High Dynamic Range Transient Simulation of Microwave Circuits

Sonali Luniya*, Michael B. Steer * and Carlos Christoffersen†
*Department of Electrical and Computer Engineering, North Carolina State University, Raleigh, North Carolina 27606.
†Department of Electrical Engineering, Lakehead University, Thunder Bay, Ontario P7B 5E1, Canada.

***Abstract*— Advances in communication hardware, urges a need for simulation tools, which can deal with large mixed-signal circuits. For the first time, a transient circuit analysis using state variables, with high dynamic range is presented. The dynamic range of the analysis is experimentally verified by a two-tone time domain simulation on a X-band MMIC.**

I. Introduction

Communications RF hardware is evolving to support multiple channels and multiple simultaneous functions with signals of different format and widely differing carrier frequencies. Design approaches and the simulation tools that have been developed for existing RF and microwave circuits are poorly equipped to deal with these signals. A particular characteristic of the signals in these circuits is that they cannot be described as a single modulated carrier. A conventional harmonic balance and transient envelope simulators cannot capture their response. Nevertheless complex multifunctional circuits will be developed and fielded as the use of common hardware, while being more difficult to design, promises to be more cost and size effective. Already multi carrier systems are being implemented in cellular base stations. Carriers in these circuits are close in frequency but even then it is only approximately correct to represent them as a single carrier albeit with complex modulation. The situation is worse when the channels are widely separated say when supporting the reception of GPS and cellular communications concurrently. It is not even reasonable to talk about peak-to-average ratios with such signals. New communications standards such as WCDMA and future 4G and 5G systems have stringent specifications in terms of out of band emissions and in band distortion. Many of these systems also transmit and receive simultaneously. The net result of this is that many receiver and transmit signals must support dynamic ranges of 120 dB or more. Here dynamic range is defined as the ability to detect a small signal in the presence of a major larger signal. The maximum ratio of these defining the dynamic range. A predictive simulator must be capable of achieving dynamic ranges considerably in excess of this.

In this paper we report for the first time a transient circuit simulator that is capable of achieving very high dynamic ranges which are sufficient to predictively model multifunctional systems. We examine the underlying cause of the dynamic range limiting process in transient circuit simulators. As well as increasing dynamic range the new simulation algorithm requires fewer time-steps and is faster than the conventional SPICE-like time stepping algorithm. The main issue is estimating error which in turn is used to choose time steps in the simulation. The better estimate of error made in the new procedure leads to the appropriate choice of time step. In summary, in the SPICE approach the waveform arrived at through nonlinear iteration is compared to a linear extrapolation from the last time point. While a good measure of error with digital waveforms it is a particularly bad estimate when sinusoidal signals are involved. Straight line extrapolations do not follow the peaks and troughs of sinusoids and as a result an excessive number of time points are chosen at the extremes of a sinusoid. For one the numerical noise resulting from time step selection that is too small affects dynamic range. We compare the results of two different nonlinear iterations — a much better estimate of error. As an example we consider the transient simulation of an X-band MMIC. The key result of this work is that transient simulation is now a viable simulation approach to use in modeling microwave circuits with complex signals. There is still the problem of long run times but the issue of limited dynamic range has been solved.

A. Error Control Techniques

The time step used in time-domain analysis using numerical integration to determine the circuit response at one instance of time given the circuit's response at a previous instance of time, depends on the circuit activity. The time step dynamically changes according to the rate of transitions of the voltages and currents. This ensures accuracy and convergence for circuits with large and rapid voltage and current transitions. During times of low circuit activity the time step is increased to reduce simulation time [3]. Consider the following differential equation,

$$x^{'} = f(x,t) \tag{1}$$

where x is an unknown variable, t is time and $f(x,t)$ is a given function. If x_0 is the state variable at time t_0, x_1 is the state variable at time $t_1 = t_0 + h$, where h is the timestep. Different integration methods predict different values of x. The colloquial wisdom is that Backward Euler integration tends to overdamp the solution whereas Trapezoidal tends to underdamp the solution. The task of an error correcting algorithm is to maintain a minimum error by finding the optimum time step. The equivalent integral function of Equation (1) is

$$x(t_1) = x(t_0) + \int_{t_0}^{t_1} f(x,t)dt \tag{2}$$

where t_0 and t_1 are two time points as defined above. The difference between the two time steps is very small. Hence the integral equation can be discretized as,

$$x_1 \approx x(t_1) \approx x(t_0) + x^{'}(t_1 - t_0) \tag{3}$$

and $x_0 \approx x(t_0)$. Which gives,

$$x_1 = x_0 + hx^{'}. \tag{4}$$

Therefore the generic expression is

$$x_n = x_{n-1} + hx^{'}. \tag{5}$$

This indicates that the future value can be computed based on the current value. The different integration methods differ in the method used to estimate $x^{'}$. The generic integration formula used by the different integration formulae can be reduced to

$$x_n^{'} = ax_n + b_{n-1} \tag{6}$$

where a is a constant and b_{n-1} depends on the previous values of x. Backward Euler integration is a first order differential method as the value of the state variable at any instant depends only on the value of the state variable at the previous instant. Setting $x^{'} = x_n^{'}$ in Equation (6) yields, the coefficients of the generic integration formula as,

$$a = \frac{1}{h}\ ,\ b_{n-1} = \frac{-1}{h}x_{n-1} \tag{7}$$

Trapezoidal integration is a second order differential method as the value of the state variable at any instant depends on the value of the state variables at previous two instances. It approximates the derivative by setting $x^{'} = \frac{(x_n^{'} + x_{n-1}^{'})}{2}$ making the coefficients of the generic integration formulae as,

$$a = \frac{2}{h}\ ,\ b_{n-1} = \frac{-2}{h}x_{n-1} - x_{n-1}^{'}. \tag{8}$$

Due to the difference in the integration formulas, each method will produce a different result when used to discretize a given function. The performance of a method is determined by its *accuracy* and *stability*. Since the numerical integration solution is only an approximation to the exact solution, a finite amount of error, known as local truncation error *(LTE)* may be introduced at each time point. How the *LTE* accumulates over a large number of time points is a measure of the stability of an integration method. If a method is unstable it will diverge from the exact solution over a large number of timepoints. The accuracy and stability of an integration method depends on the function it is applied to and time step used. Decreasing the step size improves the accuracy. It forces the simulator to solve more points which consequently results in longer simulation time. Decreasing the time step also increases the chance of stepping into or close to a model discontinuity and failing to converge. Trapezoidal integration suffers from a failure mechanism called *trapezoidal overshoot* [4]). Trapezoidal oscillation occurs when the integration step size is too large to follow the curvature of a given function. The result is a predicted solution that appears to oscillate around the correct solution from one time point solution to the next. All integration methods suffer from a failure mechanism called *accumulated error*, which occurs in periodic circuits and long transient simulations. If the overestimated/underestimated errors do not cancel each other out, the accumulated error tends to increase with each new timepoint.

B. Predictors

The Backward Euler and Trapezoidal discretization formulae use the derivative at the next point. But at the beginning of the analysis this value is not known and cannot be reasonably estimated. Therefore in order to start the calculations an approximate value must be computed. This is done in various ways, the simplest being the result of the previous step. This is used in the implementation. Another possibility for initializing analysis at the next time step is to use Forward Euler formula,

$$x_1 = x_0 + hx_0^{'}. \tag{9}$$

Once the predicted value is inserted into the corrector (say Backward Euler or Trapezoidal), iteration is performed to correct the mismatch. This iteration is usually performed using Newton's iteration. Predictors are however not absolutely necessary, a better prediction will result in fewer iterations. Since predictors are very simple to use, their application is highly desirable. They also help in estimating the errors committed and in controlling the step size h.

C. Algorithm

Conventional time step control algorithms use a straight line extrapolation i.e Forward Euler Integration, to calculate the initial guess of the state variable at the next time step. The difference between the extrapolated value and the converged value is the estimated local truncation error (*LTE*). This technique works well with digital circuits, but for inductive and sinusoidal circuits the estimated *LTE* is extremely poor giving an extremely poor estimate of the timestep [5]. Newton Raphson iterations converge only if the initial guess is close to the solution. Hence to achieve convergence this technique reduces the timestep to a very small value. In RF circuits which are mostly sinusoidal circuits, this technique does not work very well. A new approach to estimate the *LTE* is developed. The initial guess of the state variable at the next time step is calculated using Backward Euler Integration method. This solution is the initial guess for the Trapezoidal Integration, which gives the final solution at the time step. The difference between the final Trapezoidal solution and the Backward Euler solution is the estimated *LTE*. The Backward Euler and Trapezoidal Integration are both predictor-corrector techniques. In this approach the prediction-correction steps are applied twice, giving a solution closer to the actual solution. Thus the new approach eliminates the unnecessary cut in time steps. The value of the next timestep is predicted by:

$$t_f = \sqrt{\frac{\max\left(|x_{be}|, |x_{tr}|\right) * RELTOL + ABSTOL}{LTE}} \tag{10}$$

$$t_{new} = t_{old} * \sqrt{t_f * TRTOL * 2.0} \tag{11}$$

where, x_{be} is the voltage/current predicted by Backward Euler Integration and x_{tr} is the voltage/current calculated at the current time step using Trapezoidal Integration. The predicted value between iterations cannot change more than *RELTOL* percent of the final value. *ABSTOL* complements *RELTOL* at zero crossings, where the relative tolerance goes to zero, making the solution to be infinitely accurate. The new timestep is inversely proportional to the current *LTE*. In sinusoidal circuits, the calculated *LTE* using the new time step control is smaller compared to the conventional extrapolation approach, resulting in larger and fewer time steps, without any loss in accuracy. As this algorithm makes a better prediction than the conventional approach, it is able to follow curves with less time points.

II. Time Step Control Implementation

The formulation of the system equations begins with the partitioned network of [2]with the nonlinear elements replaced by variable voltage or current sources.

The system of linear equations representing the error function using sparse matrices for such a formulation is [1][2]:

$$\begin{pmatrix} [\mathbf{G}+\mathbf{C}a] & -[\mathbf{T}^T\mathbf{J}_i] \\ \mathbf{T} & -\mathbf{J}_v \end{pmatrix} \begin{pmatrix} \mathbf{u}_n^{(j+1)} \\ \mathbf{x}_n^{(j+1)} \end{pmatrix} = \begin{pmatrix} [s_{f,n} - \mathbf{C}a\mathbf{b}_{n-1}] & +\mathbf{T}^T[\mathbf{i}_{NL}(\mathbf{x}_n^j) - \mathbf{J}_i\mathbf{x}_n^j] \\ \mathbf{v}_{NL}(\mathbf{x}_n^j) & -\mathbf{J}_v\mathbf{x}_n^j \end{pmatrix}. \tag{12}$$

where, $\mathbf{G}$ consists of all conductors and frequency dependent MNAM stamps, $\mathbf{C}$ consists of capacitors and inductor values and other values that are associated with dynamic elements, $\mathbf{T}$ is the incidence matrix, s_f vector is due to the independent sources in the circuit,$\mathbf{v}_{NL}$, $\mathbf{i}_{NL}$ are the voltage and current vectors at the common ports of the nonlinear subnetwork, $\mathbf{u}_n$ is the vector of nodal voltages, $\mathbf{x}_n$ is the vector of state variables, $\mathbf{J}_v$, $\mathbf{J}_i$ are the voltage and current Jacobian matrices. From Equation (12) it can be seen that an equivalent linear circuit is formed for every nonlinear element. The circuit is only equivalent to the nonlinear element at the j^{th} iteration because its element values (but not its topology) change by discrete amounts at every iteration [7][6]. This circuit is repeatedly solved with updated element values till convergence is achieved and is solved at every time step. The above equation is solved using LU factorization. The size of the resulting algebraic system of linear equations is (n_m+n_s)x(n_m+n_s), where n_m is equal to the number of non-reference nodes in the circuit plus number of additional required variables and n_s is the number of state variables [4]. $\mathbf{G}+\mathbf{C}a$ is updated every time the time step is changed. $\mathbf{T}$ is constant. The right hand side vectors, $\mathbf{J}_i$ and $\mathbf{J}_v$ change at every Newton iteration and every time step, changing only the values of the elements in the equivalent linear circuit by a discrete amount.

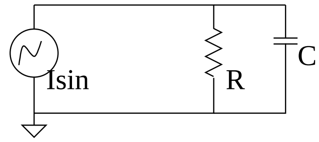

Fig. 1. Simple RC circuit.

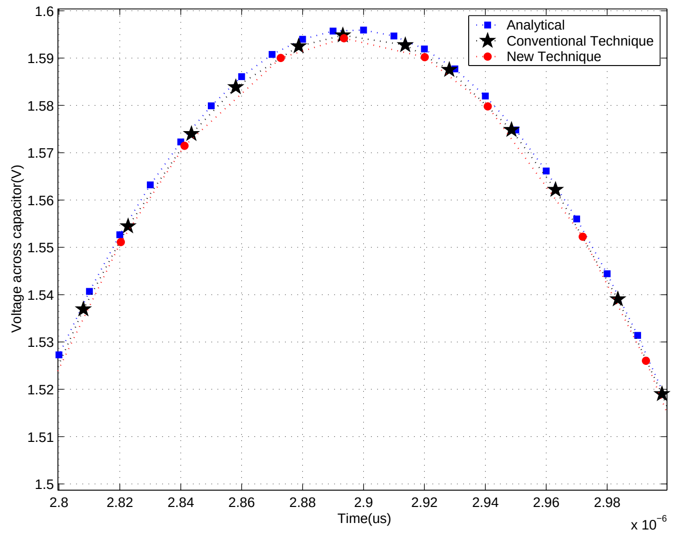

Fig. 2. Comparison of voltage across capacitor between conventional, new and analytical calculations.

III. Simulation and Results

A. Simple RC network

Consider a simple RC circuit, shown in Figure 1. The circuit was simulated on an ULTRA-SPARC 1 workstation with a maximum timestep of 0.1μs. The circuit was simulated for 10μs. Figure 2 compares the voltage across the capacitor using the the two techniques, and are in excellent agreement with the analytical solution. As expected, the absolute error (difference between the analytical solution and simulated solution) introduced in the final solution reduces with the relative tolerance, *RELTOL*, as shown in Figure 3. At high relative tolerance the absolute error introduced by the conventional technique, is approximately two times that of the new technique. To maintain similar amount of tolerance the conventional approach shows

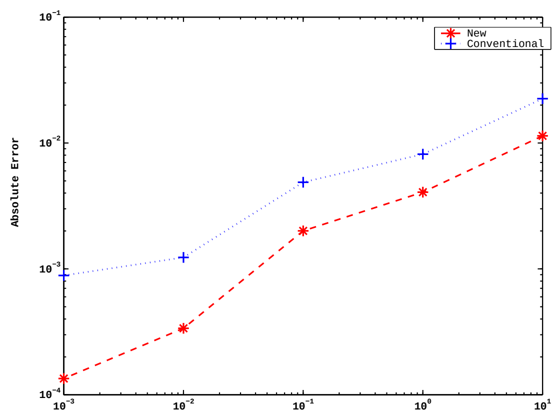

Fig. 3. Comparison of absolute error as a function of relative tolerance

% RELTOL	Conventional Approach	New Approach	% Improvement
0.1	224	159	29.01
0.01	493	350	29.00
0.001	668	476	28.7

TABLE I
COMPARISON OF NUMBER OF TIMEPOINTS OF A TRANSIENT SIMULATION OF A SIMPLE RC NETWORK.

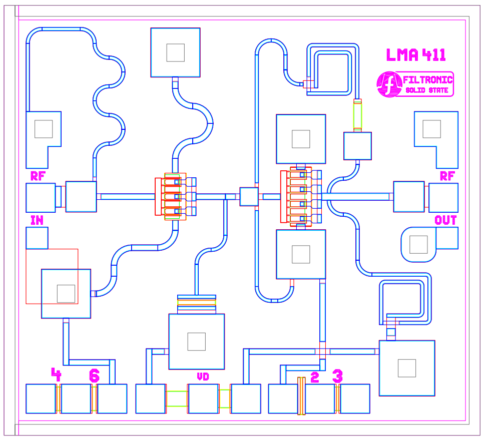

Fig. 4. Layout of the LMA411 X-band MMIC.

bunching of data points at the peak of the sinusoidal signal. Table I compares the total number of timepoints required by the two methods. The straight line extrapolation overshoots the sinusoid at every timepoint, resulting in a bigger *LTE*, effectively reducing the predicted timestep. Whereas with the new approach, at the peak of sinusoid, the predicted value is very close to the final value, reducing the *LTE* and increasing the timestep. The overall improvement in the number of timepoints is approximately 29.0%.

B. LNA MMIC

To test the dynamic range of the new technique a two tone test was done on a Filtronic Solid State (LMA411 MMIC) high dynamic range low noise PHEMT amplifier. The LNA operates from 8.5 to 14GHz. The amplifier is reactively matched at the two ports which provides a 18dB nominal gain with a 1-dB gain compression power output of +17dBm. It can be used as a pre-driver amplifier for phased array radar as well as commercial communications applications.

In this paper, the PHEMTs of the LNA were modeled using the Curtice-Cubic model of a MESFET. The transmission lines were modeled as generalized transmission lines. As this is a time domain analysis the transmission lines were modeled as RLGC elements. A 10GHz sinusoid with 6-V drain DC bias and 0-V gate DC bias was applied to the MMIC. To test the dynamic range, a two tone test, with the first tone at 10GHz and a input power level of -20dB and a second tone at 11GHz was performed. The input power level of the second tone was varied from -20dB to -120dB. A second tone 100dB below

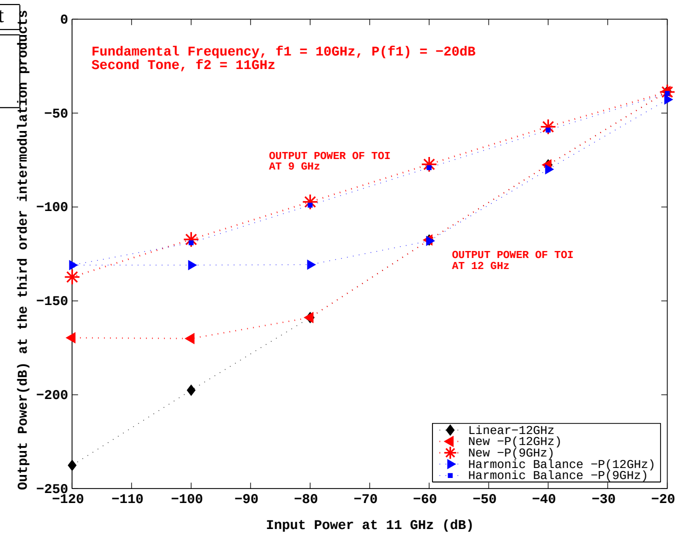

Fig. 5. Comparison between the new technique and commercial harmonic balance simulator, of the IM3 output power levels for the LMA411 MMIC.

the carrier, will have a IM3 product more than 100dB below the carrier, which will help to test the maximum possible dynamic range of the technique. As desired, a gain of 16dB was seen at the output of the two tones. The output power levels of the IM3 products at 9GHz and 12GHz are plotted in Figure 5. The output power level at 9GHz, has a slope of 1, whereas the output power level at 12GHz, has a slope of 2. The new technique performs better than commercial harmonic balance simulators at low input power levels. The commercial harmonic balance simulator has a numerical noise floor of -130dB. Below -60dB input power the commercial simulator hits the numerical noise floor, showing a maximum dynamic range of 125dBc. Whereas, since the new technique follows sinusoidal curves better, it can detect very small signals, as the *LTE* is small enough not to encapsulate tiny signals. Consequently the new technique has a numerical noise floor of approximately -170dB. It hits the numerical noise floor below -80dB input power. The ultimate numerical noise floor is defined by the fourier transforms performed on the time domain signal. The dynamic range of the new technique is approximately 165dBc.

IV. CONCLUSION

The importance of this paper was the development of a new timestep control technique for a transient analysis using state variable based device models to achieve high dynamic range. The new technique has a better estimate of error, helping to achieve a dynamic range of approximately 165dBc. The new technique achieves similar computation accuracy as the conventional time stepping technique, but with a smaller number of timepoints. The transient analysis was developed in a general purpose simulator, *f*REEDA™ (`http://freeda.org`) and is targeted towards circuits operating at RF and microwave frequencies.

References

[1] S. Luniya, *SPICE like sparse transient analysis*, Masters thesis, Department of Electrical and Computer Engg, North Carolina State University, 2002.

[2] C. E. Christoffersen and M. B. Steer,*Generalized circuit formulation for the transient simulation of circuits using wavelet, convolution and time-marching techniques*, Proc. of the 15th European Conference on Circuit Theory and Design, Aug 2001, pp. 205-208.

[3] M. B. Steer, *"Transient and Steady-State Analysis of Nonlinear RF and Microwave Circuits"*, ECE718 class notes, 2003.

[4] J. Vlach and K. Singhal, *Computer Methods for Circuit Analysis and Design*, Chapman and Hall, 1994.

[5] Ron. M. Kielkowski, *Inside SPICE*, 2nd ed, McGraw Hill, 1998.

[6] R. J. Schwarz and B. Freiland, *Linear Systems*, McGraw Hill, 1965.

[7] L. O. Chua and P. Lin, *Computer Aided Analysis of Electronic Circuits*, Prentice-Hall, 1975.