

Review Article

Radio Frequency Numerical Simulation Techniques Based on Multirate Runge-Kutta Schemes

Jorge F. Oliveira^{1,2} and José C. Pedro²

¹ School of Technology and Management, Polytechnic Institute of Leiria, Morro do Lena, Alto Vieiro, Apartado 4163, 2411-901 Leiria, Portugal

² Institute of Telecommunications, University of Aveiro, Campus Universitário de Santiago, 3810-193 Aveiro, Portugal

Correspondence should be addressed to Jorge F. Oliveira, jorge.oliveira@ipleiria.pt

Received 20 August 2011; Revised 20 October 2011; Accepted 4 November 2011

Academic Editor: Oluwole D. Makinde

Copyright © 2012 J. F. Oliveira and J. C. Pedro. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Electronic circuit simulation, especially for radio frequency (RF) and microwave telecommunications, is being challenged by increasingly complex applications presenting signals of very different nature and evolving on widely separated time scales. In this paper, we will briefly review some recently developed ways to address these challenges, by describing some advanced numerical simulation techniques based on multirate Runge-Kutta schemes, which operate in the one-dimensional time and also within multidimensional frameworks.

1. Introduction

In recent years, radio frequency (RF) and microwave electronic circuit simulation has been conducted to an increasingly demanding scenario of heterogeneous broadband and strongly nonlinear wireless communication circuits, presenting a wide variety of slowly varying and fast changing state variables (node voltages and branch currents). In such circuits, the baseband analog blocks, the digital blocks, and the RF blocks may be all intricately mixed. Classical simulation tools are not capable of handling this kind of circuits in an efficient way because they do not perform any distinction between nodes or blocks within the circuit, considering their time constants and/or excitation regimes. Thus, all the blocks in the circuits are treated in the same way, which means that the same numerical algorithm is required to simultaneously compute the response of the digital blocks, the baseband analog blocks, and the RF blocks. Taking into account that signals in different blocks (different parts of the circuit) have completely different formats and evolve on widely separated time scales

(which may differ from three, or more, orders of magnitude), it is easy to conclude that the application of the same numerical scheme to all the blocks will result in high inefficiency.

To cope with this scenario, some innovative time-domain techniques have been proposed in the literature in the last few years. In order to benefit from the different rates of variation of slowly varying and fast-varying currents and voltages (circuits' state variables) in different parts of the circuit, several advanced numerical techniques based on multirate Runge-Kutta schemes have been proposed to operate within one-dimensional and multivariate frameworks. Such techniques include diverse circuit partitioning strategies, which allow the simulator to automatically split the network into subcircuits according to the different time rates of change of their state variables. In order to provide a general overview on these simulation techniques, this paper is organized as follows.

After this brief introduction, Section 2 provides some general background material on standard time-domain circuit simulation techniques commonly used for computing the numerical solution of ordinary unirate problems. Then, Section 3 is essentially devoted to the presentation of some mathematical details of the numerical simulation algorithms based on multirate Runge-Kutta schemes. Finally, Section 4 concludes this paper.

2. Time-Domain Circuit Simulation Fundamentals

2.1. Mathematical Model of an Electronic Circuit

The behavior of an electronic circuit can be described with a system of equations involving voltages, currents, charges, and fluxes. This system of equations can be constructed from a circuit description, using, for example, nodal analysis, which involves applying the Kirchhoff current law to each node in the circuit, and applying the constitutive or branch equations to each circuit element. In general, systems generated this way have the form

$$\mathbf{p}(\mathbf{y}(t)) + \frac{d\mathbf{q}(\mathbf{y}(t))}{dt} = \mathbf{x}(t), \quad (2.1)$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ and $\mathbf{y}(t) \in \mathbb{R}^n$ stand for the excitation (independent voltage and current sources) and state variable (node voltages and branch currents) vectors, respectively. $\mathbf{p}(\mathbf{y}(t))$ represents memoryless linear or nonlinear elements, as resistors, nonlinear voltage-controlled current sources, and so forth, while $\mathbf{q}(\mathbf{y}(t))$ models dynamic linear or nonlinear elements, as capacitors (represented as linear or nonlinear voltage-dependent electric charges), or inductors (represented as linear or nonlinear current-dependent magnetic fluxes).

The system of (2.1) is, in general, a *differential algebraic equations'* (DAE) system. For achieving an intuitive explanation of the mathematical formulation of (2.1), let us consider the basic illustrative example depicted in Figure 1. This circuit is composed of a current source connected to a linear inductance and two nonlinear circuit elements commonly used when modeling semiconductor devices (a nonlinear capacitance and a nonlinear voltage-dependent current source). These nonlinearities are assumed as quasistatic [1, 2] and thus

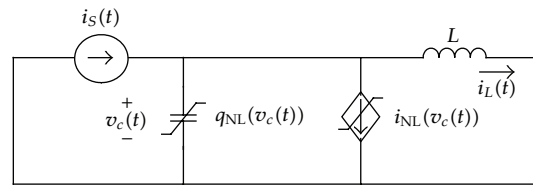


Figure 1: Nonlinear dynamic circuit example.

are described by algebraic constitutive relations of voltage-dependent charge and voltage-dependent current. A nodal analysis of this circuit leads to the following system of equations in the capacitor voltage $v_C(t)$ and the inductor current $i_L(t)$:

$$\begin{aligned} i_{NL}(v_C(t)) + i_L(t) + \frac{d}{dt}q_{NL}(v_C(t)) &= i_s(t), \\ v_C(t) + \frac{d}{dt}(-L \cdot i_L(t)) &= 0. \end{aligned} \quad (2.2)$$

This system can be seen as a particular case (in \mathbb{R}^2) of the DAE system of (2.1), in which the excitation vector, $\mathbf{x}(t)$, and the vector of state variables, $\mathbf{y}(t)$, are given by

$$\mathbf{x}(t) = \begin{bmatrix} i_s(t) \\ 0 \end{bmatrix}, \quad \mathbf{y}(t) = \begin{bmatrix} v_C(t) \\ i_L(t) \end{bmatrix}. \quad (2.3)$$

The DAE system of (2.1) may obviously be written in other forms. For instance, if we apply the chain differentiation rule to the dynamic term of its left hand side, we can obtain

$$\frac{d\mathbf{q}(\mathbf{y})}{d\mathbf{y}} \frac{d\mathbf{y}(t)}{dt} = \mathbf{x}(t) - \mathbf{p}(\mathbf{y}(t)), \quad (2.4)$$

or

$$\mathbf{M}(\mathbf{y}(t)) \frac{d\mathbf{y}(t)}{dt} = \mathbf{x}(t) - \mathbf{p}(\mathbf{y}(t)), \quad (2.5)$$

in which $\mathbf{M}(\mathbf{y}(t))$ is usually known as the *mass matrix*. If this matrix is nonsingular, then the DAE system of (2.4) may degenerate into the following *ordinary differential equations'* (ODE) system:

$$\frac{d\mathbf{y}(t)}{dt} = \mathbf{M}(\mathbf{y}(t))^{-1}[\mathbf{x}(t) - \mathbf{p}(\mathbf{y}(t))], \quad (2.6)$$

which can be rewritten in the classical form

$$\frac{d\mathbf{y}(t)}{dt} = \mathbf{f}(\mathbf{t}, \mathbf{y}(t)), \quad (2.7)$$

commonly used in the mathematical literature. When $\mathbf{M}(\mathbf{y}(t))$ is singular, the DAE system of (2.4) will not degenerate into an ODE system, but it is often possible to express it as a set of algebraic equations combined with a set of differential equations of the form (2.7).

From the above, we conclude that, in many cases, electronic circuits may be described by ODE systems instead of DAE systems. For example, if we return to the simple nonlinear dynamic circuit of Figure 1 and rewrite (2.2) as

$$\begin{aligned} \frac{dq_{\text{NL}}(v_C)}{dv_C} \frac{dv_C(t)}{dt} &= i_s(t) - i_{\text{NL}}(v_C(t)) - i_L(t), \\ L \frac{di_L(t)}{dt} &= v_C(t), \end{aligned} \quad (2.8)$$

or, in its vector-matrix form, as

$$\begin{bmatrix} \frac{dq_{\text{NL}}(v_C)}{dv_C} & 0 \\ 0 & L \end{bmatrix} \begin{bmatrix} \frac{dv_C(t)}{dt} \\ \frac{di_L(t)}{dt} \end{bmatrix} = \begin{bmatrix} i_s(t) - i_{\text{NL}}(v_C(t)) - i_L(t) \\ v_C(t) \end{bmatrix}, \quad (2.9)$$

we can easily see that if $dq_{\text{NL}}(v_C)/dv_C \neq 0$, then the mass matrix is nonsingular, and the circuit may be described by an ODE system expressed in the classical form of (2.7), which, in this case, will simply result in

$$\begin{aligned} \frac{dv_C(t)}{dt} &= \left[\frac{dq_{\text{NL}}(v_C)}{dv_C} \right]^{-1} [i_s(t) - i_{\text{NL}}(v_C(t)) - i_L(t)], \\ \frac{di_L(t)}{dt} &= \frac{1}{L} v_C(t). \end{aligned} \quad (2.10)$$

2.2. Transient Analysis: Time-Step Integration

Obtaining the solution of (2.1) over a specified time interval $[t_0, t_{\text{Final}}]$ with a specific initial condition $\mathbf{y}(t_0) = \mathbf{y}_0$ is what is usually known as an *initial value problem*, and computing such solution is frequently referred to as *transient analysis*. The most natural way to evaluate $\mathbf{y}(t)$ is to numerically time-step integrate (2.1) directly in time domain. One possible way to do that consists in simply converting the differential equations into difference equations, in which the time derivatives are approximated by appropriate incremental ratios. With this strategy, the nonlinear differential equations' system of (2.1) is converted into a purely nonlinear algebraic system.

The above formulation derives directly from the intuitive idea that derivatives can be approximated, and thus simply replaced, by finite-difference schemes. Although this technique can be used to compute the transient response of a generic electronic circuit described by (2.1), there is an alternative strategy which is more often employed to find the solution of initial value problems. Such strategy consists in using initial value solvers, as linear multistep methods [2–5] or Runge-Kutta methods [3–5] (the most popular time-step integrators). Both of these classes of methods can provide a wide variety of explicit and implicit numerical schemes, with very distinct properties in terms of order (accuracy) and

numerical stability. However, since a substantial part of the research work reviewed in this paper is based on modern multirate Runge-Kutta schemes, we will restrict our presentation to only these numerical techniques. Linear multistep methods will not be addressed.

2.3. Runge-Kutta Methods

In view of the fact that the well-established theory of numerical integration is oriented toward the solution of standard ODEs, we will now consider the form of (2.7) for the mathematical description of a circuit's operation. So, let us consider a generic initial value problem with n state variables, expressed in its classical form by the system of (2.7) and the initial condition $\mathbf{y}(t_0) = \mathbf{y}_0$, that is,

$$\frac{d\mathbf{y}(t)}{dt} = \mathbf{f}(\mathbf{t}, \mathbf{y}(t)), \quad \mathbf{y}(t_0) = \mathbf{y}_0, \quad t_0 \leq t \leq t_{\text{Final}}, \quad \mathbf{y}(t) \in \mathbb{R}^n. \quad (2.11)$$

Definition 2.1 (Runge-Kutta (RK) method). A standard s -stage RK method expressed by its Butcher tableau $(\mathbf{b}, \mathbf{A}, \mathbf{c})$ [5]

$$\begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array} \quad (2.12)$$

for obtaining the numerical solution of (2.11) at the time instant $t_1 = t_0 + h$ is defined as [3, 5]

$$\mathbf{y}_1 = \mathbf{y}_0 + h \sum_{i=1}^s b_i \mathbf{k}_i \approx \mathbf{y}(t_0 + h), \quad (2.13)$$

where

$$\mathbf{k}_i = \mathbf{f} \left(t_0 + c_i h, \mathbf{y}_0 + h \sum_{j=1}^s a_{ij} \mathbf{k}_j \right), \quad i = 1, 2, \dots, s. \quad (2.14)$$

The algorithm defined by (2.13), (2.14) allows the numerical solution \mathbf{y}_i at any generic time instant t_i to be evaluated from its previous calculated value \mathbf{y}_{i-1} . If we have $a_{ij} = 0$ for $j \geq i$, $i = 1, 2, \dots, s$, then each of the \mathbf{k}_i in (2.14) is explicitly given in terms of the previously computed \mathbf{k}_j , $j = 1, 2, \dots, i-1$, and the method is then an explicit Runge-Kutta method. If this is not the case, then the method is implicit and, in general, it is necessary to solve a nonlinear system of $n \times s$ algebraic equations to simultaneously compute all the \mathbf{k}_i . In general, any iterative technique (e.g., fixed-point iteration or Newton-Raphson iteration) may be used to solve the nonlinear system of (2.14).

Runge-Kutta methods are universally utilized for time-step integrating initial value problems and differ from linear multistep methods in several aspects. Since they present a genuine one-step format, one of their main advantages is that there is no difficulty in changing the steplength h in a dynamic time-step integration process (in opposition to multistep methods where considerable difficulties may be encountered when we want to change steplength [5]). It must be noted that small step sizes can provide a good accuracy in the simulation results but may conduct to large computation times. On the contrary, large step sizes will reduce the computation time but will definitely conduct to poorer accuracy. A good compromise between accuracy and simulation time is achieved when h is dynamically selected according to the solution's rate of change. The automatic step size control is based on the estimation of local errors, for which diverse techniques can be used, such as extrapolation techniques or embedded RK formulas. All theoretical and technical details of implementation of these techniques, as well as many other aspects of the RK methods, such as consistency, convergence, order conditions, numerical stability, and so forth, can be seen, for example, in [3] or [5].

2.4. Conventional Transient Circuit Simulation: SPICE

In the above, we have seen that the most natural way of simulating an electronic circuit is to numerically time-step integrate the ordinary differential system describing its operation. So, it should be of no surprise that this straightforward technique was used in the first digital computer programs of circuit analysis and is still nowadays the most widely used numerical method for that purpose. It is present in all SPICE (which means simulation program with integrated circuit emphasis) [6] or SPICE-like computer programs.

SPICE was initially developed at the Electronics Research Laboratory of the University of California, Berkeley, in the early 1970s. The real popularity of SPICE started with SPICE2 [6] in 1975, which was a much-improved program than its original version (SPICE1), containing several analyses (AC analysis, DC analysis, DC transfer curve analysis, transient analysis, etc.) and device models needed to design integrated circuits of that time. Other versions of SPICE have been developed along the years, and today many commercial simulators are based on SPICE. However, its application to RF circuits may cause some problems resulting from the specific behavior of RF systems. To understand this, we must recall that RF signals are typically narrowband. This means that a data signal with a relatively low bandwidth is transmitted at a very high carrier frequency. To simulate a sufficient portion of the data signal, for example, to estimate bit error rate in modern wireless systems, a large number of carrier periods must be time-step integrated, and thus a huge number of time samples is required (a large consumption of memory and computational time).

2.5. Steady-State Analysis

Although some simulation tools focus on transient analysis (SPICE-like simulation), the steady-state behavior of the circuits is of primary interest to RF and microwave designers. The main reason for this is that wireless systems are expected to handle a sinusoidal RF carrier modulated by a slowly varying envelope, so that most aspects of system performance are easier to characterize and verify in the carrier's steady state. For instance, harmonic or intermodulation distortion, noise, power, and transfer characteristics such as gain, or

impedance are examples of quantities that are best defined in the frequency domain and thus measured when the circuit is in steady state.

Time-step integration engines, as the ones mentioned above, and which are tailored for finding the circuit's transient response, are not adequate for computing the steady-state. As discussed above, time-step integration is a numerical technique intended to obtain the solution of an initial value problem, as it evaluates $\mathbf{y}(t)$ for a set of successive time instants (time steps) given an initial condition $\mathbf{y}(t_0)$. However, if the objective is the determination of the steady state, there is no alternative other than to pass through the lengthy process of integrating all transients, and expecting them to vanish. In circuits presenting extremely different time constants, or high Q resonances, as is typically the case of narrowband RF and microwave circuits, time-step integration can be very inefficient. Indeed, in such cases, frequencies in steady-state response are much higher than the rate at which the circuit approaches this regime, or, in other words, the ratio between the state variables' highest and the lowest frequency components is very large. Thus, the number of discretization time steps used by the numerical integration scheme will be enormous, because the time interval over which the differential equations must be numerically integrated is set by the lowest frequency, or by how long the circuit spends in achieving the steady state, while the size of the time steps is determined by the highest frequency component.

If the steady-state response of a circuit consists of generic waveforms presenting a common period, then the circuit is said to be in *periodic steady-state*. Computing the periodic steady state response of an electronic circuit involves finding the initial condition, $\mathbf{y}(t_0)$, for the differential system that describes the circuit's operation, such that the solution at the end of one period matches the initial condition, that is, $\mathbf{y}(t_0) = \mathbf{y}(t_0 + T)$, where T is the period. Problems of this form, that is, those of finding the solution to a system of ordinary differential equations that satisfies constraints at two or more distinct points in time, are referred to as *boundary value problems*. In this particular case, we have a periodic boundary value problem that can be formulated as

$$\begin{aligned} \mathbf{p}(\mathbf{y}(t)) + \frac{d\mathbf{q}(\mathbf{y}(t))}{dt} &= \mathbf{x}(t), \\ \mathbf{y}(t_0) &= \mathbf{y}(t_0 + T), \quad t_0 \leq t \leq t_0 + T, \quad \mathbf{y}(t) \in \mathbb{R}^n, \end{aligned} \quad (2.15)$$

or, in the classical ODE form, as

$$\begin{aligned} \frac{d\mathbf{y}(t)}{dt} &= \mathbf{f}(t, \mathbf{y}(t)), \\ \mathbf{y}(t_0) &= \mathbf{y}(t_0 + T), \quad t_0 \leq t \leq t_0 + T, \quad \mathbf{y}(t) \in \mathbb{R}^n, \end{aligned} \quad (2.16)$$

where the condition $\mathbf{y}(t_0) = \mathbf{y}(t_0 + T)$ is known as the *periodic boundary condition*.

Solving (2.16) involves computing a numerical solution that simultaneously satisfies the differential system and the two-point periodic boundary condition. Certainly, there is no shortage of mathematical literature describing methods for solving boundary value problems. However, a particular technique has been found especially useful for electronic circuit problems. This technique is known as the *shooting-Newton* method [7].

2.6. The Shooting-Newton Method

The time-domain method most commonly used for numerically evaluating the periodic steady-state solution of an electronic circuit is the *shooting method*. Shooting solves boundary value problems by computing the solution to a succession of initial value problems with progressively improved guesses of an initial condition, which ultimately results in the steady state. In a circuit's steady-state simulation, shooting begins by simulating the circuit for one period using some guessed initial condition (generally determined from a previous DC analysis). Then, the computed solution at the end of the period is checked, and if it does not agree with the initial condition, the initial condition is wisely modified. The circuit is then resimulated with the adjusted initial condition, and this process is repeated until the solution after one period matches the initial condition.

In order to provide some mathematical details on the implementation of the shooting method, let us consider (2.16). Now suppose that we want to numerically time-step integrate the differential system in (2.16) with an initial value solver. As stated above, time-step integration is tailored for transient analysis but is inadequate for computing steady-state responses. The problem comes from the fact that we do not know a priori which initial condition $\mathbf{y}(t_0)$ must be considered that will lead to the steady-state solution in the period T , that is, that will satisfy the periodic boundary condition $\mathbf{y}(t_0) = \mathbf{y}(t_0 + T)$. So, we are trying to solve a boundary value problem with an initial value solution technique. One possible way to convert the initial value solution procedure into a boundary value problem solver consists of guessing the initial estimate of $\mathbf{y}(t_0)$, or shooting for $\mathbf{y}(t_0)$, time-step integrating the differential system from $t = t_0$ until $t = t_0 + T$, comparing the resulting $\mathbf{y}(t_0 + T)$ with $\mathbf{y}(t_0)$, and then wisely updates the initial estimate. So, shooting is an iterative solver that uses an initial value technique to solve a boundary value problem. In the end, it relies on finding the solution of

$$\mathbf{y}(t_0) = \mathbf{y}(t_0 + T) \iff \mathbf{y}(t_0) - \mathbf{y}(t_0 + T) = 0. \quad (2.17)$$

Let us now define $\mathbf{y}(t_0 + T) = \phi(\mathbf{y}(t_0), T)$ and rewrite (2.17) as

$$\phi\mathbf{y}((t_0), T) - \mathbf{y}(t_0) = 0, \quad (2.18)$$

where ϕ is the *state-transition function* [7, 8]. An easy way to solve (2.18) consists in using the fixed-point iteration solver, which, in this case, would simply result in

$$\mathbf{y}^{[r+1]}(t_0) = \phi(\mathbf{y}^{[r]}(t_0), T). \quad (2.19)$$

However, shooting with the fixed-point iteration technique is obviously equivalent to integrating the original differential system from $t = t_0$ until all transients decay. So, it is generally a useless technique because the convergence to the periodic steady-state solution may be extremely slow. A well-known tactic to accelerate the route to steady state, that is, to accelerate the convergence of (2.18) to its solution, is the so-called *shooting-Newton* technique. As happens with any other shooting technique, shooting Newton is based on guessing initial conditions. However, it can take advantage of the fact that, although electronic circuits can be strongly nonlinear, their state-transition functions are usually only mildly nonlinear.

This means that slight perturbations on the initial condition (starting state) produce almost proportional perturbations in the subsequent time states. Taking this into account, it is easy to conclude that (2.18) can be iteratively solved in an efficient way with the Newton's method, which in this case will lead us to

$$\phi(\mathbf{y}^{[r]}(t_0), T) - \mathbf{y}^{[r]}(t_0) + \left[\frac{\partial \phi(\mathbf{y}(t_0), T)}{\partial \mathbf{y}(t_0)} - I \right] \Big|_{\mathbf{y}(t_0)=\mathbf{y}^{[r]}(t_0)} [\mathbf{y}^{[r+1]}(t_0) - \mathbf{y}^{[r]}(t_0)] = 0, \quad (2.20)$$

where I is the $n \times n$ identity matrix. The only entity of (2.20) that is difficult to compute is the Jacobian of the state-transition function (usually referred to as the *sensitivity matrix*). In order to compute this matrix, we must take into consideration the chain differentiation rule. In fact, since $\phi(\mathbf{y}(t_0), T)$ is nothing but the numerical value \mathbf{y}_K , with K being the total number of time steps in the interval $[t_0, t_0 + T]$, which depends on the previous value \mathbf{y}_{K-1} , which, itself, depends on \mathbf{y}_{K-2} , and so forth, the sensitivity matrix can be given by

$$\frac{\partial \phi(\mathbf{y}(t_0), T)}{\partial \mathbf{y}(t_0)} = \frac{\partial \mathbf{y}_K}{\partial \mathbf{y}_{K-1}} \cdot \frac{\partial \mathbf{y}_{K-1}}{\partial \mathbf{y}_{K-2}} \cdots \frac{\partial \mathbf{y}_1}{\partial \mathbf{y}_0}. \quad (2.21)$$

It is easy to see that all the matrices in (2.21) can be individually computed along the time-step integration process. For concreteness, let us suppose that a standard Runge-Kutta method (2.13), (2.14) is being used to perform time-step integration on the consecutive iterations of the shooting method. If, for example, we want to evaluate the first matrix, then we have

$$\frac{\partial \mathbf{y}_1}{\partial \mathbf{y}_0} = I + h \sum_{i=1}^s b_i \frac{\partial \mathbf{k}_i}{\partial \mathbf{y}_0}. \quad (2.22)$$

Now, if we rewrite (2.14) as

$$\mathbf{k}_i = \mathbf{f} \left(t_0 + c_i h, \mathbf{y}_0 + h \sum_{j=1}^s a_{ij} \mathbf{k}_j \right) = \mathbf{f}(t_0 + c_i h, \mathbf{Y}_i), \quad i = 1, 2, \dots, s, \quad (2.23)$$

we obtain

$$\frac{\partial \mathbf{k}_i}{\partial \mathbf{y}_0} = \frac{\partial \mathbf{f}}{\partial \mathbf{y}} \Big|_{t_0+c_i h, \mathbf{Y}_i} \cdot \frac{\partial \mathbf{Y}_i}{\partial \mathbf{y}_0}, \quad i = 1, 2, \dots, s, \quad (2.24)$$

with

$$\frac{\partial \mathbf{Y}_i}{\partial \mathbf{y}_0} = I + h \sum_{j=1}^s a_{ij} \frac{\partial \mathbf{k}_j}{\partial \mathbf{y}_0}. \quad (2.25)$$

Although solving (2.20) and computing the sensitivity matrix may involve some extracomputational cost, shooting Newton converges to the steady-state solution much faster

than the normal time-step integration procedure (shooting with fixed-point iteration). This is the reason why it is the time-domain steady-state engine most widely used in circuit simulation.

3. Numerical Simulation Algorithms Based on Multirate Runge-Kutta Schemes

3.1. Time-Step Integration with Different Step Sizes

As stated above, dynamic behavior of some electronic circuits involves signals with widely separated rates of variation. Analog circuits presenting extremely different time constants, coupled systems of analog and digital networks, or combined technologies of RF and baseband analog (or even digital) blocks in the same circuit are typical examples where the corresponding state variables may evolve according to very distinct time scales. In such cases, node voltages or branch currents presenting slow (*latent*) and very fast (*active*) time evolution rates may coexist in the same problem. This phenomenon, in which some of the state variables are varying very slowly (or even being practically constant) within a specific time interval, while other variables exhibit fast variations in that interval, is frequently referred to as *time-domain latency* [9–15]. Another common situation where time-domain latency can be found refers to purely digital circuits. For example, in large-scale integration (LSI) or very large-scale integration (VLSI) applications, usually, only a small part of the circuits (a small number of state variables) is active in a certain time interval, whereas the majority is latent (a large number of state variables may possibly remain practically constant within that interval). In summary, many kinds of electronic circuits may exhibit time-domain latency.

As described in Section 2, time-step integration is a conventional technique that is used by SPICE-like computer programs for simulating electronic circuits. However, when integrating differential systems whose components (state variables) evolve according to different time rates, one would like to use numerical schemes that do not expend unnecessary work on slowly changing components. In fact, in such cases, traditional time-step integrators, like standard Runge-Kutta or linear multistep methods—which use the same step size for all system's components—become inefficient. To cope with this, some modern multirate Runge-Kutta (MRK) schemes have been proposed in the literature in the recent years [11–14]. These powerful schemes split the differential system of (2.11) into coupled active and latent subsystems

$$\begin{aligned} \frac{dy_A(t)}{dt} &= \mathbf{f}_A(t, \mathbf{y}_A, \mathbf{y}_L), & \mathbf{y}_A(t_0) &= \mathbf{y}_{A,0}, \\ \frac{dy_L(t)}{dt} &= \mathbf{f}_L(t, \mathbf{y}_A, \mathbf{y}_L), & \mathbf{y}_L(t_0) &= \mathbf{y}_{L,0}, \end{aligned} \quad (3.1)$$

with

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_A \\ \mathbf{y}_L \end{bmatrix}, \quad \mathbf{y}_A \in \mathbb{R}^{n_A}, \quad \mathbf{y}_L \in \mathbb{R}^{n_L}, \quad n_A + n_L = n, \quad (3.2)$$

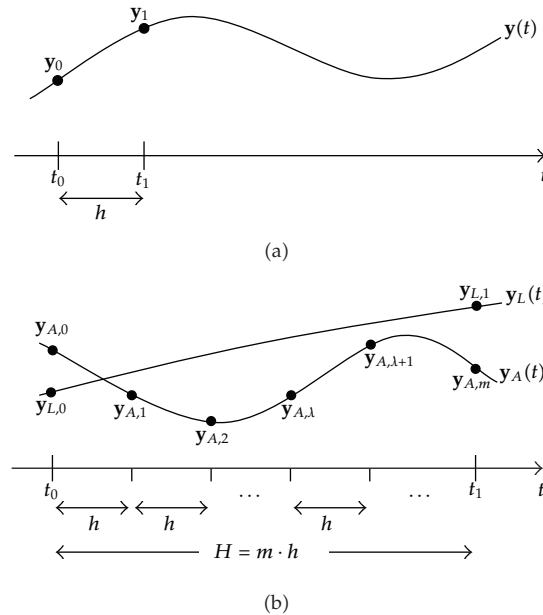


Figure 2: (a) Single step in a standard Runge-Kutta integrator. (b) Micro- and macrosteps in a multirate Runge-Kutta integrator [16].

where y_A is the active (fast-varying) state-variable components' vector, and y_L is the latent (slowly varying) state-variable components' vector. The active components are then integrated with a small step size h (*microstep*), while the latent components are integrated with a much larger step size H (*macrostep*). The number of microsteps within each macrostep is an integer that will be denoted by m , that is to say, $H = m \cdot h$. This is illustrated in Figure 2.

In summary, the vector of active state variables is calculated at each of the time instants $t_0 + h, t_0 + 2h, t_0 + 3h, \dots, t_1 = t_0 + mh$, defined in the fine grid $t = t_0 + \lambda h, \lambda = 1, 2, \dots, m$, whereas the vector of latent state variables is evaluated only in the coarse time instant $t_1 = t_0 + H$. A general definition of a multirate Runge-Kutta time-step integrator is presented in the following.

Definition 3.1 (multirate Runge-Kutta (MRK) method). Consider two Runge-Kutta methods of s and \bar{s} stages, that can be, but do not necessarily have to be, the same, expressed by their Butcher tableaus $(\mathbf{b}, \mathbf{A}, \mathbf{c})$ and $(\bar{\mathbf{b}}, \bar{\mathbf{A}}, \bar{\mathbf{c}})$:

$$\begin{array}{c|ccc} c_1 & a_{11} & a_{12} & \cdots & a_{1s} \\ c_2 & a_{21} & a_{22} & \cdots & a_{2s} \\ \vdots & \vdots & \vdots & & \vdots \\ c_s & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & b_1 & b_2 & \cdots & b_s \end{array} \quad \begin{array}{c|ccc} \bar{c}_1 & \bar{a}_{11} & \bar{a}_{12} & \cdots & \bar{a}_{1\bar{s}} \\ \bar{c}_2 & \bar{a}_{21} & \bar{a}_{22} & \cdots & \bar{a}_{2\bar{s}} \\ \vdots & \vdots & \vdots & & \vdots \\ \bar{c}_{\bar{s}} & \bar{a}_{\bar{s}1} & \bar{a}_{\bar{s}2} & \cdots & \bar{a}_{\bar{s}\bar{s}} \\ \hline & \bar{b}_1 & \bar{b}_2 & \cdots & \bar{b}_{\bar{s}} \end{array} \quad (3.3)$$

The resulting multirate Runge-Kutta method for obtaining the numerical solution of the partitioned system of (3.1), using a microstep h for the active components and a macrostep H for the latent components, is defined as follows [11, 12]:

(i) the active (fast-varying) components \mathbf{y}_A are given by

$$\mathbf{y}_{A,\lambda+1} = \mathbf{y}_{A,\lambda} + h \sum_{i=1}^s b_i \mathbf{k}_{A,i}^\lambda \approx \mathbf{y}_A(t_0 + (\lambda + 1)h), \quad \lambda = 0, 1, \dots, m-1, \quad (3.4)$$

where

$$\mathbf{k}_{A,i}^\lambda = \mathbf{f}_A \left(t_0 + (\lambda + c_i)h, \mathbf{y}_{A,\lambda} + h \sum_{j=1}^s a_{ij} \mathbf{k}_{A,j}^\lambda, \tilde{\mathbf{Y}}_{L,i}^\lambda \right), \quad i = 1, 2, \dots, s, \quad (3.5)$$

with

$$\tilde{\mathbf{Y}}_{L,i}^\lambda \approx \mathbf{y}_L(t_0 + (\lambda + c_i)h), \quad c_i = \sum_{j=1}^s a_{ij}, \quad (3.6)$$

(ii) the latent (slowly varying) components \mathbf{y}_L are given by

$$\mathbf{y}_{L,1} = \mathbf{y}_{L,0} + H \sum_{i=1}^{\bar{s}} \bar{b}_i \mathbf{k}_{L,i} \approx \mathbf{y}_L(t_0 + H), \quad (3.7)$$

where

$$\mathbf{k}_{L,i} = \mathbf{f}_L \left(t_0 + \bar{c}_i H, \tilde{\mathbf{Y}}_{A,i}, \mathbf{y}_{L,0} + H \sum_{j=1}^{\bar{s}} \bar{a}_{ij} \mathbf{k}_{L,j} \right), \quad i = 1, 2, \dots, \bar{s}, \quad (3.8)$$

with

$$\tilde{\mathbf{Y}}_{A,i} \approx \mathbf{y}_A(t_0 + \bar{c}_i H), \quad \bar{c}_i = \sum_{j=1}^{\bar{s}} \bar{a}_{ij}. \quad (3.9)$$

As can be seen, the coupling between the active and latent subsystems is performed by the intermediate stage values $\tilde{\mathbf{Y}}_{L,i}^\lambda$ and $\tilde{\mathbf{Y}}_{A,i}$. There are several strategies for computing these values, as the ones suggested in [9, 10], which are based on interpolation and/or extrapolation techniques, or the ones more recently proposed in [11, 12] or [13], which are based on coupling coefficients.

It must be noted that, in general, the partition into fast and slow subsystems, as well as the number of microsteps within a macrostep, may vary throughout the integration process. Technical details of MRK code implementation, such as active-latent partitioning strategies, step size control tools, or even stiffness detection stratagems, are described in detail, for example, in [11] or [15].

3.2. Multitime Formulation

Signals handled by wireless communication systems can be usually described by a high-frequency RF carrier modulated by some kind of baseband information signals, as amplitude and/or phase signals. The general form of an amplitude and phase-modulated signal can be expressed as

$$x(t) = e(t) \cos(\omega_C t + \phi(t)), \quad (3.10)$$

where $e(t)$ and $\phi(t)$ are, respectively, the slowly varying amplitude and phase baseband signals (the complex envelope in a constellation diagram), modulating the $\cos(\omega_C t)$ fast-varying carrier.

Circuits driven by envelope- (amplitude and/or phase-) modulated signals, or presenting themselves state variables of this type, are common in RF and microwave applications. Since the baseband signals have a spectral content of much lower frequency than the carrier, that is, because they are typically slowly varying signals, while the carrier is a fast-varying entity, simulating nonlinear circuits—in a computationally efficient way—containing this kind of signals is often a very challenging task. Because the aperiodic nature of the envelope modulation signals obviates the use of any steady-state technique, one might think that conventional time-step integration (SPICE-like transient simulation) would be the natural method for simulating such circuits, as long as the time scale of the signals' slowly varying components is comparable to the larger time constants involved. However, the large time constants of the bias networks determine long transient regimes and, as a result, the obligation of simulating a large number of carrier periods. In addition, computing the RF carrier oscillations long enough to obtain information about its envelope properties is, itself, a colossal task. Time-step integration is thus inadequate for simulating this kind of problems.

In this section, we will introduce a powerful strategy for analyzing nonlinear circuits handling multirate signals, that is, signals containing two, or more, entities running on widely separated time scales. This strategy is suitable to deal with the above-described amplitude and/or phase-modulated signals, as also with any other kind of multirate signals. It uses multiple time variables to describe multirate behavior, and it is based on the fact that multirate signals can be represented much more efficiently if they are defined as functions of two or more time variables, that is, if they are defined as *multivariate functions* [17, 18]. As we will see, with this multivariate formulation, circuits will be no longer described by ordinary differential systems in the one-dimensional time t , but, instead, by partial differential systems.

The advantages of using multivariate representations are easily illustrated by considering a bidimensional problem. Thus, let us consider, for example, an amplitude modulated RF carrier of the form

$$x(t) = e(t) \sin(2\pi f_C t), \quad (3.11)$$

where $e(t)$ is an envelope, slowly varying in time, while $\sin(2\pi f_C t)$ is a fast-varying RF carrier. As explained, simulating a circuit with this kind of stimulus, and using conventional time-step integration schemes, tends to be highly inefficient because it requires time steps closely spaced in time (for representing each fast RF cycle accurately) and during a very long

time window determined by the envelope. So, let us now consider a bivariate representation for $x(t)$, defined as

$$\hat{x}(t_1, t_2) = e(t_1) \sin(2\pi f_C t_2), \quad (3.12)$$

where t_1 is the slow envelope time scale, and t_2 is the fast carrier time scale. In this particular case, $\hat{x}(t_1, t_2)$ is a periodic function with respect to t_2 but not to t_1 , that is,

$$\hat{x}(t_1, t_2) = \hat{x}(t_1, t_2 + T_2), \quad T_2 = \frac{1}{f_C}. \quad (3.13)$$

Figures 3 and 4 depict the univariate and bivariate forms defined in (3.11) and (3.12), respectively, for a $[0, 50 \mu\text{s}]$ time interval and a rectangular region $[0, 50 \mu\text{s}] \times [0, T_2]$. An envelope $e(t) = 5 \text{ sinc}(40\,000(t - 35 \times 10^{-6})) + 2.5$ and a carrier frequency of $f_C = 1 \text{ MHz}$, were considered in this basic illustrative example.

By comparing Figures 3 and 4, it can be seen that $\hat{x}(t_1, t_2)$ does not have as many undulations as $x(t)$, allowing thus a more compact representation with fewer samples. Furthermore, due to the periodicity of $\hat{x}(t_1, t_2)$ in t_2 , we know that its plot repeats over the rest of this time axis. Thus, the bivariate form plotted in Figure 4 contains all the information necessary to recover the original univariate form depicted in Figure 3. In order to get a realistic idea of the savings that can be achieved by using the bivariate formulation, let us suppose that 20 points were sufficient to represent the slowly varying envelope signal $e(t)$ accurately in the $[0, 50 \mu\text{s}]$ interval, and that 20 points were also used per each carrier cycle. In such case, the total number of samples required to represent the bivariate form $\hat{x}(t_1, t_2)$ will be $20 \times 20 = 400$. On the other hand, since there are 50 carrier cycles in the $[0, 50 \mu\text{s}]$ interval, the number of samples required to represent the univariate form $x(t)$ will be $50 \times 20 = 1000$. Now, if we consider a realistic RF scenario, with a much higher value for the carrier frequency (e.g., $f_C = 1 \text{ GHz}$), then the number of points necessary to represent $x(t)$ will considerably increase to $50000 \times 20 = 10^6$, while the number of points necessary to represent $\hat{x}(t_1, t_2)$ will remain exactly the same ($20 \times 20 = 400$).

Let us now consider a general nonlinear RF circuit described by the differential algebraic equations' (DAE) system of (2.1), and let us suppose that this circuit is driven by an envelope-modulated signal of the form (3.11). Taking the above considerations into account, we will adopt the following procedure: for the slowly varying parts (envelope time scale) of the expressions of $\mathbf{x}(t)$ and $\mathbf{y}(t)$, t is replaced by t_1 ; for the fast-varying parts (RF carrier time scale), t is replaced by t_2 . This results in bivariate representations for the excitation $\hat{\mathbf{x}}(t_1, t_2)$, and the solution $\hat{\mathbf{y}}(t_1, t_2)$ and the application of this bivariate strategy to the system of (2.1) converts it into the following *multirate partial differential algebraic equations'* (MPDAE) system [17, 18]:

$$\mathbf{p}(\hat{\mathbf{y}}(t_1, t_2)) + \frac{\partial \mathbf{q}(\hat{\mathbf{y}}(t_1, t_2))}{\partial t_1} + \frac{\partial \mathbf{q}(\hat{\mathbf{y}}(t_1, t_2))}{\partial t_2} = \hat{\mathbf{x}}(t_1, t_2). \quad (3.14)$$

The mathematical relation between (2.1) and (3.14) establishes that if $\hat{\mathbf{x}}(t_1, t_2)$ and $\hat{\mathbf{y}}(t_1, t_2)$ satisfy (3.14), then the univariate forms $\mathbf{x}(t) = \hat{\mathbf{x}}(t, t)$ and $\mathbf{y}(t) = \hat{\mathbf{y}}(t, t)$ satisfy (2.1) [17]. Therefore, univariate solutions of (2.1) are available on diagonal lines $t_1 = t$, $t_2 = t$, along the bivariate solutions of (3.14), that is, $\mathbf{y}(t)$ may be retrieved from its bivariate form $\hat{\mathbf{y}}(t_1, t_2)$,

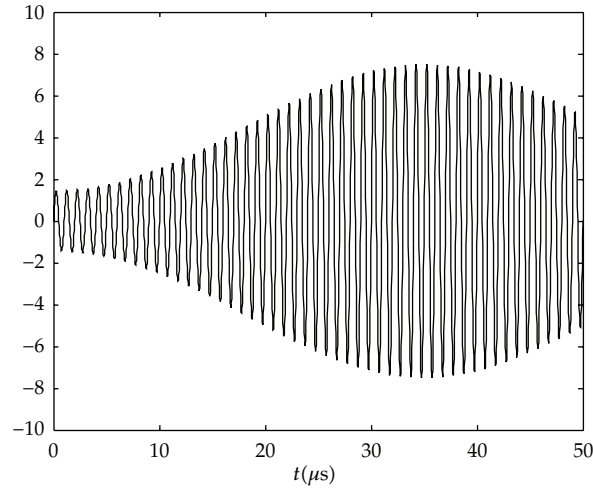


Figure 3: Amplitude-modulated RF carrier in the univariate time.

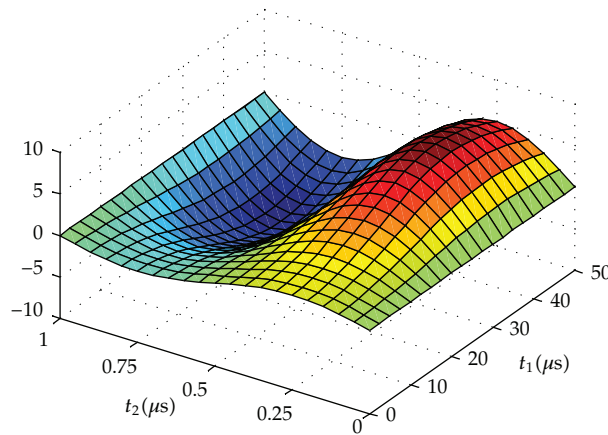


Figure 4: Bivariate representation of the amplitude-modulated RF carrier.

by simply setting $t_1 = t_2 = t$. Consequently, if we want to obtain the univariate solution in the generic $[0, t_{\text{Final}}]$ interval, due to the periodicity of the problem in the t_2 dimension, we will have

$$\mathbf{y}(t) = \hat{\mathbf{y}}(t, t \bmod T_2), \tag{3.15}$$

on the rectangular domain $[0, t_{\text{Final}}] \times [0, T_2]$, where $t \bmod T_2$ represents the remainder of division of t by T_2 . Equation (3.15) defines the sawtooth path illustrated in Figure 5.

The multivariate formulation can be employed to solve many types of multirate problems. It can be adopted to deal with either weakly or strongly nonlinear regimes, as well as with a large class of multirate signals. The main advantage of this MPDAE approach is that it can result in significant improvements in simulation speed when compared to DAE-based alternatives [16–22].

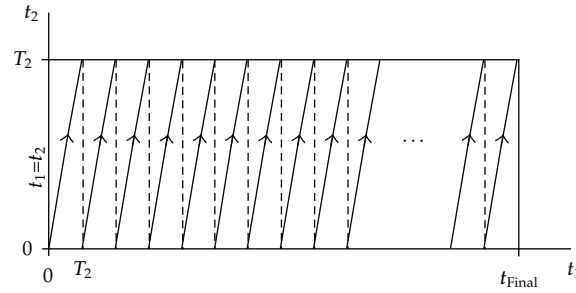


Figure 5: Sawtooth path in the $t_1 t_2$ plane, for recovering the univariate solution from its bivariate form.

3.3. Envelope following Methods Using RK Schemes

In order to compute the bivariate solutions, some initial/boundary conditions must be added to (3.14). These are determined by the existence, or not, of periodicity in the signal components. As stated above, a typical case of significant practical interest is the envelope-modulated regime, which leads to an initial-boundary value problem. Indeed, envelope-modulated responses to excitations of the form of (3.10) correspond to a combination of initial and periodic boundary conditions in (3.14). This means that the bivariate forms of these solutions can be obtained by numerically solving the following initial-boundary value problem [17]:

$$\begin{aligned} \mathbf{p}(\hat{\mathbf{y}}(t_1, t_2)) + \frac{\partial \mathbf{q}(\hat{\mathbf{y}}(t_1, t_2))}{\partial t_1} + \frac{\partial \mathbf{q}(\hat{\mathbf{y}}(t_1, t_2))}{\partial t_2} &= \hat{\mathbf{x}}(t_1, t_2), \\ \hat{\mathbf{y}}(0, t_2) &= \mathbf{g}(t_2), \\ \hat{\mathbf{y}}(t_1, 0) &= \hat{\mathbf{y}}(t_1, T_2), \end{aligned} \quad (3.16)$$

on the rectangle $[0, t_{\text{Final}}] \times [0, T_2]$. $\mathbf{g}(\cdot)$ is a given initial-condition function defined on $[0, T_2]$, satisfying $\mathbf{g}(0) = \mathbf{g}(T_2) = \mathbf{y}(0)$ [17], and the periodic boundary condition $\hat{\mathbf{y}}(t_1, 0) = \hat{\mathbf{y}}(t_1, T_2)$ is due to the periodicity of the problem in the t_2 fast carrier time scale.

The *envelope transient over shooting* [16, 17, 19, 23, 24] is an efficient time-domain method that can be used to obtain the numerical solution of circuits described by initial-boundary value problems of the form (3.16). This method is a particular implementation of a general technique that is often referred to as *envelope following* [25] and consists in replacing the derivatives of the t_1 slow aperiodic time scale by finite-difference approximations (e.g., the Backward Euler rule), to then obtain a set of successive boundary value problems with periodic boundary conditions

$$\mathbf{p}(\hat{\mathbf{y}}_i(t_2)) + \frac{\mathbf{q}(\hat{\mathbf{y}}_i(t_2)) - \mathbf{q}(\hat{\mathbf{y}}_{i-1}(t_2))}{h_{1,i}} + \frac{d\mathbf{q}(\hat{\mathbf{y}}_i(t_2))}{dt_2} = \hat{\mathbf{x}}(t_{1,i}, t_2), \quad \hat{\mathbf{y}}_i(0) = \hat{\mathbf{y}}_i(T_2), \quad (3.17)$$

where $\hat{\mathbf{y}}_i(t_2) = \hat{\mathbf{y}}(t_{1,i}, t_2)$, $h_{1,i} = t_{1,i} - t_{1,i-1}$, and T_2 is the period in the periodic fast time scale t_2 . This means that once $\hat{\mathbf{y}}_{i-1}(t_2)$ is known, the solution on the next slow time instant $\hat{\mathbf{y}}_i(t_2)$ is obtained by solving (3.17). Thus, for obtaining the whole solution $\hat{\mathbf{y}}(t_1, t_2)$ in the entire domain $[0, t_{\text{Final}}] \times [0, T_2]$, a total of K_1 boundary value problems have to be solved, with

K_1 being the number of steps in t_1 . With the envelope transient over shooting technique, each of the periodic boundary value problems of (3.17) is solved using the shooting method described in Section 2.6, where RK methods can be used to perform the successive time-step integrations in the consecutive shooting iterations.

3.4. Envelope following Methods Using MRK Schemes

Recently, a very powerful computer-aided design tool especially conceived for the efficient time-domain simulation of highly heterogeneous nonlinear RF circuits has been proposed [16, 19]. This technique is based on an ingenious modification of the above-described envelope transient over shooting technique. It splits the circuits into two distinct parts, according to the time rates of change of their state variables, and, instead of using classical time-step integrators, it uses modern multirate Runge-Kutta (MRK) schemes to perform time-step integration with different step sizes in each of the consecutive shooting iterations needed to solve (3.17). Indeed, the system of (3.17) is partitioned into the following active and latent subsystems:

$$\begin{aligned} \mathbf{p}_A(\hat{\mathbf{y}}_{A,i}(t_2), \hat{\mathbf{y}}_{L,i}(t_2)) + \frac{\mathbf{q}_A(\hat{\mathbf{y}}_{A,i}(t_2), \hat{\mathbf{y}}_{L,i}(t_2)) - \mathbf{q}_A(\hat{\mathbf{y}}_{A,i-1}(t_2), \hat{\mathbf{y}}_{L,i-1}(t_2))}{h_{1,i}} \\ + \frac{d\mathbf{q}_A(\hat{\mathbf{y}}_{A,i}(t_2), \hat{\mathbf{y}}_{L,i}(t_2))}{dt_2} = \hat{\mathbf{x}}(t_{1,i}, t_2), \\ \mathbf{p}_L(\hat{\mathbf{y}}_{A,i}(t_2), \hat{\mathbf{y}}_{L,i}(t_2)) + \frac{\mathbf{q}_L(\hat{\mathbf{y}}_{A,i}(t_2), \hat{\mathbf{y}}_{L,i}(t_2)) - \mathbf{q}_L(\hat{\mathbf{y}}_{A,i-1}(t_2), \hat{\mathbf{y}}_{L,i-1}(t_2))}{h_{1,i}} \\ + \frac{d\mathbf{q}_L(\hat{\mathbf{y}}_{A,i}(t_2), \hat{\mathbf{y}}_{L,i}(t_2))}{dt_2} = \hat{\mathbf{x}}(t_{1,i}, t_2), \end{aligned} \quad (3.18)$$

where

$$\hat{\mathbf{y}}_i(t_2) = \begin{bmatrix} \hat{\mathbf{y}}_{A,i}(t_2) \\ \hat{\mathbf{y}}_{L,i}(t_2) \end{bmatrix}, \quad \hat{\mathbf{y}}_{A,i}(t_2) \in \mathbb{R}^{n_A}, \quad \hat{\mathbf{y}}_{L,i}(t_2) \in \mathbb{R}^{n_L}, \quad n_A + n_L = n. \quad (3.19)$$

$\hat{\mathbf{y}}_{A,i}(t_2)$ is the active (fast-varying) state variable components' vector at the slow time instant $t_{1,i}$, and $\hat{\mathbf{y}}_{L,i}(t_2)$ is the latent (slowly varying) state variable components' vector at the same slow time instant. The active components will be integrated in t_2 with a small step size h_2 (microstep), while the latent components will be integrated with a much larger step size $H_2 = m \cdot h_2$ (macrostep).

In summary, this powerful technique (envelope transient over shooting with MRK) can be seen as a multirate scheme (different time-step integration sizes to state variables that present significantly disparate rates of change) coupled with a multirate excitation regime (multiple time-scale representations). Consequently, it is able to benefit from the circuits' heterogeneities, and also from the stimuli time-rate disparities, to significantly reduce the computational effort required for simulating the circuits.

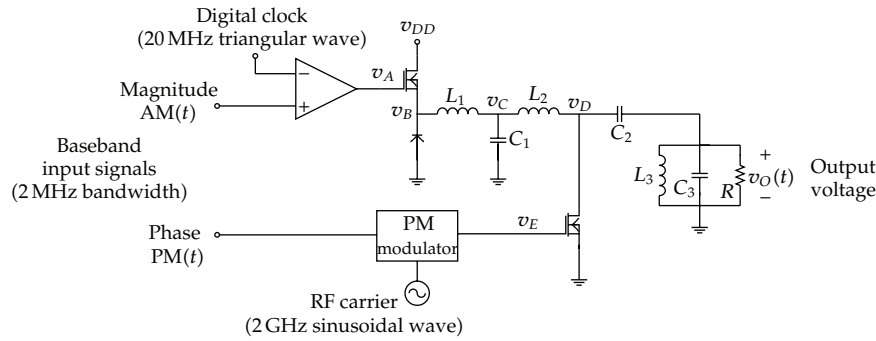


Figure 6: Simplified power amplifier schematic used in wireless polar transmitters [16].

3.5. Performance of the Methods

The performance and the efficiency of all the numerical algorithms based on MRK schemes described in this paper were already successfully attested through their application to several illustrative examples of practical relevance. Indeed, electronic circuits with distinct configurations and levels of complexity were especially selected to illustrate the significant gains in computation speed that can be achieved when simulating the circuits with these methods.

Until 2006, multirate Runge-Kutta methods were only used to obtain the numerical solution of univariate initial value problems (transient electronic circuit simulation in the one-dimensional time). For example, in [11, 13], or [15], multirate Runge-Kutta methods were used to obtain, in an efficient way, the numerical solution of two digital electronic circuits: a digital inverter chain and a pulse generator. Significant reductions in the computational effort were obtained with the methods when compared to standard RK schemes.

Since 2007, multirate Runge-Kutta methods were also included in numerical algorithms conceived for operating in multivariate frameworks, that is, for solving multidimensional problems described by partial differential systems. For example, in [16, 19] multirate Runge-Kutta methods were used within a bidimensional envelope following technique to efficiently compute the numerical solution of two RF circuits operating in two distinct time scales: a resistive field effect transistor (FET) mixer and a polar power amplifier used in wireless transmitters. Later, in [20, 21] multirate Runge-Kutta methods were also used within a three-dimensional envelope following technique to simulate a polar power amplifier operating in three separated time scales. The efficiency gains provided by the use of different step sizes (MRK schemes) within the classical univariate time-step integration, but mostly within the bivariate formulation, were also evidenced in [22].

In order to provide a realistic idea of the efficiency (in terms of computational speed) of the methods reviewed in this paper, we decided to include in this section a brief comparison between standard RK and multirate MRK schemes, within the univariate and bivariate formulations. For that, we considered the RF polar transmitter PA described in [16], and depicted in Figure 6, as our illustrative application example. The most relevant components' values of this circuit are $v_{DD} = 25$ V, $L_1 = 2$ μ H, $C_1 = 3.2$ nF, $L_2 = 40$ nH, $C_2 = 16$ pF, $L_3 = 0.4$ nH, $C_3 = 16$ pF, and $R = 50$ Ω . The MOSFETs (metal-oxide semiconductor field effect transistors) are represented by the following simplified nonlinear device model:

$$i_{DS}(v_{GS}, v_{DS}) = \beta \frac{1}{2} [v + \ln(e^v + e^{-v})] \tanh(\alpha v_{DS}), \quad v = K_T(v_{GS} - V_T), \quad (3.20)$$

Table 1: Computation times simulation of the RF polar transmitter PA depicted in Figure 6.

Simulation time interval	Univariate time-step integration		Bivariate envelope following methods	
	with RK (SPICE)	with MRK	ETS with RK	ETS with MRK
[0, 200 ns]	39 m 30 s	10 m 08 s	5 m 44 s	32 s
[0, 800 ns]	>2h	41 m 55 s	26 m 01 s	2 m 38 s

with $\alpha = 1 \text{ V}^{-1}$, $\beta=0.25 \text{ A/V}$, $K_T = 2$, and $V_T = 3 \text{ V}$. Similarly, the diode current-voltage characteristic is given by

$$i_D(v_D) = I_S \left(e^{v_D/\eta V_{\text{Temp}}} - 1 \right), \quad (3.21)$$

where $I_S = 1 \mu\text{A}$, $\eta = 2$, and $V_{\text{Temp}} = 0.026 \text{ V}$.

In this circuit, we have a combination of periodic (RF carrier and digital clock) and aperiodic [AM(t) and PM(t)] forcing functions, of very distinct time scales, with a mixture of heterogeneous state variables with widely disparate rates of variation. For instance, while voltages and currents in the output band-pass filter are all very fast (active state variables), voltages and currents in the AM branch are much slower (latent state variables). The circuit was simulated in MATLAB, and numerical computation times for two different simulation time intervals are presented in Table 1. These results were obtained using (i) a time-step integrator based on a RK scheme (SPICE-like simulation), (ii) a time-step integrator based on an MRK scheme, (iii) an envelope transient over shooting (ETS) algorithm using an RK scheme and (iv) an ETS algorithm using an MRK scheme. Standard RK and modern MRK schemes of order 3 based on the Bogacki-Shampine formulas [26] were used to perform the necessary numerical time-step integration.

Table 1 evidences the efficiency gains that can be achieved with the use of MRK schemes, in comparison to classical RK ones, within the univariate and the bivariate formulations. It also shows the advantage of operating with multiple time variables, instead of working in the natural one-dimensional time. It must be noted that these gains in computational speed were obtained without compromising the accuracy of the results. Indeed, the maximum discrepancy between the solutions obtained with any of the methods under analysis was of the order 10^{-8} for all the circuit's state variables.

4. Conclusion

Despite the scientific field of numerical simulation of electronic problems has appeared in the 1970s with the advent of SPICE, this subject is still today a hot topic. Indeed, serious difficulties arise when we have heterogeneous circuits composed of different types of blocks and/or operating in multiple time scales. In this paper, we have briefly reviewed some ways to circumvent these difficulties, operating strictly in the time domain, and which consist in using multirate Runge-Kutta schemes encapsulated in (i) one-dimensional engines and (ii) multidimensional algorithms (by decoupling the components of the signals into different time dimensions). A key aspect that contributes to the success of these numerical methods is the partitioning of the circuits into subcircuits according to the time rates of change of their state variables.

Acknowledgment

The authors would like to acknowledge the financial support provided by the Portuguese Science and Technology Foundation (FCT) under the Project SOPAS-PTDC/EEA-TEL/114530/2009.

References

- [1] S. A. Maas, *Nonlinear Microwave and RF Circuits*, Artech House, Norwood, Mass, USA, 2nd edition, 2003.
- [2] P. J. Rodrigues, *Computer-Aided Analysis of Nonlinear Microwave Circuits*, Artech House, Norwood, Mass, USA, 1998.
- [3] E. Hairer, S. P. Nørsett, and G. Wanner, *Solving Ordinary Differential Equations. I*, vol. 8 of *Springer Series in Computational Mathematics*, Springer, Berlin, Germany, 1987.
- [4] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations. II*, vol. 14 of *Springer Series in Computational Mathematics*, Springer, Berlin, Germany, 1991.
- [5] J. D. Lambert, *Numerical Methods for Ordinary Differential Systems*, John Wiley & Sons Ltd., Chichester, UK, 1991.
- [6] L. Nagel, *SPICE2: A Computer Program to Simulate Semiconductor Circuits*, Electronics Research Laboratory, University of California, Berkeley, Calif, USA, 1975.
- [7] K. Kundert, J. White, and A. Sangiovanni-Vincentelli, *Steady-State Methods for Simulating Analog and Microwave Circuits*, Kluwer Academic, Norwell, Mass, USA, 1990.
- [8] J. C. Pedro and N. B. Carvalho, *Intermodulation Distortion in Microwave and Wireless Circuits*, Artech House, Norwood, Mass, USA, 2003.
- [9] M. Günther and P. Rentrop, "Multirate ROW methods and latency of electric circuits," *Applied Numerical Mathematics*, vol. 13, no. 1-3, pp. 83–102, 1993.
- [10] M. Günther and P. Rentrop, "Partitioning and multirate strategies in latent electric circuits," *International Series of Numerical Mathematics*, vol. 117, pp. 33–60, 1994.
- [11] A. Kværnø and P. Rentrop, *Low Order Multirate Runge-Kutta Methods in Electric Circuit Simulation*, IWRMM Universität Karlsruhe, 1999.
- [12] A. Kværnø, "Stability of multirate Runge-Kutta schemes," *International Journal of Differential Equations and Applications*, vol. 1A, no. 1, pp. 97–105, 2000.
- [13] M. Günther, A. Kværnø, and P. Rentrop, "Multirate partitioned Runge-Kutta methods," *BIT*, vol. 41, no. 3, pp. 504–514, 2001.
- [14] A. Bartel, M. Günther, and A. Kværnø, "Multirate methods in electrical circuit simulation," in *Progress in Industrial Mathematics at ECMI 2000*, pp. 258–265, Springer, 2002.
- [15] J. Oliveira and A. Araújo, "Envelope transient simulation of nonlinear electronic circuits using multirate Runge-Kutta algorithms," *WSEAS Transactions on Electronics*, vol. 3, no. 2, pp. 77–84, 2006.
- [16] J. F. Oliveira and J. C. Pedro, "An efficient time-domain simulation method for multirate RF nonlinear circuits," *IEEE Transactions on Microwave Theory and Techniques*, vol. 55, no. 11, pp. 2384–2392, 2007.
- [17] J. Roychowdhury, "Analyzing circuits with widely separated time scales using numerical PDE methods," *IEEE Transactions on Circuits and Systems*, vol. 48, no. 5, pp. 578–594, 2001.
- [18] T. Mei, J. Roychowdhury, T. S. Coffey, S. A. Hutchinson, and D. M. Day, "Robust, stable time-domain methods for solving MPDEs of fast/slow systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 2, pp. 226–238, 2005.
- [19] J. F. Oliveira and J. C. Pedro, "A new time-domain simulation method for highly heterogeneous RF circuits," *Proceedings of the 37th European Microwave Conference (EUMC '07)*, pp. 1161–1164, 2007.
- [20] J. F. Oliveira and J. C. Pedro, "An innovative time-domain simulation technique for strongly nonlinear heterogeneous RF circuits operating in diverse time-scales," in *Proceedings of the 38th European Microwave Conference (EuMC '08)*, pp. 1557–1560, Amsterdam, The Netherlands, October 2008.
- [21] J. F. Oliveira and J. C. Pedro, "A multiple-line double multirate shooting technique for the simulation of heterogeneous RF circuits," *IEEE Transactions on Microwave Theory and Techniques*, vol. 57, no. 2, pp. 421–429, 2009.
- [22] J. F. Oliveira and J. C. Pedro, "Advanced time-domain techniques for strongly nonlinear RF circuit simulation: recent developments and remaining challenges," in *Proceedings of the International Conference on Computer as a Tool (EUROCON '11)*, Lisbon, Portugal, April 2011.

- [23] C. E. Christoffersen and J. Alexander, "An adaptive time step control algorithm for nonlinear time domain envelope transient," in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, vol. 2, pp. 883–886, Ontario, Canada, May 2004.
- [24] A. Brambilla and P. Maffezzoni, "Envelope-following method to compute steady-state solutions of electrical circuits," *IEEE Transactions on Circuits and Systems*, vol. 50, no. 3, pp. 407–417, 2003.
- [25] E. Ngoya and R. Larcheveque, "Envelop transient analysis: a new method for the transient and steady state analysis of microwave communication circuits and systems," in *IEEE MTT-S International Microwave Symposium Digest*, vol. 3, pp. 1365–1368, San Francisco, Calif, USA, June 1996.
- [26] P. Bogacki and L. F. Shampine, "A 3(2) pair of Runge-Kutta formulas," *Applied Mathematics Letters*, vol. 2, no. 4, pp. 321–325, 1989.