Compound BDF multirate transient analysis applied to circuit simulation

Compound BDF Multirate Transient Analysis Applied to Circuit Simulation

B. Tasić[*][1], **A. Verhoeven**[2], **E. J. W. ter Maten**[1,2], and **T. G. J. Beelen**[1]

[1] Philips Research Laboratories, High Tech Campus 48, WA01, 5656 AE Eindhoven, The Netherlands
[2] Technische Universiteit Eindhoven, Den Dolech 2, 5600 MB, The Netherlands

Transient analysis is an important circuit simulation technique. The circuit model, which is a system of differential-algebraic equations, is solved for a given initial condition using numerical time integration techniques. Multirate methods are efficient if the dynamical behaviour of the circuit model is not uniform.

1 Introduction

Analog electrical circuits are usually modelled by differential-algebraic equations of the following type

$$\frac{d}{dt}\left[\mathbf{q}(t,\mathbf{x})\right] + \mathbf{j}(t,\mathbf{x}) = \mathbf{0}, \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^d$ represents the state of the circuit. A common analysis is the transient analysis, which computes the solution $\mathbf{x}(t)$ of this non-linear DAE along the time interval $[0, T]$ for a given initial state. In the Philips analog simulator Pstar, this initial value problem is solved by means of an implicit integration method, such as the variable order, variable time step BDF method.

Often, parts of electrical circuits have various time behaviour, i.e. different time scales. This characteristic can be exploited in order to increase the simulation speed without decreasing the required overall accuracy.

2 Multirate time integration

In contrast to classical (single-rate) integration methods, multirate time integration methods integrate different parts of the circuit by using different step sizes or even different numerical schemes. Besides the coarse time-grid, defined by $\{T_n, 0 \leq n \leq N\}$ with stepsizes $H_n = T_n - T_{n-1}$, a refined time-grid $\{t_{n-1,m}, 1 \leq n \leq N, 0 \leq m \leq q_n\}$ is also used with stepsizes $h_{n,m} = t_{n,m} - t_{n,m-1}$ and multirate factors q_n. If the two time-grids are synchronized, $T_n = t_{n,0} = t_{n-1,q_n}$ holds for all n.

---
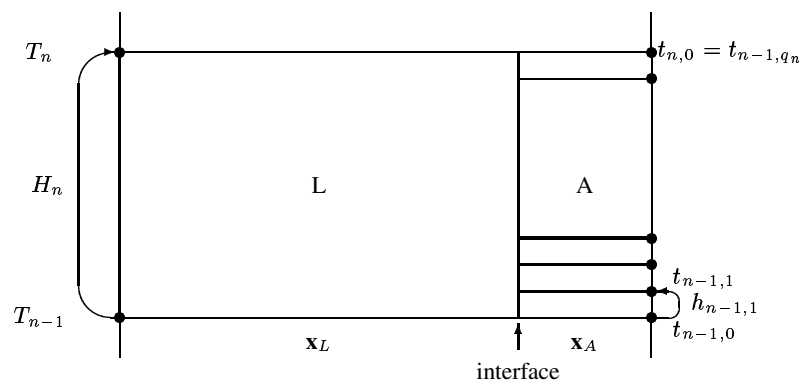
* Corresponding author: e-mail: bratislav.tasic@philips.com, Phone: +31 40 2749055, Fax: +31 40 2744700

3   Partitioning

For a multirate method it is necessary to partition variables and equations into an active (A) and a latent (L) part. We define $\mathbf{B}_A \in \mathbb{R}^{d_A \times d}$ and $\mathbf{B}_L \in \mathbb{R}^{d_L \times d}$, with $d_A + d_L = d$, to be the partitioning matrices satisfying $\mathbf{B}_A \mathbf{B}_A^T = \mathbf{I}, \mathbf{B}_L \mathbf{B}_L^T = \mathbf{I}, \mathbf{B}_A \mathbf{B}_L^T = \mathbf{O}, \mathbf{B}_L \mathbf{B}_A^T = \mathbf{O}$. Then the variables can be split into an active (A) and a latent (L) part as $\mathbf{x} = \mathbf{B}_A^T \mathbf{x}_A + \mathbf{B}_L^T \mathbf{x}_L$, allowing us to transform the equation (1) into the following partitioned system

$$\frac{d}{dt}\left[\mathbf{q}_L(t, \mathbf{x}_A, \mathbf{x}_L)\right] + \mathbf{j}_L(t, \mathbf{x}_A, \mathbf{x}_L) = \mathbf{0}, \tag{2}$$

$$\frac{d}{dt}\left[\mathbf{q}_A(t, \mathbf{x}_A, \mathbf{x}_L)\right] + \mathbf{j}_A(t, \mathbf{x}_A, \mathbf{x}_L) = \mathbf{0}. \tag{3}$$

Of course, it is also possible to extend these partitions by introducing more than two sub-systems, corresponding to various time scales.

The partitioning can be done by the user or automatically. Since the number of unknowns in modern circuits varies from several hundreds up to tens of thousands, it is clear that obtaining the permutation matrices $\mathbf{B}_A$ and $\mathbf{B}_L$ requires a significant effort. In Figure 1 the automatic dynamical partitioning principle is shown. The simulation starts with a standard single-rate integration, during which the information about the circuit behaviour is obtained. At a certain time point, where the multirate conditions (small part of the circuit changes more rapidly than the rest) are satisfied, the partitioning process is activated and the multirate time integration is initiated. If the multirate behaviour is lost after a time period, the time integration is switched back to the standard single-rate one and the new error data is started to be collected and analysed. The algorithm repeats this procedure until the end of the simulation.
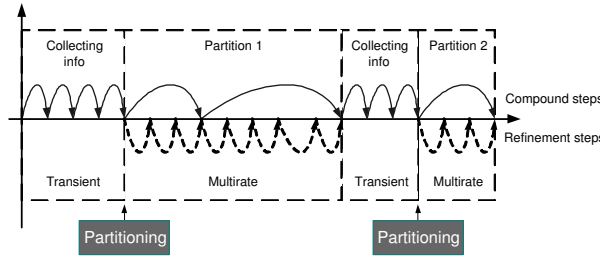

Fig. 1   Automatic dynamical partitioning of the circuit.

4   BDF Compound-Fast multirate algorithm

There are several multirate approaches for partitioned systems (see [1, 2, 3, 5, 6]) but we will consider the Compound-Fast version of the variable order BDF method, introduced in [4]. This method can be summarised by the following four steps.

1. Compound step. During the compound step the complete system (1) is integrated at the coarse time-grid by means of BDF time discretisation. Hence we solve

$$\alpha_n \, \mathbf{q}(T_n, \mathbf{x}_n) + H_n \, \mathbf{j}(T_n, \mathbf{x}_n) + \mathbf{b}_n = \mathbf{0}, \tag{4}$$

where $\mathbf{b}_n$ is a vector that represents the history of the numerical integration and α_n is a parameter that depends on the variable step (H_n) and the BDF order (k) and it is defined by $\alpha_n = H_n \sum_{m=1}^{k} \frac{1}{T_n - T_{n-m}}$.

2. Interpolation of the interface. The latent interface variables (i.e. interface currents in our approach, although voltages can also be used) are interpolated at the refined time-grid. Clearly, various interpolation techniques can be used, e.g. piecewise interpolation methods. Natural choice of the interpolation method and the order mainly depends on the integration order, corresponding to the available data history.

3. Refinement phase. The active part is integrated at the refined time-grid by the variable order BDF scheme. In fact, during the refinement phase, we solve a new initial value problem for a much smaller perturbed DAE (3). It solves for each time-point $t_{n-1,m}$ the nonlinear equation

$$\alpha_{n-1,m}\, \mathbf{q}_A(t_{n-1,m}, \mathbf{x}_{n-1,m}^A, \hat{\mathbf{x}}_{n-1,m}^I) + h_{n-1,m}\, \mathbf{j}_A(t_{n-1,m}, \mathbf{x}_{n-1,m}^A, \hat{\mathbf{x}}_{n-1,m}^I) + \mathbf{b}_{n-1,m} = \mathbf{0}, \quad (5)$$

where $\mathbf{x}^A$ denotes active unknowns only and $\hat{\mathbf{x}}_{n-1,m}^I$ is the vector of the interpolated latent interface currents at $t_{n-1,m}$, i.e. interpolated values of the latent unknowns coupled to the active part.

4. Active unknowns update. The active part of the solution and corresponding electrical variables (derived from the active unknowns) at the coarse time-grid are updated by the refined values.

5 Adaptive multirate step size control

Adaptive stepsize control of H_n and $h_{n-1,m}$ can be used to ensure

$$r_n^C < \mathrm{TOL}_C, \quad r_{n-1,m}^A < \mathrm{TOL}_A, \tag{6}$$

for every n and m. Here $r_n^C := \|\mathbf{B}_L \mathbf{r}_n^C\| + \tau \|\mathbf{B}_A \mathbf{r}_n^C\|$ represents the weighted local discretisation error (LDE) norm of the compound circuit in which $\mathbf{r}_n^C$ is the vectorial LDE norm. The factor $\tau \in [0, 1]$ is used to relax the active part of the LDE norm allowing larger compound time-steps. The value $r_{n-1,m}^A$ represents the refinement LDE norm and TOL_C, TOL_A are given compound and refinement tolerances respectively.

Since the interface of two circuit parts is interpolated, the interpolation error norm, say r_n^I, has to be estimated and included in the time-step control. Clearly, this estimate depends on the previous compound steps (the interpolation is performed on the coarse time grid) and on the type and the order of the interpolation method used. Hence the step size controller can be defined by

$$H_{n+1} = \theta\, \min\left\{ \left(\frac{\mathrm{TOL}_C}{r_n^C}\right)^{\frac{-1}{k+1}}, \left(\frac{\mathrm{TOL}_I}{r_n^I}\right)^{\frac{-1}{k+1}} \right\} H_n, \quad h_{n-1,m+1} = \theta\, \left(\frac{\mathrm{TOL}_A}{r_{n-1,m}^A}\right)^{\frac{-1}{k+1}} h_{n-1,m}, \quad (7)$$

where $\theta \in (0, 1)$ is a safety factor and TOL_I represents the defined tolerance of the interpolation error. Of course, all above mentioned tolerances can have the same value. In [7] more details are given about how control theory can be applied to design proper stepsize controllers.

6 Numerical results

The implementation of the multirate time integration algorithm in Pstar, allows us to obtain results for various circuits. We consider some theoretical examples, designed to meet multirate conditions, such as the linear chain and the matrix circuit shown in Figures 2a and 2b respectively. Moreover, we consider two practical examples, coming from the actual circuit design, the high-speed operational transconductance amplifier (HSOTA) and the charge pump, see Figures 2c and 2d. The linear chain circuit consists of two sources of various frequencies (ratio 500), a small active nonlinear part, a filter to electrically decouple two parts and a large resistance chain (1000 models) that increase the size of the latent part and yet do not affect the dynamical behaviour. Since the circuit satisfies multirate conditions to a large extent, the speed-up factor (as compared to the single-rate performance) is relatively large. The matrix circuit has similar properties, and contains 5×10 digital inverters and sources of various frequencies (ratio 100). In HSOTA and the charge pump circuits there are relatively large bias blocks that have practically constant dynamics, allowing small numbers of compound steps. Numerical results are summarised in Table 1. One should note that the speed-up factor depends on the dynamical behavior as well as the partition sizes. The latter causes smaller speed-up factors for HSOTA and the charge pump circuits, although they have an even more appropriate multirate time behaviour (compare columns for N_C and N_R).

(a) Nonlinear chain   (b) Matrix circuit

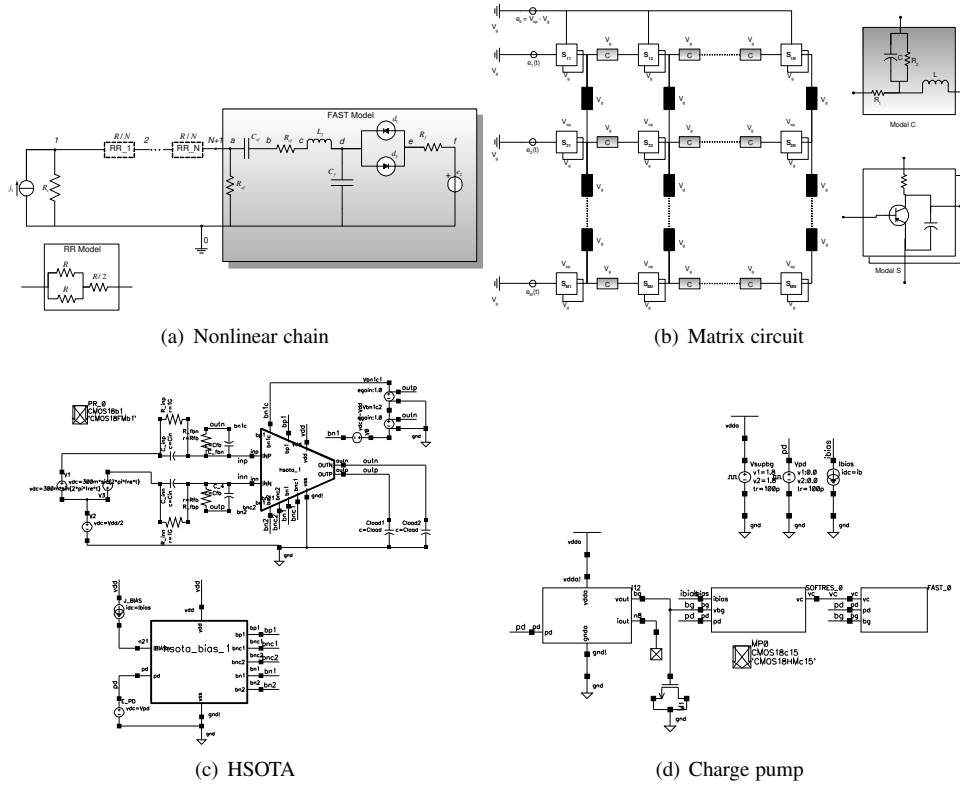

(c) HSOTA   (d) Charge pump

Fig. 2   Circuit examples.

Table 1   Numerical results. Notation: d- number of unknowns, N_C- number of compound steps, N_R- number of refinement steps, N_S- number of single-rate steps, d_A- number of active unknowns, S- speed-up factor.

Circuit name	d	N_C	N_R	N_S	d_A/d	S
Nonlinear chain	2010	303	16732	9588	0.7%	10 - 12
Matrix circuit	277	71	2810	2018	7 %	13
HSOTA	61	68	14092	14068	50%	1.6 - 2
Charge pump	249	151	10284	7419	12%	4 - 4.5

References

[1] A. Bartel. *Generalised Multirate: Two ROW-type Versions for Circuit Simulation* MSc Thesis, TU Darmstadt & IWRMM Universität Karlsruhe, 2000.

[2] A. Bartel, M. Günther. *A multirate W-method for electrical networks in state-space formulation*, J. of Comput. and Applied Maths., Vol. 147, pp. 411-425, 2002.

[3] C.W. Gear, D.R. Wells. *Multirate linear multistep methods*, BIT, 24 (1984), 484-502.

[4] A. El Guennouni, A. Verhoeven, E.J.W. ter Maten, T.G.J. Beelen. *Aspects of Multirate Time Integration Methods in Circuit Simulation Problems*, In: Proc. ECMI-2004 (Eindhoven, The Netherlands), pp 579-584, Springer.

[5] V. Savcenco, W.H. Hundsdorfer, J.G. Verwer. *A multirate time stepping strategy for parabolic PDE*, Report MAS-E0516, CWI, Amsterdam (2005).

[6] M. Striebel, M. Günther. *A charge oriented mixed multirate method for a special class of index-1 network equations in chip design*, Applied Numerical Mathematics, Vol.53, pp. 489-507, 2005.

[7] A. Verhoeven. *Automatic control for adaptive time stepping in electrical circuit simulation*, MSc Thesis, Technische Universiteit Eindhoven, Technical Note TN-2004/00033, Philips Research Laboratories, Eindhoven (2004).