Computerized Evaluation of Parameters for HEMT
DC and Microwave S Parameter models

A Thesis Presented to
The Faculty of the
Fritz J. and Dolores H. Russ
College of Engineering and Technology
Ohio University

In Partial Fulfillment
of the Requirement for the Degree
Master of Science

by
Lu Chen
November, 1995
ACKNOWLEDGEMENTS

I would like to take this opportunity to thank the individuals who have contributed to this thesis.

First I express my deepest gratitude to my advisor and committee chairman, Dr. Robert Curtis, for his encouragement, guidance and patience. Then I wish to acknowledge my sincere gratitude to Dr. Ebrahim Mokari, my former advisor, for his leading me to this challenging field and for his support. I am also grateful to my committee members, Dr. John Tague, Dr. Roger Radcliff and Dr. Thomas Wolf, for their invaluable suggestions and time.

Last I want to thank my family: my father, Jinbo Chen and my mother, Shirui Ji, for their nurturing and teaching which gave a solid foundation to my education; my wife, Hailin Yin, for her love, encouragement and understanding; and my little girl, Katherine Chen, for her positive attitude which gives me a peaceful environment.
**TABLE OF CONTENTS**

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table of Contents</td>
<td>i</td>
</tr>
<tr>
<td>Symbol List</td>
<td>1</td>
</tr>
<tr>
<td>CHAPTER 1 INTRODUCTION</td>
<td></td>
</tr>
<tr>
<td>1.1 Importance of Hemt modeling</td>
<td>4</td>
</tr>
<tr>
<td>1.2 Recent Research on HEMT Device Modelling</td>
<td>5</td>
</tr>
<tr>
<td>1.3 The Scope of The Research</td>
<td>6</td>
</tr>
<tr>
<td>CHAPTER 2 DC NONLIEAR MODEL</td>
<td></td>
</tr>
<tr>
<td>2.1 Large Signal Models of HEMT Devices</td>
<td>8</td>
</tr>
<tr>
<td>2.1-1 The HEMT Curtice Model</td>
<td>9</td>
</tr>
<tr>
<td>2.1-2 The HEMT Materka-Kacprzak Model</td>
<td>11</td>
</tr>
<tr>
<td>2.1-3 The HEMT Advanced Curtice Model</td>
<td>11</td>
</tr>
<tr>
<td>2.2 The Extraction of The Model Parameters</td>
<td>12</td>
</tr>
<tr>
<td>2.2-1 Picking the Parameters Using Curve Fitting Technique</td>
<td>13</td>
</tr>
<tr>
<td>A Extraction of the HEMT Curtice Model Parameters</td>
<td>13</td>
</tr>
<tr>
<td>B Extraction of the HEMT Advanced Curtice Model Parameters</td>
<td>15</td>
</tr>
<tr>
<td>C Extraction of the HEMT Materka-Kacprzak Model Parameters</td>
<td>13</td>
</tr>
<tr>
<td>2.2-2 Optimization</td>
<td>16</td>
</tr>
<tr>
<td>2.3 Results</td>
<td>17</td>
</tr>
<tr>
<td>2.4 Further Applications of the Modeling Method</td>
<td>19</td>
</tr>
</tbody>
</table>
CHAPTER 3 AC MODEL

3.1 The AC Equivalent Circuit for a GaAs FETs ........................................... 34

3.2 Extraction of the AC Model Element ....................................................... 35

3.2-1 Extraction of the Extrinsic Elements .................................................. 35

3.2-2 Extraction of the Intrinsic Elements ................................................... 39

3.2-3 Choosing the Initial Parameters for the Optimization ....................... 41

3.3 Optimization and Results ....................................................................... 42

CHAPTER 4 The SIMPLIFICATION OF THE AC MODEL

4.1 The Weight of Each Element in the Model ............................................. 55

4.1-1 The Examine Method ......................................................................... 56

4.1-2 The Two Fingered Device .................................................................. 57

A. The Extrinsic Elements ......................................................................... 57

B. The Intrinsic Elements ......................................................................... 58

4.1-3 The Four Fingered Device .................................................................. 58

4.2 The Simplified HEMT AC Model and Optimized Results ....................... 59

4.2-1 The First Simplified Model ................................................................. 59

4.2-2 The Second Simplified Model ............................................................. 61

4.3 Conclusion ............................................................................................ 62

CHAPTER 5 CONCLUSION

5.1 DC Modeling .......................................................................................... 97

5.2 AC High Frequency Modeling ............................................................... 98

5.3 Simplified Models ................................................................................... 98
APPENDIX

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A. Numerical Method</td>
<td>99</td>
</tr>
<tr>
<td>B. Optimization Programs</td>
<td>105</td>
</tr>
<tr>
<td>C. DC Model Functions</td>
<td>113</td>
</tr>
<tr>
<td>D. Complete AC Model Source Code For Optimization</td>
<td>121</td>
</tr>
</tbody>
</table>

REFERENCE ................................................................. 131
List of Symbols in the Models

**DC model symbols:**

- $I_{ds}$: drain source current
- $I_{ds\text{fet}}$: drain source current when $V_{TO < V_T < V_{gs} < V_{pf}}$
- $\beta$: the transconductance parameter
- $I_{ds}$: the saturation current at zero gate source voltage
- $\alpha$: DC model fitting parameter
- $\lambda$: the output conductance parameter
- $\psi$: the exponent of empirical transconductance degradation parameter
- $\xi$: the empirical transconductance degradation parameter
- $\mu_{\text{crit}}$: the mobility parameter at the critical field for mobility degradation
- $\gamma$: the parameter describes the effective pinch-off potential
- $V_{TO}$: the threshold voltage of the device
- $V_T$: the modified pinch-off potential
- $V_{pf}$: the gate voltage where the transconductance begins to degrade
- $V_{gs}$: the gate source voltage
- $V_{ds}$: the drain source voltage
AC model symbols:

- $C_{pgs}$: the pad capacitance between gate and source
- $C_{pgd}$: the pad capacitance between gate and drain
- $C_{pds}$: the pad capacitance between drain and source
- $L_g$: the parasitic gate inductance
- $L_d$: the parasitic drain inductance
- $L_s$: the parasitic source inductance
- $R_g$: the parasitic gate resistance
- $R_d$: the parasitic drain resistance
- $R_s$: the parasitic source resistance
- $C_{gs}$: the intrinsic capacitance between gate and source
- $C_{gd}$: the intrinsic capacitance between gate and drain
- $C_{ds}$: the intrinsic capacitance between drain and source
- $R_{gs}$: the intrinsic resistance between gate and source (or $R_i$)
- $R_{gd}$: the intrinsic resistance between gate and drain
- $R_{ds}$: the intrinsic resistance between drain and source
- $G_m$: the intrinsic voltage control current source
- $\tau$: the time delay of $G_m$
- $g_m$: the magnitude of $G_m$
CHAPTER 1

INTRODUCTION

The high electron mobility transistor (HEMT) is a heterostructure field effect device. Here, the electrons have high mobility because of the velocity transport properties in a potential well of lightly doped n type semiconductor material. A cross-sectional view of a conventional HEMT structure is presented in fig 1.1. As in an ordinary FET, the surface of the semiconductor structure contains three metal electrodes: the source, gate and drain. The gate is a Schottky barrier, while the source and drain are ohmic contacts. Compared to an normal FET, the HEMT structure is more complex because of more layers of different materials and doping. Although such a complexity causes fabrication difficulties, added costs and lower yield, the structure improves significantly the device noise figure and the high frequency performance.

A two dimensional electron gas (2-DEG) takes an important role in a HEMT device. The thicknesses of n-type Aluminum-Gallium-Arsenium (AlGaAs) and the undoped AlGaAs depletion layer are critical in determining device behavior. A high carrier concentration (2-DEG) resides in a narrow region along the Gallium-Arsenium (GaAs) side of the AlGaAs/GaAs boundary, the heterojunction. Electrons traveling in this region do not collide with ionized donors since the GaAs is undoped. As electron mobility is highest for lightly doped material, the 2-DEG transport properties are favorable for best response time and high frequency operation.
The 2-DEG density is controlled by the gate bias. Increasing the negative bias applied to the gate decreases the depth of the potential well at the AlGaAS/GaAs boundary that causes the carrier density of the electron gas to decrease and channel conductivity to go down.

With the discovery of a two dimensional electron gas (2-DEG) at the n-AlGaAs heterostructure interface in the low-band-gap GaAs side in 1978 [1], a new solid state device was created to be applied to high speed, high frequency and low noise circuits. This new device is named according to its characteristics: the high electron mobility transistor (HEMT), the selectively doped heterojunction transistor (SDHT), the modulation-doped FET (MODFET) and the two-dimensional electron gas FET (TEGFET). Research on the HEMT is roughly divided into two parts. The first is where the device is analyzed and designed for different applications such as monolithic microwave integrated circuits (MMIC) and other high frequency or high speed areas. Among these, the techniques of device fabrication have improved. The operating frequency of the MMIC has reached 120 GHz [18]. The second field to consider is the modelling of the device in different ways to simulate its operation. Different models can be developed to fit the characteristics of the device. This thesis takes up the second problem.

1.1 The Importance of HEMT Modeling

A circuit model for a solid state device is needed to create software for simulators
to effectively reproduce the physical properties of the device. A good model narrows the gap between the circuit on paper and the real product which means the design and fabrication process is shortened. A HEMT is a more complex than other FET devices, so modeling it is a difficult job. This thesis models HEMT devices with three DC models (Curtice, advanced Curtice and Materka-Kacprzak) and presents several AC HEMT models.

1.2 Recent Research on HEMT Device Modeling

Modeling a device is done in several steps: 1. model DC characteristics - this should use a nonlinear model to approach the real device; 2. model AC characteristics - this should give an equivalent circuit to substitute for the device in an AC small signal analysis; 3. noise modeling - this also needs an equivalent circuit; 4. model nonlinear bias dependent elements, which mainly include diodes, resistances and capacitances. This research uses measured data to develop the first two kinds of model.

For the DC HEMT characteristics, many ways are used to fit the behavior of the device. Based on the similarities and differences between MESFET's and HEMT's, the MESFET models are modified to predict the HEMT's behavior [14] (see chapter 2 for details). Another method just uses the MESFET model to fit the characteristics [12].

For the AC models, a completed equivalent circuit model is shown at fig 3.1 [7]. The widely used FET model does not have the three contact pad capacitances and the feedback resistance named $R_{gd}$ in figure 3.1. But in the high frequency range, these
effects cannot be ignored. Some models consider a more complex parasitic structure [11]. Others ignore the feedback effect of $R_{eg}$ [10]. One model, called the sliced model [27], consists of several units, and each unit has the same elements used in the completed AC intrinsic model in chapter 3 of this thesis.

1.3 The Scope of the Research

This research has three steps to model HEMT electronic properties based on the measured data. The measured data includes the DC characteristics and microwave S parameters. So the procedure and results of the research are presented in three chapters to cover this contents.

Chapter 2 introduces the DC modeling. Three DC models (HEMT Curtice model, HEMT Materka-Macprzak model and advanced Curtice model) are used to model a HEMT device with a 0.25 micron long two fingered gate which has a 100 micron gate width. A method to extract the parameters is given. An optimization method is used in the modeling. Another two devices (two fingered and four fingered with 0.25 micron gate lengths, 50 micron gate widths) are modeled by applying this method.

Chapter 3 presents the modeling of AC characteristics. A complete AC model[7] is used to model two HEMT devices. A method for an initial approximation of the parameters is introduced and the detailed process of optimization is given.

Chapter 4 gives a procedure to simplify the AC models in chapter 3. The weight of each element in the completed model is examined first. Two simplified models are then
obtained.

Figure 1.1 A cross section view of conventional HEMT device structure
CHAPTER 2

DC NONLINEAR MODELING

This chapter starts with an introduction of the equations for three HEMT large signal models: HEMT Curtice model, HEMT Makerka-Kacprzak model and HEMT advanced Curtice model. Following that, the extracting and optimizing the DC model parameters for device C (2x0.25x100 gate) are presented. Finally, another two devices (two fingered and four fingered devices, called Device A and B, with 50 micron gate width and 0.25 micron gate length) are modeled by using this method.

2.1 Large Signal Models of HEMT Devices

GaAs MESFET models successfully predict large signal performance by predicting the voltage dependence of device characteristics. Of device's bias dependent nonlinear behavior, the transconductance is the most critical to the accurate prediction of many important large signal effects. The structure of each of the models has the same elements here [7] (figure 2.1). They have the same characteristics at low gate voltages. In
Figure 2.1 The large signal model of a HEMT
contrast to a MESFET, however, the HEMT’s transconductance begins to decrease rapidly at some gate bias level. Based on this point, the HEMT models can be modified from the MESFET models.

In figure 2.1, there are three main bias dependent elements \( (C_{gd}, C_{gs} \text{ and } I_d) \). Among them, drain current is the most important electrical property. The other two nonlinear capacitors do not affect the DC characteristics and will be ignored in the DC analysis.

Many different models have been developed to fit the DC characteristics of HEMT’s including the drain current and transconductance. The expressions of \( I_d \) in figure 2.1 differ in detail according to the different curve fitting techniques. Three large signal HEMT models are considered in this chapter. They are the HEMT Curtice model, the HEMT Materka-Kacprzak model and the HEMT Advanced Curtice Model.

2.1-1 The HEMT Curtice Model

The Curtice model [16], one of the earliest large signal models for the MESFET devices, is modified to express the characteristics of the HEMT. To account for the difference between these two kinds of devices, the drain current function is divided into two regions depending on the gate voltage. The first region, at very negative gate voltages, has the same expression as the MESFET model (\( I_{dsfet} \), the drain to source current in the equation). The second is modified according to the character of the
HEMT's transconductance. The curve fitting multipliers, \( \tanh(\alpha V_{ds}) \) and \( 1 + \lambda V_{ds} \), are kept in both regions. The equations are [2]:

\[
\begin{align*}
I_{ds_{\text{def}}}&=\beta(V_{gs}-V_{T0})^2\tanh(\alpha V_{ds})(1+\lambda V_{ds}) & V_{TO} < V_{gs} < V_{pf} & 2.1a \\
I_{ds}&=I_{ds_{\text{def}}}-\frac{\xi}{\psi+1}(V_{gs}-V_{pf})^{\psi+1}f(V_{ds}) & V_{pf} < V_{gs} < 0 & 2.1b \\
I_{ds}&=0 & V_{gs} \leq V_{T0} < V_{pf} & 2.1c
\end{align*}
\]

where \( I_{ds_{\text{def}}} \) is the DC drain to source current when \( V_{gs} < V_{pf} \). It has the same expression as \( I_{ds} \) in the MESFET Curtice model. The function \( f(V_{ds}) = \tanh(\alpha V_{ds}) x (1+\lambda V_{ds}) \), is a nonlinear separable function of \( V_{ds} \). \( \beta \) is the transconductance parameter used to fit the transconductance to actual device behavior. \( \alpha \) is another model fitting parameter. \( \lambda \) is used to model the device output conductance which is the slope of the drain to source current with respect to the drain-source voltage. \( V_{T0} \) is defined as the threshold voltage of the device. \( V_{pf} \) is a gate voltage at which the transconductance begins to degrade. \( \xi \) and \( \psi \) are both the empirical transconductance degradation parameters. But exponent \( \psi \) determines the rate of degradation. The last three parameters are also used in the other two models below.
2.1-2 The HEMT Materka-Kacprzak Model

Just as in the HEMT Curtice model, the HEMT Materka-Kacprzak model is modified from the MESFET model [17] which has \( I_{\text{dsfet}} \) as the drain to source current. The equations are expressed as [2]:

\[
\begin{align*}
I_{\text{dsfet}} &= I_{\text{ds}} (1 - \frac{V_{gs}}{V_T})^2 \tanh\left( \frac{\alpha V_{gs}}{V_{gs} - V_T} \right) & V_T < V_{gs} < V_{pf} & 2.2a \\
I_{ds} &= I_{\text{dsfet}} - \xi (V_{gs} - V_{pf}) \tanh\left( \frac{\alpha V_{gs}}{V_{gs} - V_T} \right) & V_{pf} < V_{gs} < 0 & 2.2b \\
I_{ds} &= 0 & V_{gs} < V_T < V_{pf} & 2.2c
\end{align*}
\]

where \( V_T = V_{TO} + \gamma V_{ds} \). Here \( V_T \) is the modified pinch-off potential. This replaces \( V_{TO} \) in the Curtice model. \( \gamma \) describes the effective pinch-off when combined with \( V_{ds} \). \( I_{\text{ds}} \) is the saturation current at zero gate-source voltage.

2.1-3 The HEMT Advanced Curtice Model

The Advanced Curtice model [18] is modified in three ways: \( V_{TO} \) is converted to \( V_T \) from the Materka-Kacprzak model, \( \beta \) is converted to \( \beta_{\text{eff}} \) and the square-law term is
converted to an exponential parameter VGEXP. The HEMT advanced Curtice model equations are:

\[
\begin{align*}
I_{d_{\text{defet}}} &= \beta_{\text{eff}} V_{\text{gs}}^{\text{VGEXP}} \tanh(\alpha V_{ds}) (1 + \lambda V_{ds}) & \text{for } V_T < V_{gs} < V_{pf} \quad 2.3a \\
I_d &= I_{d_{\text{defet}}} - \beta_{\text{eff}} \xi (V_{gs} - V_{pf}) \tanh(\alpha V_{ds}) & \text{for } V_{pf} < V_{gs} < 0 \quad 2.3b \\
I_d &= 0 & \text{for } V_{gs} < V_T < V_{pf} \quad 2.3c
\end{align*}
\]

where \( \beta_{\text{eff}} = \beta / (1 + \mu_{\text{crit}} V_{gs}) \), and \( V_{gs} = V_{gs} - V_T. \mu_{\text{crit}} \) is the mobility parameter at the critical field for mobility degradation. \( \beta_{\text{eff}} \) predicts transconductance with respect to the gate-source voltage. The other parameters are same as the parameters defined in the two previous models.

### 2.2 Extracting The Model Parameters

A method is given here to find the parameters for the models in detail. First, the curve fitting technique is used to pick the parameters which bases on the character of the model equations. They are then become the initial values of the optimization. Second, an optimized algorithm[6] is used to reduce the error between the model and the measured data. The algorithm and programs are given in appendix A through C. Finally, a detailed optimization procedure is given.
2.2.1 Picking the Parameters Using Curve Fitting Technique

In this thesis, two kinds of measured data from a device which will be called Device C are used to model it: the drain current versus drain voltage characteristics for different gate voltages (fig 2.2) and the drain current versus gate voltage characteristics for a fixed drain to source voltage (fig 2.3). These will also be used to calculate the transconductance versus the gate voltage. A curve fitting technique is introduced to select the parameters based on these measured data.

In order to model a HEMT device, a good initial estimate is very important for the optimization process. The HEMT Curtice model is considered first as it is the simplest model. The other two models, which are more complex, can be constructed based on this model’s results.

A. Extracting The HEMT Curtice Model Parameters

From the basic DC relations between drain current and gate voltage, the parameter \( V_{TO} \) can be obtained directly. For large fixed \( V_{gs} \), \( \lambda \) is the slope of the \( I_{ds} \) versus \( V_{ds} \) curve when \( V_{ds} \) is large. For the Curtice HEMT model, the equation goes as \( (V_{gs} - V_{TO})^2 \) when \( V_{gs} \) is less than \( V_{pf} \) with \( V_{ds} \) held constant. So the curve of the square root of drain current versus gate voltage is plotted from the measured data in figure 2.4. This curve is used to obtain three parameters: a) \( V_{TO} \), b) \( V_{pf} \) can be found where the curve starts to deviate...
from a straight line. c) $\beta$ is obtained from:

$$\sqrt{I_{ds}} = \sqrt{\beta \tanh(\alpha V_{ds}) (1 + \lambda V_{ds}) (V_{gs} - V_{TO})} \quad 2.4$$

which gives the slope of the curve.

In addition, $\tanh(\alpha V_{ds})$ can be assumed to be 1 if $\alpha V_{ds}$ is large enough.

The way to get $\alpha$ is by a calculation to get an initial value and then by curve fitting. The function $\tanh(\alpha V_{ds})$ does not approach one if $V_{ds}$ is small. A value of $\alpha$ can be obtained by using the measured data points with $V_{gs}$ in the range of equation 2.1a and the values of $\beta$ and $\lambda$ calculated above. These are put into equation 2.4 to obtain $\alpha$. The characteristic curves using this value of $\alpha$ are compared to the experimental data and $\alpha$ is adjusted by trial and error to make the computed curves of $I_{ds}$ vs $V_{ds}$ match the actual ones as closely as possible.

The other two parameters, $\xi$ and $\psi$, are difficult to obtain directly from the measured data. So here, the original equations are examined again. From these it is seen that the equation 2.1a minus the measured data ( $I_{ds}$ in equation 2.1b ) should be equal to the last term in equation 2.1b ( call this the deviation current ). As all the parameters ( $\beta, \lambda, \alpha$ and $V_{TO}$ ) have been chosen for $I_{ds}$ above, the curve of the deviation current versus $V_{gs}$ is calculated from this and the measured data when $V_{ds}$ is constant. For equation 2.1b, there are three parameters, $V_{pt}$, $\xi$ and $\psi$, which determine the deviation current function of $V_{gs}$ ( when $V_{ds}$ is constant ). To get $\xi$ and $\psi$, the deviation current $I_{ds} - I_{dsfet}$ is first plotted from the data and the values of $\beta, \lambda, \alpha$ and $V_{TO}$. Next the second
Figure 2.2 The DC characteristics of HEMT C for different gate voltages.

Figure 2.3 The DC transfer characteristic of HEMT C when Vds=2.5v.
Figure 2.4 Squareroot of \( I_d \) versus \( V_{gs} \) for HEMT C at \( V_{ds}=2.5 \text{v} \) (eqn. 2.4)

Figure 2.5 Fitting data curves of HEMT C based on eqn 2.1b
term in equation 2.1b is plotted on the same graph. \( \xi \) and \( \psi \) are adjusted until the two curves match for small \( |V_{gs}| \). This kind of curve fitting is another way to estimate \( V_{pf} \).

B. Extracting The HEMT Advanced Curtice Model Parameters

These preceeding results are used to extract the parameters of the other two models. For the advanced Curtice, the remaining parameters \( \gamma, \mu_{crit} \) and \( VGEXP \) are hard to obtain, a quick and simple way is introduced here. Based on the relationship between the Curtice and the advanced Curtice model, it is convenient to simplify the HEMT advanced Curtice model to Curtice model. That is: \( \mu_{crit} \) is assumed 0.0 which lets \( \beta_{eff} \) equal \( \beta \), \( VGEXP \) is set to 2 and \( \gamma \) is set to 0.0. To start the process, the optimized values of the HEMT Curtice model are picked as the initial values of the advanced Curtice model.

C. Extracting The HEMT Materka-Kacprzak Model Parameters

Lastly, the Materka-Kacprzak model is considered. The values of the parameters for this case can been obtained from the combination of the other two models considered above, but some calculation is needed to transfer its parameters to this model. \( I_{ds} \times V_{T}^2 \) replaces \( \beta \), \( \alpha/(V_{gs}-V_{T}) \) replaces \( \alpha \) in the Curtice model, and \( V_{T} \) comes from the advanced Curtice model. The initial values of this model still come from the optimized results of
the other two models.

2.2.2 Optimization

Curve fitting is a good technique to directly and easily model the device, but the accuracy of the model is not satisfied. Furthermore, the parameters are not easy to adjust by hand and it will also cost lots of time. An optimization program DCMOD was written in C and the Levenberg-Marquardt method [6] is used to solve this problem (see appendix A for details).

The parameters are picked in accordance with the method in section 2.2, and then optimized based on the discussion in appendix A. DCMOD was written to optimize the parameters (see appendix B). The program DCMOD includes three parts: the main program, an optimization routine and a function part based on the HEMT model. As mentioned above, $V_{TO}$ and $V_{pf}$ come directly from the measured data. These two parameters can be used to decrease the time of optimization. First keeping $V_{TO}$ and $V_{pf}$ as constants and making the other parameters changeable, a group of new values is obtained. Next using these new values and letting $V_{pf}$ be variable, a more refined set is obtained. Lastly, all the parameters are put into the optimization program with their current values, and a final optimization is run. Such steps should make $V_{TO}$ and $V_{pf}$ close to their optimum values and avoid the problem of falling into a local minimum which is unreasonably far from the graphical approximation.
2.3 Results

There are two kinds of error which can be used to analyze the results, the absolute error and the relative error. In this work the relative error is picked. This error is defined as:

\[
Err = \frac{1}{N} \sum_{i=1}^{N} \left( \frac{I_{ds\text{(dat)i}} - I_{ds\text{(mod)i}}}{I_{ds\text{(dat)i}}} \right)^2
\]

where \( I_{ds\text{(dat)i}} \) is the measured drain current and \( I_{ds\text{(mod)i}} \) is the drain current calculated from the model.

Note that the error calculation for the HEMT Curtice model is very large when \( V_{gs} \) is at -0.8V and -1.0V as the data are much smaller than the calculated values of the model. So the error results in this thesis ignore these points for all the models in order to give a valid comparison. Another point is that the calculated errors given in the tables will be large values because the drain current \( I_d \) is a small value at each bias point.

The proposed method has been applied to extract DC parameters for a 0.25 micron gate length HEMT device. The drain current versus drain-source voltage curves created using the HEMT Curtice model are illustrated in fig 2.6. The measured \( I_d \) (dotted lines) is compared to the values calculated using the model with the optimized parameters (solid lines). Notice that the HEMT Curtice model fits the data well except at low gate voltage levels. The initial and optimized parameters are presented in Table 2.1. The error is 0.59
after optimization. Figure 2.7 gives the transconductance calculated from this model where it is plotted along with the measured data.

In figure 2.8 is presented the I-V curves of the HEMT Materka-Kacprzak model. Compared to the HEMT Curtice model, the data fit is improved at low gate voltages. The threshold voltage \( V_{TO} \) of the model is decreased. However, at higher gate voltages, it becomes worse. Table 2.2 gives the parameters of this model for the initial and optimized values. The error is less than the Curtice model. It is 0.57. Figure 2.9 gives the transconductance calculated using this model.

The best results obtained here are for the HEMT advanced Curtice model. Figure 2.10 gives the current versus voltage curves. The curves of the measured data and the calculated values of this model are close together for higher gate voltage levels as well as lower ones. The error is 0.56 after optimization, which is the smallest error in these three models. The parameters are presented in table 2.3. The calculated transconductance is presented in figure 2.11.

### 2.4 Further Applications of the Modeling Method

With the method introduced above, two other HEMT devices were modeled. Device A is a two fingered HEMT with a 50 micron gate width and device B has four fingers, also with a 50 micron gate width and 0.25 micron length. This data includes only I-V characteristics of the devices, so the results are shown in these kinds of curves.
For the two fingered device (A), fig 2.12 gives the optimized HEMT Curtice model. Fig 2.13 and fig 2.14 are for the HEMT Materka-Kacprzak model and the HEMT advanced Curtice model respectively. Device B is considered in fig 2.15, fig 2.16 and fig 2.17. For these two devices, the curves are similar to the device considered above. Error values are slightly higher because fewer data points were available. For lower gate voltages, the Materka-Kacprzak model is better than the Curtice model; the best fitting of the models is the HEMT advanced Curtice model. The parameters are given in table 2.4, 2.5 and 2.6 for each.
Table 2.1 The initial parameters and final values after optimization using the HEMT Curtice model.

<table>
<thead>
<tr>
<th>parameters (dimension)</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$ (A/V²)</td>
<td>0.024</td>
<td>0.020</td>
</tr>
<tr>
<td>$\alpha$ (1/V)</td>
<td>2.62</td>
<td>2.62</td>
</tr>
<tr>
<td>$\lambda$ (1/V)</td>
<td>0.18</td>
<td>0.18</td>
</tr>
<tr>
<td>$\xi$ (A/V^5)</td>
<td>0.084</td>
<td>0.061</td>
</tr>
<tr>
<td>$\psi$</td>
<td>0.6</td>
<td>1.67</td>
</tr>
<tr>
<td>$V_{pf}$ (V)</td>
<td>-0.40</td>
<td>-0.40</td>
</tr>
<tr>
<td>$V_{to}$ (V)</td>
<td>-1.05</td>
<td>-1.00</td>
</tr>
<tr>
<td>error</td>
<td>0.73</td>
<td>0.59</td>
</tr>
</tbody>
</table>

Table 2.2 The initial parameters and final values after optimization using the HEMT Materka-Kacprzak model.

<table>
<thead>
<tr>
<th>parameters (dimension)</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{ds}$ (A)</td>
<td>0.02</td>
<td>0.031</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>-0.083</td>
<td>-0.14</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>1.0</td>
<td>0.91</td>
</tr>
<tr>
<td>$\psi$</td>
<td>3.4</td>
<td>2.40</td>
</tr>
<tr>
<td>$\xi$ (A/V^6)</td>
<td>1.44</td>
<td>0.096</td>
</tr>
<tr>
<td>$V_{pf}$ (V)</td>
<td>-0.28</td>
<td>-0.29</td>
</tr>
<tr>
<td>$V_{to}$ (V)</td>
<td>-0.78</td>
<td>-0.63</td>
</tr>
<tr>
<td>error</td>
<td>0.69</td>
<td>0.56</td>
</tr>
</tbody>
</table>
Table 2.3 The initial parameters and final values after optimization using the HEMT advanced Curtice model.

<table>
<thead>
<tr>
<th>parameters ( dimension )</th>
<th>initial</th>
<th>final(optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$ ( $A/V^{VGEXP}$ )</td>
<td>0.020</td>
<td>0.063</td>
</tr>
<tr>
<td>$\alpha$ ( 1/V )</td>
<td>2.62</td>
<td>2.81</td>
</tr>
<tr>
<td>$\lambda$ ( 1/V )</td>
<td>0.18</td>
<td>-0.057</td>
</tr>
<tr>
<td>$\mu_{crit}$ ( 1/V )</td>
<td>0</td>
<td>0.84</td>
</tr>
<tr>
<td>$VGEXP$</td>
<td>2</td>
<td>2.309</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>0</td>
<td>-0.083</td>
</tr>
<tr>
<td>$\xi$ ( $A/V^w$ )</td>
<td>2.03</td>
<td>2.40</td>
</tr>
<tr>
<td>$\psi$</td>
<td>2.67</td>
<td>4.57</td>
</tr>
<tr>
<td>$V_{pf}$ ( V )</td>
<td>-0.40</td>
<td>-0.178</td>
</tr>
<tr>
<td>$V_{to}$ ( V )</td>
<td>-1.00</td>
<td>-0.78</td>
</tr>
<tr>
<td>error</td>
<td>0.59</td>
<td>0.557</td>
</tr>
</tbody>
</table>

Table 2.4 The parameters of the HEMT Curtice model for two fingered 50 micron gate width and 0.25 micron gate length HEMT.

<table>
<thead>
<tr>
<th>parameters</th>
<th>Initial</th>
<th>final(optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$ ( $A/V^2$ )</td>
<td>0.029</td>
<td>0.11</td>
</tr>
<tr>
<td>$\alpha$ ( 1/V )</td>
<td>3.15</td>
<td>3.15</td>
</tr>
<tr>
<td>$\lambda$ ( 1/V )</td>
<td>0.26</td>
<td>0.24</td>
</tr>
<tr>
<td>$\xi$ ( $A/V^{w+1}$ )</td>
<td>0.064</td>
<td>0.32</td>
</tr>
<tr>
<td>$\psi$</td>
<td>0.6</td>
<td>4.79</td>
</tr>
<tr>
<td>$V_{pf}$ ( V )</td>
<td>-0.65</td>
<td>-0.66</td>
</tr>
<tr>
<td>$V_{to}$ ( V )</td>
<td>-1.3</td>
<td>-1.26</td>
</tr>
<tr>
<td>error</td>
<td>1.894</td>
<td>0.53</td>
</tr>
</tbody>
</table>
Table 2.5 The parameters of the HEMT Materka-Kaczprzak model for two fingered 50 micron gate width HEMT.

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final(optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{ds}$ (A)</td>
<td>0.029</td>
<td>0.033</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>0.783</td>
<td>-0.18</td>
</tr>
<tr>
<td>$\alpha$ (1/V)</td>
<td>3.15</td>
<td>1.028</td>
</tr>
<tr>
<td>$\xi$ (A/V^3)</td>
<td>0.57</td>
<td>0.064</td>
</tr>
<tr>
<td>$\psi$</td>
<td>0.26</td>
<td>2.21</td>
</tr>
<tr>
<td>$V_{pf}$ (V)</td>
<td>-0.45</td>
<td>-0.47</td>
</tr>
<tr>
<td>$V_{to}$ (V)</td>
<td>-0.8</td>
<td>-0.710</td>
</tr>
<tr>
<td>error</td>
<td>1.51</td>
<td>0.52</td>
</tr>
</tbody>
</table>

Table 2.6 The parameters for the HEMT advanced Curtice model for two fingered 50 micron gate width HEMT.

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final(optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$ (A/V^{VGEXP})</td>
<td>0.011</td>
<td>0.034</td>
</tr>
<tr>
<td>$\mu_{crit}$ (1/V)</td>
<td>0</td>
<td>1.027</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>-0.18</td>
<td>-0.091</td>
</tr>
<tr>
<td>VGEXP</td>
<td>2</td>
<td>1.38</td>
</tr>
<tr>
<td>$\lambda$ (1/V)</td>
<td>0.24</td>
<td>0.052</td>
</tr>
<tr>
<td>$\alpha$ (1/V)</td>
<td>3.15</td>
<td>2.99</td>
</tr>
<tr>
<td>$\psi$</td>
<td>5.79</td>
<td>6.54</td>
</tr>
<tr>
<td>$\xi$ (A/V^3)</td>
<td>5.095</td>
<td>-4.02</td>
</tr>
<tr>
<td>$V_{pf}$ (V)</td>
<td>-0.47</td>
<td>-0.43</td>
</tr>
<tr>
<td>$V_{to}$ (V)</td>
<td>-0.71</td>
<td>-0.78</td>
</tr>
<tr>
<td>error</td>
<td>0.56</td>
<td>0.42</td>
</tr>
</tbody>
</table>
Table 2.7 The parameters for the HEMT Curtice model for four fingered 50 micron gate width HEMT.

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final(optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$ (A/V^2)</td>
<td>0.029</td>
<td>0.010</td>
</tr>
<tr>
<td>$\alpha$ (1/V)</td>
<td>3.15</td>
<td>2.84</td>
</tr>
<tr>
<td>$\lambda$ (1/V)</td>
<td>0.26</td>
<td>0.23</td>
</tr>
<tr>
<td>$\xi$ (A/V^{w+1})</td>
<td>0.064</td>
<td>0.20</td>
</tr>
<tr>
<td>$\psi$</td>
<td>0.6</td>
<td>5.02</td>
</tr>
<tr>
<td>$V_{pf}$ (V)</td>
<td>-6.5</td>
<td>-0.67</td>
</tr>
<tr>
<td>$V_{to}$ (V)</td>
<td>-1.3</td>
<td>-1.23</td>
</tr>
<tr>
<td>error</td>
<td>2.93</td>
<td>0.57</td>
</tr>
</tbody>
</table>

Table 2.8 The parameters for the HEMT Materka-Kacprzak model for four fingered 50 micron gate width HEMT.

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final(optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{ds}$ (A)</td>
<td>0.029</td>
<td>0.031</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>0.783</td>
<td>-0.17</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>3.15</td>
<td>0.89</td>
</tr>
<tr>
<td>$\xi$ (A/V^w)</td>
<td>0.57</td>
<td>0.047</td>
</tr>
<tr>
<td>$\psi$</td>
<td>0.26</td>
<td>2.19</td>
</tr>
<tr>
<td>$V_{pf}$ (V)</td>
<td>-0.45</td>
<td>-0.49</td>
</tr>
<tr>
<td>$V_{to}$ (V)</td>
<td>-0.8</td>
<td>-0.66</td>
</tr>
<tr>
<td>error</td>
<td>1.62</td>
<td>0.56</td>
</tr>
</tbody>
</table>
Table 2.9 The parameters for the HEMT advanced Curtice model for four fingered 50 micron gate width HEMT.

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta \ (A/V^{\text{VGEXP}})$</td>
<td>0.010</td>
<td>0.037</td>
</tr>
<tr>
<td>$\mu_{\text{crit}} \ (1/V)$</td>
<td>0</td>
<td>1.24</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>-0.172</td>
<td>-0.095</td>
</tr>
<tr>
<td>VGEXP</td>
<td>2</td>
<td>1.53</td>
</tr>
<tr>
<td>$\lambda \ (1/V)$</td>
<td>0.23</td>
<td>0.038</td>
</tr>
<tr>
<td>$\alpha \ (1/V)$</td>
<td>2.84</td>
<td>2.70</td>
</tr>
<tr>
<td>$\psi$</td>
<td>6.023</td>
<td>7.98</td>
</tr>
<tr>
<td>$\xi \ (A/V^v)$</td>
<td>3.387</td>
<td>-24.88</td>
</tr>
<tr>
<td>$V_{\text{pf}} \ (V)$</td>
<td>-0.49</td>
<td>-0.47</td>
</tr>
<tr>
<td>$V_{\text{to}} \ (V)$</td>
<td>-0.66</td>
<td>-0.75</td>
</tr>
<tr>
<td>error</td>
<td>0.566</td>
<td>0.44</td>
</tr>
</tbody>
</table>
Figure 2.6 The DC characteristics of the HEMT Curtice model (solid line) and the measured data (dashed line) for Device C.

Figure 2.7 The transconductance of the HEMT Curtice model (solid line) and the measured data (dashed line) for Device C.
Figure 2.8 The DC characteristics of the HEMT Materka-Kacprzak model (solid line) and the measured data (dashed line) for Device C.

Figure 2.9 The transconductance of the HEMT Materka-Kacprzak model (solid line) and the measured data (dashed line) for Device C.
Figure 2.10 The DC characteristics of the HEMT-Advanced Curtice model (solid line) and the measured data (dashed line) for Device C.

Figure 2.11 The transconductance of the HEMT Advanced Curtice model (solid line) and the measured data (dashed line) for Device C.
Figure 2.12 The DC characteristics of the HEMT Curtice model (solid line) and the measured data (dashed line) for Device A.

Figure 2.13 The DC characteristics of the HEMT Materka-Kacprzak model (solid line) and the measured data (dashed line) for Device A.
Figure 2.14 The DC characteristics of the HEMT Advanced Curtice model (solid line) and the measured data (dashed line) for Device A.

Figure 2.15 The DC characteristics of the HEMT Curtice model (solid line) and the measured data (dashed line) for Device B.
Figure 2.16 The DC characteristics of the HEMT Materka-Kacprzak model (solid line) and the measured data (dashed line) for Device B.

Figure 2.17 The DC characteristics of the HEMT Advanced Curtice model (solid line) and the measured data (dashed line) for Device B.
CHAPTER 3

AC MODELING

This chapter first presents a complete AC model for GaAs HEMTs. Then the parameters' extraction and the optimized results of the AC model are presented in detail. The measured S-parameters for two HEMT devices are used to optimize the model and the error in matching the measured data is calculated. LIBRA, the software, is used to show the curves of S-parameters in Smith chart form.

3.1 The AC Equivalent Circuit for A GaAs Field Effect Transistor

AC small signal modeling is considered as linear because the circuit is at a fixed operating point with small voltage and current variations. To study the AC characteristics of HEMT devices, an accurate, sophisticated equivalent circuit is necessary. Here, the HEMT AC model used by (Hughes et al) [7] is considered (fig3.1).

The model is divided into two sections: the intrinsic device and extrinsic elements dependent on the device's physical structure. The extrinsic elements of this model include three contact pad capacitances ($C_{pgs}$, $C_{pgd}$ and $C_{pds}$), three parasitic inductances ($L_g$, $L_s$ and $L_d$) and three parasitic resistances ($R_g$, $R_s$ and $R_d$). The intrinsic components consist of seven elements: three capacitances ($C_{gs}$, $C_{gd}$ and $C_{ds}$), three resistances ($R_{gs}$, $R_{gd}$ and $G_{ds}$) and a voltage controlled current source with parameters, $g_m$ and $\tau$. The
major distinction between the intrinsic and the extrinsic elements is that the extrinsic components are bias invariant, while the intrinsic ones are bias dependent. So the intrinsic elements are more complex and critical than the extrinsic ones in the model and they are hard to predict. In the next section, a method is introduced to extract the parameters from the geometry of the device and the measured data.

3.2 Extraction of The AC Model Elements

In order to model a HEMT device, good initial values for the elements are needed for the optimization process. Regularly, some kinds of measurements are used to obtain the parasitic resistances. The other extrinsic elements are ignored for the initiation of estimation as they have only a small roles in the model (see chapter 4 for details). Here, equations are given to extract them based on the device material and geometry. The intrinsic elements can be extracted from S-parameters if the extrinsic components are de-embedded.

3.2-1. Extraction of The Extrinsic Elements

As mentioned above, there are nine parasitic elements in the extrinsic section. They can be calculated by studying the FET's physics and geometry [3, 8] (fig 3.2). It should be mentioned that fig 3.2 is the simplest geometrical structure in an FET. There is only one gate finger shown in the figure. Multi-fingered devices have more complex
structures, so the equations are given based on the multi-fingered case:

\[ C_{psg} = \frac{m \varepsilon Z_G h_B}{L_{SG}} \]  

\[ C_{pgd} = \frac{m \varepsilon Z_G h_B}{L_{GD}} \]  

\[ C_{pds} = \frac{m \varepsilon Z_G h_D}{L_{GD} + L_{SG} + L_G} \]  

\[ L_G = \frac{\mu_0 dZ_G}{m^2 L_G} \]  

\[ L_B = \frac{\mu_0 dZ_G}{3m^2 L_G} \]
\[ L_d = \frac{\mu_d dZ_g}{2m^2 L_g} \]  

\[ R_g = \frac{\rho Z_g}{3m^2 hL_g} \]  

\[ R_s = \frac{L_{GD}}{Wq\mu_1 n_{eff} - \frac{I\mu_1}{V_{s1}}} \]  

\[ R_d = \frac{L_{GD}}{Wq\mu_1 n'_{eff} - \frac{I\mu_1}{V_{s1}}} \]  

where

\[ I = Wq n_a \left( \frac{dV_c}{dx} \right) - \frac{\mu_1}{V_{s1}} \left( \frac{dV_c}{dx} \right) \]
The right side of the equations (3.1 to 3.9) includes the electrical characteristics of the semiconductor material, physical structure, dimensions and doping of the HEMT. The symbols in these equations are listed below:

\( \mu_0 \) : permeability of free space

\( \mu_1 \) : low field electron mobility in the channel

\( \varepsilon \) : permittivity of free space

\( d \) : depletion depth

\( m \) : the number of fingers

\( Z_G \) : the gate width

\( L_m \) : the metallurgical gate length

\( \rho \) : resistivity of the gate strip

\( h \) : the height of the gate strip

\( q \) : electron charge

\( L_{GD} \) : the gate drain separation

\( L_{SG} \) : the gate source separation

\( h_s \) : the effective height of the metal at source and gate

\( h_d \) : the effective height of the metal at drain and source

\( h_G \) : the effective height of the metal at gate and drain

\( W \) : the thickness of the N-layer under the gate

\( n_{\text{eff}} \) and \( n'_{\text{eff}} \) : effective carrier concentration for Rs and Rd separately

\( \frac{dV_e}{dx} \) : the field inside the channel

\( v_{st} \) : Scattering - limited velocity of electron in the material
3.2-2. Extraction of The Intrinsic Elements

With the extraction of extrinsic elements of the small signal model, an algorithm has been set up to obtain intrinsic parameters. All parameters are based on a set of measured S-parameters at a given bias point. A procedure that extracts the intrinsic parameter elements for the circuit of fig 3.1 is:

1) Convert S-parameters to the Y-parameters
2) Remove $C_{pgs}$, $C_{pgd}$ and $C_{pds}$ from the Y-parameters
3) Convert these Y-parameters to the Z-parameters
4) Remove $L_g$, $L_s$, $L_d$ and $R_g$, $R_s$, $R_d$ from the Z-parameters
5) Convert the modified Z-parameters back to the Y-parameters.

The final modified Y-parameters are in the form of a 2x2 Y-matrix. From the new Y-parameters, one is able to compute the intrinsic elements by applying the following equations [7]:

$$Y_{gs} = Y_{11} + Y_{12} \quad 3.10$$

$$Y_{gd} = -Y_{12} \quad 3.11$$

$n_e$: the 2-dimensional electron gas density.
where \( Y_{gs}, Y_{gd}, Y_{ds} \) and \( Y_{gm} \) are the Y-parameters defined in fig 3.3.

\[
Y_{ds} = Y_{22} + Y_{12} \quad 3.12
\]

\[
Y_{gm} = Y_{21} - Y_{12} \quad 3.13
\]

\[
C_{gs} = \frac{-1}{\operatorname{Im}(\frac{1}{Y_{gs}})} \omega \quad 3.14
\]

\[
R_{gs} = \operatorname{Re}(\frac{1}{Y_{gs}}) \quad 3.15
\]

\[
C_{gd} = \frac{-1}{\operatorname{Im}(\frac{1}{Y_{gd}})} \omega \quad 3.16
\]

\[
R_{gd} = \operatorname{Re}(\frac{1}{Y_{gd}}) \quad 3.17
\]
\[ C_{ds} = \frac{\text{Im}(Y_{ds})}{\omega} \]  
\[ G_{ds} = \text{Re}(Y_{ds}) \]

\[ \tau = -\frac{\text{phase}[Y_{gm} \times (1 + j\omega \tau_{gm})]}{\omega} \]

\[ g_m = \text{mag}[Y_{gm} \times (1 + j\omega \tau_{gm})] \]

where \( \tau_{gm} = R_{gs} C_{gs} \).

3.2-3. Choosing The Initial Parameters for The Optimization

It should be mentioned that there are no real HEMT devices available to measure for this work. All work done is based on the measured S-parameter data obtained from outside, so the method of picking the initial values cannot rely on special measurements as in [7]. The low frequency S parameters are used first because the parallel capacitances and series inductances will have less contribution to them. At first these parasitic reactive elements are assumed to be zero. This allows other evaluation of Z parameters directly since step 2 in section 3.2-2 is not needed. In step 4, the inductances are assumed to be zero while the initial parasitic resistances are chosen based on typical values and
consideration of the geometry of the device and equations 3.7 to 3.9. After step 4 and step 5, the initial values for the intrinsic elements are calculated by equations 3.10 to 3.21. Then these values are put into the software for minimization. At this stage, it is found that resistances come out negative. Therefore, the extrinsic parameters were given some rough adjustment by trial and error before optimization to lower the error (calculated by equations 3.22 and 3.23 below) to acceptable limits. In this stage, the S parameter data for all frequencies is used. After these adjustments are finished, the result is the initial column in table 3.1.

3.3 Optimization and Results

The data is measured at different frequencies $\omega_k$, $k = 1, \ldots, N$. The measurement were taken every 0.5 GHz from 0.5 GHz to 26.5 GHz and therefore $N$ is 53. The S-parameters are measured based on a two port network, so they form a 2x2 matrix. To analyze optimization results of the HEMT equivalent circuit, we must first define an error function between the measured data and the values calculated for a model. The definition of the error in the S-parameters will be given by [7]:

$$ Err_{S_{ij}}(k) = \frac{1}{N} \sum_{k=1}^{N} \left| \frac{S_{dat_{ij}}(k) - S_{mod_{ij}}(k)}{S_{dat_{ij}}(k)} \right| $$  \hspace{1cm} (3.22)

where $Err_{S_{ij}}$ is the error in $S_{ij}$, $k$ is the kth data point. $N$ is the number of data points used,
Sdat,ij is the measured $S_{ij}$ and Smod,ij is the calculated value of $S_{ij}$ from the model under consideration. The total error of the device (ErrS) is the average error of the four S-parameters given by:

$$ErrS = \frac{ErrS_{11} + ErrS_{12} + ErrS_{21} + ErrS_{22}}{4}$$

With the initial estimation and calculation of the model's elements, the procedure of optimization is made in the same way as the method introduced in appendix A; the programs used are in appendix B and D. The error function is set up by equations 3.22 and 3.23 based on the model elements and the measured data.

For device A (the two fingered device), the initial values of the parameters are picked with the method introduced as before. Then table 3.1 gives the parameters and the optimized results:

- The total error (ErrS) from 1.75% reduces to 1.06% after optimization
- Error of $S_{11}$ goes from 1.4% to 0.68%
- $S_{12}$ from 3.2% to 2.45%
- $S_{21}$ from 1.35% to 0.48%
- Error of $S_{22}$ drops from 1% to 0.64%.

The optimized results are presented graphically in fig 3.4 though fig 3.6 in Smith chart form, where the calculated and measured data are both plotted. LIBRA, a CAD tool designed to analyze and simulate microwave devices and circuits [15], was used to
create these figures.

Device B (the four fingered device) has different parameters than device A (the two fingered HEMT). The effective area of the gate is doubled in Device B, and so are the drain and the source areas. Therefore, the parasitic resistances were picked as half the values of device A for the initial guess. The intrinsic elements are also calculated from the measured S-parameters using the lower frequency data as before. And the rough adjustment technique is again used on the extrinsic parameters to reduce the error.

Table 3.2 gives the initial values and optimized results for device B:

- The total error ($\text{Err}_S$) decreases from 1.6% before to 0.74% after optimization
- Error of $S_{11}$ goes from 2.26% to 0.66%
- $S_{12}$ from 0.95% to 1.06%
- $S_{21}$ from 1.17% to 0.52%
- Error of $S_{22}$ drops from 2.11% to 0.72%.

The comparison of model calculations and the measured data are given in figures 3.7 though 3.9 in Smith chart form.
Table 3.1 The initial parameters and final values after optimization for the two-fingered device

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{pgs}(fF)$</td>
<td>0.8</td>
<td>0.28</td>
</tr>
<tr>
<td>$C_{pds}(fF)$</td>
<td>1.0</td>
<td>0.67</td>
</tr>
<tr>
<td>$C_{pgd}(fF)$</td>
<td>1.0</td>
<td>0.89</td>
</tr>
<tr>
<td>$L_s(pH)$</td>
<td>35</td>
<td>47.69</td>
</tr>
<tr>
<td>$L_s(pH)$</td>
<td>1.0</td>
<td>1.147</td>
</tr>
<tr>
<td>$L_d(pH)$</td>
<td>26.8</td>
<td>33.02</td>
</tr>
<tr>
<td>$R_s(ohm)$</td>
<td>0.98</td>
<td>1.51</td>
</tr>
<tr>
<td>$R_s(ohm)$</td>
<td>0.95</td>
<td>1.50</td>
</tr>
<tr>
<td>$R_s(ohm)$</td>
<td>2.35</td>
<td>1.55</td>
</tr>
<tr>
<td>$C_{pg}(fF)$</td>
<td>13.88</td>
<td>13.78</td>
</tr>
<tr>
<td>$R_{pg}(ohm)$</td>
<td>42.26</td>
<td>45.75</td>
</tr>
<tr>
<td>$C_{pg}(fF)$</td>
<td>79.69</td>
<td>79.48</td>
</tr>
<tr>
<td>$R_{f}(ohm)$</td>
<td>1.10</td>
<td>1.56</td>
</tr>
<tr>
<td>$G_m(mS)$</td>
<td>29.23</td>
<td>29.13</td>
</tr>
<tr>
<td>$\tau(psec)$</td>
<td>1.16</td>
<td>1.09</td>
</tr>
<tr>
<td>$C_{gd}(fF)$</td>
<td>38.00</td>
<td>3.82</td>
</tr>
<tr>
<td>$G_{gd}(mS)$</td>
<td>2.95</td>
<td>3.10</td>
</tr>
<tr>
<td>ERRS$_{11}$(%)</td>
<td>1.41</td>
<td>0.68</td>
</tr>
<tr>
<td>ERRS$_{12}$(%)</td>
<td>3.21</td>
<td>2.45</td>
</tr>
<tr>
<td>ERRS$_{21}$(%)</td>
<td>1.35</td>
<td>0.48</td>
</tr>
<tr>
<td>ERRS$_{22}$(%)</td>
<td>1.02</td>
<td>0.64</td>
</tr>
<tr>
<td>ERRS(%)</td>
<td>1.75</td>
<td>1.06</td>
</tr>
</tbody>
</table>
Table 3.2 The initial parameters and final values after optimization for the four fingered device

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_{pgs}$ (fF)</td>
<td>0.1</td>
<td>0.11</td>
</tr>
<tr>
<td>$C_{psd}$ (fF)</td>
<td>0.2</td>
<td>0.16</td>
</tr>
<tr>
<td>$C_{pgd}$ (fF)</td>
<td>0.2</td>
<td>0.22</td>
</tr>
<tr>
<td>$L_e$ (pH)</td>
<td>40.0</td>
<td>48.38</td>
</tr>
<tr>
<td>$L_s$ (pH)</td>
<td>0.1</td>
<td>0.13</td>
</tr>
<tr>
<td>$L_d$ (pH)</td>
<td>40.0</td>
<td>33.04</td>
</tr>
<tr>
<td>$R_{g}$ (ohm)</td>
<td>0.2</td>
<td>0.26</td>
</tr>
<tr>
<td>$R_s$ (ohm)</td>
<td>0.5</td>
<td>0.48</td>
</tr>
<tr>
<td>$R_d$ (ohm)</td>
<td>0.5</td>
<td>0.54</td>
</tr>
<tr>
<td>$C_{pgd}$ (fF)</td>
<td>28.13</td>
<td>28.22</td>
</tr>
<tr>
<td>$R_{gd}$ (ohm)</td>
<td>21.35</td>
<td>21.33</td>
</tr>
<tr>
<td>$C_{gs}$ (fF)</td>
<td>140.26</td>
<td>140.30</td>
</tr>
<tr>
<td>$R_{g}$ (ohm)</td>
<td>1.07</td>
<td>1.41</td>
</tr>
<tr>
<td>$G_m$ (mS)</td>
<td>56.08</td>
<td>56.09</td>
</tr>
<tr>
<td>$\tau$ (psec)</td>
<td>1.66</td>
<td>1.42</td>
</tr>
<tr>
<td>$C_{gd}$ (fF)</td>
<td>54.01</td>
<td>54.01</td>
</tr>
<tr>
<td>$G_{gd}$ (mS)</td>
<td>5.92</td>
<td>5.92</td>
</tr>
<tr>
<td>ERRS$_{11}$ (%)</td>
<td>2.26</td>
<td>0.66</td>
</tr>
<tr>
<td>ERRS$_{12}$ (%)</td>
<td>0.95</td>
<td>1.06</td>
</tr>
<tr>
<td>ERRS$_{21}$ (%)</td>
<td>1.17</td>
<td>0.52</td>
</tr>
<tr>
<td>ERRS$_{22}$ (%)</td>
<td>2.11</td>
<td>0.72</td>
</tr>
<tr>
<td>ERRS (%)</td>
<td>1.62</td>
<td>0.74</td>
</tr>
</tbody>
</table>
Figure 3.1 A complete small signal model of a HEMT. Here $G_m = g_m \exp(-j\omega t)$. 
Figure 3.2 The extrinsic geometry of a HEMT

Figure 3.3 The intrinsic structure in Y parameters of the HEMT model
Figure 3.4 S parameters S11 and S22: the calculations from the model and the measured data for Device A in Smith chart form.
Figure 3.5 S parameters S12: the calculations from the model and the measured data for Device A in Smith chart form.
Figure 3.6  S parameters S21: the calculations from the model and the measured data for Device A in Smith chart form.
Figure 3.7  S parameters $S_{11}$ and $S_{22}$: the calculations from the model and the measured data for Device B in Smith chart form.
Figure 3.8  S parameters S12: the calculations from the model and the measured data for Device B in Smith chart form.
Figure 3.9 S parameters S21: the calculations from the model and the measured data for Device B in Smith chart form.
CHAPTER 4

THE SIMPLIFICATION OF THE AC MODEL

This chapter examines possible simplifications of the AC model discussed in the last chapter. The analysis of each element of the model is presented for two different HEMT devices: one has two gate fingers (named device A) and the other has four fingers (named device B). Two simplified AC models are derived based on different considerations. The optimized results of the modeling are given at the end of the chapter.

4.1 The Weight of Each Element in The Model

There are sixteen elements and seventeen parameters in the AC small signal model. They are divided into two sections: extrinsic and intrinsic elements, as mentioned in chapter 3. The extrinsic elements include all parasitic parameters: $C_{pgs}$, $C_{pgd}$, $C_{pda}$, $L_e$, $L_s$, $L_d$, $R_e$, $R_s$ and $R_g$. The intrinsic ones include $C_{gs}$, $C_{gd}$, $C_{ds}$, $R_j$, $R_{gd}$, $G_{ds}$ and a voltage controlled current source which has two parameters: $g_m$ and $\tau$. Each parameter in the model affects the error with a different weight. Here, a method is used to explore this. Then the method is applied to the two devices.
4.1-1 Examining Parameter Sensitivity

With the optimized results obtained in chapter 3, an analysis of the importance of each element in the model is easily done. A parameter is picked to sweep through a range of values (from zero to double its optimized value for most parameters). Meanwhile, the other elements keep their optimized values. The higher and lower frequencies, 1 GHz and 26 GHz, are used to show the changing of the error as the parameter is changed. The error curves are calculated by comparing the results to the measured data in each case (see equations 3.21 and 3.22). Care is taken that the swept range covers the parameter values having minimum error. The scaling of an element value is normalized by dividing by the optimized value given in the last chapter, so the reference point is always one. This gives information about the sensitivity of the S-parameter to the element being varied. If there is little change in the error, it indicates that the particular element may be eliminated or at least set automatically to a nominal value. For each device, the extrinsic elements are considered first, then the intrinsic ones.

Here, two points should be mentioned. First, only two frequency values are picked to show the results of varying parameters. On the other hand optimization is over the whole frequency range for the measured data, so the optimized values are not necessarily the minimum error values at every frequency. Second, these curves are constructed to express the error change characteristics versus the values of the elements, so to pick the exact values from the curves for minimization purposes does not make sense. The only purpose of this analysis is to show the importance of the elements, with the view to see
which elements are critical to the model and which can be left out. For this, the curves are helpful.

4.1-2 The Two Fingered Device

A. The Extrinsic Elements

The extrinsic elements include three capacitances, three inductances and three resistances. Among the capacitances:

- The errors due to \( C_{p_g} \) go down as the capacitance value tends to zero for both frequencies. The error at the upper frequency is smaller than for the lower frequency in the swept range, but the error increases faster for the upper one. The curves are shown in fig. 4.1.

- For \( C_{p_d} \) (fig 4.2), the minimum error is at \( C = 0 \) for 1 GHz and at some point larger than the reference value for 26 GHz.

- The minimum error for \( C_{p_s} \) is unchanged for the lower frequency and occurs when \( C \) is at zero for the upper frequency, as shown in Fig 4.3.

For the inductances:

- The performance of \( L_s \) is similar to \( C_{p_s} \) (fig 4.4). The errors decrease as the value goes down for both frequencies.

- \( L_d \) and \( L_s \) are similar curves (fig 4.5 and fig 4.6). The minimum error is near the reference value for the upper frequency curve and is almost constant at the lower frequency.
The error curves of the parasitic resistances are:

- The curves of $R_s$ are shaped as expected (fig 4.7). For both frequency curves, the minimum error is around the reference value.

- The $R_d$ curves shows unusual behavior (fig 4.8), the minimum error point shifts very much to the right of the optimization point.

- From fig 4.9, when $R_s$ is scanned from 0 to double the optimized value of the element, the error for the lower frequency curve goes down slowly as $R_s$ increases and the minimum error point for the higher frequency curve is shifted to the right of the reference point.

**B. The Intrinsic Elements**

The intrinsic elements are usually assumed independent of frequency in a FET model; this means their values are unchanged when the frequency is changed. From figures 4.10 through 4.17, it is evident that the minimum error point in each case almost matches the overall optimized value for both frequencies. The exceptions are the two resistors $R_{gs}$ and $R_{gd}$.

**4.1-3 The Four fingered device**

Figures 4.18 through 4.34 give the same kind of curves for the four fingered device (Device B). As this device has two more gate fingers, the structure is more complex.
The results show that there are some similarities in the characteristics of the two devices. First, the slopes of the curves are similar at 1 GHz and 26 GHz. Second, the slope is large for 26 GHz. Third, the intrinsic elements show that they are optimum at both frequencies. From the curves, it is seen the minimum error is around the reference point (normalized value equals one). The main difference shown is that there is a larger error difference between the upper frequency and lower in the four fingered device than in the two fingered.

4.2 The Simplified HEMT AC Model and Optimized Results

It is clear from figures 4.1 though 4.34 that each element affects the error of the model in its own way. Some curves have a small slope but some have a steep slope. This means that some elements are more important than the others. The important elements are in the intrinsic section, except \( L_e \). In order to simplify the model, elements which do not result in a large error when they are changed are considered for removal because they are not expected to affect the error too much. Here, based on the optimized results from the last chapter and analyzed above, two simplified models are considered.

4.2-1 The First Simplified Model

From the results above, it is clear that the eight extrinsic and two intrinsic elements of the AC model can be removed. The figures 4.1 through 4.34 show that most extrinsic
elements and the two intrinsic resistances will not increase the error very much if they are left out. Here, \( L_d, L_s, R_e, R_d, R_s, C_{pg}, C_{pgd}, C_{ps}, R_{gs}, \) and \( R_{gd} \) are considered for removal. Omitting any of them will not increase the error more than 4%. The simplified model then just contains \( L_e, C_{gs}, C_{ds}, C_{gd}, g_m, \) tau and \( G_{ds}, \) (shown in fig 4.35). When an optimization was done with this new model, the results were almost as good as for the complete equivalent circuit.

The new model is simpler and it keeps all the most important parts of the FET model. Table 4.1 and table 4.2 show the optimized results of this model. In table 4.1, the initial error for the two fingered device (device A) is just 5.5% and it is decreased to 4.8% after another optimization. Compared to the complex model developed in chapter 3, the error increases but it is still acceptable. The errors of device A are:

- from 0.68% to 2.55% for \( S_{11} \)
- from 2.45% to 14.4% for \( S_{12} \)
- from 0.48% to 0.54% for \( S_{21} \)
- from 0.64% to 1.75% for \( S_{22}. \)

The S-parameters curves for Device A are given in fig 4.37 through 4.39 in Smith chart form.

In table 4.2, the total error of Device B increases from 0.8% in the complex model to 4.24% in the simplified model. The error of each S-parameter is:

- from 0.65% to 2.26% for \( S_{11} \)
- from 1.06% to 9.2% for \( S_{12} \)
- from 0.52% to 0.63% for \( S_{21} \)
• form 0.72% to 4.85% for $S_{22}$.

The S-parameters curves are shown from fig 4.40 to 4.42.

### 4.2-2 The Second Simplified Model

From the tables, the error of $S_{12}$ for the simplified model in fig 4.35 is bigger than the others. A second model can be made with the goal of reducing that error. Examine the figures from 4.1 to 4.34 again. $R_{gs}$ is found to cause more error than any other elements which have been taken away, so it is put back again to create another simplified model (fig 4.36). After optimization, the error of $S_{12}$ is decreased.

Table 4.3 shows that the errors are decreased compared to the first model:

- the error of $S_{12}$ decreases from 14.4% to 9.16%
- total error drops from 4.8% to 2.86%
- from 2.55% to 0.84% for $S_{11}$
- from 0.54% to 0.49% for $S_{21}$
- from 1.75% to 0.94% for $S_{22}$

Compared to the first model, all the errors are decreased, so the model is improved, although it is not as good as the complete equivalent circuit.

Table 4.4 gives the results of device B:

- the error of $S_{12}$ goes down from 9.2% to 2.5%
- the total error drops from 4.24% to 1.97%
- from 2.26% to 0.74% for $S_{11}$
• from 0.63% to 0.56% for $S_{21}$

• from 4.85% to 4.05% for $S_{22}$.

The model gives improved accuracy from the first simplified model as above.

4.3 Conclusion

In order to predict the performance of the device, a suitable model is necessary. In chapter 3, a complete HEMT model was presented. After optimization, it predicts a real device's behavior well. For Device A, the ErrS is 1% and for Device B, ErrS is 0.8%. In this chapter, two simplified models are given based on the complex one. For the first model ErrS is 4.8% for Device A and 4.24% for Device B and the second model has $ErrS = 2.85\%$ for A and 1.97% for B; they are all at reasonable error levels.

For the S-parameter data used for this work, the error of the measured data for each point of $S_{11}$, $S_{21}$ and $S_{22}$ is between 1 to 3% and is much bigger for $S_{12}$ as it is a smaller value compared to the other parameters. The errors in our optimized models are of this order of magnitude.

The new simple models are good for practical usage. They are accurate enough to keep the average error below 5%, and they can fit the requirements of design and manufacture to model and simulate. They are also simpler. Each simple model has just seven or eight parameters. The advantage is obvious; due to the simpler construction, modeling and simulation are more efficient and rapid.
Table 4.1 The first simplified HEMT model (Fig 4.35) for Device A. The initial values come from the optimized parameters in Chapter 3.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_e$ (pH)</td>
<td>47.69</td>
<td>38.98</td>
</tr>
<tr>
<td>$C_{gs}$ (fF)</td>
<td>79.48</td>
<td>74.56</td>
</tr>
<tr>
<td>$C_{gd}$ (fF)</td>
<td>13.78</td>
<td>16.77</td>
</tr>
<tr>
<td>$C_{ds}$ (fF)</td>
<td>38.18</td>
<td>34.04</td>
</tr>
<tr>
<td>$G_{ds}$ (mS)</td>
<td>3.1</td>
<td>3.13</td>
</tr>
<tr>
<td>$g_m$ (mS)</td>
<td>29.13</td>
<td>27.86</td>
</tr>
<tr>
<td>$\tau$ (psec)</td>
<td>1.09</td>
<td>1.62</td>
</tr>
<tr>
<td>ERRS$_{11}$ (%)</td>
<td>3.98</td>
<td>2.55</td>
</tr>
<tr>
<td>ERRS$_{12}$ (%)</td>
<td>7.38</td>
<td>14.39</td>
</tr>
<tr>
<td>ERRS$_{21}$ (%)</td>
<td>7.17</td>
<td>0.54</td>
</tr>
<tr>
<td>ERRS$_{22}$ (%)</td>
<td>3.46</td>
<td>1.75</td>
</tr>
<tr>
<td>ERRS (%)</td>
<td>5.50</td>
<td>4.81</td>
</tr>
</tbody>
</table>
Table 4.2 The first simplified HEMT model (Fig 4.35) for Device B. The initial values come from the optimized parameters in Chapter 3.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_g$ (pH)</td>
<td>48.38</td>
<td>45.90</td>
</tr>
<tr>
<td>$C_{gs}$ (fF)</td>
<td>140.30</td>
<td>134.66</td>
</tr>
<tr>
<td>$C_{gd}$ (fF)</td>
<td>28.22</td>
<td>30.83</td>
</tr>
<tr>
<td>$C_{ds}$ (fF)</td>
<td>54.01</td>
<td>47.12</td>
</tr>
<tr>
<td>$G_{ds}$ (mS)</td>
<td>5.92</td>
<td>6.05</td>
</tr>
<tr>
<td>$g_m$ (mS)</td>
<td>56.09</td>
<td>54.93</td>
</tr>
<tr>
<td>$\tau$ (psec)</td>
<td>1.42</td>
<td>1.92</td>
</tr>
<tr>
<td>ERRS_{11} (%)</td>
<td>3.77</td>
<td>2.26</td>
</tr>
<tr>
<td>ERRS_{12} (%)</td>
<td>3.84</td>
<td>9.21</td>
</tr>
<tr>
<td>ERRS_{21} (%)</td>
<td>3.65</td>
<td>0.63</td>
</tr>
<tr>
<td>ERRS_{22} (%)</td>
<td>8.88</td>
<td>4.85</td>
</tr>
<tr>
<td>ERRS (%)</td>
<td>5.04</td>
<td>4.24</td>
</tr>
</tbody>
</table>
Table 4.3 The second simplified model (Fig 4.36) when $R_{gd}$ is added back to the first model for Device A

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_g$ (pH)</td>
<td>47.69</td>
<td>55.09</td>
</tr>
<tr>
<td>$C_{gs}$ (fF)</td>
<td>79.48</td>
<td>74.17</td>
</tr>
<tr>
<td>$C_{gd}$ (fF)</td>
<td>13.78</td>
<td>16.14</td>
</tr>
<tr>
<td>$C_{ds}$ (fF)</td>
<td>38.18</td>
<td>35.53</td>
</tr>
<tr>
<td>$R_{gd}$ (ohm)</td>
<td>45.75</td>
<td>47.94</td>
</tr>
<tr>
<td>$G_{ds}$ (mS)</td>
<td>3.1</td>
<td>3.01</td>
</tr>
<tr>
<td>$g_m$ (mS)</td>
<td>29.13</td>
<td>27.75</td>
</tr>
<tr>
<td>$\tau$ (psec)</td>
<td>1.09</td>
<td>1.65</td>
</tr>
<tr>
<td>ERRS$_{11}$ (%)</td>
<td>2.64</td>
<td>0.84</td>
</tr>
<tr>
<td>ERRS$_{12}$ (%)</td>
<td>7.13</td>
<td>9.16</td>
</tr>
<tr>
<td>ERRS$_{21}$ (%)</td>
<td>6.94</td>
<td>0.49</td>
</tr>
<tr>
<td>ERRS$_{22}$ (%)</td>
<td>2.38</td>
<td>0.94</td>
</tr>
<tr>
<td>ERRS (%)</td>
<td>4.77</td>
<td>2.86</td>
</tr>
</tbody>
</table>
Table 4.4 The second simplified model (Fig 4.36) when $R_{gd}$ is added back to the first model for Device B

<table>
<thead>
<tr>
<th>parameters</th>
<th>initial</th>
<th>final (optimized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$L_b$ (pH)</td>
<td>48.38</td>
<td>54.30</td>
</tr>
<tr>
<td>$C_{gs}$ (fF)</td>
<td>140.30</td>
<td>135.26</td>
</tr>
<tr>
<td>$C_{gd}$ (fF)</td>
<td>28.22</td>
<td>29.64</td>
</tr>
<tr>
<td>$C_{ds}$ (fF)</td>
<td>54.01</td>
<td>51.39</td>
</tr>
<tr>
<td>$R_{gd}$ (ohm)</td>
<td>21.33</td>
<td>20.19</td>
</tr>
<tr>
<td>$G_{ds}$ (mS)</td>
<td>5.92</td>
<td>6.04</td>
</tr>
<tr>
<td>$g_m$ (mS)</td>
<td>56.09</td>
<td>54.97</td>
</tr>
<tr>
<td>$\tau$ (psec)</td>
<td>1.42</td>
<td>1.96</td>
</tr>
<tr>
<td>$\text{ERRS}_{11}$ (%)</td>
<td>1.99</td>
<td>0.74</td>
</tr>
<tr>
<td>$\text{ERRS}_{12}$ (%)</td>
<td>3.09</td>
<td>2.52</td>
</tr>
<tr>
<td>$\text{ERRS}_{21}$ (%)</td>
<td>4.34</td>
<td>0.56</td>
</tr>
<tr>
<td>$\text{ERRS}_{22}$ (%)</td>
<td>6.20</td>
<td>4.05</td>
</tr>
<tr>
<td>$\text{ERRS}$ (%)</td>
<td>3.90</td>
<td>1.97</td>
</tr>
</tbody>
</table>
Figure 4.1 The percent total error (Errs) at 1GHz and 26GHz as the normalized Cpgs is changed for device A.

Figure 4.2 The percent total error (Errs) at 1GHz and 26GHz as the normalized Cpgd is changed for device A.
Figure 4.3 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cpedia is changed for device A.

Figure 4.4 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Ls is changed at for device A.
Figure 4.5 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Lg is changed for device A.

Figure 4.6 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Ld is changed for device A.
Figure 4.7 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rs is changed for device A.

Figure 4.8 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rd is changed for device A.
Figure 4.9 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rg is changed for device A.

Figure 4.10 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cgs is changed for device A.
Figure 4.11 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cgd is changed for device A.

Figure 4.12 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cds is changed for device A.
Figure 4.13 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rgs is changed for device A.

Figure 4.14 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rgd is changed for device A.
Figure 4.15 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Gds is changed for device A.

Figure 4.16 The percent total error (ErrS) at 1GHz and 26GHz as the normalized gm is changed for device A.
Figure 4.17 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Tau is changed for device A.

Figure 4.18 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cpgs is changed for device B.
Figure 4.19 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cpgd is changed for device B.

Figure 4.20 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cpds is changed for device B.
Figure 4.21 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Ls is changed for device B.

Figure 4.22 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Lg is changed for device B.
Figure 4.23 The percent total error at $(\text{Err S})$ 1GHz and 26GHz as the normalized $L_d$ is changed for device B.

Figure 4.24 The percent total error $(\text{Err S})$ at 1GHz and 26GHz as the normalized $R_s$ is changed for device B.
Figure 4.25 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rd is changed for device B.

Figure 4.26 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rg is changed for device B.
Figure 4.27 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cgs is changed for device B.

Figure 4.28 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cgd is changed for device B.
Figure 4.29 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Cds is changed for device B.

Figure 4.30 The percent total error (ErrS) at 1GHz and 26GHz as the normalized Rgs is changed at for device B.
Figure 4.31 The percent total error (ErrS) 1GHz and 26GHz as the normalized Rgd is changed at for device B.

Figure 4.32 The percent total error (ErrS) 1GHz and 26GHz as the normalized Gds is changed for device B.
Figure 4.33 The percent total error (ErrS) at 1GHz and 26GHz as the normalized $g_m$ is changed for device B.

Figure 4.34 The percent total error (ErrS) at 1GHz and 26GHz as the normalized $\tau$ is changed for device B.
Figure 4.35 The first simplified AC model of a HEMT

Figure 4.36 The second simplified AC model of a HEMT
Figure 4.37 S parameters $S_{11}$ and $S_{22}$: the calculation of the first simplified model and the measured data for Device A in Smith chart form.
Figure 4.38 $S_{12}$: the calculation of the first simplified model and the measured data for Device A in Smith chart form
Figure 4.39 $S_{21}$: the calculation of the first simplified model and the measured data for Device A in Smith chart form.
Figure 4.40 S parameters $S_{11}$ and $S_{22}$: the calculation of the first simplified model and the measured data for Device B in Smith chart form
Figure 4.41 S parameter $S_{12}$: the calculation of the first simplified model and the measured data for Device B in Smith chart form.
Figure 4.42 S parameter $S_{21}$: the calculation of the first simplified model and the measured data for Device B in Smith chart form
Figure 4.43 S parameters $S_{11}$ and $S_{22}$: the calculation of the second simplified model and the measured data for Device A in Smith chart form.
Figure 4.44 $S$ parameters $S_{12}$: the calculation of the second simplified model and the measured data for Device A in Smith chart form
Figure 4.45 $S$ parameter $S_{21}$: the calculation of the second simplified model and the measured data for Device A in Smith chart form
Figure 4.46 S parameters $S_{11}$ and $S_{22}$: the calculation of the second simplified model and the measured data for Device B in Smith chart form.
Figure 4.47 $S$ parameter $S_{12}$: the calculation of the second simplified model and the measured data for Device B in Smith chart form
Figure 4.48  $S_{21}$ : the calculation of the second simplified model and the measured data for Device B in Smith chart form
CHAPTER 5

CONCLUSION

In this research, three nonlinear DC models and a linear AC model were used to model HEMT devices. A complete procedure has been established to extract and optimize the models based on the measurements of its I-V characteristics and small-signal S-parameters. Two simplified models were introduced based on the AC small signal model which introduces only slightly more error than the complete model.

5.1 DC Modeling

The study of HEMT's has been approached according to the transistor's most important properties for microwave applications. In DC models, an algorithm has been introduced by using conventional I-V curve fitting. Based on this technique, a detailed procedure is given to extract the parameters of the Hemt Curtice model. From this latter model, the Hemt advanced Curtice model and the Hemt Materka-Kacprzak model are extracted. An optimized method is introduced and a definition of the error is given. The optimized results are presented in both tables and curves in chapter 2. From the tables and curves, it is seen that the best model of the three for the devices examined is the Hemt Advanced Curtice model.
5.2 AC High Frequency Modeling

In the AC model, a conventional microwave model is presented. Since the external parasitic resistors can be estimated or measured in the lab quite accurately, an algorithm is introduced to extract the internal elements with the lower frequency data, which means the contact pad capacitances and parasitic inductances can be ignored for the initial estimate. The optimized results are obtained for two devices having different geometric configurations. A measurement of the error in the S-parameters is defined and used to determine how well the model performs. The parameter tables and the Smith charts of the S-parameters are shown in Chapter 3.

5.3 Simplified Models

As the AC small signal model is very complex, two more simplified models are proposed by considering the different effects of each element on the average error in the model mentioned above. For the simplified models, almost all the parasitic elements are neglected except the gate inductance. In one model, the only intrinsic element eliminated is the gate input resistor. For the other simplified model, the intrinsic feedback resistor \( R_{gd} \) is eliminated too. The optimized results of these models are in chapter 4. Both models show only slightly increased error from the more complete equivalent circuit, therefore, it is advantageous to use them for the evaluation of circuits.
APPENDIX A

OPTIMIZATION METHOD

Nonlinear optimization is the process of finding the minimum or maximum of a nonlinear function of one or several independent variables. A defined merit function is used to determine best-fit parameters by finding its minimum. With given trial values for the parameters, a procedure is developed to improve the trial solution. The procedure is then repeated until the merit function stops decreasing. Here the algorithm \[6\] is explained in detail.

First, a function $F$ is assumed with trial parameters $a_1, a_2, ..., a_m$, $y = F(x, a)$. $x$ is a vector which makes up the independent variable of the function $F$. A merit function $\chi$ is defined as:

$$
\chi^2(a) = \sum_{i=1}^{N} \left\{ \frac{Y_i - y(x_i, a)}{\sigma_i} \right\}^2
$$

where $a$ is the $M$ dimensional vector made up of the $M$ parameters to be found: $a_1, a_2, ..., a_M$.

$M$ is the number of the parameters

$x_i$ is the $i$th data point of the function

$y$ is the analytical function of $x$
$y_i$ are measured data points at each $x_i$.

$N$ is the number of data points.

$\sigma_i$ is the individual standard deviation of the measured data. It is assumed that $\sigma_i$ is the same for all data points in this thesis and the value is arbitrarily set to one.

The merit function $\chi^2(a)$ is minimized by adjusting $a$ to obtain the closest possible match between the data and the function $F(x,a)$. The gradient of $\chi^2(a)$ with respect to the parameter vector $a$ will be zero at the minimum point, where the error is least. Take the partial derivative of $\chi^2(a)$ with respect to each parameter:

$$\frac{\partial \chi^2(a)}{\partial a_k} = -2 \sum_{i=1}^{N} \frac{y_i - y(x_i, a)}{\sigma_i^2} \frac{\partial y(x_i, a)}{\partial a_k} \quad k=1, 2, \ldots, M \quad A.2$$

Take an additional partial derivative:

$$\frac{\partial^2 \chi^2}{\partial a_i \partial a_1} = 2 \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left\{ \frac{\partial y(x_i, a)}{\partial a_k} \frac{\partial y(x_i, a)}{\partial a_1} - [y_i - y(x_i, a)] \frac{\partial^2 y(x_i, a)}{\partial a_k \partial a_1} \right\} \quad A.3$$

The Taylor series of $\chi^2$ is:

$$\chi^2(a) = \chi^2(a_0) + \sum_i \frac{\partial \chi^2(a_0)}{\partial a_i} \Delta a_i + \frac{1}{2} \sum_{i,j} \frac{\partial^2 \chi^2(a_0)}{\partial a_i \partial a_j} \Delta a_i \Delta a_j + \ldots \quad A.4$$
where $a_0$ is the particular starting point, and $\Delta a$ in the vector from the start to the current location $a_{\text{cur}}$: $\Delta a = a_{\text{cur}} - a_0$.

If $|\Delta a|$ is small, and only the first three terms are used, then the function can be converted to:

$$
\chi^2(a) = \chi^2(a_0) - b \cdot (\Delta a) + \frac{1}{2} (\Delta a) \cdot D \cdot (\Delta a)
$$

where $b = -\nabla \chi^2$ is minus the gradient of the merit function at $a_0$, and

$$
D_{ij} = \frac{\partial^2 \chi(a)}{\partial a_i \partial a_j} \text{ evaluated at } a_0
$$

$D$ is the Hessian matrix.

Here the gradient is a column vector defined as:

$$
\nabla f = \left[ \frac{\partial f}{\partial a_1}, \frac{\partial f}{\partial a_2}, \ldots, \frac{\partial f}{\partial a_M} \right]^T
$$

The gradient of $\chi^2$ using then can be taken on equation A.5. The results is:

$$
\nabla \chi^2 = D \cdot (a_{\text{cur}} - a_0) - b
$$
The equation implies:

$$\delta (\nabla \chi^2) = D(\delta \alpha)$$  \hspace{1cm} A.6a

where $\delta \alpha = a_2 - a_1$ and $a_1$ and $a_2$ represent any two vector near $a_0$. Note $\nabla (\chi^2) = 0$ when $a = a_{\min}$ is at the minimum of $\chi^2$.

If the $M \times M$ matrix $D$ has an inverse, then the relationship of the parameters can be obtained from this fact:

$$a_{\min} = a_{\text{cur}} + D^{-1} [-\nabla \chi^2 (a_{\text{cur}})]$$  \hspace{1cm} A.7

where $a_{\text{cur}}$ is a vector of the current trial parameters near $a_0$.

Define:

$$\beta_k = -\frac{1}{2} \frac{\partial \chi^2}{\partial a_k}$$  \hspace{1cm} A.8

and

$$\alpha_{kj} = \frac{1}{2} \frac{\partial^2 (\chi^2)}{\partial a_k \partial a_j}$$  \hspace{1cm} A.9
Substitute the definition A.8 and A.9 in equation A.6a:

\[ \sum_{j=1}^{M} \alpha_{kj} \delta a_j = \beta_k \]  \hspace{1cm} \text{A.10}

where \( \delta a = a_{\text{min}} - a_{\text{cur}} \), and \( \delta(\nabla \chi^2) = \nabla \chi^2(a_{\text{min}}) - \nabla \chi^2(a_{\text{cur}}) = -\nabla \chi^2(a_{\text{cur}}) \).

Using equation A.3, to obtain the \( a_{ki} \) terms, we will ignore the second term, because we are close to the optimum \( a \), where \( y_i = y(x_i, a) \):

\[ a_{ki} = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left[ \frac{\partial y(x_i, a)}{\partial a_k} \cdot \frac{\partial y(x_i, a)}{\partial a_i} \right] \]  \hspace{1cm} \text{A.11}

An alternative approach is the steepest descent method. In this approach, one searches for the minimum by going opposite the direction of the gradient:

\[ a_{\text{next}} = a_{\text{cur}} - \text{constant} \times \nabla \chi^2(a_{\text{cur}}) \]  \hspace{1cm} \text{A.12}

The constant must not be chosen too large, or the minimum will be passed. If it is too small on the other hand, it will take many more steps than needed.

Marquardt’s method is to construct a diagonal matrix instead of the constant in A.12, and to pick the constants in eqn A.12 based on the diagonal elements of \( D \). C is a diagonal
matrix with elements:

\[ constant_{jj} = \frac{1}{c\alpha_{jj}} \quad A.13 \]

where \( c \) is a nondimensional positive factor.

Note the \( \alpha_{jj} \) are automatically positive because of equation A.11.

Modify equation A.12 by replacing the constant with the matrix \( C \), then:

\[(a_{next} - a_{cur})^T = -C\nabla_2 \chi^2 (a_{cur})\]

and

\[C^{-1}(a_{next} - a_{cur})^T = -\nabla_2 \chi^2 (a_{cur})\]

this leads to:

\[c'\alpha_{jj}\delta a_j = \beta_j \quad j = 1, \ldots, M \quad A.14\]

In the Levenberg-Marquardt method, we use a mixture of the steepest descent approach and the minimization based on equation A.10. In other words, we construct a
new matrix which combines elements of A.10 and A.14 for the new diagonal elements $\alpha'_{jj}$:

$$\alpha'_{jj} = \alpha_{jj} + c\alpha_{jj} = \alpha_{jj}(1 + c) \quad \text{A.15}$$

and

$$\alpha'_{jk} = \alpha_{jk} \quad (j \neq k) \quad \text{A.16}$$

where $j$ and $k$ are 1, 2, ..., $M$

Then replace $\alpha$ by $\alpha'$ in eqn A.10:

$$\sum_{i=1}^{M} \alpha'_{ik} \delta \alpha_{i} = \beta_{k} \quad \text{A.17}$$

The $\alpha'_{jj}$ and $\beta_{k}$ can be computed with $c < 1$ and used to solve for $\delta \alpha_{1}, ..., \delta \alpha_{M}$. If it is found $\chi^2(a)$ increases, then $c$ is increased by a factor of 10 to more closely follow the path of steepest descent (eqn A.12). If $\chi^2(a)$ is less for the new values in $a$, then we are on the right track and we can decrease $c$ by a factor of 10 so that the solution is closer to equation A.7 (minimization of a quadratic function) for the next change in $a_{\text{cur}}$. 


Optimization Programs

The following programs[6] which are based on Levenberg-Marquardt method are used for optimization between the parameters of the models and the measured data. Here, \( x[1..ndata] \) and \( y[1..ndata] \) are sets of points with individual standard deviations \( \text{sig}[1..ndata] \), \( a[1..ma] \) are the coefficients of nonlinear function. \( ma \) is the number of parameters and \( \text{lista}[1..ma] \) numbers the parameters which let first \( mfit \) elements to be adjusted. \( \text{covar}[1..ma][1..ma] \) and \( \text{alpha}[1..ma][1..ma] \) are used to give working space and \( \text{alamda} \) is matrix of adjustment factors. \( \text{chisq} \) is the result of the merit function and \( \text{funcs} \) is the nonlinear function and partials.

```c
#include <stdio.h>
#include <math.h>

void mrqmin(x,y,sig,ndata,a,ma,lista,mfit,covar,alpha,chisq,funcs,alamda)
float x[],y[],sig[],a[],**covar,**alpha,*chisq,*alamda;
int ndata,ma,lista[],mfit;
void (*funcs)();

{  
    int k, kk,j,ihit;
    static float *da,*atry,**oneda,*beta,ochisq;
    float *vector(),**matrix();
    void mrq cof(),gaussj(),covsrt(),nerror(),free_matrix(),free_vector();
    if (*alamda < 0.0)
    {  
        oneda=matrix(1,mfit,1,1);
        atry=vector(1,ma);
        beta=vector(1,ma);
    }
}```
da=vector(1,ma);
kk=mfit+1;
for (j=1;j<=ma;j++)
{
    ihit=0;
    for (k=1;k<=mfit;k++)
        if (lista[k] == j) ihit++;
    if (ihit == 0)
        lista[kk++]=j;
    else if (ihit > 1) nrerror("Bad Lista permutation in MRQMIN-1");
}
if (kk != ma+1) nrerror("bad Lista permutation in MRQMIN-2");
*alamda=0.001;
mrqcof(x,y,signdata,a,ma,lista,mfit,alpha,beta,chisq,funcs);
ochisq=(*chisq);
for (j=1;j<=mfit;j++)
{
    for (k=1;k<=mfit;k++)
        covar[j][k]=alpha[j][k];
    covar[j][j]=alpha[j][j]*(1.0+(*alamda));
    oneda[j][1]=beta[j];
}
gaussj(covar,mfit,onedav,1);
for (j=1;j<=mfit;j++)
    da[j]=onedav[j][1];
if (*alamda == 0.0)
{
    covsrt(covar,ma,lista,mfit);
    free_vector(beta,1,ma);
    free_vector(da,1,ma);
    free_vector(atry,1,ma);
    free_matrix(onedav,1,mfit,1,1);
    return;
}
for (j=1;j<=ma;j++) atrv[j]=a[j];
for (j=1;j<=mfit;j++)
    atrv[lista[j]] = a[lista[j]]+da[j];
mrqcof(x,y,signdata,atry,ma,lista,mfit,covar,da,chisq,funcs);
if (*chisq < ochisq)
{
    *alamda *= 0.1;
    ochisq=(*chisq);
    for (j=1;j<=mfit;j++)
void mrqcof(x,y,signdata,a,ma,lista,mfit,alpha,beta,chisq,funcs)
float x[],y[],sig[],a[],**alpha,beta[],*chisq;
int ndata,ma,lista[],mfit;
void (*funcs)();
{
    int k,j,i;
    float ymod,wt,sig2i,dy,*dyda,*vector();
    void free_vector();
    dyda=vector(1,ma);
    ymod=0.0;
    for (j=1;j<=mfit;j++) {
        for (k=1;k<=j;k++) alpha[j][k]=0.0;
        beta[j]=0.0;
    }
    *chisq=0.0;
    for (i=1;i<=ndata;i++) {
        (*funcs)(x[i],a,&ymod,dyda,ma);
        sig2i=1.0/(sig[i]*sig[i]);
        dy=y[i]-ymod;
        for (j=1;j<=mfit;j++) {
            wt=dyda[lista[j]]*sig2i;
            for (k=1;k<=j;k++)
                alpha[j][k] += wt*dyda[lista[k]];
            beta[j] += dy*wt;
        }
        (*chisq) += dy*dy*sig2i;
    }
}
for (k=1;k<=mfit;k++) alpha[j][k]=covar[j][k];
beta[j]=da[j];
a[lista[j]]=atry[lista[j]];
else
{
    *alamda *= 10.0;
    *chisq=ochisq;
}
return;
}
```c
for (j=2;j<=mfit;j++)
for(k=1;k<=j-1;k++) alpha[k][j]=alpha[j][k];
free_vector(dyda,1,ma);
}

void covsrt(covar,ma,lista,mfit)
float **covar;
int ma,lista[],mfit;
{
    int i,j;
    float swap;
    for (j=1;j<ma;j++)
        for (i=j+1;i<=ma;i++)
            covar[i][j]=0.0;
    for ( i=1;i<=mfit;i++)
        for (j=i+1;j<=mfit;j++)
            {          
                if (lista[j] > lista[i])
                    covar[lista[j]][lista[i]]=covar[i][j];
                else
                    covar[lista[i]][lista[j]]=covar[i][j];
            }
    swap=covar[1][1];
    for (j=1;j<ma;j++)
        {          
            covar[1][j]=covar[j][j];
            covar[j][j]=0.0;
        }
    covar[lista[1]][lista[1]]=swap;
    for (j=2;j<=mfit;j++)
        covar[lista[j]][lista[j]]=covar[1][j];
    for (j=2;j<=ma;j++)
        for (i=1;i<j-1;i++)
            covar[i][j]=covar[j][i];
}
```
```c
#include <math.h>
#define SWAP(a,b) {float temp=(a);(a)=(b);(b)=temp;}

tvoid gaussj(a,n,b,m)
float **a,**b;
int n,m,
{ int *indxc,*indxr,*ipiv;
int i,icol,irow,j,k,l,ll,*ivector();
float big , dum,pivinv;
void nrerror(),free_ivector();
indxc=ivector(1,n);
indxr=ivector(1,n);
ipiv=ivector(1,n);
for (j=1;j<=n;j++) ipiv[j]=0;
for (i=1;i<=n;i++)
{ big=0.0;
for (j=1;j<=n;j++)
if (ipiv[j] != 1)
for (k=1;k<=n;k++)
if (ipiv[k] == 0)
{ if (fabs(a[j][k]) >= big)
{ big=fabs(a[j][k]);
irow=j;
icol=k;
}
}
else if (ipiv[k] > 1)
 nrerror("GAUSSJ: Singular Matrix-1");
} ++(ipiv[icol]);
if (irow != icol)
{ for (l=1;l<=n;l++) SWAP(a[irow][l],a[icol][l]);
for (l=1;l<=m;l++) SWAP(b[irow][l],b[icol][l]);
}
indxr[i]=irow;
indxc[i]=icol;
if (a[icol][icol] == 0.0)
{ nrerror("GAUSSJ: Singular Matrix-2");
}
```

pivinv=1.0/a[icol][icol];
a[icol][icol]=1.0;
for (l=1;l<=n;l++) a[icol][l] *= pivinv;
for (l=1;l<=m;l++) b[icol][l] *= pivinv;
for (ll=1;ll<=n;ll++)
    if (ll != icol)
    {
        dum=a[ll][icol];
a[ll][icol]=0.0;
        for (l=1;l<=n;l++) a[ll][l] -= a[icol][l]*dum;
        for (l=1;l<=m;l++) b[ll][l] -= b[icol][l]*dum;
    }
}
for (l=n;l>=1;l--)
{
    if (indxr[l] != indxc[l])
    for(k=1;k<=n;k++)
      SWAP(a[k][indxr[l]],a[k][indxc[l]]);
}
free_ivector(ipiv,1,n);
free_ivector(indxr,1,n);
free_ivector(indxc,1,n);

#include <stdio.h>
#include <malloc.h>

void nrerror(error_text)
char error_text[];
{
    void exit();
    fprintf(stderr,"Numerical Recipes run-time error...\n");
    fprintf(stderr,"\n",error_text);
    fprintf(stderr,"...now exiting to system...\n");
    exit(1);
}

float *vector(nl,nh)
int nl, nh;
{
    float *v;
    v = (float *) malloc((unsigned) (nh-nl+1)*sizeof(float));
    if (!v) nrerror("allocation failure in vector()");
    return v-nl;
}

float ** matrix(nrl, nrh, ncl, nch)
int nrl, nrh, ncl, nch;
{
    int i;
    float ** m;
    m = (float **) malloc((unsigned) (nrh-nrl+1)*sizeof(float));
    if (!m) nrerror("allocation failure in matrix()");
    m -= nrl;
    for (i = nrl; i <= nrh; i++)
    {
        m[i] = (float *) malloc((unsigned) (nch-ncl+1)*sizeof(float));
        if (!m[i]) nrerror("allocation failure 2 in matrix()");
        m[i] -= ncl;
    }
    return m;
}

int * ivector(nl, nh)
int nl, nh;
{
    int * v;
    v = (int *) malloc((unsigned) (nh-nl+1)*sizeof(int));
    if (!v) nrerror("allocation failure in ivector()");
    return v-nl;
}

void free_vector(v, nl, nh)
float * v;
int nl, nh;
{
    free ((char*) (v+nl));
}
void free_ivector(v, nl, nh)
int *v, nl, nh;
{
    free((char*) (v + nl));
}

void free_matrix(m, nrl, nrh, ncl, nch)
float **m;
int nrl, nrh, ncl, nch;
{
    int i;
    for(i = nrh; i >= nrl; i--) free((char*) (m[i] + ncl));
    free((char*) (m + nrl));
}
Appendix C

DC Model Functions

The following gives source code for three functions of DC models. These include the HEMT Curtice model, HEMT Materka-Kacprzak model and HEMT Advanced Curtice model. Each program consists of the model function and partial derivative with respect to each parameter to be optimized.

/* The function of hemt Curtice model */


    a[2] = \psi, a[6] = V_{pf}, x = V_{ds}, z = V_{gs} */

#include <math.h>
void model(x,a,y,dyda,na)
float x,a[],y,dyda[];
int na;
{
    float fac;
    float tt;
    static float z=-2.0;
    static int j=1;

    if (z<=a[6])
    {
        if (z<=a[7])
        {
            \*y=0.0;
            dyda[1]=0.0;
            dyda[2]=0.0;
            dyda[3]=0.0;
            dyda[4]=0.0;
            dyda[5]=0.0;
        }
dyda[6]=0.0;
dyda[7]=0.0;

} else
{
  y=a[1]*(z-a[7])*(z-a[7])*tanh(a[3]*x)*(1+a[4]*x);
dyda[1]= y/a[1];
dyda[2]=0.0;
if(x==0.0) dyda[3]=0.0;
else
  dyda[3]=y*x/(tanh(a[3]*x)*cosh(a[3]*x)*cosh(a[3]*x));
dyda[4]=y*x/(1+a[4]*x);
dyda[5]=0.0;
dyda[6]=0.0;
dyda[7]=-y*2/(z-a[7]);
}
else
{
  fac=a[1]*(z-a[7])*(z-a[7])*tanh(a[3]*x)*(1+a[4]*x);
tt=-a[5]*pow(z-a[6],a[2]+1)*(1+a[4]*x)*tanh(a[3]*x)/(1+a[2]);
  y=fac+tt;
dyda[1]=fac/a[1];
dyda[2]=tt*(log(z-a[6])-1/(1+a[2]));
if(x==0.0) dyda[3]=y;
else
  dyda[3]=y*x/(tanh(a[3]*x)*cosh(a[3]*x)*cosh(a[3]*x));
dyda[4]=y*x/(1+a[4]*x);
dyda[5]=tt/a[5];
dyda[6]=tt*(1+a[2])/(z-a[6]);
dyda[7]=-fac*2/(z-a[7]);
}
if(x==4.0)
{
  z+=0.2;
  j+=1;
}
if(x==4.0 && j==12)
{
  j=1;
  z=-2.0;
}
The function of hemt Materka-Kacprzak model

assume:

```c
#include <math.h>
void model2(x,a,y,dyda,ma)
int ma;
float x,a[7],*y,dyda[];
{
    int i;
    float fac,fa,tt,vt;
    static float z=-2.0;
    static int j=1;
    static int k=0;
    float vds=2.5;

    k+=1;
    if(k<=99)
        {
*      y=0.0;
      vt=a[7]+a[2]*x;
      if(z <= a[6])
        {
        if(z <= vt)
            {
*            y=0.0;
            for (i=1;i<=7;i++) dyda[i]=0.0;
            }
        else
            {
            if(x==0.0)
                {
*              y=0.0;
              for(i=1;i<=7;i++) dyda[i]=0.0;
                }
            else
                {
* f ac=a[1]*(1-z/vt)*(1-z/vt)*tanh(a[3]*x/(z-vt));
* y=f ac;
 dyda[1]=*y/a[1];
 tt=tanh(a[3]*x/(z-vt))*cosh(a[3]*x/(z-vt))*
 cOh s(a[3]*x/(z-vt));
```
```c
if (x == 0.0) {
    *y = 0.0;
    for (i = 1; i <= 7; i++) dyda[i] = 0.0;
} else {
    fa = a[5] * pow(z-a[6], a[4]) * tanh(a[3] * x/(z-vt));
    *y = fac - fa;
         * cosh(a[3] * x/(z-vt));
    dyda[1] = fac / a[1];
    dyda[2] = 2 * fac * z * x/(vt * vt * (1 - z/vt))
              + *y * a[3] * x * x / (tt * (z-vt) * (z-vt));
    dyda[3] = *y * x / ((z-vt) * tt);
    dyda[4] = -fa * log(z-a[6]);
    dyda[5] = -fa / a[5];
}
```

else
{
    vt=a[7]+a[2]*vds;
    if(x <= a[6])
    {
        if(x <= a[7])
        {
            *y=0.0;
            for (i=1;i<=7;i++) dyda[i]=0.0;
        }
        else
        {
            fac=a[1]*(1-x/vt)*(1-x/vt)*tanh(a[3]*vds/(x-vt));
            *y=fac;
            dyda[1]=*y/a[1];
            tt=tanh(a[3]*vds/(x-vt))*cosh(a[3]*vds/(x-vt))
                *cosh(a[3]*vds/(x-vt));
            dyda[2]=*y*vds*(2*x/(vt*vt*(1-x/vt))
                +a[3]*vds/((x-vt)*(x-vt)*tt));
            dyda[3]=*y*vds/((x-vt)*tt);
            dyda[4]=0.0;
            dyda[5]=0.0;
            dyda[6]=0.0;
        }
    }
    else
    {
        fac=a[1]*(1-x/vt)*(1-x/vt)*tanh(a[3]*vds/(x-vt));
        fa=a[5]*pow(x-a[6],a[4])*tanh(a[3]*vds/(x-vt));
        *y=fac-fa;
        tt=tanh(a[3]*vds/(x-vt))*cosh(a[3]*vds/(x-vt))
            *cosh(a[3]*vds/(x-vt));
        dyda[1]=fac/a[1];
        dyda[2]=2*fac*x*vds/(vt*vt*(1-x/vt))
            +*y*a[3]*vds*vds/(tt*(x-vt)*(x-vt));
        dyda[3]=*y*vds/((x-vt)*tt);
        dyda[4]=fa*log(x-a[6]);
        dyda[5]=fa/a[5];
        dyda[6]=fa*a[4]/(x-a[6]);
    }
}
The function of hemt advanced Curtice model


#include <math.h>
void model3(x,a,y,dyda,ma)
float x,a[],*y,dyda[];
int ma;
{
  int i;
  static int j=1;
  static float z=-2.0;
  float fac,tt,den,vg,be;

  *y=0.0;
  vg=z-a[10]-a[3]*x;
  if(z<=a[9])
    if(vg<=0.0)
      {
        *y=0.0;
        for(i=1;i<=10;i++) dyda[i]=0.0;
      }
    else
      {
        if(x==0.0)
          {
            *y=0.0;
            for(i=1;i<=10;i++) dyda[i]=0.0;
          }
        else
          {
            be=a[1]/(1+a[2]*vg);
            fac=be*exp(a[4])*(1+a[5]*x)*tanh(a[6]*x);
            den=tanh(a[6]*x)*cosh(a[6]*x)*cosh(a[6]*x);
            *y=fac;
          }
      }
  else
    {
      *y=0.0;
      for(i=1;i<=10;i++) dyda[i]=0.0;
    }
}
if(k==136) k=0;
}
dyda[1]=\*y/a[1];
dyda[2]=\*y*vg/(1+a[2]*vg);
dyda[3]=\*y*(a[2]*x/(1+a[2]*vg)-a[4]*x/vg);
dyda[4]=\*y*log(vg);
dyda[5]=\*y*x/(1+a[5]*x);
dyda[6]=\*y*x/den;
dyda[7]=0.0;
dyda[8]=0.0;
dyda[9]=0.0;
dyda[10]=dyda[3]/x;
}
else
{
    if(x == 0.0 )
    {
        for(i=1;i<=10;i++) dyda[i]=0.0;
    }
    else
    {
        if(vg <= 0)
        {
            \*y=0.0;
            for(i=1;i<=10;i++) dyda[i]=0.0;
        }
        else
        {
            be=a[1]/(1+a[2]*vg);
            fac=be*pow(vg,a[4])*(1+a[5]*x)*tanh(a[6]*x);
            tt=be*a[8]*pow(z-a[9],a[7])*(1+a[5]*x)*tanh(a[6]*x);
            den=tanh(a[6]*x)*cosh(a[6]*x)*cosh(a[6]*x);
            \*y=fac+tt;
            dyda[1]=\*y/a[1];
            dyda[2]=\*y*vg/(1+a[2]*vg);
            dyda[3]=\*y*x*a[2]/(1+a[2]*vg)-a[4]*x/vg*fac;
            dyda[4]=fac*log(vg);
            dyda[5]=\*y*x/(1+a[5]*x);
            dyda[6]=\*y*x/den;
            dyda[7]=tt*log(z-a[9]);
            dyda[8]=tt/a[8];
            dyda[9]=tt*a[7]/(z-a[9]);
            dyda[10]=dyda[3]/x;
        }
    }
}
if(x==4.0)
{
    j+=1;
    z+=0.2;
}
if(x==4.0 & & j==12)
{
    j=1;
    z=-2.0;
}
APPENDIX D

COMPLETE AC MODEL SOURCE CODE FOR OPTIMIZATION

The following program for the small signal model in fig 3.1 is used for the optimization in chapter 3. The program consists of two parts: the function and its partial derivatives.

The code for the functions for chapter 4 is based on this program, with some elements taken away, so there is no need to give a separate listing.

```c
/*  The function of hemt small model */
/*  w = frequency, y=mag(sll) */

#include <math.h>
define w x*2*3.141593
do void model1(x,b,y,dydb,na)
do float x,b[],*y,dydb[];
do int na;
{
    int i;
do float dy1,dy2,ry11,ry12,ry21,ry22,iy11,iy12,iy21,iy22,ryin,iyin;
do float sumry,sumiy,rp1,rp2,rp,rm,in,magn,magn2,my11,my12,my21,my22;
do float iny11,iny12,iny21,iny22,rya11,iya11;
do float rya12,iya12,rya21,iya21,rya22,iya22;
do float ryt11,iyt11,iyt12,iyt21,iyt22,ryt21,iyt22;
do float rds,ids,magds,magds2,rns11,ins11;
```
float dy12,dy22;
float dary11[20],daiy11[20],dary12[20],daiy12[20],dary21[20],daiy21[20];
float dasumry[20],dasy[20],darp[20],daip[20],darm[20],dain[20],dan[20];
float daryny1[20],dairy11[20],dairy12[20],dairy21[20],dairy22[20];
float dary12[20],dainty12[20],dairy21[20],dainty21[20],dairy22[20],dainty22[20];
float darya12[20],dairy12[20],dairy21[20],dairy22[20];
float daryal2[20],dairyal2[20],darya21[20],daiyal2[20],darya22[20],daiyal2[20];
float dbnry[20],dbny1[20],dbny2[20],dbnry2[20],dbny2[20];
float dbny1[20],dbny2[20],dbny1[20],dbny2[20],dbny1[20],dbny2[20];
float dbny1[20],dbny2[20],dbny1[20],dbny2[20],dbny1[20],dbny2[20];
float dbnry[20],dbny1[20],dbny2[20],dbnry2[20],dbny2[20];
float dbnry[20],dbny1[20],dbny2[20],dbnry2[20],dbny2[20];
float dbnry[20],dbny1[20],dbny2[20],dbnry2[20],dbny2[20];
float dbnry[20],dbny1[20],dbny2[20],dbnry2[20],dbny2[20];
float dbnry[20],dbny1[20],dbny2[20],dbnry2[20],dbny2[20];

/* intrinsic y-parameters */

\[
\begin{align*}
dy21 &= (\cos(w^*b[15]) - w^*b[13]*b[12]*\sin(w^*b[15]))*b[14]/dy1 \\
    &+ w^*b[11]*b[10]*b[10]/dy2; \\
iy11 &= w^*b[12]/dy1 + w^*b[10]/dy2; \\
iy12 &= -w^*b[11]*b[10]*b[10]/dy2; \\
iy21 &= -(w^*b[13]*b[12]*\cos(w^*b[15]) + \sin(w^*b[15]))*b[14]/dy1 - w^*b[10]/dy2; \\
iy22 &= w^*b[11]*b[10]*b[10]/dy2 + b[17]; \\
iy22 &= w^*b[10]/dy2 + w^*b[16];
\end{align*}
\]

/* y-parameters consider parasitic resistors and inductors */

\[
\begin{align*}
ryin &= ry11*ry22-iy11*iy22-ry12*ry21+iy12*iy21; \\
iyin &= ry22*iy11+ry11*iy22-ry12*iy21-ry21*iy12; \\
surny &= ry11+ry22+ry12+ry21; \\
surny &= iy11+iy22+iy12+iy21; \\
rp &= rp^1*ryin-rp2*iyin; \\
im &= rp^1*iyin+rp2*ryin; \\
rn &= 1 + b[8]*surny+w^*b[5]*surny+b[7]*ry11+ry22+b[9] \\
    - w^*b[4]*iy11-w^*b[6]*iy22+rp; \\
in &= b[8]*surny+w^*b[5]*surny+w^*b[4]*ry11+ry22+w^*b[6] \\
    + b[7]*iy11+iy22+b[9]*ip; \\
magn &= rn^*rn+in^*in;
\end{align*}
\]
magn2 = magn * magn;
my22 = ry22 + (b[8] + b[7]) * ryin - w * (b[5] + b[4]) * iyin;
in22 = iy22 + (b[8] + b[7]) * iyin + w * (b[5] + b[4]) * ryin;

/* y-parameters of model */
ryal1 = (rn * rny1 + in * iny1) / magn;
iyal1 = (rn * iny1 + in * rny1) / magn;
rya12 = (rn * rny12 + in * iny12) / magn;
iya12 = (rn * iny12 + in * rny12) / magn;
rya21 = (rn * rny21 + in * iny21) / magn;
iya21 = (rn * iny21 + in * rny21) / magn;
rya22 = (rn * rny22 + in * iny22) / magn;
iya22 = (rn * iny22 + in * rny22) / magn;

/* s-parameters of model */

rds = 1 + ryt11 + ryt22 + ryt11 * ryt22 - iyt11 * iyt22 - ryt21 * ryt12 + iyt12 * iyt21;
ids = iyt11 + iyt22 + iyt11 * ryt22 + ryt11 * iyt22 - ryt21 * iyt12 - iyt21;
magds = rds * rds + ids * ids;
magds2 = magds * magds;

/* s11 */
rms11 = 1 - ryt11 + ryt22 - ryt11 * ryt22 + iyt11 * iyt22 + ryt12 * ryt21 - iyt12 * iyt21;
ins11 = iyt11 + iyt22 + ryt11 * iyt22 - ryt21 * ryt12 + ryt22 * iyt21 + iyt12 * iyt21;
*y = sqrt((rms11 * ms11 + ins11 * ins11) / magds);

/* s12 */
rms12 = -2 * ryt12;
ins12 = -2 * iyt12;
\*y=sqrt((\text{ms12}*\text{ms12}+\text{ins12}*\text{ins12})/\text{magds});
*/

/* s21
\text{ms21}=-2*\text{ryt21};
\text{ins21}=-2*\text{iyt21};
\*y=sqrt((\text{ms21}*\text{ms21}+\text{ins21}*\text{ins21})/\text{magds});
*/

/* s22
\text{ms22}=1+\text{ryt11}-\text{ryt22}-\text{ryt11}^*\text{ryt22}+\text{iyt11}^*\text{iyt22}+\text{ryt12}^*\text{ryt21}-\text{iyt12}^*\text{iyt21};
\text{ins22}=\text{iyt11}^*\text{iyt22}-\text{ryt11}^*\text{ryt22}-\text{iyt11}^*\text{ryt22}+\text{ryt12}^*\text{iyt21}+\text{iyt12}^*\text{ryt21};
\*y=sqrt((\text{ms22}*\text{ms22}+\text{ins22}*\text{ins22})/\text{magds});
*/

/* partial of intrinsic */

dy12=dy1*dy1;
dy22=dy2*dy2;
for\(i=10;i<=17;i++\)
{
    \text{dary11}[i]=0;
    \text{daiy11}[i]=0;
    \text{dary12}[i]=0;
    \text{daiy12}[i]=0;
    \text{dary21}[i]=0;
    \text{daiy21}[i]=0;
    \text{dary22}[i]=0;
    \text{daiy22}[i]=0;
}
dary11[10]=2*w*w*b[11]*b[10]/\text{dy22};
daiy11[10]=w*(2-\text{dy2})/\text{dy22};
dary12[10]=-dary11[10];
daiy12[10]=-daiy11[10];
dary21[10]=-dary11[10];
daiy21[10]=-daiy11[10];
dary22[10]=dary11[10];
daiy22[10]=daiy11[10];
dary11[11]=w*w*b[10]*b[10]*(2-\text{dy2})/\text{dy22};
dary11[12]=2*w*w*b[13]*b[12]/dy12;
daiy11[12]=w*(2-dy1)/dy12;
dary21[12]=w*b[13]*b[14]*(sin(w*b[15])*(2-dyl)
+cos(w*b[15])*2*w*b[12]*b[13])/dy12;
daiy21[12]=~*(2-dyl)/dy12;
dq21[12]=w*b[13]*b[14]*(cos(w*b[15])*(2-dyl)
-sin(w*b[15])*2*w*b[12]*b[12])/dy12;
daiy21[13]=w*b[12]*b[13]*(cos(w*b[15])*(2-dyl)
-sin(w*b[15])*2*w*b[12]*b[12])/dy12;
dary21[14]=(cos(w*b[15])-w*b[13]*b[12]*sin(w*b[15]))ldy1;
daiy21[14]=-(w*b[13]*b[12]*cos(w*b[15])+sin(w*b[15]))/dyl;
dary21[15]=-w*b[14]*(sin(w*b[15])+b[12]*b[12]*w*cos(w*b[15]))ldy1;
daiy21[15]=w*b[14]*(b[12]*b[12]*w*sin(w*b[15])-cos(w*b[15]))/dyl;
daiy22[16]=w;
dary22[17]=1;
for(i=10;i<=17;i++)
{
  daryin[i]=dary11[i]*ry22+ry11*dary22[i]-daiy11[i]*iy22
    -iy11*daiy22[i]-dary12[i]*ry21-ry12
   *dary21[i]+daiy12[i]* iy21+iy12*daiy21[i];
  daiyin[i]=daiy11[i]*ry22+iy11*dary22[i]+dary11[i]*iy22
     +ry11*daiy22[i]-dary12[i]*iy21+ry12
    *daiy21[i]-daiy12[i]  * ry21-iy12*dary21[i];
  dasumry[i]=dary11[i]+dary22[i]+dary12[i]+dary21[i];
  dasumiy[i]=daiy11[i]+daiy22[i]+daiy12[i]+daiy21[i];
  darp[i]=rp1*daryin[i]-rp2*daiyin[i];
  daip[i]=rp1*daiyin[i]+rp2*daryin[i];
  darn[i]=b[8]*dasumry[i]-w*b[5]*dasumiy[i]+b[7]*dary11[i]+b[9]*dary22[i]
  -w*b[4]*daiy11[i]-w*b[6]*daiy22[i]+ darp[i];
  dain[i]=b[8]*dasumiy[i]+w*b[5]*dasumry[i]+w*b[4]*dary11[i]+w*b[6]*dary22[i]
  +b[7]*daiy11[i]+b[9]*daiy22[i]+ dain[i];
  dan[i]=2*(rn*darn[i]+in*dain[i]);
  dary12[i]=dary12[i]-b[8]*daryin[i]+w*b[5]*daiyin[i];
  daiy12[i]=daiy12[i]-b[8]*daiyin[i]-w*b[5]*daryin[i];
  dary21[i]=dary21[i]-b[8]*daryin[i]+w*b[5]*daiyin[i];
  daiy21[i]=daiy21[i]-b[8]*daiyin[i]-w*b[5]*daryin[i];
}
darya11[i]=(darn[i]*rny11+rn*darya11[i]+dain[i]*iny11+in*dainy11[i])
/magn-rya11*dan[i]/magn;
daiya11[i]=(darn[i]*iny11+rn*dainy11[i]-dain[i]*rny11-in*darny11[i])/magn
-iyal1*dan[i]/magn;
darya2[i]=(darn[i]*rny12+rn*darya2[i]+dain[i]*iny12+in*dainy2[i])/magn
-rya2*dan[i]/magn;
daiya2[i]=(darn[i]*iny12+rn*dainy2[i]-dain[i]*rny12-in*darya2[i])/magn
-iyal2*dan[i]/magn;
darya21[i]=(darn[i]*rny21+rn*darya21[i]+dain[i]*iny21+in*dainy21[i])/magn
-rya21*dan[i]/magn;
daiya21[i]=(darn[i]*iny21+rn*dainy21[i]-dain[i]*rny21-in*darya21[i])/magn
-iyal2*dan[i]/magn;
darya22[i]=(darn[i]*rny22+rn*darya22[i]+dain[i]*iny22+in*dainy22[i])/magn
-rya22*dan[i]/magn;
daiya22[i]=(darn[i]*iny22+rn*dainy22[i]-dain[i]*rny22-in*darya22[i])/magn
-iyal2*dan[i]/magn;
dards[i]=50*(darya11[i]+darya22[i]+darya11[i]*ryt22+ryt11*darya22[i]
-daiya11[i]*iyt22-iyt11*daiya22[i]-darya12[i]*ryt21-ryt12*darya21[i]
+daiya12[i]*iyt21-iyt12*daiya21[i]);
daids[i]=50*(daiya11[i]+daiya22[i]+daiya11[i]*ryt22+iyt11*darya22[i]
+darya11[i]*iyt22+ryt11*daiya22[i]-darya12[i]*iyt21
-ryt12*daiya21[i]-daiya12[i]*ryt21-iyt12*darya21[i]);

/* beginning of partial for s11 */
darns11[i]=50*(-darya11[i]+darya22[i]-darya11[i]*ryt22-ryt11*darya22[i]
+daiya11[i]*iyt22+iyt11*daiya22[i]+darya12[i]*ryt21
+ryt12*darya21[i]-daiya12[i]*iyt21-ryt12*daiya21[i]);
dains11[i]=50*(-daiya11[i]+daiya22[i]-daiya11[i]*ryt22-ryt11*daiya22[i]
-darya11[i]*ryt22-iyt11*darya22[i]+darya12[i]*iyt21
+ryt12*daiya21[i]+daiya12[i]*ryt21+iyt12*darya21[i]);
ndyda[i]=(rns11*darns11[i]+ins11*dains11[i])*magds
-(rns11*rns11+ins11*ins11)*(rds*dards[i]+ids*daids[i]);
/* end of s11 */

/* s12 */
darns12[i]=100*darya12[i];
dains12[i]=100*daiya12[i];
ndyda[i]=(rns12*darns12[i]+ins12*dains12[i])*magds
-(rns12*rns12+ins12*ins12)*(rds*dards[i]+ids*daids[i]);
/* s21

darns21[i]=100*darya21[i];
dains21[i]=100*daiya21[i];
ndyda[i]=(rms21*darns21[i]+ins21*dains21[i])*magds
     -(rms21*ms21+ins21*ins21)*(rds*dards[i]+ids*daids[i]);
*/

/* s22

darns22[i]=50*(darya11[i]-darya22[i]-darya11[i]*ryt22-ryt11*darya22[i]
         +daiya11[i]*iyt22+iyt11*daiya22[i]+darya12[i]*ryt21
         +ryt12*darya21[i]-daiya12[i]*iyt21-iyt12*daiya21[i]);
dains22[i]=50*(daiyall[i]-daiya22[i]-darya11[i]*iyt22-ryt11*daiya22[i]
         -daiya11[i]*ryt22+iyt11*darya22[i]+darya12[i]*iyt21
         +ryt12*daiya21[i]-daiya12[i]*ryt21+iyt12*darya21[i]);
ndyda[i]=(rms22*darns22[i]+ins22*dains22[i])*magds
     -(rms22*ms22+ins22*ins22)*(rds*dards[i]+ids*daids[i]);
*/

dydb[i]=ndyda[i]/(*y*magds2);
}

/* partial of extrinsic */

for(i=1;i<=9;i++)
{
    dydb[i]=0.0;
}
for(i=4;i<=9;i++)
{

\begin{verbatim}

dbrp[i]=dbrp1[i]*ryin-dbrp2[i]*iyin;
dbin[i]=dbrp1[i]*iyin+dbrp2[i]*ryin;

} 
dbrn[4]=-w*iy11+dbrp[4];
dbin[4]=w*ry11+dbip[4];
dbrn[5]=-w*sumiy+dbrp[5];
dbin[5]=w*sumry+dbip[5];
dbrn[6]=dbrp[6]-w*iy22;
dbrn[7]=ry11+dbrp[7];
dbin[7]=iy11+dbip[7];
dbrn[8]=sumry+dbrp[8];
dbin[8]=sumiy+dbip[8];
dbrn[9]=ry22+dbrp[9];
dbin[9]=iy22+dbip[9];
for(i=4;i<=9;i++)
{

dbrny11[i]=0;
dbiny11[i]=0;
dbrny12[i]=0;
dbiny12[i]=0;
dbrny21[i]=0;
dbiny21[i]=0;
dbrny22[i]=0;
dbiny22[i]=0;
}
dbrny22[4]=-w*iyin;
dbiny22[4]=w*ryin;
dbrny11[5]=-w*iyin;
dbiny11[5]=w*ryin;
dbrny12[5]=-w*iyin;
dbiny12[5]=w*ryin;
dbrny21[5]=w*iyin;
dbiny21[5]=-w*ryin;
dbrny22[5]=w*iyin;
dbiny22[5]=-w*ryin;
dbrny11[6]=-w*iyin;
dbiny11[6]=w*ryin;
dbrny22[7]=ryin;
dbiny22[7]=iyin;
dbrny11[8]=ryin;
dbiny11[8]=iyin;
dbrny12[8]=ryin;
dbiny12[8]=iyin;
\end{verbatim}
dbrny21[8]=-ryin;
dbiny21[8]=-iyin;
dbrny22[8]=-ryin;
dbiny22[8]=-iyin;
dbrny11[9]=-ryin;
dbiny11[9]=-iyin;
dbrns11[1]=50*w*iyt22;
dbins11[1]=-50*w*(ryt22+1);
dbrns11[2]=50*w*iyt11;
dbins11[2]=50*w*(1-ryt11);
dbrns11[3]=50*w*(iyt11+iyt22+iyt12+iyt21);
dbins11[3]=-50*w*(ryt11+ryt22+ryt12+ryt21);
dbrds[1]=-50*w*iyt22;
dbids[1]=50*w+50*w*ryt22;
dbrds[2]=50*w*iyt11;
dbids[2]=50*w+50*w*ryt11;
dbrds[3]=-50*w*(iyt11+iyt22+iyt12+iyt21);
dbids[3]=50*w*(2+ryt11+ryt22+ryt12+ryt21);
for(i=4;i<=9;i++)
{
    dbn[i]=2*rn*dbrn[i]+2*in*dbin[i];
dbrya11[i]=(dbrn[i]*rny11+rn*dbrny11[i]+dbin[i]*iny11
       +in*dbiny11[i])/magn-rya11*dbn[i]/magn;
dbiya11[i]=(dbrn[i]*iny11+rn*dbiny11[i]-dbin[i]*my11
       -in*dbrny11[i])/magn-ya11*dbn[i]/magn;
dbrya12[i]=(dbrn[i]*rny12+rn*dbrny12[i]+dbin[i]*iny12
       +in*dbiny12[i])/magn-rya12*dbn[i]/magn;
dbiya12[i]=(dbrn[i]*iny12+rn*dbiny12[i]-dbin[i]*my12
       -in*dbrny12[i])/magn-ya12*dbn[i]/magn;
dbrya21[i]=(dbrn[i]*rny21+rn*dbrny21[i]+dbin[i]*iny21
       +in*dbiny21[i])/magn-rya21*dbn[i]/magn;
dbiya21[i]=(dbrn[i]*iny21+rn*dbiny21[i]-dbin[i]*my21
       -in*dbrny21[i])/magn-ya21*dbn[i]/magn;
dbrya22[i]=(dbrn[i]*rny22+rn*dbrny22[i]+dbin[i]*iny22
       +in*dbiny22[i])/magn-rya22*dbn[i]/magn;
dbiya22[i]=(dbrn[i]*iny22+rn*dbiny22[i]-dbin[i]*my22
       -in*dbrny22[i])/magn-ya22*dbn[i]/magn;
dbrds[i]=-50*(dbryall[i]+dbrya22[i]+dbrya11[i]*ryt22+ryt11*dbrya22[i]
       -dbiya11[i]*iyt22-iyt11*dbiya22[i]-dbrya12[i]*ryt12
       -ryt12*dbrya21[i]+dbiya12[i]*iyt12+iyt12*dbiya21[i]);
dbids[i]=-50*(dbiya11[i]+dbiya22[i]+dbiya11[i]*ryt22+iyt11*dbrya22[i]
       +dbrya11[i]*iyt22+ryt11*dbiya22[i]-dbrya22[i]*iyt21
       -ryt12*dbiya21[i]-dbiya12[i]*ryt21-iyt21*dbrya21[i]);
/* s11 partial */

dbrns11[i]=50*(-dbrya11[i]+dbrya22[i]-dbrya11[i]*ryt22-ryt11*dbrya22[i]
+dbiya11[i]*iyt22+iyt11*dbiya22[i]+dbrya12[i]*ryt21
+ryt12*dbrya21[i]-dbiya12[i]*iyt21-ryt12*dbiya21[i]);
dbins11[i]=50*(-dbiya11[i]+dbiya22[i]-dbrya11[i]*iyt22-ryt11*dbiya22[i]
-dbrya11[i]*ryt22-iyt11*dbrya22[i]+dbrya12[i]*iyt21
+ryt12*dbiya21[i]+dbya12[i]*ryt21+iyt12*dbrya21[i]);

/* end */

/* s12 */
dbrns12[i]=-100*dbrya12[i];
dbins12[i]=-100*dbiya12[i];
/*

/* s21 */
dbrns21[i]=-100*dbrya21[i];
dbins21[i]=-100*dbiya21[i];
/*

/* s22 */
dbrns22[i]=50*(dbrya11[i]-dbrya22[i]-dbrya11[i]*ryt22-ryt11*dbrya22[i]
+dbiya11[i]*iyt22+iyt11*dbiya22[i]+dbrya12[i]*ryt21
+ryt12*dbrya21[i]-dbiya12[i]*iyt21-ryt12*dbiya21[i]);
dbins22[i]=50*(dbiya11[i]-dbiya22[i]-dbrya11[i]*iyt22-ryt11*dbiya22[i]
-dbrya11[i]*ryt22-iyt11*dbrya22[i]+dbrya12[i]*iyt21
+ryt12*dbiya21[i]+dbya12[i]*ryt21+iyt12*dbrya21[i]);
*/
}
for(i=1;i<=9;i++)
{

/* s11 */
ndydb[i]=(rns11*dbrns11[i]+ins11*dbins11[i])*magds
-(rns11*rns11+ins11*ins11)*(rds*dbrds[i]+ids*dbids[i]);

/* s12 */
ndydb[i]=(rns12*dbrns12[i]+ins12*dbins12[i])*magds
-(rns12*rns12+ins12*ins12)*(rds*dbrds[i]+ids*dbids[i]);
/*

/* s21 */
ndydb[i]=(rns21*dbrns21[i]+ins21*dbins21[i])*magds
-(rns21*rns21+ins21*ins21)*(rds*dbrds[i]+ids*dbids[i]);
ndydb[i] = (rns22*dbrrns22[i] + ins22*dbins22[i]) * magds
- (rns22*rns22 + ins22*ins22) * (rdns*dbrdns[i] + ids*dbids[i]);

dydb[i] = ndydb[i] / (*y*magds2);
REFERENCES


Ohio University, Athens, Ohio 45701.


[15] LIBRA, EEsof corp., 5601 Lindero Canyon Road, Westlake Village, CA 91362.


[23] R. Allam, L. Kolanowski, D. Theron, and Y. Crosnier, "Large signal Model for Analysis and Design of HEMT Gate Mixer", IEEE Microwave and


