Proceedings of the 4th WSEAS/IASME Int. Conf. on System Science and Simulation in Engineering, Tenerife, Spain, December 16-18, 2005 (pp90-93)

Speed Versus Stability of Multi-Rate Runge-Kutta Methods in Transient Electronic Circuit Simulation

JORGE OLIVEIRA, ADÉRITO ARAÚJO*
Department of Electrical Engineering
Technology and Management School - Polytechnic Institute of Leiria
Morro do Lena, Alto Vieiro, Apartado 4163, 2411-901 Leiria

*Department of Mathematics
University of Coimbra
Apartado 3008, 3001-454 Coimbra
PORTUGAL

Abstract: - Dynamical behavior of some electronic circuits involves signals with widely separated rates of variation. Numerical solution of ordinary differential systems describing such circuits may be achieved in an efficient way using multi-rate methods, which use different step sizes for each subsystem. In this paper we will test the performance of two multi-rate Runge-Kutta algorithms in terms of numerical stability and computational speed. Being similar to the previous study done in [6], the results for linear stability analysis here presented are much more coherent with the characteristics of the methods.

Key-Words: - Electronic circuit simulation, transient analysis, multi-rate Runge-Kutta methods, speed, stability.

1 Introduction

Transient analysis of an electronic circuit is usually expressed by an initial value problem of the form

$$y'(t) = f(y), \quad y(t_0) = y_0, \quad y \in \mathbb{R}^n, \quad t > t_0 \quad (1)$$

where $y(t)$ is the solution. This system of ordinary differential equations (ODEs) can be numerically solved by time-step integration, which is a classical technique that is commercially used by all SPICE like simulators. However, when integrating systems whose components evolve at different time scales one would like to use numerical methods that do not expend unnecessary work on slowly changing components. In such cases traditional time-step integrators become inefficient and numerical schemes with different time-step sizes are required. For example, in highly integrated electronic circuits normally only a small part of the elements is active, whereas the major part is latent. This latency can be exploited by multi-rate methods, which integrate components of the slow subsystem with a larger step length than the fast subsystem.

Let us consider (1). If we split this system into active and latent subsystems we obtain

$$\begin{aligned} y_A'(t) &= f_A(y_A, y_L), \quad y_A(t_0) = y_{A,0} \\ y_L'(t) &= f_L(y_A, y_L), \quad y_L(t_0) = y_{L,0} \end{aligned} \quad (2)$$

with

$$y = \begin{bmatrix} y_A \\ y_L \end{bmatrix}, \quad y_A \in \mathbb{R}^{n_A}, \quad y_L \in \mathbb{R}^{n_L}, \quad n_A + n_L = n,$$

where y_A is the active components vector and y_L the latent components vector. The efficiency of the methods presented in this paper is verified only if there is a small number of fast changing components, i.e., if y_A is a small subset of y. It is so because while the active components y_A are integrated with a small step size h (microstep), the latent components y_L are integrated with a large step size H (macro-step). The number of microsteps within a macrostep is m, thus

$$h = (1/m) \cdot H, \quad m \in \mathbb{N}.$$

Throughout the integration process the partition into fast and slow components may vary with time, as well as m.

2 Multi-Rate Runge-Kutta Methods

Let us consider two Runge-Kutta (RK) methods [5], [8], that can but do not have to be the same, expressed by their Butcher tableaus (b, A, c) and $(\underline{b}, \underline{A}, \underline{c})$, for integrating y_A and y_L, respectively. The resulting multi-rate Runge-Kutta (MRK) method for the numerical solution of (2) is defined as follows [6]:

- the active components y_A are given by

$$y_A(t_0 + (\lambda+1)h) \approx y_{A,\lambda+1} = y_{A,\lambda} + h\sum_{i=1}^{s} b_i k_{A,i}^{\lambda},$$
$$\lambda = 0, 1, \ldots, m-1,$$

$$k_{A,i}^{\lambda} = f_A\left(y_{A,\lambda} + h\sum_{j=1}^{s} a_{ij} k_{A,j}^{\lambda}, \tilde{Y}_{L,i}^{\lambda} \right), \quad i = 1,2,\ldots s,$$

where $\tilde{Y}_{L,i}^{\lambda} \approx y_L\left(t_0 + (\lambda + c_i)h\right)$ and $c_i = \sum_{j=1}^{s} a_{ij}$,

- the latent components y_L are given by

$$y_L(t_0 + H) \approx y_{L,1} = y_{L,0} + H\sum_{i=1}^{\bar{s}} \bar{b}_i k_{L,i},$$

$$k_{L,i} = f_L\left(\tilde{Y}_{A,i}, y_{L,0} + H\sum_{j=1}^{\bar{s}} \bar{a}_{ij} k_{L,j} \right), \quad i = 1,2,\ldots,\bar{s},$$

where $\tilde{Y}_{A,i} \approx y_A\left(t_0 + \bar{c}_i H\right)$ and $\bar{c}_i = \sum_{j=1}^{\bar{s}} \bar{a}_{ij}$.

As we can see the coupling between active and latent subsystems is performed by the intermediate stage values $\tilde{Y}_{A,i}$ and $\tilde{Y}_{L,i}$. There are several strategies for computing this values, like for example the ones suggested by Günther and Rentrop in [3] and [4], but the algorithms studied and tested in this paper are the ones more recently proposed by Kværnø and Rentrop: MRKI and MRKII. MRKII is more robust than MRKI and all details of these two algorithms can be viewed in [7]. They were omitted here for brevity. We have also considered for (b, A, c) and $(\bar{b}, \bar{A}, \bar{c})$ the same method: the Bogacki-Shampine embedded Runge-Kutta method [1].

3 Stability

Numerical stability properties of various multi-rate schemes have been discussed by several authors. Unfortunately, most of these discussions (including the one presented in [6]) are not very detailed, nor very conclusive, and until now a concise theory is missing.

The absolute stability properties of an integration method are usually studied by applying the method to the scalar test equation $y' = \alpha y$, with $\alpha \in \mathbb{C}^-$. Doing so for a standard Runge-Kutta method, the solution after one step h is given by

$$y_1 = R(h\alpha) y_0,$$

where $R(h\alpha)$ is the stability function of the method. The method is stable if and only if $|R(h\alpha)| < 1$ [8]. However, multi-rate schemes require at least two components, so the linear differential system

$$\begin{bmatrix} y_A' \\ y_L' \end{bmatrix} = \mathrm{A} \begin{bmatrix} y_A \\ y_L \end{bmatrix}, \quad \mathrm{A} = \begin{bmatrix} \alpha_{11} & \alpha_{12} \\ \alpha_{21} & \alpha_{22} \end{bmatrix} \in \mathbb{R}^{2\times 2}, \quad (3)$$

may be an appropriate test problem. If the assumptions

$$\alpha_{11}, \alpha_{22} < 0, \quad \gamma = \frac{\alpha_{12}\alpha_{21}}{\alpha_{11}\alpha_{22}} < 1,$$

are satisfied, then no extra conditions are required to ensure that both eigenvalues of A have negative real parts. The parameter γ can be seen as a measure for the coupling between the equations and we also define $\kappa = \alpha_{11} / \alpha_{22}$ as a measure for the stiffness of the system.

The numerical solution of (3) performed by the algorithms MRKI or MRKII after one compound step (1 macrostep H for y_L and m microsteps h for y_A) can be expressed by

$$\begin{bmatrix} y_{A,m} \\ y_{L,1} \end{bmatrix} = \mathrm{K} \begin{bmatrix} y_{A,0} \\ y_{L,0} \end{bmatrix},$$

where K is a matrix that depends on H, h, A and the version of the MRK algorithm. It doesn't depend on the initial condition and for fixed step lengths H and h it remains constant throughout the integration process. Thus, for example, from $y_1 = \mathrm{K} y_0$ and $y_2 = \mathrm{K} y_1$ it is possible to find K. The method is stable if and only if the spectral radius $\rho(\mathrm{K})$ of K satisfies $\rho(\mathrm{K}) < 1$.

Step sizes h and H are chosen to ensure stability for the uncoupled system $(\alpha_{12} = \alpha_{21} = 0)$, i.e., to ensure that the stability functions $R_A(h\alpha_{11})$ and $R_L(H\alpha_{22})$ satisfy the conditions $|R_A(h\alpha_{11}) < 1|$ and $|R_L(H\alpha_{22}) < 1|$. In the case of the Bogacki-Shampine method that means [8] $-2.54 < h\alpha_{11} < 0$ and $-2.54 < H\alpha_{22} < 0$, that is to say, $H < 2.54$ and $m \geq \kappa$ if, with no loss of generality, we make $\alpha_{22} = -1$. The question is to know how the coupling γ between the two systems affects the stability of the MRK and in what way it depends on κ, H and m.

Experimental results obtained computationally in *MATLAB*® are shown in Fig.1, where we have plots of some stability regions for $\kappa = 1$ and $\kappa = 10$. The methods are stable below the boundaries and unstable above and as it can be seen these stability regions become smaller with increasing κ, $|\gamma|$ and m. We have also tested other values of κ larger than 10 and we obtained similar results with smaller regions (we just omitted them here for brevity). From the above we conclude that increasing the stiffness of the system, the coupling between the parts, or the number of microsteps, we force the methods to use a smaller macrostep H. As we can see in Fig.1, MRKII is more robust than MRKI because it has larger stability regions. According to the performance of both algorithms, this result is more coherent than the one presented by Kværnø in [6].

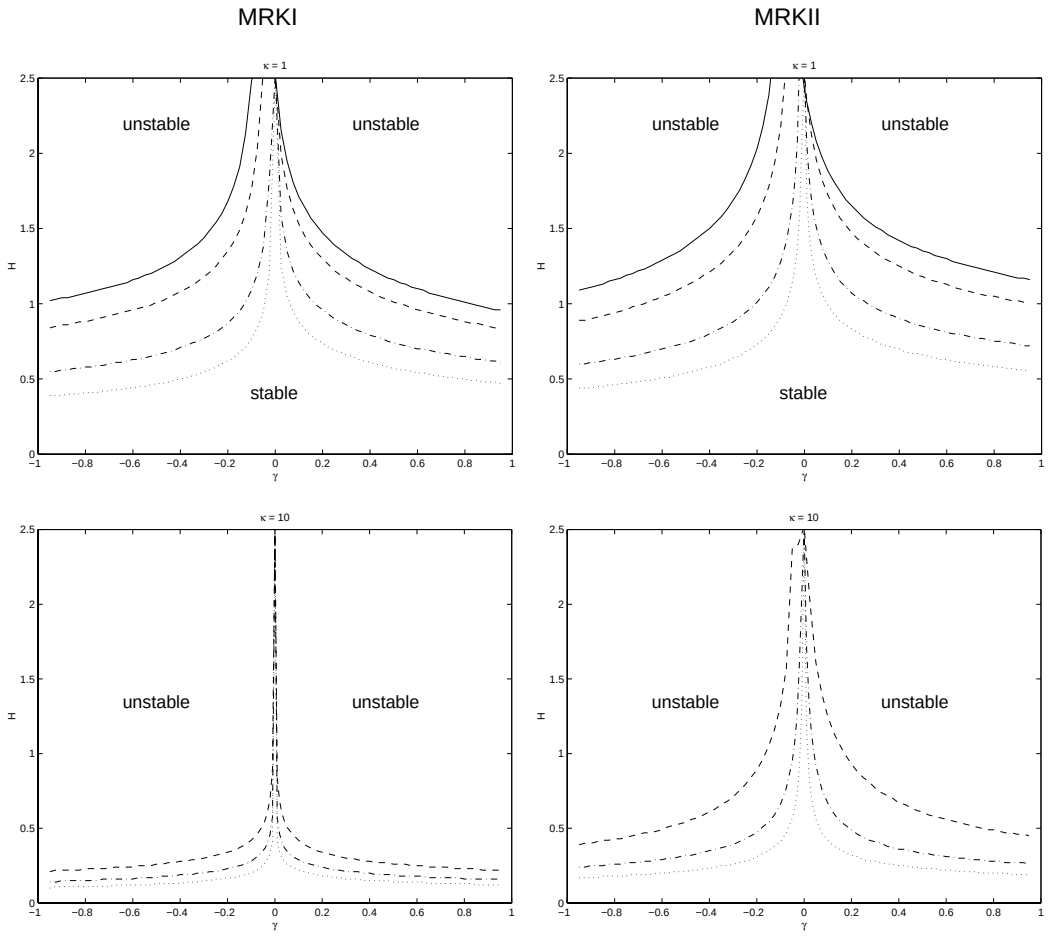

Fig.1. Stability regions: — *m*=8, --- *m*=12, -·- *m*=36, ··· *m*=100

4 Sample Application

4.1 Electronic Pulse Generator

In order to test the performance and the efficiency of our multi-rate algorithms an electronic pulse generator with MOSFETs was simulated with MRKI and MRKII. The schematic of this circuit is shown in Fig.2.

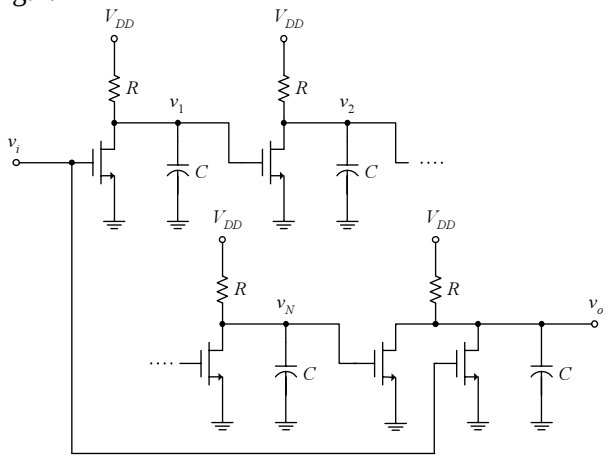

Fig.5. Pulse generator

This kind of circuit is commonly found in digital systems and it generates a positive pulse on its output when detects a transition from state 1 (high) to state 0 (low) on its input. It is constituted by an odd number N of logical inverters connected in chain, followed by a NOR gate. The output of each inverter is the input of the next inverter and the charging and discharging of the capacitors C produces delays in the run time of the signal. Thus, v_N is the logical negation of a phase shifted version of the input v_i, and the output v_O is the logical NOR between v_i and v_N.

In all our tests we have considered V_{DD}=5V, R=4.7kΩ and C=0.2pF. The mathematical model we have adopted for the MOSFETs was the one suggested by Kværnø and Rentrop in [7].

4.2 Numerical Simulation Results

The circuit with $N = 51$ inverters was simulated in *MATLAB*® from $t = 0$ to $t = 40$ ns, for an input v_i transition at $t = 1$ ns. The numerical solution v_O is shown in Fig.3 and the overall results of this simulation are presented in Table 1. In both algori-

thms, MRKI and MRKII, we have included step size control, stiffness detection and partitioning strategies, whose technical details can be seen in [9].

	time[1]	number of steps			error in v_O	
	(sec)	rejected	macro	micro	$\|\cdot\|_\infty$	$\|\cdot\|_{L^2}$
MRKI	1.37	58	142	406	0.0048	0.0038
MRKII	2.15	60	135	616	0.0034	0.0029

Table 1. Numerical simulation results

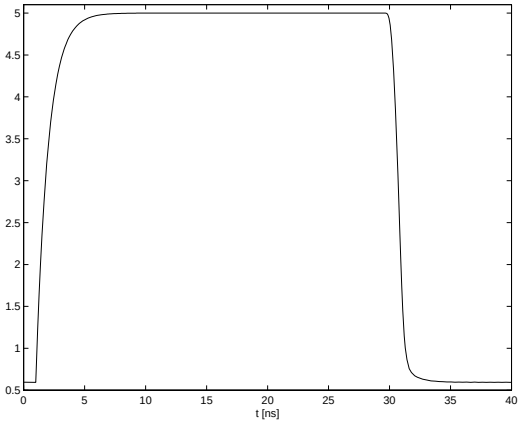

Fig.3. Numerical solution v_O

5 Conclusions

After several simulation tests with different quantities of logical inverters and different integration intervals, we can say that in general both MRK methods show a good performance when solving our sample circuit. However, the MRKII algorithm leads to an increase of the computational work, once the total time for obtaining the numerical solution was in all cases bigger than in MRKI. In Section 3 we saw that the MRKII was a more stable method, nevertheless this stability gain implies a consequent loss of computational speed. So, due to our opinion the MRKII algorithm must be chosen over MRKI only in stiff problems or when the coupling from the active to the latent part is strong.

References:

[1] P. Bogacki and L. Shampine, A (2)3 pair of Runge-Kutta formulas, *Applied Math. Letters*, No.2, 1989, pp. 1-9.

[2] M. Günther, A. Kværnø and P. Rentrop, Multirate partitioned Runge-Kutta methods, *BIT*, Vol.3, No.41, 2001, pp. 504-514.

[3] M. Günther and P. Rentrop, Multirate ROW methods and latency of electric circuits, *Applied Numerical Mathematics*, No.13, 1993, pp. 83-102.

[4] M. Günther and P. Rentrop, Partitioning and multirate strategies in latent electric circuits, *International Series of Numerical Mathematics*, No.117, 1994, pp. 33-60.

[5] E. Hairer, S. Nørsett and G. Wanner, *Solving Ordinary Differential Equations I: Nonstiff Problems*, Springer-Verlag, Berlin, 1987.

[6] A. Kværnø, Stability of multirate Runge-Kutta schemes, *International Journal of Differential Equations and Applications*, No.1A, 2000, pp. 97-105.

[7] A. Kværnø and P. Rentrop, Low order multirate Runge-Kutta methods in electric circuit simulation, *IWRMM Universität Karlsruhe*, Preprint No.99/1, 1999.

[8] J. Lambert, *Numerical Methods for Ordinary Differential Systems: The Initial Value Problem*, John Wiley & Sons, West Sussex, 1991.

[9] J. Oliveira, *Métodos Multi-Ritmo na Análise e Simulação de Circuitos Electrónicos não Lineares*, Master Thesis, Department of Mathematics, University of Coimbra, 2004.

[1] Computation time (AMD Athlon 1.8 GHz, 256MB RAM).