Silicon Valley Professor of Electrical and Computer Engineering. From 2002 - 2005, he was supported by a project funded by the C-SPAN Archives located in West Lafayette, Indiana. During this time, he developed video and text analysis tools for the Archives. In the Spring 2004, Anthony and his four member team placed fifth among 50 competitors at the Burton D. Morgan Entrepreneur Business Competition in West Lafayette, Indiana. From 2005 - 2007, he was supported by a project funded by the U.S. Army MURI program. During this time, he developed methods to identify RF devices. In the Spring 2007, Anthony was a member of a three person team that entered the CERIAS Research Poster Competition, winning first place.

During his Ph.D. studies, Anthony has worked at the Army Research Labs (ARL) in Adelphi, Maryland. While at ARL, he developed tools to extract features of moving targets from synthetic aperture radar (SAR) images and designed classification systems to classify the features.
Journal Papers:


Conference Papers:


