Transient and Convolution Simulation

September 2002
Notice

The information contained in this document is subject to change without notice.

Agilent Technologies makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Agilent Technologies shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty

A copy of the specific warranty terms that apply to this software product is available upon request from your Agilent Technologies representative.

Restricted Rights Legend

Use, duplication or disclosure by the U. S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DoD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

Agilent Technologies
395 Page Mill Road
Palo Alto, CA 94304 U.S.A.

## Contents

1 Transient and Convolution Simulation  
   Performing a Transient or Convolution Simulation .................................................... 1-2  
   Example (ADS only) ................................................................................................. 1-3  
   Transient/Convolution Simulation Overview .............................................................. 1-7  
      Transient Analysis............................................................................................... 1-7  
      Convolution Analysis .......................................................................................... 1-7  
      Transient/Convolution Simulation Process .......................................................... 1-8  
      Integration Methods Used in Transient/Convolution Simulation ....................... 1-10  
      Reusing Transient Simulation Solutions ............................................................ 1-11  
   Monte-Carlo Noise in the Time Domain ................................................................. 1-12  
   Transient Simulation Controller .............................................................................. 1-13  
      Time Setup ......................................................................................................... 1-14  
      Integration ......................................................................................................... 1-15  
      Convolution ........................................................................................................ 1-17  
      Convergence ...................................................................................................... 1-19  
      Options ............................................................................................................... 1-21  
      Noise .................................................................................................................. 1-22  
      Freq .................................................................................................................... 1-23  
   Troubleshooting a Simulation .................................................................................... 1-25  
      Avoiding Simulation Errors in Transient Analysis .............................................. 1-25  
      Solving Convergence Problems ......................................................................... 1-26  
      Comparing Time-Domain and Steady-State Results ......................................... 1-26  
      Setting Max Frequency and Other Convolution Parameters ........................... 1-27  
      Dealing with Noncausal Frequency Responses ................................................. 1-28  
      Using Measured and Simulated S-Parameter Data .............................................. 1-29

Index
Chapter 1: Transient and Convolution Simulation

The Transient/Convolution Simulation component, in the Simulation-Transient palette, enables you to:

• Perform a SPICE-type transient time-domain analysis on a circuit.
• Perform nonlinear transient analysis on circuits that include the frequency-dependent loss and dispersion effects of linear models. Such analyses are known as convolution analyses.

Refer to these topics:

• “Performing a Transient or Convolution Simulation” on page 1-2 has the minimum setup requirements for a transient simulation.
• “Example (ADS only)” on page 1-3 is a detailed setup for performing a transient simulation, using a Gilbert cell mixer as the example.
• “Transient/Convolution Simulation Overview” on page 1-7 is a brief explanation of transient and convolution simulations.
• “Transient Simulation Controller” on page 1-13 provides details on the tabs and fields in the Transient simulation controller.
• “Troubleshooting a Simulation” on page 1-25 offers suggestions on how to improve a simulation.

**Note** You must have the Transient simulator (included with the RFIC Designer Pro and Premier suites) and the Convolution simulator (included with the RFIC Designer Premier suite only) licenses to run these types of simulations. You may build the examples without the appropriate license, but cannot run the simulations.
Performing a Transient or Convolution Simulation

Start by creating your design, then add current probes and identify the nodes from which you want to collect data.

For a successful analysis:

• When selecting sources, you can use either frequency-domain or time-domain sources. Transient sources are under the Sources-Time Domain palette. They are identified by the small ‘t’ in their names (for example, VtStep:Voltage Source: Step).

• Add the Trans component to the schematic. Double-click to edit it. Fill in the fields under the Time Setup tab:
  • Enter the start and stop times.
  • Enter the Max time step. This is the largest time step that will be used in the simulation. It should be small enough to adequately sample the highest frequency expected in the circuit.

• To achieve the most accurate model, select the Convolution tab and set the Max Frequency and max impulse sample points. The other parameters here are related to generating impulse responses for convolution analysis, but in general accept the defaults. For more information, refer to the sections “Convolution Analysis” on page 1-7 and “Setting Max Frequency and Other Convolution Parameters” on page 1-27.

• If frequency is not defined by a frequency source, select the Freq tab and set the fundamentals and order.

• The parameters under the Integration tab set truncation, integration techniques, and charge accuracy. For information on integration techniques, refer to “Integration Methods Used in Transient/Convolution Simulation” on page 1-10.

• The parameters under the Convergence tab are used to improve convergence. For more information, refer to “Solving Convergence Problems” on page 1-26.

• You can use previous simulation solutions to speed the simulation process. Refer to “Reusing Transient Simulation Solutions” on page 1-11.

• It is recommended that parameters not specifically mentioned here be left at the default values. For more information about each parameter, click Help in the Trans dialog box.
Example (ADS only)

Figure 1-1 illustrates the setup for a basic transient/convolution simulation.

Note This design, TRAN1.dsn, is in the Examples directory under Tutorial/SimModels_prj. The results are in TRAN1.dds.
Transient and Convolution Simulation

To perform a basic transient/convolution simulation:

1. From the Component Palette, choose **Sources-Time Domain** or **Sources-Freq Domain** and select appropriate sources. Place the source at the input of the component or circuit under test, then define input power and edit other parameters as required. In a circuit employing a mixer, provide a source for the local oscillator (LO).

   **Note** Transient sources, available in the Time Domain Sources palette, are available for these simulations, and are identified by the small “t” in their names (for example, VtStep:Voltage Source: Step). However, standard frequency sources (in the Freq Domain Source palette) can also be used.

2. Ensure that the inputs and outputs of nodes at which you want data to be reported are appropriately labeled.

3. If they are needed, select from among various transient measurement equations in the Transient Simulation palette, place these on the schematic, and edit them as required. (Refer to the topic “Troubleshooting a Simulation” in the DC Simulation documentation.) Their results can be plotted later in a Data Display window.

4. From the Component Palette, choose **Simulation-Transient**. Select and place the Tran simulation component on the schematic. You can run a basic simulation by editing the parameters **StopTime** and **MaxTimeStep** on the schematic.

   **Note** The Max time step value should be small enough to adequately sample the highest frequency expected in the circuit. The simulator may use a smaller timestep if needed but will never use a larger value.

5. To edit the frequency of the fundamental and any other frequencies to be considered, select the Freq tab. This frequency information is required only if the frequency is not set explicitly in the frequency domain sources.

6. The parameters that make it possible to obtain the most accurate model are **Max Frequency** and **Max impulse sample points**, under the Convolution tab of the simulation component. It is recommended that you leave the other parameters under this tab at their default settings and edit them only in special cases.
7. For details regarding all Transient parameters, double-click the simulation component to edit it, select the tab of interest, and click the Help button. Many parameters in this simulator apply only to special cases.

8. Launch the simulation. The data resulting from this simulation will be identified by TRAN1. The large trace is a plot of $V_{if}$ versus time, the output of the filter showing all harmonic components. The small sinusoid is a plot of $V_{load}$ following the filter. The filter output has been given enough time to approach a steady-state amplitude.

9. A plot of $fs(V_{if},...,4\text{ns},16\text{ns})$ shows a time-to-frequency transform from 4 to 16 ns for the mixer output before filtering. (The seven commas represent $fs$ parameters not used here.) Run the simulation longer to observe the noise floor drop.
10. Finally, an fs plot of the filter output (Vload) between 24 and 32 ns shows the response after steady-state amplitude has been approximated. Essentially only the bandpass frequency remains.
Transient/Convolution Simulation Overview

The transient and convolution simulators are SPICE-like in their operation. They solve a set of integro-differential equations that express the time dependence of the currents and voltages of the circuit under analysis. The result is a nonlinear analysis with respect to time and, possibly, a swept variable.

The main difference between the transient and convolution options lies in how each analysis characterizes the distributed and frequency-dependent elements of a circuit, as discussed below.

Transient Analysis

A transient analysis is performed entirely in the time-domain, and so is unable to account for the frequency-dependent behavior of distributed elements such as microstrip elements, S-parameter elements, and so on. Therefore, in a transient analysis, such elements must be represented by simplified, frequency-independent models such as lumped equivalents, transmission lines with constant loss and no dispersion, short circuits, open circuits, and the like. These assumptions and simplifications are usually very reasonable at low frequencies.

Convolution Analysis

A convolution analysis represents all the distributed elements in the frequency domain and hence accounts for their frequency-dependent behavior. The characterization of many RF and microwave distributed elements is best accomplished in the frequency domain, because the exact time-domain equivalents for these elements cannot always be obtained.

Convolution converts the frequency-domain information from all the distributed elements to the time domain, effectively resulting in the impulse response of those elements. The time-domain input signals at an element's terminals are convolved with the impulse-response of the element to yield the output signals. Elements that have exact lumped equivalent models—including nonlinear elements—are characterized entirely in the time domain without using impulse responses.
Transient and Convolution Simulation

**Note**  In a convolution analysis, all elements are characterized by means of the full frequency-domain model, through the use of either an exact time-domain model or convolution. However, there may be minor differences between the results of a convolution simulation and the results of a transient simulation of the same circuit.

A convolution analysis requires both a convolution license and a transient license, and is performed whenever a convolution license is available. If the simplified approximate models are preferred, in this situation (for speed), set the Use Approximate Models When Available option to yes.

**Transient/Convolution Simulation Process**

The following steps describe how both the transient and convolution simulators operate:

1. The user specifies a time-sweep range, tolerances, and iteration limits.
2. A DC analysis is conducted to determine the system solution at zero time.
3. Inside the simulator, a breakpoint table is constructed to deal with frequency-domain-devices and data. Independent source waveforms frequently have sharp transitions that may not normally coincide with the time step calculated by the program. Such is the case with the piecewise linear sources. The breakpoint table contains a sorted list of all the transition points of the independent sources. During the simulation, whenever the next time point is sufficiently close to one of the breakpoints, the time step is adjusted to land exactly on the breakpoint. This prevents unnecessary time-step reductions in the vicinity of the transitions.
4. An internal control variable updates the current time, and the values of the independent sources are calculated at that time.
5. An attempt is made to solve the system of equations through numerical integration and a finite number of Newton-Raphson iterations. If the number of iterations exceeds Max iterations per time point, then the time step is reduced by a factor of Integration coefficient mu divided by 8. If this new time step is acceptable, the analysis is repeated from step 4. If Integration coefficient mu = 0, backward-Euler numerical integration is used. Otherwise, trapezoidal numerical integration is used.
6. Following convergence, the local truncation error is calculated. The default Trapezoidal integration method is used to estimate the error, unless Gear’s method is selected.

7. The time step interval is calculated. By default, the time step is computed for transient analysis by means of the truncation error estimate method.

8. The error tolerance is compared with the value in the Local truncation error over-est factor field (under the Integration tab). If the error is within acceptable limits, the results are stored and analysis continues at the next time point. Otherwise, the analysis is repeated at a smaller time step.

9. Steps 3 through 9 are repeated until the user-specified time-sweep range has been analyzed.
Integration Methods Used in Transient/Convolution Simulation

Like SPICE, this simulator uses the trapezoidal integration method described by the following equation as the default method for calculating derivatives at each time step in the simulation.

$$x_{n+1} = \frac{2}{\Delta t} (x_{n+1} - x_n) - x_n$$

For most circuits, this method will succeed. For those that do not, the simulator also supports Gear's backward difference method:

$$x_{n+1} = \alpha_0 x_{n+1} + \alpha_1 x_n + \alpha_2 x_{n-1} + \ldots + \alpha_k x_{n+1-k}$$

In this equation, the index $k$ is called the order of the integration.

For most circuits, Gear's method is no more accurate than the default trapezoidal integration technique. However, if a circuit analysis fails to converge, Gear's method may succeed where trapezoidal integration fails. In particular, oscillator circuits and any circuit that is characterized by stiff state equations may benefit from Gear's method.


If Time Step Control is set to TruncError and Max Gear order (under the Integration tab) is set to a number between two and six, the simulator will use Gear's method along with an adaptive stepsize algorithm that picks the largest possible step size at each point in the simulation. For each time step, the order of Gear's method will be chosen (up to the value of Max Gear order) to maintain accuracy with the largest possible time step. This potentially speeds up simulations with no loss in accuracy. If Gear Integration is selected with fixed timestep, then the integration will always be done at the fixed order given by Max Gear order.

The integration order at each time step is output to the dataset as the variable tranorder. This data is used by the fs() function, in data display, to do accurate interpolation of the data when an FFT is required. For the default trapezoidal integration, this will normally have a value of two, except at source-induced breakpoints where it will be one.
Reusing Transient Simulation Solutions

Transient simulations can be set to generate a harmonic balance solution that can then be used as an initial guess for a harmonic balance simulation. For example, in circuits such as dividers, harmonic balance cannot usually directly converge on a solution since multiple mathematical, but useless, solutions exist. By first running a Transient simulation and generating a harmonic balance solution file and then using this as an initial guess for a harmonic balance simulation, the harmonic balance simulation can then converge to the desired solution.

To save a solution to a file:

1. Add a transient component to your schematic if one is not present, and double-click to edit it.

2. Select the **Freq** tab. Fill in the frequency fields as you would for a harmonic balance simulation. The Frequency values can still be used, independent of this solution mode, to define the fundamental frequencies and _freqN variables used in sources. The Order and Maximum order information is used to determine the number of frequencies for which to compute a harmonic balance solution.

3. Enable **Write Solution** and enter a file name and extension. The default is the name of the project.

4. For best results, especially in multi-tone applications, enable **Apply Window**. This applies a Hanning window to the time domain data. For a good Fourier transform, the duration of the transient simulation must be longer than the period of the smallest specified frequency. A factor of 4 is often used here. It is also best to make sure that the transient simulation has reached an approximately steady state solution. This does not have to be precise, since this solution is only providing an initial guess value to the harmonic balance simulation. The start time of the transient simulation can be used to indicate when this approximate steady state solution has been reached. The internal Fourier transform will then not begin until after this start time.

5. To get the best results for the solution, the transient analysis should be run in a fixed timestep control mode.
Monte-Carlo Noise in the Time Domain

Noise in transient simulations can be added as pseudo-random voltages and currents at each timepoint. The noise has a Gaussian amplitude distribution. The noise can be frequency- or bias-dependent, as appropriate for each component. All components that generate noise in a linear or harmonic balance simulation generate the appropriate noise, with the following exceptions:

- S-parameter data from files that include a noise parameter block with NFmin, Rn and Sopt
- Behavioral components for amplifiers and mixers
- Noisy2Port component
- NoiseCorr noise correlation component
- SDD
- Mextram504 and Hicum transistor models

The noise is added to the signal, so there is no easy way to separate the signal from the noise. The full, nonlinear circuit equations are applied to this composite signal of random voltages and currents, so no small-signal assumptions about the relative size of the noise are required.

The Noise bandwidth parameter (under the Transient Noise tab) enables the generation of noise in transient analysis. Because transient analysis uses a variable timestep method, the noise must be limited to a bandwidth of less than 1/(2*MaxTimeStep). If this parameter is left blank or specified as zero, no noise is generated.

The Noise scale parameter is used to multiplicatively scale all of the noise sources in the circuit. This can be useful when the noise levels are very low, as this allows them to be increased so they are visible above the numerical noise in the simulation. However, if the noise is increased too much, it can change the nonlinear operation of the circuit.
Transient Simulation Controller

The Transient (Tran) Simulation controller enables you to define these aspects of the simulation:

- **Time Setup** sets parameters related to time and frequency.
- **Integration** selects an integration mode and sweep offset, turns on source and resistor noise, and sets device-fitting parameters. Options for oscillator analysis are also available.
- **Convolution** sets parameters related to setting up for convolution analysis.
- **Convergence** sets parameters related to achieving convergence.
- **Options** sets parameters related to simulation reporting levels and saving operating point level data.
- **Noise** sets noise bandwidth and scale.
- **Freq** sets parameters for the fundamental frequencies and computing an harmonic balance solution as an initial guess.
- **Output (in ADS)**—Selectively save simulation data to a dataset. For details, refer to the topic “Selectively Saving and Controlling Simulation Data” in the chapter “Simulation Basics” in the Circuit Simulation documentation. In RFDE, use Outputs > Save Options in the Analog Design Environment window.
- **Display (in ADS)**—Control the visibility of simulation parameters on the Schematic. For details, refer to the topic “Displaying Simulation Parameters on the Schematic” in the chapter “Simulation Basics” in the Circuit Simulation documentation. RFDE does not have a similar feature.

**Setting Initial Conditions with InitCond Components**

There are two elements for setting initial conditions in a transient simulation InitCond and InitCondByName. InitCond and InitCondByName are used to provide an initial DC value for transient analysis only. These elements attach the specified voltage source with a series resistor to the specified node(s) to force a value. The DC solution for the entire circuit is then calculated. This DC solution is then used as the starting state for the transient analysis.
Transient and Convolution Simulation

**Time Setup**

Following is information on the parameters related to time and frequency.

<table>
<thead>
<tr>
<th>Time Setup</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start time</td>
<td>The time at which the simulator begins outputting time-point results.</td>
</tr>
<tr>
<td>Stop time</td>
<td>The time at which the simulator stops outputting time-point results. You must specify this parameter.</td>
</tr>
<tr>
<td>Max time step</td>
<td>The largest time step to be taken in the simulation. This parameter must be specified by the user.</td>
</tr>
<tr>
<td>Min time step</td>
<td>The smallest time step to be taken in the simulation. Generally the default value is satisfactory. In RFDE, enter this value in Transient Options &gt; Time Step.</td>
</tr>
<tr>
<td>Limit timestep for Transmission Line</td>
<td>Where transmission lines are involved, setting this option further limits the time step to half of the shortest transmission line’s delay time. In RFDE, enable this option in Transient Options &gt; Time Step.</td>
</tr>
</tbody>
</table>
Integration

Following is information on setting up the Integration portion of the simulation.

Table 1-2. Tran Simulation Integration Options

<table>
<thead>
<tr>
<th>Integration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time step control method</td>
<td>In RFDE, set these parameters in Transient Options &gt; Time Step</td>
</tr>
<tr>
<td>Fixed</td>
<td>Selects a fixed time-step method. The simulation is performed with a uniform, constant time step that is specified by Max time step (under the Time Setup tab). It is quicker than the other methods. However, it is not as robust, because it cannot select a smaller time step when convergence problems are encountered.</td>
</tr>
<tr>
<td>Iteration count</td>
<td>Uses the number of Newton-Raphson iterations that were needed to converge at a time point as a measure of the rate of change of the circuit. If the number of iterations is less than an internal threshold, the time step is doubled; if the number is greater than Max iterations per time step, the time step is scaled by a factor of Integration coefficient mu divided by 8 (see below). This method has a minimal computational overhead, but does not take into account the true rate of change of circuit variables.</td>
</tr>
<tr>
<td>Trunc Error</td>
<td>Uses the current estimate of local truncation error to determine an appropriate time step. Although it takes longer than Iteration count, it sets a meaningful error bound on computed output values.</td>
</tr>
<tr>
<td>Local truncation error over-est factor</td>
<td>A value against which the simulation's error tolerance is compared. In transient analysis each time step is computed by means of the truncation-error estimate method. If the error is within acceptable limits, the results are stored and analysis continues at the next time point. Otherwise, the analysis is repeated at a smaller time step.</td>
</tr>
<tr>
<td>Charge accuracy</td>
<td>Determines the charge tolerance used in computing the local truncation error.</td>
</tr>
<tr>
<td>Integration</td>
<td></td>
</tr>
<tr>
<td>Trapezoidal</td>
<td>Integrates between time points by assuming they are connected by line segments. The local truncation error is then related to the difference between the areas determined by the present and previous time points.</td>
</tr>
<tr>
<td>Gear's</td>
<td>Integrates by assuming that the time points are connected by a polynomial curve. The order of the polynomial is controlled by the Max Gear order parameter. Lower-order polynomials tend to create greater truncation error, while higher-order polynomials can become unstable.</td>
</tr>
</tbody>
</table>
Table 1-2. Tran Simulation Integration Options

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Gear order</td>
<td>Determines the maximum order of the polynomial when Gear’s method is used. The default is 2. This is available only when Gear’s is selected.</td>
</tr>
<tr>
<td>Integration coefficient μ</td>
<td>A coefficient that determines the degree of mixing of the trapezoidal (μ = 0.5) and backward-Euler (μ = 0.0) methods when the trapezoidal method is used. This is available only when Trapezoidal is selected. The integration order at each timestep is output to the dataset as the variable tranorder. This data is used by the fs() function in the data display server to do accurate interpolation of the data when an FFT is required. For the default trapezoidal integration, this will normally have a value of 2, except at source-induced breakpoints where it will be 1.</td>
</tr>
</tbody>
</table>
Convection

Following is information on setting up the Convection portion of the simulation.

<table>
<thead>
<tr>
<th>Table 1-3. Tran Simulation Convection Options</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Convolution</strong></td>
</tr>
<tr>
<td><strong>Note:</strong> With the exception of Max Frequency, we recommend that the remaining parameters under this tab be left at their default settings and edited only in special cases.</td>
</tr>
<tr>
<td><strong>Use approximate models when available</strong></td>
</tr>
<tr>
<td>Causes the simulator to bypass impulse-based convolution (when that option is available to the user). Instead, it uses models that, although somewhat less accurate, can provide faster simulations. These approximations neglect effects such as frequency-dependent loss and dispersion, but include the basic delay and impedance. These models are the default, if no convolution license is available.</td>
</tr>
<tr>
<td><strong>Approximate short transmission lines</strong></td>
</tr>
<tr>
<td>Specifies a limit on the time delay below which transmission lines will be approximated instead of modeled as a delay line. This allows the user to analyze very short transmission lines with a Laplace transform approximation. Also, it does not require the simulator to take the very small time steps normally associated with short transmission lines. Only single, two terminal, transmission lines (like MLIN and TLIN, but not MCLIN and TLIN4) can be approximated this way.</td>
</tr>
<tr>
<td><strong>Max Frequency</strong></td>
</tr>
<tr>
<td>The maximum frequency to which a frequency-domain device is evaluated to obtain its impulse response. By default, the program chooses this value iteratively and separately for each device, in order to obtain a good impulse response. The user does not normally need to set this, unless it is necessary to prevent devices from being evaluated in an invalid region.</td>
</tr>
<tr>
<td><strong>Delta frequency</strong></td>
</tr>
<tr>
<td>The frequency interval between samples in the evaluation of frequency-domain-defined devices.</td>
</tr>
<tr>
<td><strong>Max impulse sample points</strong></td>
</tr>
<tr>
<td>The maximum number of points allowed in an impulse response.</td>
</tr>
<tr>
<td><strong>Relative impulse truncation factor</strong></td>
</tr>
<tr>
<td>The relative truncation factor for the impulse response.</td>
</tr>
<tr>
<td><strong>Absolute impulse truncation factor</strong></td>
</tr>
<tr>
<td>The absolute truncation factor for the impulse response.</td>
</tr>
<tr>
<td><strong>Convolution interpolation order</strong></td>
</tr>
<tr>
<td>The interpolation order used during convolution simulation.</td>
</tr>
<tr>
<td><strong>Convolution mode</strong></td>
</tr>
<tr>
<td><strong>Discrete</strong></td>
</tr>
<tr>
<td>Causes a periodic extension of the frequency response that is modeled from DC to Max Frequency. This is many times faster than PWL Continuous (see below).</td>
</tr>
</tbody>
</table>
### Table 1-3. Tran Simulation Convolution Options

<table>
<thead>
<tr>
<th>PWL Continuous</th>
<th>Always leads to a low-pass response. This may be desirable where a low-pass response is modeled.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smoothing window type</td>
<td>Applies to time-domain impulse responses that are derived from noncausal frequency functions (such as Hilbert transforms, integrators, or differentiates). A smoothing window reduces ripple in the approximation caused by discontinuities in the frequency function. The window is a multiplier function that is applied to data prior to their conversion (by means of an inverse Fourier transform) to an impulse response. A significant advantage of a window is its ability to reduce the exaggerated effects of noise in high-dynamic-range systems, resulting in a more realistic noise floor. It is also useful in the analysis of digital filters, by removing nonphysical artifacts and allowing a filter’s response to be viewed more realistically.</td>
</tr>
<tr>
<td>Rectangle</td>
<td>Applies a function that truncates both noncausal and trailing responses.</td>
</tr>
<tr>
<td>Hanning</td>
<td>Applies a function that is smoother than that provided by Rectangle.</td>
</tr>
<tr>
<td>Non-causal fcn imp response length</td>
<td>Determines the length of the impulse response for non-causal impulse responses. The unit is the number of sampling points. Non-causal impulse responses are caused by elements like ideal phase shifters. These are modeled, but with an added delay based on this value.</td>
</tr>
</tbody>
</table>
Convergence

Following is information on setting up the Tran Convergence portion of the simulation.

Table 1-4. Tran Simulation Convergence Options

<table>
<thead>
<tr>
<th>Convergence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use user-specified initial conditions</td>
<td>Instructs the simulator to use user-specified initial conditions to compute the initial charges and fluxes in the circuit. The simulator will skip time=0 dc analysis. When this option is enabled, the simulator uses the currents specified in the InitCond field of the inductors and the branch voltages specified in the InitCond field of the capacitors, as well as the node voltages specified in both the Simulation-Transient InitCond and Simulation-Transient InitCondByName components to compute the initial charges and fluxes in the circuit. The voltage given in the capacitor's InitCond field will take precedence over the voltage given in the Simulation-Transient InitCond and Simulation-Transient InitCondByName components if there is a conflict. When the option is disabled and the Simulation-Transient InitCond and/or Simulation-Transient InitCondByName components are placed on the schematic, the simulator will start the transient analysis with a DC analysis at time=0 and try to force those node voltages identified in the Simulation-Transient InitCond and/or Simulation-Transient InitCondByName components to be close to the voltages specified. The values, if any, given in the InitCond field of the capacitors and inductors, however, will not be used.</td>
</tr>
<tr>
<td>Connect all nodes to GND via GMIN during initial DC analysis</td>
<td>Useful in conditions where open circuits would result in a singular matrix error (where the diagonal term is missing). When selected, this option instructs the simulator to insert a resistor valued at 1e12 ohms between any node and ground, whether at the circuit or the device level. (&quot;GMIN&quot; stands for minimum conductance.) This allows the user to still obtain useful results.</td>
</tr>
<tr>
<td>Perform KCL check for convergence</td>
<td>Checks to verify how closely Kirchoff's Current Law is satisfied at each node. This also depends on how the current tolerances are set (under the Convergence tab of the Options component). This option is selected by default.</td>
</tr>
<tr>
<td>Check only delta voltage for convergence</td>
<td>Looks for only the voltage difference between two consecutive iterations. This less-stringent check saves both time and memory. This option is selected by default.</td>
</tr>
</tbody>
</table>
### Table 1-4. Tran Simulation Convergence Options (continued)

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Check for strange behavior at every timestep</td>
<td>Looks for inordinate device currents or voltages and returns a warning message when it finds any for devices or models which support the Overload Alert and have their limit values specified. Although this can be a little slower, it is useful where out-of-bounds conditions are suspected.</td>
</tr>
<tr>
<td>Skip device evaluation if volt chg are small between iters</td>
<td>Instructs the simulator to bypass the full evaluation of a device if it finds small voltage changes between iterations. This can improve simulation time in digital circuits containing many devices. What is a “small” voltage change is determined by how voltage tolerances are set (under the Convergence tab of the Options component).</td>
</tr>
<tr>
<td>Max iterations per time step</td>
<td>The maximum number of iterations allowed for each time step.</td>
</tr>
<tr>
<td>Max iterations @ initial DC</td>
<td>The maximum number of iterations allowed during DC analysis, before the stepping of the source begins.</td>
</tr>
</tbody>
</table>
Options

Following is information on setting up the Tran Options portion of the simulation.

Table 1-5. Tran Simulation Options

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Levels (in ADS)</td>
<td>Select the degree of simulation information to be reported.</td>
</tr>
<tr>
<td>Annotation (in RFDE)</td>
<td></td>
</tr>
<tr>
<td>Status level</td>
<td>Prints information about the simulation in the Status/Summary part of the Message Window.</td>
</tr>
<tr>
<td></td>
<td>- 0 reports little or no information, depending on the simulation engine.</td>
</tr>
<tr>
<td></td>
<td>- 1 and 2 yield more detail.</td>
</tr>
<tr>
<td></td>
<td>- Use 3 and 4 sparingly since they increase process size and simulation times considerably.</td>
</tr>
<tr>
<td></td>
<td>The type of information printed may include the sum of the current errors at each circuit node, whether convergence is achieved, resource usage, and where the dataset is saved. The amount and type of information depends on the status level value and the type of simulation.</td>
</tr>
<tr>
<td>Device operating point level</td>
<td>Enables you to save all the device operating-point information to the dataset. In ADS, if this simulation performs more than one Transient analysis (from multiple Transient controllers), the device operating point data for all Transient analyses will be saved, not just the last one. Default setting is None.</td>
</tr>
<tr>
<td>Brief</td>
<td>Saves device currents, power, and some linearized device parameters.</td>
</tr>
<tr>
<td>Detailed</td>
<td>Saves the operating point values which include the device’s currents, power, voltages, and linearized device parameters.</td>
</tr>
<tr>
<td>Output solutions</td>
<td></td>
</tr>
<tr>
<td>Output solutions at all interval time points</td>
<td>Causes the simulator to save simulation results at all internal timepoints; this option is on by default. Deselecting this option causes results to be saved at least as often as the Max timestep option but some of the intermediate points will be suppressed. For simulations that take many small timesteps due to automatic timestep control, but whose output is still well-sampled at Max timestep, this can make the resulting datasets smaller and make the post-processing of the data faster.</td>
</tr>
</tbody>
</table>
Transient and Convolution Simulation

Noise

Following is information on setting up the Tran Noise portion of the simulation.

Table 1-6. Tran Simulation Noise Options

<table>
<thead>
<tr>
<th>Noise</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise bandwidth</td>
<td>Enables the generation of pseudorandom noise at each timestep. NoiseBandwidth controls the bandwidth of the generated noise and must be less than 1/(2*MaxTimeStep). If this parameter is not specified or zero, no noise is generated.</td>
</tr>
<tr>
<td>Noise scale</td>
<td>A multiplicative scaling applied to all generated noise.</td>
</tr>
</tbody>
</table>
**Freq**

Following is information on setting up the frequency portion of the simulation.

Table 1-7. Trans Simulation Freq Options

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>The frequency of the fundamental(s). Change by typing over the entry in the field. Select the units (None, Hz, kHz, MHz, GHz) from the drop-down list.</td>
</tr>
<tr>
<td>Order</td>
<td>The maximum order (harmonic number) of the fundamental(s) that will be considered. Change by typing over the entry in the field.</td>
</tr>
<tr>
<td>Select (in ADS)</td>
<td>Contains the list of fundamental frequencies. Use the Edit field to add fundamental frequencies to this window.</td>
</tr>
<tr>
<td>Maximum order</td>
<td>The maximum order of the intermodulation terms in the simulation. The combined order is the sum of the individual frequency orders that are added or subtracted to make up the frequency list. For example, assume there are two fundamentals and Order (see below) is 3.</td>
</tr>
<tr>
<td></td>
<td>If Maximum order is 0 or 1, no mixing products are simulated. The frequency list consists of the fundamental and the first, second, and third harmonics of each source.</td>
</tr>
<tr>
<td></td>
<td>If Maximum order is 2, the sum and difference frequencies are added to the list.</td>
</tr>
<tr>
<td></td>
<td>If Maximum order is 3, the second harmonic of one source can mix with the fundamental of the others, and so on.</td>
</tr>
<tr>
<td>Edit (in ADS)</td>
<td>Edit the Frequency and Order fields, then use the buttons to Add the frequency to the list displayed under Select.</td>
</tr>
<tr>
<td>Add</td>
<td>Allows you to add an item.</td>
</tr>
<tr>
<td>Cut</td>
<td>Allows you to delete an item.</td>
</tr>
<tr>
<td>Paste</td>
<td>Allows you to take an item that has been cut and place it in a different order.</td>
</tr>
</tbody>
</table>
### Table 1-7. Trans Simulation Freq Options (continued)

<table>
<thead>
<tr>
<th>Fundamental Frequencies (in RFDE)</th>
<th>Add and edit fundamental frequencies by specifying these parameters:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Frequency-The frequency of the fundamental(s).</td>
</tr>
<tr>
<td></td>
<td>Order-The maximum order (harmonic number) of the fundamental(s) that will be considered.</td>
</tr>
<tr>
<td></td>
<td>Add enters the values from the input fields into the table.</td>
</tr>
<tr>
<td></td>
<td>Delete removes selected frequency from the table.</td>
</tr>
<tr>
<td></td>
<td>Update changes frequency selected in table with contents of the input fields.</td>
</tr>
<tr>
<td></td>
<td>Clear removes contents of the input fields.</td>
</tr>
<tr>
<td>Compute HB Solution (in ADS)</td>
<td>Transient simulations can also be programmed to generate a harmonic balance solution that can then be used as an initial guess for an HB simulation. For example, in circuits such as dividers, HB cannot usually directly converge on a solution since multiple mathematical, but useless, solutions exist. By first running a Transient simulation and generating an HB solution file and then using this as an initial guess for an HB simulation, the HB simulation can then converge to the desired solution.</td>
</tr>
<tr>
<td>Write Initial Guess for HB (in RFDE)</td>
<td>The Frequency values can still be used, independent of this HB solution mode, to define the fundamental frequencies and _freqN variables used in sources. The Order and Maximum order information is just used to determine the number of frequencies for which to compute an HB solution.</td>
</tr>
<tr>
<td></td>
<td>To get the best results for the HB solution, the Transient analysis should be run in fixed timestep control mode. Also, especially in multi-tone applications, you should activate Apply Window.</td>
</tr>
<tr>
<td>Write Solution (ADS)</td>
<td>Enables the generation of the HB solution.</td>
</tr>
<tr>
<td>File</td>
<td>A file name would also normally be provided, but as with the HB analysis, a default one will be generated if none is given.</td>
</tr>
<tr>
<td>Apply Window</td>
<td>Applies a Hanning window to the time domain data. To get a reasonable Fourier transform, the duration of the Transient simulation must be longer than the period of the smallest specified frequency. A factor of 4 is often used here. You should make sure that the Transient simulation has reached an approximately steady state solution. This does not have to be precise, since this is just providing an initial guess value to the HB simulation. The start time of the Transient simulation can be used, in this compute HB solution mode, to indicate when this approximate steady state solution has been reached. The internal Fourier transform will then not begin until after this start time.</td>
</tr>
</tbody>
</table>
Troubleshooting a Simulation

This section presents suggestions for using this simulation tool and improving the accuracy of results.

Avoiding Simulation Errors in Transient Analysis

This section lists a variety of steps that can be taken to avoid errors in simulation.

1. Check the circuit’s schematic diagram carefully, and turn on the topology checker if it has been turned off. Consider using DC_Block and DC_Feed components where applicable.

2. Check the parameter Min time step. This parameter sets the smallest time step that the simulator is allowed to take, and should be smaller than the fastest rise time in the circuit. The default value of Min time step is Max time step/10^{12}. (Stop time is the last time point in the simulation.) The default value should be satisfactory.

3. Check to verify that the absolute and relative current and voltage tolerances (in the Options component) are not too small.

4. The models for some frequency-dependent devices have a high-frequency limit, beyond which they are not valid. Unless the very-high-frequency response of the device is important, the simulation results will still be valid.

5. If a lossy inductor model is included in the circuit, and the inductance has been set to zero, you may need to replace the lossy inductor with a resistor.

6. A lossy inductor model cannot be used with an initial condition. To solve this problem, replace the lossy inductor model with a lossless inductor in series with a resistor.

7. The simulator supports user-defined models that can have any impedance. However, it is easy for users to define nonphysical or noncausal components for which there is no correct answer. If a component has a constant reactance that does not vary with frequency (or has a nonzero reactance at DC), then the component is mathematically nonphysical. In these cases, the simulator will produce an answer that may not be physically realistic. To eliminate this problem, change the component’s definition.
8. Sometimes, in the case of user-defined devices, the simulator cannot handle certain types of time dependencies with guaranteed accuracy. These devices often work correctly, but the simulation results should be checked carefully.

9. The simulator cannot support S-parameter ports with zero impedances. To use a source with a zero impedance, use a simple voltage source instead.

**Solving Convergence Problems**

Nonconvergence is a numerical problem encountered by the simulator when it cannot reach a solution, within a given tolerance, after a given number of iterations.

There is no single solution for solving convergence problems in transient and convolution analysis. Several ways to approach those problems are listed below.

- Make sure that you have specified an appropriate value for Max time step.
- Vary Integration coefficient $\mu$ in the case of high-Q circuits.
- Increase the time-point iteration limit, Max iterations per timestep.
- If the circuit includes a GaAsFET, reduce the value of the transit time $\tau$ in the device model itself.
- Adjust the current and voltage relative and absolute tolerances in the Options component. Note the following guidelines:

  The relative tolerance parameters have a greater effect on simulation speed and accuracy than the absolute tolerance parameters.

  Each order-of-magnitude change in accuracy (for example, from $10^{-3}$ to $10^{-4}$) will result in approximately three times as many time points in the simulation.

**Comparing Time-Domain and Steady-State Results**

Certain circuit elements such as microstrip, discontinuities, and so on, are represented by simplified models in a transient analysis. A few planar discontinuities are also treated as ideal short or open circuits. Therefore, results from transient or convolution simulation may differ from those for steady-state simulation on the same circuit, depending upon the types of elements used in the circuit. (A convolution simulation should yield results closer to those of a steady-state simulation.)

The frequency of operation also plays a major role. At low frequencies, simplified models or short-circuits may be valid approximations for certain dispersive models.
Setting Max Frequency and Other Convolution Parameters

In general, the synthesized impulse response accurately represents the frequency
domain function over the frequency range specified by the convolution control
parameter Max Frequency. The simulation techniques require negligible spectral
energy outside this frequency range. An accurate solution is not guaranteed if this
principle is not obeyed. Therefore, setting Max Frequency correctly is the most
important aspect of any transient simulation using convolution-based devices.

Setting Max Frequency is similar to choosing the number of harmonics in a harmonic
balance simulation or a time step in SPICE. In these cases the user must estimate
the value of the parameter prior to the simulation. Examination of the results reveals
whether the parameter was chosen correctly. The built-in estimation of Max
Frequency is always a good starting point.

The remaining convolution-control parameters described below are best used with
their default values for any causal device, such as a microstrip transmission line.

The Discrete convolution mode option, by causing a periodic extension of the
frequency response, generally leads to the most accurate and efficient description of
the device from DC to Max Frequency. Provided Max Frequency is set correctly, the
frequency response beyond Max Frequency is irrelevant, because no significant
spectral energy exists beyond this value. The PWL Continuous option always leads to
a low-pass response (which may be desirable in cases where a low-pass response is
being modeled). The discrete mode is many times faster than the continuous mode.

Delta frequency defines the frequency spacing with which the convolution-based
devices are sampled in the frequency domain. If Delta frequency is not specified, the
simulator adaptively samples the frequency function until an appropriate value is
determined. If you are unsure about the correct value of Delta frequency, simply allow
the simulator to decide.

Non-causal fcn imp response length adjusts the length of the impulse response
associated with the treatment of noncausal frequency responses (see discussion
below).

Smoothing window type specifies the smoothing window to be applied to the
time-domain impulse responses that are derived from noncausal frequency functions
(such as Hilbert transforms). The window reduces ripple in the approximation caused
by discontinuities in the frequency function. (Refer to “Dealing with Noncausal
Frequency Responses” on page 1-28.)
Max impulse sample points places an upper limit on the allowed impulse-response length. It is mainly used when Max Frequency is specified but Delta Frequency is not. It is necessary to increase this value (default = 4096) if you specify a frequency response that requires a long impulse response.

Relative impulse truncation factor and Absolute impulse truncation factor are used to remove redundancy from the impulse responses. Setting them to a small value leads to longer impulse responses and a more accurate description of the frequency response of a convolution-based device. The simulator uses them to make decisions about the relative sizes of individual members of an impulse response.

Dealing with Noncausal Frequency Responses

A causal system has the property that if the input is zero for \( t < t_1 \), then the output is also zero for \( t < t_1 \). The value of \( t_1 \) is usually defined to be 0. Therefore, the impulse response of a causal system is zero for negative time. This may be stated in the frequency domain by saying that the real and imaginary parts of a causal frequency response are related by the Hilbert transform. Only causal frequency-domain transfer functions can be handled directly by means of transient simulation methods.

However, there are many ideal frequency functions that are noncausal but are extremely valuable in processing signals (such as Hilbert transforms). The transient simulator provides a way to use these functions in defining the frequency response of a nonlinear device.

A frequency response is considered noncausal if it has one of the following forms:

\[
R + jI \\
R(w) + j0 \\
R + jI(w)
\]

When you select Non-causal fcn imp response length, non-causal frequency functions are approximated by a digital filter that introduces a time delay in the impulse response. This time delay is sufficient to make the apparent impulse response causal. Non-causal fcn imp response length defines the length of this filter. The Smoothing window type can be selected to reduce Gibb's phenomenon, which may be present if the non-causal frequency response contains discontinuities. Increasing non-causal leads to a more accurate description of the frequency function over the chosen band set by Max Frequency. However, increasing non-causal causes a longer delay through the filter and a longer simulation time. Only even values of non-causal are allowed. If an odd value is specified, the simulator adds 1 and issues a warning. The delay
associated with the causal approximation is given by the equation:

\[
\text{(noncausal/2-1)/(2*max Frequency)}.
\]

If a non-causal frequency response is specified which does not fit into one of the forms described above, the simulator will assume it is a causal function. This will almost certainly cause unexpected and erroneous results. Noncausal frequency responses should be avoided if possible.

The concept of causality is not an issue when using the harmonic balance simulator. All excitations and responses are sums of sinusoids and exist for all time. For this reason, any bounded frequency response can be simulated. When moving from harmonic balance to transient simulation, it must be remembered that non-causal frequency functions cannot be simulated directly. Only a band-limited approximation to these functions can be simulated. This is particularly important when using SDD weighting functions.

**Using Measured and Simulated S-Parameter Data**

The ability to handle convolution based devices allows the user of measured or simulated S-parameter data to describe a wide variety of devices and circuits. A dataset or file containing the S-parameter values can be used to integrate the frequency response into a time-domain simulation. This adds a tremendous amount of flexibility to the number and types of devices and circuits which can be used in a simulation.

When S-parameter data is used, it is important that the frequency response be adequately sampled over the entire bandwidth to ensure negligible interpolation errors when the impulse response is being calculated. The Max Frequency parameter should never be set to a value which is greater that the maximum S-parameter data frequency. Doing so will lead to erroneous results as the available data would have to be extrapolated. S-parameter data must also extend all the way down to DC.
Transient and Convolution Simulation
Index

C
convolution
parameters, 1-17
convolution analysis, 1-7
convolution simulations
See transient/convolution simulations

G
GaAsFETs
and TAU parameter in
transient/convolution simulations, 1-26
Gear’s method, 1-10
Gibb’s phenomenon, 1-28
GMIN, 1-19

H
harmonic balance simulations
initial guesses for, 1-11

I
initial conditions
setting for transient analysis, 1-19
integration methods
in transient/convolution simulations, 1-10

M
maximum order, 1-23

N
noise
Monte-Carlo in time domain, 1-12

S
simulation parameters
transient/convolution simulations, 1-13
simulation solutions
reusing, 1-11
solutions
reusing, 1-11
S-parameters
in transient simulations, 1-29

T
TAU parameter

in GaAsFETs, 1-26
transient simulations
transient analysis, 1-7
transient/convolution simulations
about, 1-1
avoiding errors, 1-28
correction parameters, 1-19
integration method, 1-10
noncausal frequency responses, 1-28
overview, 1-7
performing, 1-2
process, 1-8
reusing solutions, 1-11
setting convolution parameters, 1-27
simulation parameters, 1-13
transeient versus convolution analysis, 1-7
troubleshooting, 1-25
using S-parameter data, 1-29
transmission lines, 1-14, 1-17
truncation error
and transient/convolution simulations, 1-15