SYNOPSYS®

Sentaurus Technology Template: DC and RF Characterization of InGaAs HEMTs

Abstract

This TCAD Sentaurus™ simulation project provides a template setup for the DC and RF characterization of high electron mobility transistors (HEMTs).

For HEMTs, the following simulations are performed: I_d–V_{gs}, a family of I_d–V_{ds} curves, on-state and off-state breakdown, and RF analysis. The template is based on an InGaAs HEMT; however, the project structure and input files can be used for any HEMT with only minor modifications.

For each of the simulated I–V curves, relevant electrical parameters such as pinch-off voltage, transconductance, drain saturation current, and breakdown voltage are calculated. Furthermore, transistor figures of merit such as f_t and f_{max}, as well as other two-port network and RF gain parameters, are computed.

Introduction

This project provides standard templates for Sentaurus Device, which can be used to perform the most common types of simulation for the characterization and performance assessment of HEMTs. There are three different Sentaurus Workbench projects:

- A DC characterization project `HEMT_DC`, which performs an I_d–V_{gs} simulation and computes a family of I_d–V_{ds} curves.
- A breakdown simulation project `HEMT_BV`, which computes the on-state and off-state breakdown.
- A small-signal simulation project `HEMT_RF`, which extracts RF parameters by performing a two-port network analysis.

All three projects are based on an analytic InGaAs HEMT device structure, which is created using Sentaurus Structure Editor.

It is assumed that users are familiar with the Sentaurus tool suite, in particular, Sentaurus Workbench, Sentaurus Structure Editor, Sentaurus Device, and Sentaurus Visual. For an introduction and tutorials, refer to the TCAD Sentaurus Tutorial.

The focus of this project is to provide setups that can be used as they are or adapted to specific needs. The documentation focuses on aspects of the project setup. For details about tool uses and specific tool syntax, refer to the respective manuals.

General Simulation Setup

The simulations are organized in three Sentaurus Workbench projects. The tool flow of the projects are discussed here. The tool flow consists of Sentaurus Structure Editor, which creates the analytic HEMT structure; Sentaurus Device, which calculates the device characteristics; and the visualization tool Sentaurus Visual. For each tool, the associated Sentaurus Workbench input parameters, as well as the extracted parameters, are discussed.

All Projects

Sentaurus Structure Editor

The analytic HEMT structure is defined with Sentaurus Structure Editor. The Sentaurus Structure Editor setup is identical for all three HEMT template projects.

The generated structure, mesh, and doping information are stored in a TDR file, which is then passed to Sentaurus Device.

Project Setup: HEMT_DC

The first Sentaurus Workbench template project performs device simulations relevant for the DC characterization of HEMT devices.

Sentaurus Device

Sentaurus Device performs I_d–V_{gs} and I_d–V_{ds} sweeps. The bias conditions and the number of sweeps are controlled using the following Sentaurus Workbench parameters:

- `Vgmin` [V] defines the minimum gate voltage for the I_d–V_{gs} or I_d–V_{ds} sweeps. Here, it is set to `-1.5` for the I_d–V_{gs} sweeps and to `-0.8` for the I_d–V_{ds} sweeps.
- `Vgmax` [V] defines the maximum gate voltage for the I_d–V_{gs} or I_d–V_{ds} sweeps. Here, it is set to `0.8`.
- `Vd` [V] defines the drain bias for the I_d–V_{gs} sweeps and the final drain voltage for the I_d–V_{ds} sweeps. It is set to `0.05` for the low bias I_d–V_{gs} sweep and to `1.5` for the high bias I_d–V_{gs} sweep, as well as the I_d–V_{ds} sweeps.
- `IdVd = 0,1,2,3,...` For `IdVd = 0`, an I_d–V_{gs} sweep is performed. For `IdVd =` *N*, a family of *N* I_d–V_{ds} sweeps is simulated at gate biases equally spaced between `Vgmin` and `Vgmax`. Here, the parameter is set to `0` (I_d–V_{gs}) and `5` (I_d–V_{ds}).

Sentaurus Visual

Sentaurus Visual plots the respective I–V characteristics and extracts:

- `Vtgm` [V] is the threshold voltage, defined as the V_{gs} axis intercept of the tangent line at the maximum transconductance (g_m) point.
- `Vti` [V] is the threshold voltage, defined as V_{gs} at which I_d = 1 mA/mm.
- `gm` [mS/mm] is the maximum transconductance g_m.
- `Idmax` [mA/mm] is the maximum value of the I_d–V_{gs} curve.
- `Ron` [kΩ mm] is the on-state resistance, extracted for V_{gs} = 0 V and V_{ds} = 1 V.

Project Setup: HEMT_BV

The second Sentaurus Workbench template project performs on-state and off-state breakdown simulations of the HEMT device.

Sentaurus Device

Sentaurus Device performs breakdown simulations using two different methods. In the first approach, the adaptive *continuation* method is used to simulate the breakdown characteristics. In this method, an appropriate value of the external resistor to be applied to the drain is computed at every bias point. The drain is then ramped to a very high voltage. Depending on the gate bias, this yields the on-state or off-state breakdown.

Alternatively, the *drain-current injection* method [1] is used to perform an off-state breakdown simulation. In this method, the drain current level is held constant while the gate is closed. To maintain the drain current, the drain voltage must rise sharply when the gate closes. When breakdown sets in, the drain voltage saturates.

The bias condition and type of breakdown simulation are controlled using the Sentaurus Workbench parameters:

- `Vg` [V]: Gate bias. For the on-state breakdown I_d–V_{ds} sweep, the bias is set to `0`. For the off-state breakdown I_d–V_{ds} sweep, the bias is set to `-1.5`. For the drain-current injection method, this parameter sets the end point of the gate voltage sweep. Here, the value `–5` is used.
- `Vd` [V]: For the direct method, this parameter sets the end point of the drain bias sweep. Values of `200` and `400` are used for the on-state and off-state breakdown sweeps, respectively. For the drain-current injection method, this parameter is used pre-bias the drain, before switching to the current boundary condition. Here, a value of `0.1` is used.
- `Id` [mA/mm]: If this parameter is set to `0`, the direct method is used. For a nonzero value, the drain-current injection method is used and the value gives the current level. Here, three different current levels are used (`200`, `300`, and `500`).

Sentaurus Visual

Sentaurus Visual plots the respective I–V characteristics and extracts:

- `BVv` [V]: Breakdown voltage defined as the maximum *inner* voltage on the drain. The inner voltage is measured at the device contact directly and does not include the voltage drop across the contact resistance.
- `BVi` [V]: Breakdown voltage defined as the *inner* drain voltage at which the drain current reaches a certain value. Here, a value of 1000 mA/mm is used.
- `BVc` [V]: Breakdown voltage determined using the drain-current injection method. Here, the breakdown voltage is defined as the drain voltage reached after the gate has been biased well below the pinch-off voltage. Here, a value of V_{gs} = –1.5 V is used.

Project Setup: HEMT_RF

The third Sentaurus Workbench project is designed to investigate the RF characteristics of the HEMT device.

Sentaurus Device

Sentaurus Device performs an AC analysis for various frequency points during an I_d–V_{gs} sweep. At each gate bias, the frequency sweep is performed from 10^8 to 10^{12} Hz. The bias conditions and feedback circuit are controlled using Sentaurus Workbench parameters:

- `Rfb` [Ωμm]: For `Rfb = 0`, no external resistive feedback from drain to gate is considered. Otherwise, a resistor of the given magnitude between the drain electrode and the gate electrode is included in the circuit setup. Here, the values `0` and `1e5` are used.
- `Vgmin` [V]: Defines the minimum gate voltage for the I_d–V_{gs} sweep. Here, it is set to `-0.5`.
- `Vgmax` [V]: Defines the maximum gate voltage for the I_d–V_{gs} sweep. Here, it is set to `0.5`.
- `Vd` [V]: Defines the drain bias for the I_d–V_{gs} sweep. Here, it is set to `1.0`.
- `nPerDecade` [1]: Defines the number of points per decade for the frequency sweep from 10^8 to 10^{12} Hz (four decades). Here, it is set to `6`.

Sentaurus Visual: IdVg

The Sentaurus Visual tool instance `IdVg` plots I_d–V_{gs} and the DC transconductance.

Sentaurus Visual: AC

The Sentaurus Visual tool instance `AC` uses the following Sentaurus Workbench parameter:

- `W` [μm] defines the device width in the z-direction *W* to be used for postprocessing of the terminal currents or AC parameters such as conductance, capacitance, and the S-parameters. Here, it is set to `25`.

The Sentaurus Visual tool instance `AC` plots the following:

- I_d–V_{gs} and the DC transconductance.
- The gate-to-drain capacitance C_{gd} as a function of bias at a low frequency of 100 MHz and a high frequency of 1 THz.
- C_{gd} as a function of frequency at a low gate bias of –0.5 V and a high gate bias of 0.5 V.

Sentaurus Visual: RF

In the Sentaurus Visual tool instance `RF`:

- The RF parameter h_{21} is plotted as a function of:
  - Frequency for a gate bias of 0 V.
  - Bias for a frequency of 1 GHz.

  Both plots show the real and imaginary parts of h_{21} as well as the magnitude and the phase.
- The RF parameters S_{11} and S_{22} are plotted on a Smith chart for:
  - All frequencies and a gate bias of 0 V.
  - All bias points and a frequency of 1 GHz.
- The RF parameters S_{21} and S_{12} are plotted on a polar plot for all frequencies and a gate bias of 0 V.
- The S-parameters are written to two comma-separated value (CSV) format files, which can be read by any spreadsheet program. The file `nX_S_freq.csv` (where `X` is the node number) contains the real and imaginary parts of the S-parameters (S_{11}, S_{12}, S_{21}, and S_{22}) sorted by frequency first, with bias being the other parameter. In `nX_S_bias.csv`, the S-parameters are sorted by bias first.

Sentaurus Visual: Ft

The Sentaurus Visual tool instance `Ft` plots:

- A family of short-circuit current gain $|h_{21}|$ versus frequency curves (current gain curves) for all bias points on both the linear scale and the dB scale.
- A family of derivatives of $|h_{21}|$ [dB/decade] versus frequency curves for all bias points.
- The cut-off frequency f_t versus applied bias curves (f_t curves) for three different extraction methods:
  - `unit-gain-point`
  - `extract-at-dBPoint`
  - `extract-at-frequency`

f_t is extracted from the current gain curves.

It extracts several quantities that are represented by the following Sentaurus Workbench output variables:

- `ft0` [GHz], `Vb_ft0` [V]: Maximum f_t value over all bias points extracted for the `unit-gain-point` method and the corresponding bias point.
- `ftdB` [GHz], `Vb_ftdB` [V]: Maximum f_t value over all bias points for the `extract-at-dBPoint` method and the corresponding bias point.
- `ftfq` [GHz], `Vb_ftfq` [V]: Maximum f_t value over all bias points for the `extract-at-frequency` method and the corresponding bias point.

Sentaurus Visual: FK1

The Sentaurus Visual tool instance `FK1` plots the following as a function of frequency for a gate bias of 0 V:

- The current gain.
- The power gains: Mason's unilateral gain (MUG) U, maximum stable gain (MSG), and maximum available gain (MAG).
- Unilateral figure of merit U_f.
- The stability criteria, Rollett stability factor K, and stability condition delta Δ.

The Sentaurus Visual tool instance `FK1` also plots:

- Current gain, power gains, unilateral figure of merit, and stability criteria as a function of bias for a frequency of 1 GHz.
- A family of K versus frequency curves for all bias points.
- The cut-off frequency for stability f_{K1} versus applied bias, and the boundaries of the region of unconditional stability.

Sentaurus Visual: Fmax

The Sentaurus Visual tool instance `Fmax` uses the following Sentaurus Workbench parameter:

- `Pgain = MUG|MAG` defines the power gain to be plotted and used for extracting f_{max}. Here, it is set to `MUG`.

The Sentaurus Visual tool instance `Fmax` plots:

- A family of unilateral figure of merit curves and MUG versus frequency curves (MUG curves) for all bias points if `Pgain = MUG`.

  If `Pgain = MAG`, only MAG versus frequency curves are plotted. The MUG and MAG curves are plotted on both the linear scale and the dB scale.
- A family of derivative of MUG (`Pgain = MUG`) or MAG (`Pgain = MAG`) [dB/decade] versus frequency curves for all bias points.
- The maximum frequency of oscillation f_{max} versus bias curves (f_{max} curves) for three different extraction methods:
  - `unit-gain-point`
  - `extract-at-dBPoint`
  - `extract-at-frequency`

  f_{max} is extracted using MUG if `Pgain = MUG` or MAG if `Pgain = MAG`.
- A family of MUG (`Pgain = MUG`) or MAG (`Pgain = MAG`) [dB] versus bias curves for all frequencies.

It extracts several quantities that are represented by the following Sentaurus Workbench output variables:

- fmax0 [GHz], Vb_fmax0 [V]: Maximum f_{max} value over all bias points for the unit-gain-point method and the corresponding bias point.
- fmaxdB [GHz], Vb_fmaxdB [V]: Maximum f_{max} value over all bias points for the extract-at-dBPoint method and the corresponding bias point.
- fmaxfq [GHz], Vb_fmaxfq [V]: Maximum f_{max} value over all bias points for the extract-at-frequency method and the corresponding bias point.

Tool-specific Setups

Sentaurus Structure Editor and Sentaurus Mesh

Sentaurus Structure Editor is used to define the HEMT devices in a fully parameterized manner.

The HEMT layer structure is as follows (the actual names of the Scheme variables used in the parameterization of the HEMT structure are given in parentheses):

- Substrate: 0.8 µm GaAs (HGaAsSub)
- Channel: 10 nm InGaAs (HChannel)
- Spacer: 34.5 nm AlGaAs (HSpacer)
- Cap layer: 30 nm GaAs (HGaAsCap)
- Passivation layer: 50 nm nitride (HPass)
- Location of delta-doping layer: 31 nm (YDelta)
- Thickness of delta-doping layer: 2 nm (DeltaThick)

The gate Schottky contact is etched into the top spacer layer. This gate recess is 15 nm deep (GRecess). The gate length is 0.25 µm (Lg). At each side of the gate, a 40 nm wide oxide layer isolates the gate from the cap layer (Wsp).

The sheet doping concentration is $5.4\times10^{12}\,\mathrm{cm}^{-2}$ (SheetCharge). It is assumed that the dopants are in a tight Gaussian distribution, with a diffusion length given by the thickness of the delta-doping layer. Mole fractions and trap concentrations are defined in the input file of Sentaurus Device.

To adjust details of the devices, you can modify the top section of the Sentaurus Structure Editor input file sde_dvs.cmd. For example, the width and sheet doping concentration of the delta-doping layer are defined by the Scheme variables:

```
(define DeltaThick 0.002)   ; [um]
(define SheetCharge 5.4e12) ; [1/cm2]
```

Other geometric, doping, and meshing parameters are accessible in a similar manner.

The meshing strategy is designed to result in a high-quality mesh without excessive node counts for a large range of geometric parameters.

Most of the meshing strategy is defined using regular refinement boxes, which specify the allowed minimum and maximum mesh spacing within a specified area. It is the meshing algorithm that places the actual lines.

To control precisely the vertical mesh spacing in the delta-doping layer, mesh lines are placed explicitly using a list of y-values representing the desired locations with:

```
(sdeaxisaligned:set-parameters "yCuts"
  (list <list of coordinates>))
```

Sentaurus Structure Editor calls the meshing engine Sentaurus Mesh to generate the structure files for Sentaurus Device. Sentaurus Mesh is called from within Sentaurus Structure Editor with:

```
(sde:build-mesh "snmesh" " " "n@node@_half_msh")
```

This command generates a device structure in TDR format, containing doping and grid data. Note that only half of the HEMT structure is created by Sentaurus Structure Editor and meshed with Sentaurus Mesh. It is subsequently reflected about the vertical axis to obtain the full device. The reflection is performed in Sentaurus Structure Editor by a system call to Sentaurus Data Explorer (tdx):

```
(system:command "tdx -mtt -x -ren drain=source
  n@node@_half_msh n@node@_msh")
```

The option -x instructs Sentaurus Data Explorer to reflect the device along an axis defined by $x = x_{min}$. The given half-structure has three contacts: drain, gate, and substrate, which are defined in sde_dvs.cmd. Of these, the gate and substrate contacts touch the axis of reflection and, upon reflection, are extended and thereby preserve their names. However, the drain contact in the reflected half is named drainmirrored by default. This contact is explicitly renamed to source with the Sentaurus Data Explorer command-line option -ren. The device obtained using Sentaurus Structure Editor is shown in Figure 1 on page 7.

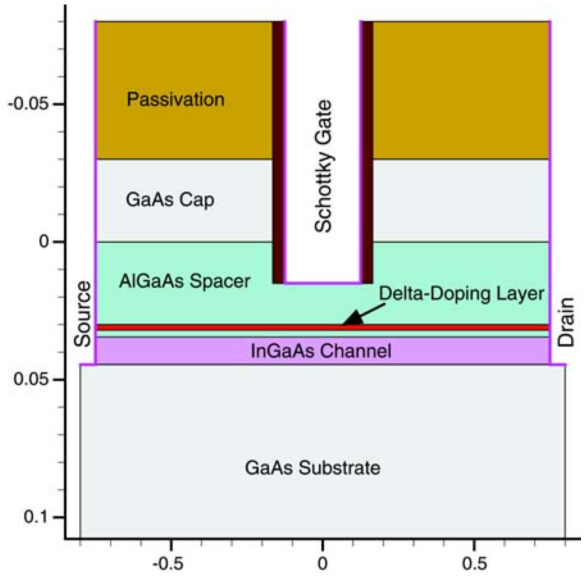

Figure 1 HEMT device generated by Sentaurus Structure Editor. For better viewing, different scales are used for the x-axis and y-axis.

Sentaurus Device

Sentaurus Device is used to simulate the various DC current-voltage as well as the RF characteristics.

The mole fraction in the $In_{(1-x)}Ga_xAs$ channel is set to x = 0.75 with:

```
Physics(Material="InGaAs"){
  MoleFraction(xFraction=0.75 Grading=0)
}
```

The mole fraction in the $Al_xGa_{(1-x)}As$ spacer layer is set to x = 0.3 with:

```
Physics(Material="AlGaAs"){
  MoleFraction(xFraction=0.30 Grading=0)
}
```

In the substrate layer, a donor trap concentration of 10^{16} cm^{-3} is defined with:

```
Physics( Region="R.GaAsSub" ){
  Traps(
    (Donor Conc=1e16 EnergyMid=0.61 fromCondBand
    eXsection=2.5e-18 hXsection=2.5e-16)
  )
}
```

Hydrodynamic transport is used for the electrons. High-field mobility saturation is selected. Furthermore, Shockley–Read–Hall (SRH), Auger, and radiative recombination models are activated. For the breakdown simulations, the avalanche generation model is used as well.

Sentaurus Device selects automatically the appropriate driving-force model for the high-field saturation model and the impact ionization model as follows:

```
Physics {
  ...
  Mobility(
    ...
    eHighFieldSaturation
  )
  Recombination(
    ...
    eAvalanche
  )
}
```

Both the high-field saturation model and impact ionization model are activated only for electrons. Since the simulation is performed using the hydrodynamic transport model for electrons, Sentaurus Device automatically selects `CarrierTempDrive` as the driving-force model for electrons.

At the Ohmic source and drain contact, a contact resistance of 150 Ω µm is set. The gate is defined as a Schottky contact with a Schottky barrier height of 0.9 eV and electron and hole recombination velocities of 10^7 cm/s :

```
Electrode {
{ Name="source" Voltage=0.0 Resistor= 150 }
{ Name="drain"  Voltage= 0.0 Resistor= 150 }
{ Name="gate"   Voltage=0.0 Schottky Barrier=0.9
eRecVelocity=1.e7 hRecVelocity=1.e7 }
...
}
```

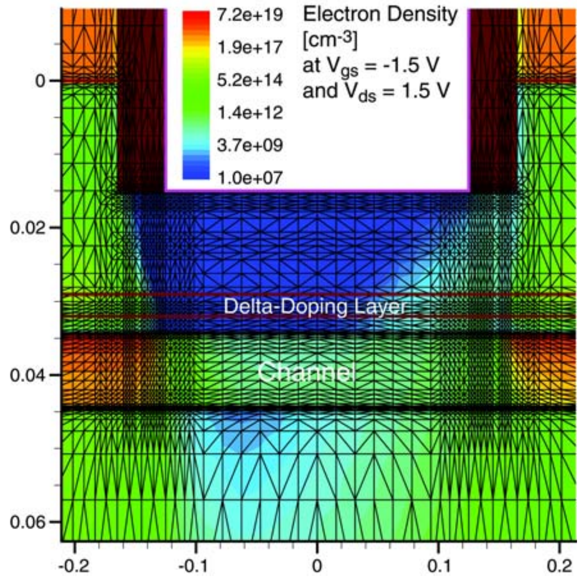

Figure 2 Electron concentration under the gate at $V_{gs} = -1.5$ V and $V_{ds} = 1.5$ V. The mesh in this critical area is shown. For better viewing, different scales are used for the x-axis and y-axis.

To track accurately the electron and hole concentrations, the error control parameter `ErrRef(Electron|Hole)` is reduced from the default value of 10^{10} cm^{-3} to 10^7 cm^{-3} for the electrons and 10^4 cm^{-3} for the holes. In regions with very low concentrations, such as the depletion region under the Schottky contact, the carrier temperature is not a well-defined quantity. It is possible that the numeric solution of the partial differential equation (PDE) results in nonphysically high temperatures in such regions, which can have a negative effect on convergence.

To suppress such artifacts, the energy relaxation time is effectively reduced in regions where the carrier density is less than `RelTermMinDensity`:

```
Math{...
  ErrRef(Electron)   = 1e7
  ErrRef(Hole)       = 1e4
  RelTermMinDensity = 1e4
  RelTermMinDensityZero = 1e7
}
```

(`RelTermMinDensityZero` controls the reduction of the energy relaxation time in low density regions in which artificial cooling occurs.)

DC Characterization

For the I_d–V_{gs} sweeps (see Figure 3), the gate and drain are first ramped together to the bias conditions defined by the Sentaurus Workbench parameters `Vgmax` and `Vd`. Then, the gate is ramped to the final bias point defined by the Sentaurus Workbench parameter `Vgmin`. (Note that, in HEMT devices, often faster convergence can be obtained by sweeping an open device into pinch-off, instead of starting at pinch-off.)

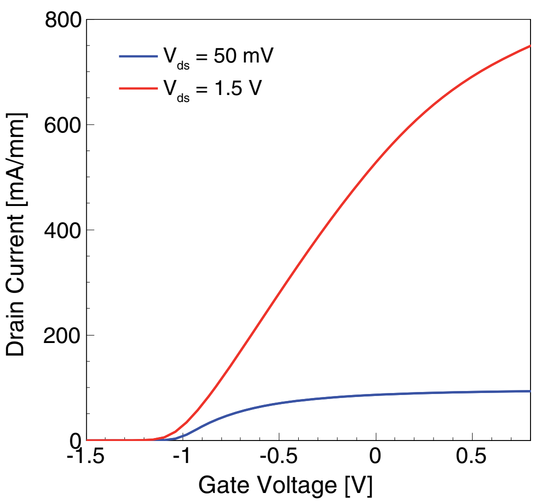

Figure 3 Drain current as function of gate voltage for a drain bias of V_{ds} = 50 mV (blue) and 1.5 V (red)

For the I_d–V_{ds} sweeps (see Figure 4), first the gate is ramped to `Vgmin` and then to `Vgmax`. During the sweep from `Vgmin` to `Vgmax`, the solution is saved at a number of equidistant bias points. This number is set by the Sentaurus Workbench parameter `IdVd`. The solutions are reloaded one by one and a drain bias sweep is performed.

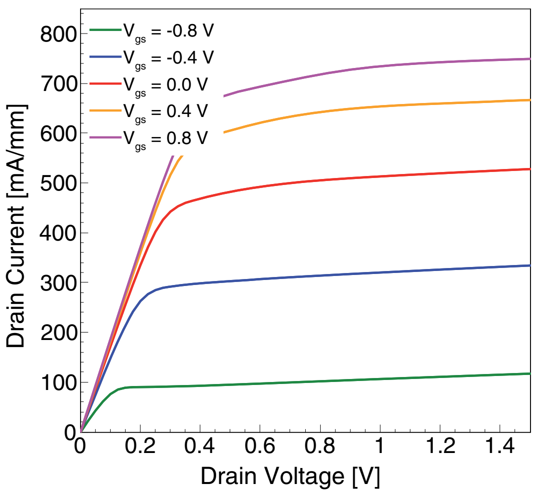

Figure 4 Drain current as function of drain voltage for a gate bias of V_{gs} = –0.8, –0.4, 0.0, 0.4, and 0.8 V

Breakdown Simulations

For the breakdown simulations, the Ohmic contact at source is set to $150\ \Omega\ \mu\text{m}$. The drain contact resistance is set to $150\ \Omega\ \mu\text{m}$ for the drain-current injection; while for the continuation method, it is estimated using a built-in algorithm. The gate is defined as a Schottky contact with a Schottky barrier height of 0.9 eV and electron and hole recombination velocities of 10^7 cm/s :

```
Electrode{
{ Name="source" Voltage=0.0 Resistor= 150 }
#if @Id@ == 0
   { Name="drain"  Voltage= 0.0 }
#else
   { Name="drain"  Voltage= 0.0 Resistor= 150 }
#endif
{ Name="gate"   Voltage=0.0 Schottky Barrier=0.9
eRecVelocity=1.e7 hRecVelocity=1.e7 }
...
}
```

As previously discussed, two different approaches are used for the breakdown simulations. In the direct approach (see Figure 5 on page 9), a load-line resistor is attached to the drain contact and the drain is ramped to a very large value. This load-line resistor is estimated at every bias point using the continuation method. After the onset of impact ionization, most of the voltage drop occurs across this resistor if the value chosen is sufficiently large. Therefore, this simple technique achieves an automatic switching from a voltage-controlled prebreakdown regime to a current-controlled post-breakdown regime.

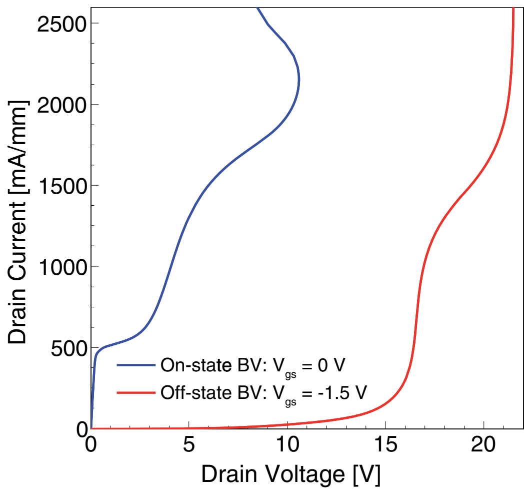

Figure 5 On-state and off-state breakdown I–V characteristics calculated using the direct method: blue curve is I_d–V_{ds} at $V_{gs} = 0$ V and $V_{ds} = 200$ V (on-state breakdown) and red curve is I_d–V_{ds} at $V_{gs} = -1.5$ V and $V_{ds} = 400$ V (off-state breakdown)

The breakdown I–V characteristics can be seen by plotting the terminal current versus the *inner* contact voltage instead of the *outer* contact voltage. The appropriate value for the attached resistor is computed using the continuation method. This technique is used here for the on-state as well as the off-state breakdown simulations.

The implementation of the continuation method is based on a dynamic load-line technique for adapting the boundary conditions along the traced curve to ensure convergence. The boundary condition consists of an external voltage applied to the load resistor. The boundary conditions are generated automatically by the algorithm.

For further details on the continuation method, refer to the *Sentaurus™ Device User Guide*.

The input file segment for the continuation method is described here:

```
NewCurrentFile="IdVd_"
Continuation (
  Name="drain"
  Normalized
  InitialVstep= 0.001
  MaxVoltage= 200
  MinVoltage= 0
  MaxCurrent= 1e3
  MinCurrent= 0
  Iadapt= 3e-14
)
```

Here, the keyword `Name` refers to the continuation electrode where the external voltage is being applied. This node cannot be connected to any other external circuit element. The keyword `Normalized` activates in Sentaurus Device a more complex step control, based on both convergence and curve smoothness.

The initial step of the simulation is mentioned using the parameter `InitialVstep`. The tracing window is specified by user-defined lower and upper values for the voltage and current of the operating point at the continuation electrode: `MinVoltage`, `MaxVoltage`, `MinCurrent`, and `MaxCurrent`. The simulation ends when the operating point is outside the tracing window.

At low-biasing voltages, the current at the continuation electrode is very small and, in general, is too noisy to trace. To overcome this problem, Sentaurus Device divides the continuation window by the parameter `Iadapt`. In the low-current regime, defined as the current between `MinCurrent` and `Iadapt`, the adaptive algorithm is switched off and a fixed value of the load resistor equal to 10^{-3} Ω is used instead. Above the current value of `Iadapt`, the automatic load-resistor algorithm is activated. For breakdown simulations, the default `Iadapt` value of 3×10^{-14} A is used.

An alternative approach for breakdown simulations uses the drain-current injection method. With the drain-current injection method [1] (see Figure 6), the HEMT is first biased to a nonzero drain bias, given by the Sentaurus Workbench parameter `Vd`, in order to have a nonzero current flow. Then, the contact boundary condition is switched from voltage controlled to current controlled with:

```
Set ( "drain" mode Current )
```

The drain current is ramped to a value given by the Sentaurus Workbench parameter `Id`. Finally, the gate is closed by ramping it to a large negative value, given by the Sentaurus Workbench parameter `Vg`.

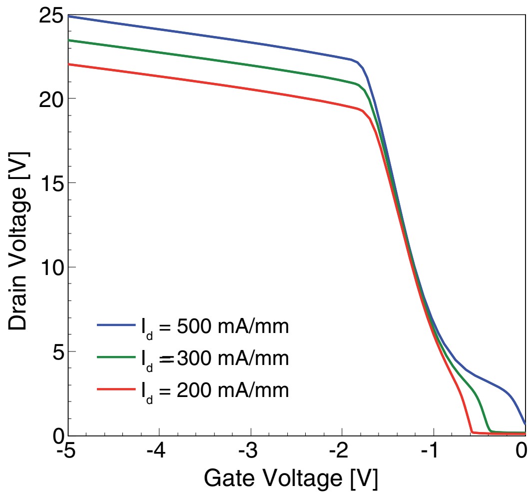

Figure 6 Off-state breakdown characteristics calculated using the drain-current injection method: drain voltage as function of gate voltage for a constant drain current level of $I_d = 500$ (blue), 300 (green), and 200 (red) mA/mm

Near pinch-off, the channel resistivity rises sharply. To maintain the current level, the drain voltage must rise sharply. At some point, impact ionization sets in and the device resistivity stabilizes. Then, the drain voltage saturates. (The residual slope of the drain voltage as a function of the gate bias originates from the resistance of the substrate layer, which now carries most of the current.)

RF Characterization

For the simulation of the RF characteristics, the same biasing and sweeping scheme as for the I_d–V_{gs} simulation is used. At equidistant bias points, Sentaurus Device performs a small-signal (AC) analysis for various frequencies, and it stores the small-signal conductances and capacitances for all contact-to-contact combinations in the Sentaurus Device AC data file. During postprocessing with Sentaurus Visual, this data is used to compute various RF characteristics and to extract various RF parameters (see RF Extractions Using Sentaurus Visual on page 11).

A mixed-mode environment is required for AC simulation in Sentaurus Device, that is, instead of simulating an isolated HEMT, the HEMT is embedded in an external circuit. Here, a two-port network circuit configuration (see Figure 7 on page 11) is used, in which voltage sources are attached to the gate (port 1) and drain (port 2) terminals. All other terminals are grounded. The circuit is defined in the `System` section:

```
System {
  HEMT hemt (gate=1 drain=2 source=0 substrate=0)
  Vsource_pset vg ( 1 0 ) { dc = 0 }
  Vsource_pset vd ( 2 0 ) { dc = 0 }
#if @Rfb@ != 0
  Resistor_pset Rfb (1 2) { resistance = @Rfb@ }
#endif
  }
```

If the Sentaurus Workbench parameter `Rfb` is set to a nonzero value, an external feedback resistor is also included in the circuit.

The AC analysis is activated with the `ACCoupled` statement:

```
Quasistationary( ...
  Goal{ Parameter=vg.dc Voltage=@Vgmax@ }
  ){ ACCoupled(
    StartFrequency= 1e8 EndFrequency= 1e12
    NumberOfPoints= @<nPerDecade*4+1>@ Decade
    Node(1 2) Exclude(vg vd)
    ACCompute( Time=(Range=(0 1) Intervals=20) )
  ){ Poisson Electron Hole eTemperature }
  CurrentPlot(Time=(Range=(0 1) Intervals= 20))
}
```

The `ACCompute` statement defines that the AC analysis will be performed at 21 evenly spaced bias points between the starting bias point `Vgmin` and the bias end point `Vgmax`. The frequency range is set to a 4 decade range from 100 MHz (`1e8`) to 1 THz (`1e12`). This range is sampled by 25 logarithmically distributed points (that is, 6 points/decade and the last frequency point).

As a result of AC analysis, Sentaurus Device stores the small-signal conductances and capacitances for all combinations of gate and drain contacts in the Sentaurus Device AC data file, which is specified in the global `File` section using the keyword `ACExtract`.

DC Extractions Using Sentaurus Visual

In the Sentaurus Workbench tool flow, Sentaurus Device is followed by the Sentaurus visualization tool Sentaurus Visual. This tool plots the corresponding I–V characteristics and extracts relevant electrical parameters, as discussed in General Simulation Setup on page 3.

Refer to the "Visualizing Simulation Results Using Sentaurus Visual" section of *Sentaurus Technology Template: CMOS Characterization,* for an example that illustrates the Sentaurus Visual commands for:

- Plotting a curve.
- Visualizing curves from multiple Sentaurus Visual nodes in a single Sentaurus Visual tool instance.

The DC parameter extractions are performed using the extraction library of Sentaurus Visual. The extraction library is loaded automatically when Sentaurus Visual starts. However, if you have disabled the automatic loading of extension libraries, you can load the extraction library explicitly with the command:

```
load_library extract
```

Refer to the *Sentaurus™ Visual User Guide* for more information on the Sentaurus Visual commands and the procedures of the extraction library. For an introduction to the extraction library, refer to the "Extraction Using Extraction Library of Sentaurus Visual" section of *Sentaurus Technology Template: CMOS Characterization*.

Scaling I–V Data

By default, for 2D simulations, Sentaurus Device uses the current units of A/µm if no explicit `AreaFactor` is specified in the `Physics` section. For HEMT devices, it is common to use the units mA/mm. Rescaling to these units is performed with the extraction library procedure `ext::LinTransList`:

```
set Ids [get_variable_data "drain TotalCurrent" \
  -dataset PLT($N)]
ext::LinTransList -out scaledIds -x $Ids -m 1e6
```

The Tcl variable `Ids` contains the drain current values in units of A/µm. The keyword `-m` specifies the unit conversion factor (`-m 1e6`). The drain current values in units of mA/mm are stored in the Tcl variable `scaledIds`.

Extracting Parameters From I_d–V_{gs} and I_d–V_{ds} Curves

The procedure `ext::ExtractVtgm` extracts the threshold voltage from the I_d–V_{gs} curve using the maximum transconductance method as follows:

```
ext::ExtractVtgm -out Vt -name "Vtgm" -v $Vgs \
  -i $scaledIds
```

The I_d–V_{gs} curve is represented by two Tcl lists: `Vgs` and `scaledIds`. The `Vgs` list contains the gate voltage points, and the `scaledIds` list contains the corresponding drain current values in units of mA/mm. The keywords `-v` and `-i` of the `ext::ExtractVtgm` procedure are set to `Vgs` and `scaledIds` (`-v $Vgs -i $scaledIds`), respectively.

Similarly, the `ext::ExtractVti`, `ext::ExtractGm`, and `ext::ExtractExtremum` procedures are used to extract the threshold voltage for a given subthreshold current level, maximum transconductance, and the maximum value of the drain current, respectively, from the I_d–V_{gs} curve.

By default, the procedure `ext::ExtractExtremum` extracts the maximum of a curve:

```
ext::ExtractExtremum -out Idmax -name "IdMax" \
  -x $Vgs -y $scaledIds
```

The on-state resistance is extracted from the I_d–V_{ds} curve using the procedure `ext::ExtractRdiff`.

Extracting Breakdown Voltage

The breakdown voltages `BVv` and `BVi` are extracted using the extraction library procedures `ext::ExtractBVv` and `ext::ExtractBVi`, respectively:

```
set Vds [get_variable_data "drain InnerVoltage" \
  -dataset PLT($N)]
ext::ExtractBVv -out BVv -name "out" -v $Vds \
  -i $scaledIds -sign 1
ext::ExtractBVi -out BVi -name "out" -v $Vds \
  -i $scaledIds -io 1000
```

The I–V curve is represented by two Tcl lists: `Vds` and `scaledIds`. The `Vds` list contains the drain inner voltage points, and the `scaledIds` list contains the corresponding drain current values in units of mA/mm.

For the procedure `ext::ExtractBVv`, `-sign 1` is used since the breakdown occurs at positive bias. For the procedure `ext::ExtractBVi`, the current level at which breakdown occurs is specified as `1000` mA/mm (`-io 1000`).

The breakdown voltage `BVc` is extracted using the procedure `ext::ExtractValue`:

```
ext::ExtractValue -out BV -name "BVc" \
  -x $Vgs -y $Vds -xo -1.5 -f "%.3f"
```

Here, the Tcl lists `Vgs` and `Vds` contain the gate voltage and the drain voltage values, respectively. The gate voltage value used to compute the breakdown voltage is specified using the keyword `-xo` as –1.5 V.

RF Extractions Using Sentaurus Visual

The RF parameter extractions are performed using the RF extraction library of Sentaurus Visual. The RF extraction library is loaded automatically when Sentaurus Visual starts. However, if you have disabled the automatic loading of extension libraries, you can load the RF extraction library explicitly with the command:

```
load_library rfx
```

Refer to the *Sentaurus™ Visual User Guide* for more information on the Sentaurus Visual commands and the procedures of the RF extraction library.

The RF extraction library assumes that the transistor can be modeled by a two-port network as shown in Figure 7.

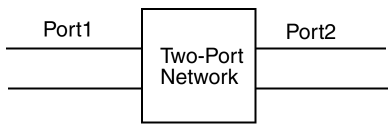

Figure 7 Schematic of two-port network

Loading Sentaurus Device AC Data File and Creating Admittance Matrix

As discussed in RF Characterization on page 10, the Sentaurus Device AC data file contains the conductance and the capacitance at each bias and frequency point for all contact-to-contact combinations included in the small-signal analysis (here, `gate` and `drain`).

The Sentaurus Device AC data file is loaded with the procedure `rfx::Load`:

```
set W @W@
rfx::Load -dataset "ACPLT($N)" -file "@acplot@" \
  -port1 1 -port2 2 -biasport "v(1)" -devicewidth $W
```

The Sentaurus Device AC data file is specified with the keyword `-file`. The port names are specified with the keywords `-port1` and `-port2`. They must agree with the node names defined in the Sentaurus Device `System` section. The keyword `-biasport` defines which port is biased and whether the biasing is a voltage or a current condition. The keyword `-devicewidth` is the device width multiplier W and it should be specified only for 2D devices. It allows you to specify the actual width of the device in the z-direction if no `AreaFactor` has been defined in Sentaurus Device. Its default value is 1 µm. Some parameters such as h_{21} scale trivially with the device width. For the unilateral figure of merit and AC parameters such as conductance, capacitance, and S-parameters, the device width is important. Here, W is set to the Sentaurus Workbench parameter `W`, which is set to `25.0`.

NOTE Avoid applying the device width scaling twice by using only one of the two scaling methods.

The `rfx::Load` procedure creates the Sentaurus Visual dataset that contains the data from the Sentaurus Device AC data file. The name of this dataset is specified using the keyword `-dataset`.

The `rfx::Load` procedure creates several namespace variables (Tcl arrays and variables) described in [2]. It creates the Tcl array `rfx::AC` that contains the conductance coefficients (a_{ij}) and the capacitance coefficients (c_{ij}) of the conductance matrix (A-matrix) and the capacitance matrix (C-matrix), respectively. This data is used to compute the admittance coefficients (Y_{ij}) of the complex admittance matrix (Y-matrix), which is stored in the Tcl array `rfx::Y`.

Identifying Bias Point or Frequency Index

Internally, the RF extraction library stores the small-signal data and the Y-matrix as a function of bias and frequency in the form of a two-dimensional array [2]. To access the small-signal data or RF parameter for a given bias or frequency, the appropriate array index (frequency index or bias index) must be given. Many of the RF extraction library procedures (for example, `rfx::GetPowerGain`) internally call the procedure `rfx::GetNearestIndex` to obtain the relevant index.

This procedure operates on a list of bias points (`rfx::bias`) or a list of frequencies (`rfx::freq`). Both the `rfx::bias` and `rfx::freq` lists are created when loading the Sentaurus Device AC data file using `rfx::Load`.

For example, to find the index of the bias point closest to 0.025 V, use:

```
rfx::GetNearestIndex -out i_bias -target 0.025 \
  -list $rfx::bias
```

The keyword `-target` specifies the bias point or the frequency whose index is required. Here, it specifies the bias point.

The information about the bias associated with the returned index is printed in the Sentaurus Visual run-time output file (batch mode) or in the Tcl Command panel (interactive mode):

```
!rfx::GetNearestIndex: Nearest Index is 11.
!  Corresponding value is 5.0E-02.
!  Target value is 0.025.
```

The actual bias point can be accessed in the Sentaurus Visual script with:

```
set BiasPoint [lindex $rfx::bias $i_bias]
```

The identification of a frequency index works in the same way.

For example, to find the index for a frequency close to 1 GHz, use:

```
rfx::GetNearestIndex -out i_freq -target 1e9 \
  -list $rfx::freq
set Frequency [lindex $rfx::freq $i_freq]
```

To access a parameter as a function of bias, specify the frequency index (see Creating Sentaurus Visual Dataset for Small-Signal Data); whereas, to access the same parameter as a function of frequency, specify the bias index.

Creating Sentaurus Visual Dataset for Small-Signal Data

The Y-matrix is converted to a hybrid (h-)matrix, a scattering (S-)matrix, or an impedance (Z-)matrix using internal matrix conversion procedures [2]. The RF parameters Y_{ij}, h_{ij}, S_{ij}, and Z_{ij} are the coefficients (or matrix elements) of the Y-, h-, S-, or Z-matrix, respectively.

The procedure `rfx::CreateDataset` creates the Sentaurus Visual dataset containing the small-signal data as a function of frequency or bias. This dataset can be used to visualize the small-signal data. The names of the variables in this Sentaurus Visual dataset are described in [2].

For example, the following commands plot C_{gd} of the HEMT device as a function of gate bias (see Figure 8 on page 13) at a frequency of 100 MHz:

```
rfx::CreateDataset -dataset "AC_bias($N)" \
  -xaxis "bias" -rfmatrix "AC"
create_plot -1d -name Plot_CV
# Specify frequency index
set index 0
set freq [lindex $rfx::freq 0]
create_curve -name CgdlfTmp($N) \
  -dataset "AC_bias($N)" \
  -axisX "frequency_${index} bias" \
  -axisY "frequency_${index} c(1,2)" \
create_curve -name Cgdlf($N) \
  -function "abs(<CgdlfTmp($N)>)"
remove_curves "CgdlfTmp($N)"
```

The Sentaurus Visual dataset created by `rfx::CreateDataset` is specified using the `-dataset` keyword. Since the keyword `-xaxis` is set to `"bias"` and the keyword `-rfmatrix` is set to `"AC"`, the dataset `"AC_bias"($)` contains a_{ij} and c_{ij} as a function of bias for all frequencies. After the dataset is created, Sentaurus Visual commands are used to create the C_{gd} versus bias curve shown in Figure 8.

If the keyword `-rfmatrix` is set to either `"Y"`, `"H"`, `"S"`, or `"Z"`, the `rfx::CreateDataset` procedure creates a dataset containing the real part, the imaginary part, the magnitude or the absolute value (linear or dB scale), and the phase of the RF parameters (all components) of the RF matrix, as a function of frequency or bias.

For example, if `-rfmatrix "Y"` is specified, the dataset contains the values for the Y-parameters Y_{11}, Y_{12}, Y_{21}, and Y_{22}.

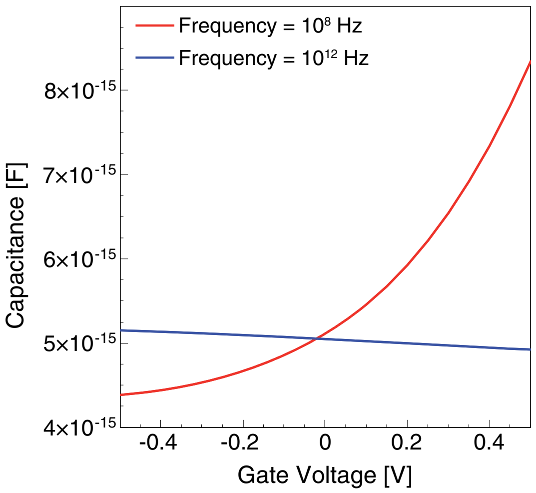

Figure 8 C_{gd} as a function of gate bias for frequency of 100 MHz (red) and 1 THz (blue), at a drain bias of V_{ds} = 1 V; data is computed without a feedback resistor

By default, the `rfx::CreateDataset` procedure computes the magnitude of an RF parameter on a linear scale. To compute the magnitude on a dB scale, specify the keyword `-dB`, which can take the values `0` (default, linear scale), `10` (10 dB scale), or `20` (20 dB scale).

Internally, `rfx::CreateDataset` calls the procedure `rfx::GetRFCList` to access a matrix element of the complex Y-, h-, S-, or Z-matrix as a function of either frequency or bias. If the keyword `-rfmatrix` is set to either `"H"`, `"S"`, or `"Z"`, the `rfx::CreateDataset` procedure internally calls the matrix conversion procedures [2].

NOTE In a Sentaurus Visual script, you must ensure that the names of the datasets created by different RF extraction library procedures are unique.

Plotting RF Parameters: Rectangular Plots

The Sentaurus Visual dataset created by `rfx::CreateDataset` can be used to plot the real part, the imaginary part, the magnitude, and the phase of an RF parameter as a function of frequency or bias. For example, to plot h_{21} as a function of frequency at a gate bias of 0 V (see Figure 9), use:

```
set RFmatrix "H"
set RFparameter "h21"
set xaxis "frequency"
set dataset "2port_${RFmatrix}_${xaxis}($N)"
create_plot -1d -name Plot_${RFparameter}_${xaxis}
rfx::CreateDataset -dataset $dataset \
  -xaxis $xaxis -rfmatrix $RFmatrix
rfx::GetNearestIndex -out i_bias -target 0.0 \
  -list $rfx::bias
set BiasPoint [lindex $rfx::bias $i_bias]

# Plot real part of h21
create_curve -name Re_${RFparameter}_${xaxis}($N) \
  -dataset $dataset \
  -axisX "bias_${i_bias} frequency"
  -axisY "bias_${i_bias} ${RFparameter}_Re"

# Plot imaginary part of h21
create_curve -name Im_${RFparameter}_${xaxis}($N) \
  -dataset $dataset \
  -axisX "bias_${i_bias} frequency" \
  -axisY "bias_${i_bias} ${RFparameter}_Im"

# Plot magnitude of h21 (linear scale)
create_curve -name Abs_${RFparameter}_${xaxis}($N) \
  -dataset $dataset \
  -axisX "bias_${i_bias} frequency" \
  -axisY "bias_${i_bias} ${RFparameter}_Abs"

# Plot phase of h21
create_curve -name Phase_${RFparameter}_${xaxis}($N)\
  -dataset $dataset \
  -axisX "bias_${i_bias} frequency" \
  -axisY2 "bias_${i_bias} ${RFparameter}_Phase"
```

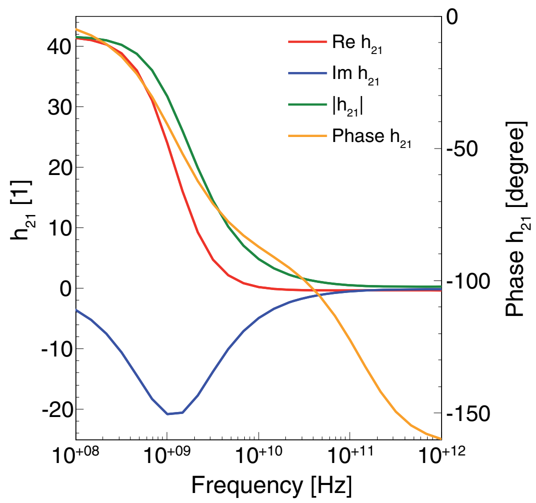

Figure 9 RF parameter h_{21} as a function of frequency for a gate bias of V_{gs} = 0 V and a drain bias of V_{ds} = 1 V: real part (red), imaginary part (blue), magnitude (green), and phase (orange). This data is computed with a feedback resistor of 100 kΩ µm.

To plot $|h_{21}|$ on a 20 dB scale, the above commands are modified as follows:

```
set dB 20
rfx::CreateDataset -dataset $dataset \
  -xaxis $xaxis -rfmatrix $RFmatrix -dB $dB

# Plot magnitude of h21 (20 dB scale)
create_curve -name Abs_${RFparameter}_${xaxis}($N) \
  -dataset $dataset \
  -axisX "bias_${i_bias} frequency" \
  -axisY "bias_${i_bias} ${RFparameter}_Abs_20dB"
```

Any RF parameter can be plotted by specifying the proper value of the Tcl variables `RFmatrix` (Y, H, S, Z) and

RFparameter, respectively. For RFparameter, valid choices are, for example, "y11", "z12", "h21", and "s22".

Plotting an RF parameter as a function of bias works in a similar way. Plotting a family of curves works in a similar way as for power gains (see Power Gains, Unilateral Figure of Merit, and Stability Criteria on page 16), except that the bias or frequency indices are looped over using a Tcl for loop.

Plotting RF Parameters: Smith Charts

For a Smith chart, the imaginary part of an RF parameter is plotted against the real part on a backdrop of two families of curves: the normalized resistance circles (R-circles) and the normalized reactance (capacitive or inductive) arcs (X-arcs).

The backdrop is generated with the procedure rfx::SmithBackdrop using two lists specified using the keywords -r and -x. The normalized resistance and reactance values are specified using -r and -x, respectively.

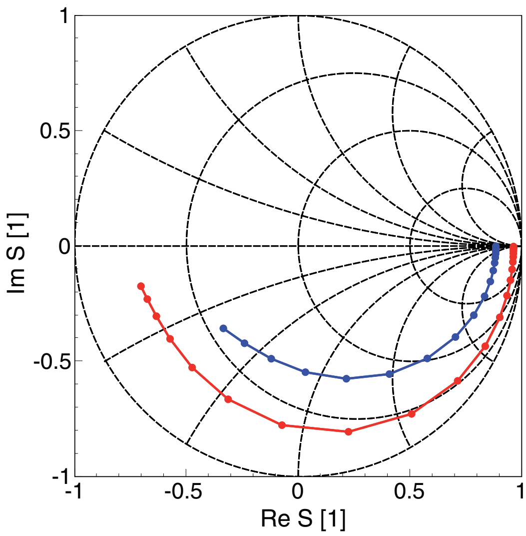

Figure 10 RF parameters S_{11} (red curve) and S_{22} (blue curve) are shown on a Smith chart for a gate bias of 0 V, a drain bias of V_{ds} = 1 V, and frequencies between 100 MHz and 1 THz. This data is computed with a feedback resistor of 100 kΩ µm.

For example, the backdrop shown in Figure 10 is generated with the command:

```
set Rs [list 0 0.3333 1.0 3.0]
set Xs [list 0.268 0.575 1 1.73 3.75]
rfx::SmithBackdrop -plot "Plot_Smith_freq" \
  -r $Rs -x $Xs
```

For example, to plot S_{11} on a Smith chart as a function of frequencies at gate bias of 0 V, use:

```
rfx::CreateDataset -dataset "2port_S_freq($N)" \
  -xaxis "frequency" -rfmatrix "S"

# Get bias index
rfx::GetNearestIndex -out i_bias -target 0.0 \
  -list $rfx::bias

create_curve -name s11($N) \
  -dataset "2port_S_freq($N)" \
  -axisX "bias_${i_bias} s11_Re" \
  -axisY "bias_${i_bias} s11_Im"
```

To plot a different RF parameter such as S_{22} on a Smith chart, replace s11 with s22 in the script segment shown above. To plot the parameter as a function of bias for a fixed frequency, use -xaxis "bias".

Plotting RF Parameters: Polar Plots

For a polar plot, the imaginary part of an RF parameter is plotted against the real part on a backdrop of a family of concentric circles and a family of lines radiating from the origin. The polar backdrop (see Figure 11) is generated with the procedure rfx::PolarBackdrop using two lists:

```
set Rs [list 0.25 0.5 0.75 1.0 1.25]
set Phis [list 0 30 60 90 120 150]
rfx::PolarBackdrop -plot "Plot_Polar" \
  -r $Rs -phi $Phis
```

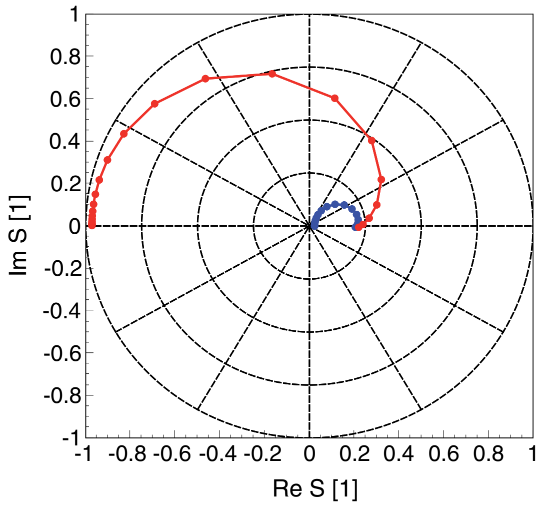

Figure 11 RF parameters S_{21} (red curve) and S_{12} (blue curve) are shown on a polar plot for a gate bias of 0 V, a drain bias of V_{ds} = 1 V, and frequencies between 100 MHz and 1 THz. This data is computed with a feedback resistor of 100 kΩ µm.

The list Rs specified using the keyword -r contains the radii of the concentric circles to be drawn, and the list Phis specified using the keyword -phi contains the angles for which lines are to be drawn.

Plotting the RF parameters works in the same way as for the Smith charts (see Plotting RF Parameters: Smith Charts).

Extracting Cut-off Frequency

As discussed in Plotting RF Parameters: Rectangular Plots on page 13, the current gain can be plotted as a function of either frequency or bias using the dataset created by the rfx::CreateDataset procedure.

[Figure 12](#) shows the current gain curves on the dB scale for V_{gs} = –0.5 V to 0.5 V for the HEMT device without feedback resistor. Plotting such a family of curves is discussed in Power Gains, Unilateral Figure of Merit, and Stability Criteria on page 16.

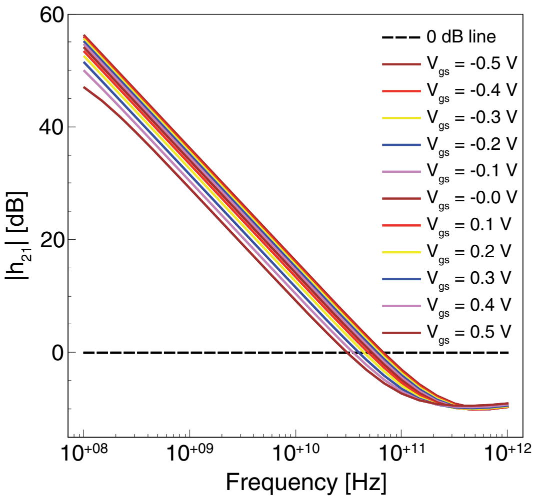

Figure 12 Current gain (dB scale) as a function of frequency for various gate voltages between V_{gs} = –0.5 V and 0.5 V at a drain bias of V_{ds} = 1 V. This data is computed without a feedback resistor.

The procedure `rfx::GetFt` extracts the cut-off frequency f_t from the current gain curves using three different methods. For details, refer to [2].

For example, the following command extracts the f_t values as a function of bias using the `unit-gain-point` method:

```
rfx::GetFt -out Ft0 -method "unit-gain-point" \
  -scale 1e9 -xscale "log" -dataset "ft0_bias($N)" \
  -slopedataset "h21_slope_freq($N)"
```

If `-slopedataset` is specified, `rfx::GetFt` creates a Sentaurus Visual dataset containing the derivative of the current gain (in units of dB/decade) as a function of frequency.

Here, `-slopedataset "h21_slope_freq($N)"` is used. The dataset `"h21_slope_freq($N)"` can be used to plot the derivative of current gain versus frequency curves (see [Figure 13](#)).

NOTE Since the derivative of the current gain curve is sensitive to the frequency interval, to increase the accuracy of the value of the derivatives, you can increase the number of frequency points by using the Sentaurus Workbench variable `nPerDecade`. This will help to identify the –20 dB/decade region more accurately and also will result in a more accurate extraction of f_t using the extrapolation methods. These remarks apply to the extraction of f_{max} as well (see Extracting Maximum Frequency of Oscillation on page 18).

If the keyword `-dataset` is specified, `rfx::GetFt` creates a Sentaurus Visual dataset containing f_t values as a function of bias. The dataset contains two variables, `ft` and `bias`, which can be used to visualize the bias dependency of f_t using Sentaurus Visual commands (see Figure 14 on page 16). The optional keyword `-scale` allows you to change the unit of f_t. Here, the unit is set to GHz by setting `-scale 1e9`.

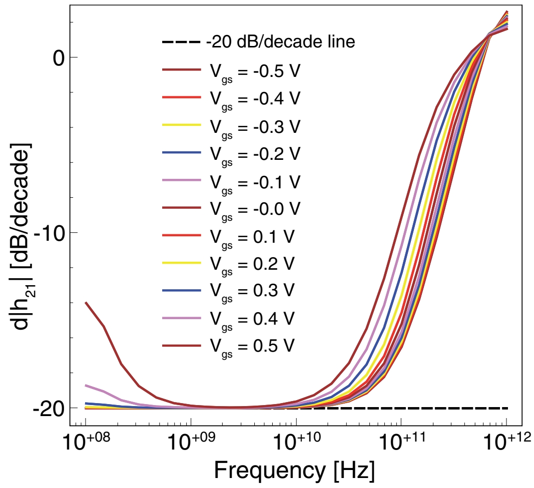

Figure 13 Derivative of current gain as a function of frequency for various gate voltages between V_{gs} = –0.5 V and 0.5 V at a drain bias of V_{ds} = 1 V. This data is computed without a feedback resistor.

The keyword `-method` specifies the extraction method:

- `-method "unit-gain-point"` returns the frequency at which $|h_{21}| = 1$ or $|h_{21}| = 0\,[\text{dB}]$.
- `-method "extract-at-dBPoint"` and `-method "extract-at-frequency"` use the extrapolation method [2] to extract f_t.

In the `extract-at-frequency` method, a frequency point is specified using the keyword `-parameter`:

```
set frequency 2e9
rfx::GetFt -out Ftfq -method "extract-at-frequency" \
  -parameter $frequency -scale 1e9 -xscale "log" \
  -dataset "ftfq_bias($N)"
```

The `extract-at-dBPoint` method looks for the current gain point where the gain has fallen by a certain number of decibels from its value at the start of the gain curve. This difference in decibels, called the *dB point*, is specified using the keyword `-parameter`:

```
set dBPoint 10
rfx::GetFt -out FtdB -method "extract-at-dBPoint" \
  -parameter $dBPoint -scale 1e9 -xscale "log" \
  -dataset "ftdB_bias($N)"
```

The extrapolation methods assume that the gain curve is flat at low frequencies and rolls off at –20 dB/decade at high frequencies. The extrapolation methods assume that the slope is –20 dB/decade at the specified frequency point or dB point as well as at the unit gain point and uses extrapolation to extract f_t.

The frequency point and the dB point can be chosen by examining the derivative of the current gain curve so that these points lie in the –20 dB/decade region. These curves also can be used to verify whether the slope is –20 dB/decade at the unity gain point.

For the HEMT device without feedback resistor, the derivative of the current gain curves (see Figure 13 on page 15) show that the slope is –20 dB/decade in the frequency range 10^9 Hz to 4×10^9 Hz. Therefore, the frequency point of 2×10^9 Hz is specified when using the `extract-at-frequency` method.

For the `extract-at-dBPoint` method, the dB point is chosen so that the frequency corresponding to the dB point is close to the frequency point specified in the method `extract-at-frequency`. Here, the dB point is specified as 25 dB. The frequency corresponding to this dB point for all the current gain curves is printed in the Sentaurus Visual run-time output file (batch mode) or in the Tcl Command panel (interactive mode). For example:

```
# ! rfx::GetFt > Method= "b": For bias= -0.5,
# ! frequency corresponding to dB point= 25 is
1.78e+09
```

The f_t curves obtained from all three methods will match if the assumption of a –20 dB/decade slope is valid.

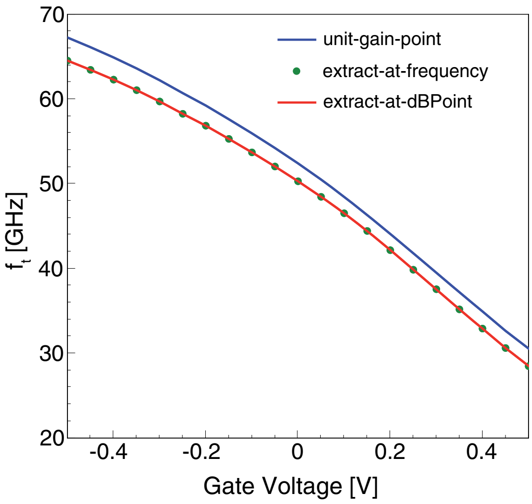

Figure 14 Cut-off frequency f_t as a function of gate bias, extracted with three different methods: unit-gain-point (blue), extract-at-frequency (green), and extract-at-dBPoint (red). This data is computed without a feedback resistor.

Since both the dB point and the frequency point were chosen to lie in the –20 dB/decade region, the f_t curves obtained from the extrapolation methods match (see Figure 14). The small difference between the f_t curves obtained from the extrapolation methods and the `unit-gain-point` method is due to a small deviation of the slope from –20 dB/decade at the unit gain point. Therefore, Figure 14 shows that the f_t curves using all three methods for the HEMT device agree well, and the results can be trusted with a high level of confidence.

As discussed in [2], no single extraction method of f_t is appropriate under all circumstances. Therefore, it is recommended to always use all three methods concurrently.

The procedure `rfx::GetFt` returns an array (specified by the keyword `-out`) with a one string-valued index. The index contains the elements `ft` and `bias`. The values of the `ft` element and the `bias` element are a list of f_t and bias values, respectively.

Here, for the `unit-gain-point` method, `-out Ft0` is used. The lists `Ft0(ft)` and `Ft0(bias)` are used to compute the maximum f_t value and the corresponding bias value using the procedure `ext::ExtractExtremum` of the extraction library [2].

Power Gains, Unilateral Figure of Merit, and Stability Criteria

The procedure `rfx::GetPowerGain` computes K, Δ, U_f, and the power gains (MSG, MAG, and MUG) as a function of frequency (at a fixed bias point) or bias (at a fixed frequency point). The power gain values are computed on both the linear scale and the 10-dB scale.

The dataset created by `rfx::GetPowerGain` can be used to plot the power gains and the stability criteria as a function of frequency or bias.

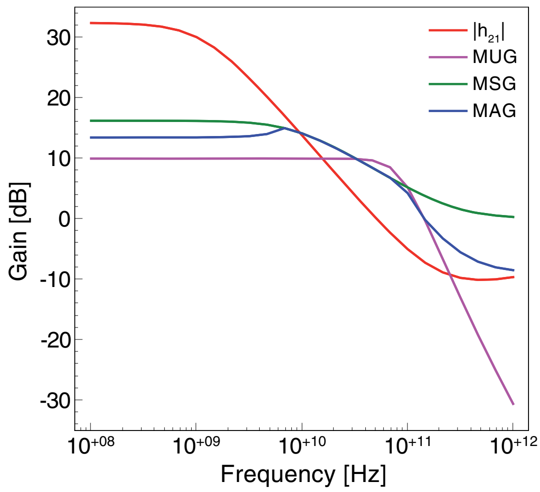

Figure 15 Current gain and various power gains as a function of frequency for a gate bias of V_{gs} = 0 V and a drain bias of V_{ds} = 1 V. This data is computed with a feedback resistor of 100 kΩ µm.

For example, to plot the power gains as a function of frequency at gate bias of 0 V (see Figure 15), use:

```
set N @node@
set target 0.0
set xaxis "frequency"
set dataset "PowerGain_${xaxis}($N)"
# Compute power gain and stability criteria
rfx::GetPowerGain -out PowerGain -dataset $dataset \
  -xaxis $xaxis -target $target -powergain "all"
# Plot MUG [dB] versus frequency
create_curve -name MUG_dB($N) -dataset $dataset \
```

```
  -axisX $xaxis -axisY MUG_dB
# Plot MSG [dB] versus frequency
create_curve -name MSG_dB($N) -dataset $dataset \
  -axisX $xaxis -axisY MSG_dB
# Plot MAG [dB] versus frequency
create_curve -name MAG_dB($N) -dataset $dataset \
  -axisX $xaxis -axisY MAG_dB
...
```

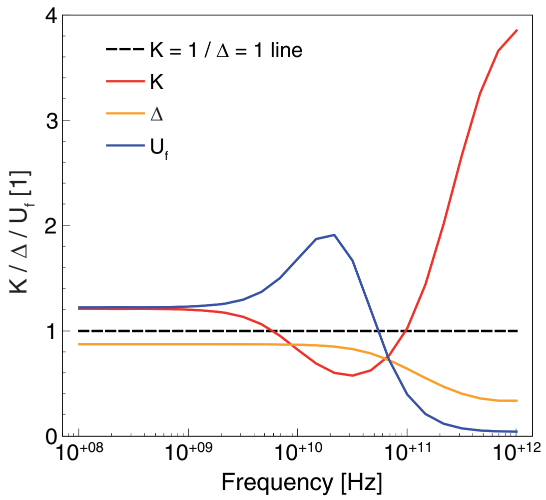

Figure 16 Unilateral figure of merit and stability criteria as a function of frequency for a gate bias of V_{gs} = 0 V and a drain bias of V_{ds} = 1 V. This data is computed with a feedback resistor of 100 kΩ µm.

To plot the unilateral figure of merit and the stability criteria as a function of frequency at a gate bias of 0 V (see Figure 16), use:

```
# Plot K versus frequency
create_curve -name K($N) -dataset $dataset \
  -axisX $xaxis -axisY K
# Plot delta versus frequency
create_curve -name delta($N) -dataset $dataset \
  -axisX $xaxis -axisY delta
# Plot unilateral figure of merit versus frequency
create_curve -name U($N) -dataset $dataset \
  -axisX $xaxis -axisY U
...
```

The power gains computed depend on the `-powergain` keyword. Here, since `-powergain "all"` is used, all the power gains are computed, along with the stability criteria and the unilateral figure of merit. These quantities are stored in the Tcl array `PowerGain` (specified by the `-out` keyword), as well as the `"PowerGain_frequency($N)"` dataset. This dataset is used to plot the power gains and the stability criteria as a function of frequency.

In the above example, the bias at which these quantities are computed is specified using the keyword `-target`. To compute the power gains and the stability criteria as a function of frequency at a particular index point, for example, the fifth bias point index, use the keyword `-index` instead of `-target`:

```
set index 5
rfx::GetPowerGain -out PowerGain -dataset $dataset \
  -xaxis $xaxis -index $index -powergain "all"
```

NOTE In the procedure `rfx::GetPowerGain`, only one of the keywords `-index` or `-target` must be specified.

To compute the power gains and the stability criteria as a function of bias at a particular frequency, for example, 1 GHz, use:

```
set xaxis "bias"
set dataset "PowerGain_${xaxis}($N)"
set target 1e9
rfx::GetPowerGain -out PowerGain -dataset $dataset \
  -xaxis $xaxis -target $target -powergain "all"
```

To plot a family of power gains or stability criteria versus frequency (bias) curves over a bias (frequency) range, the bias (frequency) indices are looped over using a Tcl loop.

For example, to plot MUG curves as a function of frequency over the complete applied bias range from –0.5 V to 0.5 V (see Figure 18 on page 18), use:

```
set N @node@
set Pgain MUG
set xaxis "frequency"
for {set i $rfx::i_biasstart} {$i <= $rfx::i_biasend}\
  {incr i} {
  rfx::GetPowerGain -out PowerGain \
    -xaxis $xaxis -index $i -powergain $Pgain \
    -dataset "${Pgain}_${xaxis}($N,$i)"
  create_curve -name ${Pgain}_dB($N,$i) \
    -dataset "${Pgain}_${xaxis}($N,$i)" \
    -axisX $xaxis -axisY "${Pgain}_dB"
}
```

Here, since `-powergain "MUG"` is used, only MUG and U_f are computed, along with the stability criteria.

Instead of bias or frequency indices, the looping can be performed over a bias list (for example, `rfx::bias`) or a frequency list (for example, `rfx::freq`), using the keyword `-target` of the `rfx::GetPowerGain` procedure. For example, MUG versus frequency curves over the complete applied bias range from –0.5 V to 0.5 V can be plotted using (see Figure 18 on page 18):

```
set N @node@
set Pgain MUG
set xaxis "frequency"
foreach BiasPoint $rfx::bias {
  rfx::GetPowerGain -out PowerGain \
    -xaxis $xaxis -target $BiasPoint \
    -powergain $Pgain \
    -dataset "${Pgain}_${xaxis}($N,$i)"
  create_curve -name ${Pgain}_dB($N,$i) \
    -dataset "${Pgain}_${xaxis}($N,$i)" \
    -axisX $xaxis -axisY "${Pgain}_dB"
}
```

Region of Unconditional Stability

An amplifier is unconditionally stable if $K > 1$ and $\Delta < 1$. These stability criteria can be plotted as a function of either frequency or bias using the `rfx::GetPowerGain` procedure (see Power Gains, Unilateral Figure of Merit, and Stability Criteria on page 16).

The procedure `rfx::GetFK1` computes f_{K1} (frequency at which $K = 1$) at all bias points from the K versus frequency curves. This procedure is useful for plotting f_{K1} versus bias and, therefore, for determining the region of unconditional stability. The boundaries of the region of unconditional stability are defined as the isolines at which $K = 1$. These boundaries are plotted using:

```
set ID "R<sub>fb</sub>= $Rfb"
# Plot the first K=1 isoline
rfx::GetFK1 -out fK1_1 -dataset "fK1_1($N)" \
  -target 1.0 -xscale "log" -occurrence 1 -scale 1e9
create_curve -name fK1_1($N) -dataset "fK1_1($N)" \
  -axisX "bias" -axisY "fK1"
set_curve_prop fK1_1($N) \
  -label "f<sub>K1_1</sub> ($ID)" \
  -color red -line_style solid -line_width 3

# Plot the second K=1 isoline
rfx::GetFK1 -out fK1_2 -dataset "fK1_2($N)" \
  -target 1.0 -xscale "log" -occurrence 2 -scale 1e9
create_curve -name fK1_2($N) -dataset "fK1_2($N)" \
  -axisX "bias" -axisY "fK1"
set_curve_prop fK1_2($N) \
  -label "f<sub>K1_2</sub> ($ID)" \
  -color blue -line_style solid -line_width 3
```

Use the optional keyword `-occurrence` if there is more than one boundary of a region of stability for a given bias point. If, for example, `-occurrence 2` is used, the location of the second occurrence of $K = 1$ is returned. If no second $K = 1$ occurrence is found, 0 is returned instead.

If only the $K = 1$ boundaries are shown, it is unclear as to which side of the boundary the amplifier is stable. Use the optional keyword `-target` to draw another isoline. For example, the isoline $K = 1.1$ is drawn with:

```
# Plot the first K=1.1 isoline
rfx::GetFK1 -out fK1.1_1 -dataset "fK1.1_1($N)" \
  -target 1.1 -xscale "log" -occurrence 1 -scale 1e9
create_curve -name fK1.1_1($N) \
  -dataset "fK1.1_1($N)" -axisX "bias" -axisY "fK1"

# Plot the second K=1.1 isoline
rfx::GetFK1 -out fK1.1_2 -dataset "fK1.1_2($N)"\
  -target 1.1 -xscale "log" -occurrence 2 -scale 1e9
create_curve -name fK1.1_2($N) \
  -dataset "fK1.1_2($N)" -axisX "bias" -axisY "fK1"
```

Figure 17 shows the region of unconditional stability for the HEMT device with a feedback resistor of $R_{fb} = 100\,k\Omega\,\mu m$. The lower frequency $K = 1$ isoline is red and the higher frequency $K = 1$ isoline is blue. To highlight in which regions the device is unconditionally stable, the isolines at which $K = 1.1$ are also shown as green dashes.

Figure 17 shows that the region of unconditional stability lies above the blue curve and below the red curve. At the gate bias of –0.5 V (0.5 V), the device is unconditionally stable below 5.5 GHz (6 GHz) and above 123 GHz (53 GHz).

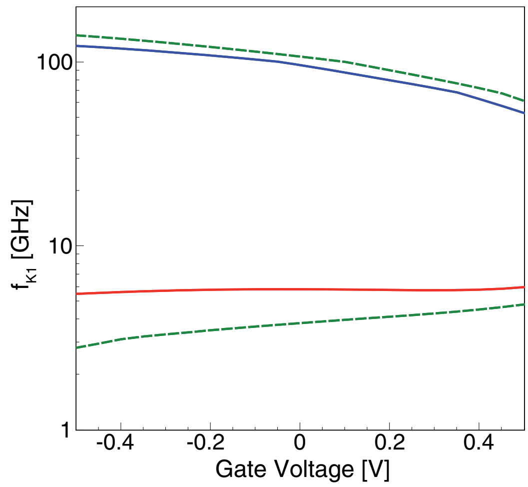

Figure 17 Region of unconditional stability: K=1 isolines are red and blue, K=1.1 isolines are green dashes. This data is computed with a feedback resistor of 100 kΩ µm.

Extracting Maximum Frequency of Oscillation

For a unilateral amplifier design approach, the unilateral figure of merit U_f should be as small as possible. U_f and MUG can be plotted as a function of either frequency or bias using the procedure `rfx::GetPowerGain` (see Power Gains, Unilateral Figure of Merit, and Stability Criteria on page 16). Figure 18 shows MUG curves for the bias range –0.5 V to 0.5 V.

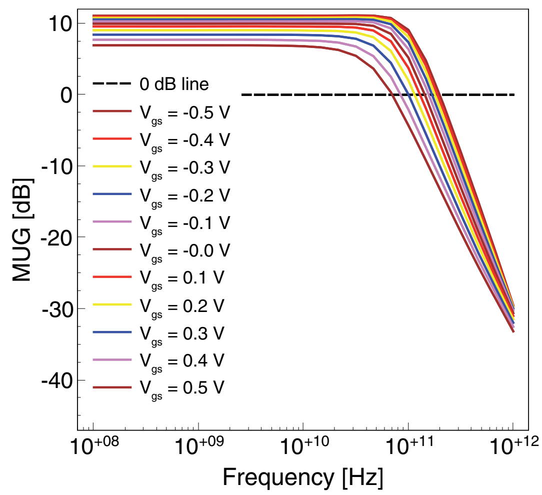

Figure 18 Mason's unilateral gain (dB scale) as a function of frequency for various gate voltages between $V_{gs} = -0.5$ V and 0.5 V at a drain bias of $V_{ds} = 1$ V. This data is computed with a feedback resistor of 100 kΩ µm.

The procedure `rfx::GetFmax` returns a list of f_{max} values. It uses the same methods and control parameters as the procedure `rfx::GetFt`. The main difference is that `rfx::GetFmax` operates on MUG or MAG (specified using the keyword `-powergain`) instead of $|h_{21}|$.

For example, f_{max} can be computed using the `unit-gain-point` method and MUG (`-powergain "MUG"`) as follows:

```
rfx::GetFmax -out Fmax0 -method "unit-gain-point"
  -scale 1e9 -xscale "log" -dataset "fmax0_bias($N)"\
  -slopedataset "MUG_slope_frequency($N)" \
  -powergain "MUG"
```

Similar to the `rfx::GetFt` procedure, the derivative of the power gain (MUG or MAG) versus frequency curves (see Figure 19) can be used to choose the frequency or the dB point for the `-parameter` keyword.

Without a feedback resistor, the MUG values for the HEMT device are negative and f_{max} is undefined. With a feedback resistor of $100\,k\Omega\,\mu m$, the MUG values are positive and f_{max} can be extracted. In addition, the U_f values of this device are much lower compared to U_f values for the HEMT device without feedback resistor. Therefore, the unilateral amplifier design approach is valid for the HEMT device with a feedback resistor of $100\,k\Omega\,\mu m$.

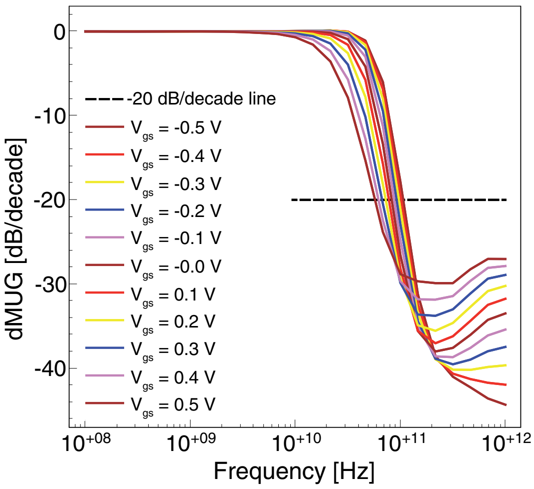

Figure 19 Derivative of Mason's unilateral gain as a function of frequency for various gate voltages between V_{gs} = –0.5 V and 0.5 V at a drain bias of V_{ds} = 1 V. This data is computed with a feedback resistor of 100 kΩ µm.

Figure 20 shows the f_{max} curves extracted using all three methods for the HEMT device. The `unit-gain-point` method gives f_{max} values between 70 and 200 GHz, which is close to the values between 61 and 185 GHz obtained from the `extract-at-dBPoint` method. The `extract-at-frequency` method predicts substantially lower f_{max} values (21–35 GHz).

The difference in results obtained from all three extraction methods is because the MUG curves (see Figure 18 on page 18) do not have a –20 dB/decade slope region (see Figure 19).

As discussed in Extracting Cut-off Frequency on page 14, the accuracy of the f_{max} values obtained from the extrapolation methods can be increased by increasing the number of frequency points.

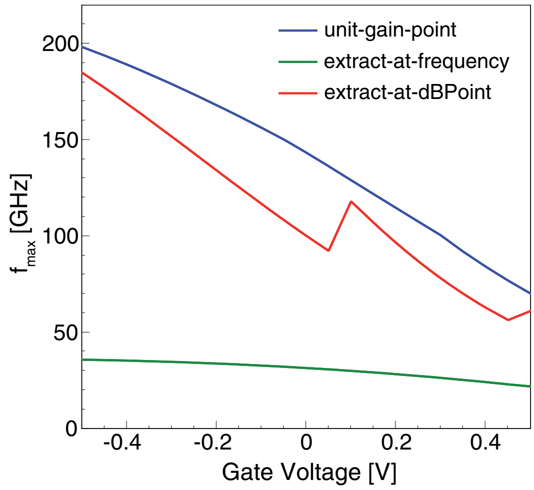

Figure 20 Maximum frequency of oscillation f_{max} as a function of gate bias, extracted with three different methods: unit-gain-point (blue), extract-at-frequency (green), and extract-at-dBPoint (red). This data is computed with a feedback resistor of 100 kΩ µm.

Exporting RF Parameters

To export the Tcl array `rfx::AC` or the Y-, Z-, H-, or S-matrix into a CSV file, use the procedure `rfx::Export`. The CSV file can be loaded into any spreadsheet application. It contains a header that lists the number of bias and frequency points as well as the value of the first and last bias and frequency points. The header is followed by a table, which contains the frequency, the bias, and the real and imaginary parts of the elements of the RF matrix.

For example, the following commands write the S-parameters S_{11}, S_{12}, S_{21}, and S_{22} in the CSV files `nX_S_freq.csv` and `nX_S_bias.csv`:

```
rfx::Export -file "n@node@_S_freq.csv" \
  -rfmatrix "S" -xaxis "frequency"
rfx::Export -file "n@node@_S_bias.csv" \
  -rfmatrix "S" -xaxis "bias"
```

The file name is set with the keyword `-file`. The keyword `-rfmatrix` defines the RF matrix to be exported. The keyword `-xaxis` specifies whether the parameters should be sorted by frequency first (`-xaxis "frequency"`) with bias being the secondary parameter, or by bias first (`-xaxis "bias"`) with frequency being the secondary parameter.

For example, the `nX_S_freq.csv` file contains the following columns:

```
bias,freq,S11_Re,S11_Im,S12_Re,S12_Im,
        S21_Re,S21_Im,S22_Re,S22_Im
```


References

[1] S. R. Bahl and J. A. del Alamo, “A New Drain-Current Injection Technique for the Measurement of Off-State Breakdown Voltage in FET’s,” *IEEE Transactions on Electron Devices*, vol. 40, no. 8, pp. 1558–1560, 1993.

[2] *Sentaurus™ Visual User Guide*, Version K-2015.06, Mountain View, California: Synopsys, Inc., 2015.

