Digital linear control theory applied to automatic stepsize control in electrical circuit simulation

EINDHOVEN UNIVERSITY OF TECHNOLOGY
Department of Mathematics and Computer Science

CASA-Report 05-04
January 2005

Digital linear control theory applied to
automatic stepsize control in
electrical circuit simulation

by

A. Verhoeven, T.G.J. Beelen,
M.L.J. Hautus and E.J.W. ter Maten

Digital linear control theory applied to automatic stepsize control in electrical circuit simulation

A. Verhoeven[1], T.G.J. Beelen[2], M.L.J. Hautus[1], and E.J.W. ter Maten[3]

[1] Technische Universiteit Eindhoven averhoev@win.tue.nl
[2] Philips Research Laboratories
[3] Technische Universiteit Eindhoven and Philips Research Laboratories

Summary. Adaptive stepsize control is used to control the local errors of the numerical solution. For optimization purposes smoother stepsize controllers are wanted, such that the errors and stepsizes also behave smoothly. We consider approaches from digital linear control theory applied to multistep BDF-methods.

Key words: adaptive stepsize control, BDF-methods, digital linear control

1 Introduction to error control

Transient simulation of electrical circuits is done by integration of the following implicit Differential-Algebraic Equation

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

where $\mathbf{q},\mathbf{j} : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ are nonlinear functions, that represent the charges and currents in the circuit, while $\mathbf{x}$ is the state vector. Because the BDF multistep-methods are the default methods used by analog circuit simulators, we will concentrate on these methods. While Runge Kutta methods often contain an embedded reference method to estimate the local error, for the k-step BDF-method this can be done by means of the prediction $\hat{\mathbf{q}}_n$ which is based on the extrapolation of the previous $k + 1$ values of $\mathbf{q}$. For the time-grid $\{t_i, i = 0, \ldots, N\}$ with timesteps $h_i = t_i - t_{i-1}$ we obtain the estimate

$$\hat{r}_n = \frac{h_n}{t_n - t_{n-k-1}}\|\mathbf{q}(t_n,\mathbf{x}_n) - \hat{\mathbf{q}}_n\|. \tag{2}$$

If this estimate $\hat{r}_n$ is larger than a given tolerance level TOL, the current step is rejected. Otherwise, the solution $\mathbf{x}_n$ is accepted and the next numerical solution can be computed at a new timepoint.

The following stepsize controller is very commonly used for integration methods of order p:

$$h_n = \left(\frac{\epsilon}{\hat{r}_{n-1}}\right)^{\frac{1}{p+1}} h_{n-1}, \tag{3}$$

where $\epsilon = \theta$ TOL. It is based on the assumption that the error estimate satisfies the model

$$\hat{r}_n = \hat{\varphi}_n h_n^{p+1}, \tag{4}$$

where φ_n is an unknown variable which is independent of h_n. This model is a good description for onestep methods and also a first order approximation for the BDF-methods.

However, it appears that the controller from eqn. (3) may produce rather irregular error and stepsize sequences, which will decrease the effectiveness in optimization.

2 Control-theoretic approach to stepsize control

It is possible to use control-theoretic techniques for error control. In [2] this idea has been applied to onestep methods where we have the simple model of eqn. (4). The logarithmic version of the onestep error model is

$$\log \hat{r}_n = (p+1)\log h_n + \log \hat{\varphi}_n. \tag{5}$$

Indeed, this implies that the sequence $\log \hat{r} = \{\log \hat{r}_n\}_{n \in \mathbb{N}}$ can be viewed as the output of a digital (i.e. discrete) linear control system, where $\log h = \{\log h_n\}_{n \in \mathbb{N}}$ is the input signal and $\log \hat{\varphi} = \{\log \hat{\varphi}_n\}_{n \in \mathbb{N}}$ is an unknown output disturbance. In general, one can denote all finite linear models for $\log \hat{r}$ by

$$\log \hat{r} = G(q)\log h + \log \hat{\varphi}, \tag{6}$$

where q is the shift-operator, with $q(\log h_n) = \log h_{n+1}$ and $G(q)$ being a rational function of q:

$$G(q) = \frac{L(q)}{K(q)} = \frac{\lambda_0 q^M + \cdots + \lambda_M}{q^M + \kappa_1 q^{M-1} + \cdots + \kappa_M}. \tag{7}$$

For the onestep model (5), we have $G(q) = p+1$. The input $\log h$ is computed on base of the previous values of the output $\log \hat{r}$ and the reference $\log \epsilon$.

All linear controllers can be denoted by

$$\log h = C(q)(\log \epsilon - \log \hat{r}), \tag{8}$$

where $C(q)$ is a rational function of q:

$$C(q) = \frac{B(q)}{A(q)} = \frac{\beta_0 q^{N-1} + \cdots + \beta_{N-1}}{q^N + \alpha_1 q^{N-1} + \cdots + \alpha_N}. \tag{9}$$

For the controller (3) we just have that $C(q) = \frac{1}{p+1}\frac{1}{q-1}$.

3 Derivation of process model for BDF-methods

Unfortunately, for the multistep BDF-methods, it is not possible to derive a linear model of the form of eqn. (6). In this case, we have the following nonlinear model for $\log \hat{r}$

$$\log \hat{r}_n = 2\log h_n + \log(h_{n-1}+h_n) + \cdots + \log(h_{n-p+1}+\cdots+h_n) + \log \hat{\varphi}_n - \log p!.$$
(10)

Note that $\log \hat{r}_n$ also depends on the previous stepsizes, because it is a multistep method. In [5] it is tried to approximate this model by the previous model for onestep methods. If the stepsizes only have small variations, also linearization can be used [1]. In [3] it is proved that the linearized model is equal to

$$\log \hat{r}_n = \sum_{k=0}^{p-1}(\gamma_p - \gamma_k)\log h_{n-k} + \log \hat{\varphi}_n, \quad \gamma_0 = -1, \gamma_k = \sum_{m=1}^{k}\frac{1}{m}.$$
(11)

This model can also be cast in the form of eqn. (6), where

$$G(q) = \frac{(1 + \gamma_p)q^{p-1} + (\gamma_p - \gamma_1)q^{p-2} + \cdots + (\gamma_p - \gamma_{p-1})}{q^{p-1}}.$$
(12)

4 Design of finite order digital linear stepsize controller

Consider the error model in eqn. (6), which is controlled by the linear controller (8). It is assumed that $G(q)$ is already be known, while $C(q)$ still must be designed. Now, the closed loop dynamics are described by the following equations:

$$\begin{cases} \log h = U_r(q)\log \epsilon + U_w(q)\log \hat{\varphi}, \\ \log \hat{r} = Y_r(q)\log \epsilon + Y_w(q)\log \hat{\varphi}. \end{cases}$$
(13)

where by (7), (9) the transfer functions satisfy

$$U_r(q) = \frac{B(q)K(q)}{A(q)K(q)+B(q)L(q)}, U_w(q) = \frac{-B(q)K(q)}{A(q)K(q)+B(q)L(q)},$$
$$Y_r(q) = \frac{B(q)L(q)}{A(q)K(q)+B(q)L(q)}, Y_w(q) = \frac{A(q)K(q)}{A(q)K(q)+B(q)L(q)}.$$
(14)

Thus, the poles of the system are determined by the $N + M$ roots of the characteristic equation

$$R(q) \equiv A(q)K(q) + B(q)L(q) = 0.$$

If the poles lay inside the complex unity circle, the closed loop system is stable. Suitable choices are $R(q) = (q - r)^{N+M}$ or $R(q) = q^{N+M} - r^{N+M}$ for $r \in [0,1)$ [3]. Assume that $A(q), B(q)$ can be factorized like $A(q) = (q - 1)^{p_A}(q + 1)^{p_R}\tilde{A}(q)$ and $B(q) = (q+1)^{p_F}\tilde{B}(q)$. Then, the order of adaptivity is equal to p_A, while the stepsize and error filter orders are p_R and p_F [2]. The coefficients of $\tilde{A}, \tilde{B}$ can be computed from

$$(q - 1)^{p_A}(q + 1)^{p_R}\tilde{A}(q)K(q) + (q + 1)^{p_F}\tilde{B}(q)L(q) = R(q).$$
(15)

5 Numerical experiments

Consider the circuit which corresponding equations are given by:

$$C\dot{V}_1 + \frac{1}{R}V_1 - \sin(\omega_1 t) - \frac{1}{R_1}(V_2 - V_1) = 0,$$
$$\frac{1}{R_1}(V_2 - V_1) - i_E = 0,$$
$$V_2 - V_3 = 0,$$
$$i_E - \frac{1}{R_2}(V_4 - V_3) = 0,$$
$$C\dot{V}_4 + \frac{1}{R}V_4 - \sin(\omega_2 t) - \frac{1}{R_2}(V_3 - V_4) = 0.$$

Parameters	Value
ω_1	$\frac{5}{2}\pi \cdot 10^3$
ω_2	$\frac{7}{4}\pi \cdot 10^3$
R	10
C	10^{-3}
R_1	1
R_2	1

A transient simulation along $[0, 0.08]$ is computed by a circuit simulator, while several stepsize controllers are used. Because the theory only holds for fixed integration order, the integration order is kept fixed at $p = 3$. By default, the simulator uses the controller of eqn. (3) with a buffer such that the stepsize remains constant for small variations (case 1). This control action is removed for the other cases, because it destroys the characteristic behaviour of the designed controller. For all controllers we have $R(q) = (q - r)^{N+M}$. The smoothness of the stepsize and error sequence is quantified by the number $s(x) = \sqrt{\sum_{m=1}^{N}(x_m - x_{m-1})^2}/\|x\|_2$. Table 1 shows the results of the several testcases. For this circuit case 4 produces the smoothest results. Note at the decline of the number of rejections for the cases 1,2 and 6 with $p_A = 1$ and $p_F = p_R = 0$. Figure 1 shows the results for cases 1 and 4.

Table 1. Numerical results

Case	N	M	p_A	p_F	p_R	r	# stepsizes	# rejections	# Newton iterations	$s(\log \hat{r})$	$s(\log h)$
1	1	0	1	0	0	0	1258	222	1480	1.17	0.57
2	1	0	1	0	0	$\frac{1}{2}$	1277	198	1475	0.82	0.38
3	2	0	2	0	0	$\frac{1}{2}$	990	609	1599	1.14	0.83
4	2	0	1	1	0	$\frac{1}{2}$	1053	0	1053	0.57	0.10
5	2	0	1	0	1	$\frac{1}{2}$	1198	0	1198	0.75	0.22
6	3	2	1	0	0	$\frac{1}{2}$	1015	0	1015	1.01	0.32

An important question is whether the new designed controllers also have a better performance if variable order is used. For many tested cases it was possible to get smoother results for a slightly increased or even decreased computational effort [3].
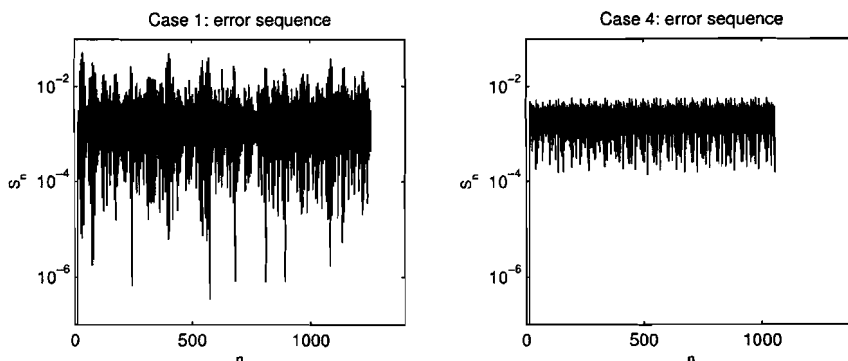
Fig. 1. Results of error sequences for cases 1 and 4 in Table 1.

6 Conclusions

If the error model is linear, control theory can indeed improve the smoothness of the results. For multistep BDF-methods applied to smooth problems, where the stepsizes have small variations, the linearized model works well. For more stiff problems it is better to use the onestep model.

The process model depends on the integration order. The designed controllers are also applicable to variable integration order. From the experiments it turns out that it is not attractive to use higher order adaptive controllers, while filtering can be attractive.

References

1. A. Sjö. *Analysis of computational algorithms for linear multistep methods*. PhD thesis, Lund University, Sweden, 1999.
2. G. Söderlind. Digital filters in adaptive time-stepping. *ACM Tr. on Math.Softw.*, V(N):1–24, 2002.
3. A. Verhoeven. *Automatic control for adaptive time stepping in electrical circuit simulation*. MSc thesis, Technische Universiteit Eindhoven, The Netherlands, 2004.
4. A. Verhoeven, T.G.J. Beelen, M.L.J. Hautus, and E.J.W. ter Maten. Digital linear control theory for automatic stepsize control. Presented at *Scientific Computing in Electrical Engineering*, Capo D'Orlando, Sicily, Italy, 2004.
5. M. Zhuang and W. Mathis. Research on stepsize control in the bdf method for solving differential-algebraic equations. In *IEEE Proceedings*, pages 229–232, 1994.