Digital linear control theory for automatic stepsize control

Digital linear control theory for automatic stepsize control

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. In transient analysis of electrical circuits the solution is computed by means of numerical integration methods. Adaptive stepsize control is used to control the local errors of the numerical solution. For optimization purposes smoother stepsize controllers can ensure that the errors and stepsizes also behave smoothly. For onestep methods, the stepsize control process can be viewed as a digital (i.e. discrete) linear control system for the logarithms of the errors and steps. For the multistep BDF-method this control process can be approximated by such a linear control system.

1 Introduction

Electrical circuits can be modelled by the following Differential-Algebraic Equation

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

where $\mathbf{q}, \mathbf{j} : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ represent the charges on capacitors and currents through resistors and sources in the circuit and $\mathbf{x}$ is the state vector. In transient analysis an Initial Value Problem has to be solved for this DAE, which is done by implicit integration methods (usually BDF methods).

The accuracy of integration methods depends on the magnitude of the stepsizes. Adaptive stepsize control is used to handle with the trade-off between the computational work load and the accuracy. Therefore, each iteration the magnitude of the local error must be estimated. If this estimate $\hat{r}_n$ is larger than a given tolerance level TOL, the current step is rejected. Otherwise, the numerical solution can be computed at a next timepoint $t_{n+1} = t_n + h_n$.

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

$$h_n = \left(\frac{\epsilon}{\hat{r}_{n-1}}\right)^{\frac{1}{p+1}} h_{n-1}. \tag{2}$$

This controller tries to keep the error $\hat{r}_n$ close to a reference level ϵ by means of the stepsize h_n. The reference level ϵ is equal to θ TOL, where $0 < \theta < 1$ is a safety factor, which reduces the number of rejections.

The stepsize controller is based on the assumption that the error estimate satisfies the model

$$\hat{r}_n = \hat{\varphi}_n h_n^{p+1}, \tag{3}$$

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 multistep BDF-methods. In practice, always some bounds and limiters are added to this controller in order to avoid numerical problems.

Important properties of a good simulator are speed, accuracy and robustness. It appears that the controller (2) produces rather irregular error and stepsize sequences, which will decrease the robustness.

2 Application of control theory

It seems attractive to use control-theoretic techniques for error control. In [1, 4] this idea has been applied to onestep methods where we have the simple model (3). Figure 1 shows the block diagram of this feedback control system. The process model $G(q)$ and the controller model $C(q)$ are described in the next subsections.

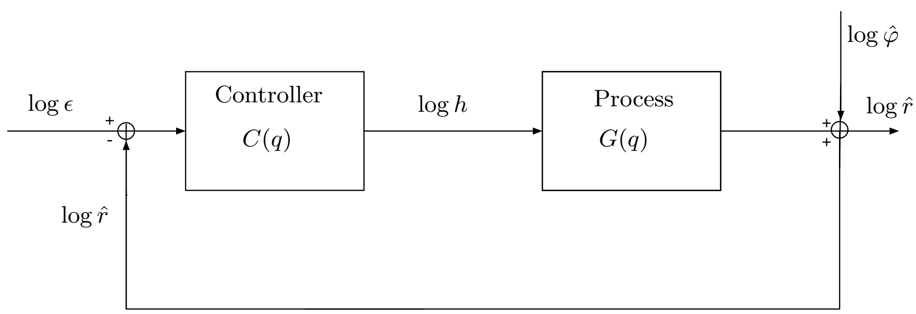


Fig. 1. Diagram of adaptive stepsize control viewed as a feedback control system.

2.1 Process model $G(q)$

The logarithmic version of the onestep error model (3) is

$$\log \hat{r}_n = (p+1) \log h_n + \log \hat{\varphi}_n. \tag{4}$$

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

$$\log \hat{r} = G(q) \log h + \log \hat{\varphi}, \tag{5}$$

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

$$G(q) = \frac{L(q)}{K(q)} = \frac{\lambda_0 q^M + \cdots + \lambda_M}{q^M + \kappa_1 q^{M-1} + \cdots + \kappa_M}. \tag{6}$$

For the onestep model, we just have that $G(q) = p + 1$. However, it is not possible to derive a linear model of this form for the multistep BDF methods. In this case for a p-step method, we have the following nonlinear model for $\log \hat{r}$ [5]

$$\log \hat{r}_n = 2 \log h_n + \log(h_{n-1} + h_n) + \cdots + \log(h_{n-p+1} + \cdots + h_n) + \log \hat{\varphi}_n - \log p!. \tag{7}$$

Note that $\log \hat{r}_n$ also depends on the previous stepsizes, because it is a multistep method. In [7] 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 [3]. In [5] it is proved that the linearized model is equal to

$$\log \hat{r}_n = (1+\gamma_p) \log h_n + (\gamma_p - \gamma_1) \log h_{n-1} + \cdots + (\gamma_p - \gamma_{p-1}) \log h_{n-p+1} + \log \hat{\varphi}_n, \tag{8}$$

where $\gamma_m = \sum_{n=1}^{m} \frac{1}{n}$ for $m \in \mathbb{N}$.

This model can also be cast in (5), where

$$G(q) = \frac{(1+\gamma_p) q^{p-1} + (\gamma_p - \gamma_1) q^{p-2} + \cdots + (\gamma_p - \gamma_{p-1})}{q^{p-1}}. \tag{9}$$

2.2 Controller model $C(q)$

The logarithmic version of the controller in eqn. (2) is

$$\log h_n - \log h_{n-1} = \frac{1}{p+1} (\log \epsilon - \log \hat{r}_{n-1}). \tag{10}$$

So, also the control action can be viewed as a linear feedback controller for the same linear system. 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

$$\log h = C(q)(\log \epsilon - \log \hat{r}), \tag{11}$$

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

$$C(q) = \frac{B(q)}{A(q)} = \frac{\beta_0 q^{N-1} + \cdots + \beta_{N-1}}{q^N + \alpha_1 q^{N-1} + \cdots + \alpha_N}. \tag{12}$$

For the controller of eqn. (2) we just have that $C(q) = \frac{1}{p+1} \frac{1}{q-1}$.

3 Design of finite order digital linear stepsize controller

Consider the error model (5), which is controlled by the linear controller (11). It is assumed that the error model is already available, while the controller still must be designed. This means that K, L are known, while A, B are unknown. Now, the closed loop dynamics are described by the following equations:

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

The transfer functions satisfy

$$\begin{aligned} U_r(q) &= \tfrac{B(q)K(q)}{R(q)},\ U_w(q) = \tfrac{-B(q)K(q)}{R(q)}, \\ Y_r(q) &= \tfrac{B(q)L(q)}{R(q)},\ \ Y_w(q) = \tfrac{A(q)K(q)}{R(q)}, \end{aligned} \tag{14}$$

where $R(q) = A(q)K(q) + B(q)L(q)$. In this section we will derive conditions for A, B such that the closed loop dynamics have some preferred properties.

3.1 Adaptivity and filter properties

The output $\log\hat{r}$ depends on the reference signal $\log\epsilon$ and the disturbance $\log\hat{\varphi}$. This means that in general the control error $\log\epsilon - \log\hat{r}$ is unequal to zero. However, there is no control error if $Y_w(q)\log\hat{\varphi} = 0$ and $Y_r(1) = 1$ [5]. If $\log\hat{\varphi}$ is a polynomial of degree $p_A - 1$ and $Y_w(q)\log\hat{\varphi} = 0$, we call the order of adaptivity p_A. It is always required that $p_A \geq 1$ in order to have no control error for a constant disturbance. For higher order adaptivity the controller is capable to follow linear or other polynomial trends of the disturbance $\log\hat{\varphi}$. It can be proved that the controller is adaptive with adaptivity order p_A if $(q-1)^{p_A}$ is a divisor of $A(q)$.

$$A(q) = (q-1)^{p_A}\hat{A}(q)$$

Because of numerical errors, the disturbance $\log\hat{\varphi}$ can contain alternating noise with frequency near π. The controller acts like a filter for the stepsizes or the errors if

$$|U_w(e^{i\omega})| = O(|\omega-\pi|^{p_F}), \quad \omega \to \pi$$

or

$$|Y_w(e^{i\omega})| = O(|\omega-\pi|^{p_R}), \quad \omega \to \pi.$$

Here p_F and p_R are the orders of the stepsize filter and the error filter. It is not possible combine an error filter with a stepsize filter. The controller is a stepsize filter of order p_F if $(q+1)^{p_F}$ is a divisor of $B(q)$.

$$B(q) = (q+1)^{p_F}\hat{B}(q)$$

The controller is an error filter of order p_R if $(q+1)^{p_R}$ is a divisor of $A(q)$.

$$A(q) = (q+1)^{p_R}\check{A}(q)$$

3.2 Position of the poles

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

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

If the poles lay inside the complex unity circle, the closed loop system is stable. The absolute values determine the reaction speed of the controllers, while the angles determine the eigenfrequencies. This means that real positive poles will produce smoother behaviour.

If the controller is adaptive, we know that the error always will be equal to the reference level if the disturbance is a low degree polynomial. However, this will be never the case in practice. Thus it is still possible that the next error will be larger than the tolerance level TOL.

Let R, S be polynomials of degree $N+M$, such that

$$\begin{aligned} S(q) &= A(q)K(q) &&= q^{N+M} + \sigma_1 q^{N+M-1} + \cdots + \sigma_{N+M} \\ R(q) &= A(q)K(q) + B(q)L(q) &&= q^{N+M} + \rho_1 q^{N+M-1} + \cdots + \rho_{N+M} \end{aligned}$$

In [5] it is proved that there are no rejections, such that $\hat{r}_n \geq$TOL if

- The disturbance $\hat{\varphi}$ satisfies the inequality:

$$\theta^{R(1)} \hat{\varphi}_n \hat{\varphi}_{n-1}^{\sigma_1} \cdots \hat{\varphi}_{n-N-M}^{\sigma_{N+M}} \leq 1. \tag{15}$$

- The coefficients of $R(q)$ satisfy: $\rho_i \leq 0, \quad i \in \{1, \dots, N+M\}$, e.g. $R(q) = q^{N+M} - r^{N+M}$.
- The previous $N+M$ stepsizes have been accepted.
- The real behaviour of the error estimate $\hat{r}$ is sufficiently modelled by the process model.

The first condition for the disturbance also depends on θ. Note that a small θ will indeed decrease the number of future rejections. The second condition is not true if all poles are real positive. However, if

$$R(q) = q^{N+M} - r^{N+M}, \tag{16}$$

this property is satisfied.

3.3 Computation of the control parameters

Assume that A, B 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 filter orders are p_R and p_F. Let $R(q)$ be the polynomial which roots are equal to the wanted poles, then the polynomials A, B are determined by

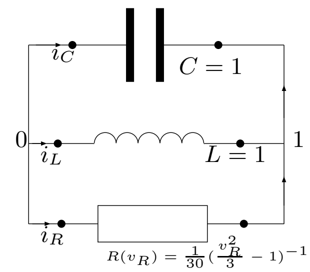
$$(q-1)^{p_A}(q+1)^{p_R}\tilde{A}(q)K(q) + (q+1)^{p_F}\tilde{B}(q)L(q) = R(q). \tag{17}$$

The coefficients of A, B are the control parameters, which can be computed from (17).

4 Numerical experiments

Consider the initial value problem (VandePol equation) for the following electrical circuit:

$$\begin{aligned} &\frac{dV_1}{dt} + i_L + 30V_1\left(\frac{V_1^2}{3} - 1\right) = 0 \qquad && V_1(0) = 0 \\ &\frac{di_L}{dt} - V_1 = 0 && i_L(0) = 1 \end{aligned}$$



This IVP is solved on $[0, 100]$ by means of the BDF2 method with tolerance level TOL = 1e-4 and reference level $\epsilon = 0.3$TOL. A frequently used controller is (2) with $p_A = 1$ and having a pole equal to zero.

$$\text{I:} \quad h_n = \left(\frac{\epsilon}{\hat{r}_{n-1}}\right)^{\frac{1}{3}} h_{n-1} \quad (p_A = 1)$$

Often, this controller is used in combination with a buffer, e.g.

$$\frac{h_n}{h_{n-1}} \in [0.8, 2] \Rightarrow h_n = h_{n-1}.$$

Consider the next second order adaptive stepsize controller, which poles are equal to 0.2. This means that it is able to predict linear trends of the disturbance $\log \hat{\phi}$.

$$\text{II:} \quad \frac{h_n}{h_{n-1}} = \left(\frac{\epsilon}{\hat{r}_{n-1}}\right)^{\frac{8}{15}} \left(\frac{\hat{r}_{n-2}}{\hat{r}_{n-1}}\right)^{-\frac{8}{25}} \frac{h_{n-1}}{h_{n-2}} \quad (p_A = 2)$$

The IVP has been solved by controller I with buffer (case 1) and Controller II (case 2). These cases require 1000, 1080 stepsizes and 1686, 2054 Newton iterations, respectively. Figure 2 shows the resulting stepsize and error sequences. The best results are obtained in case 2, because of the better adaptivity at the cost of an increase of Newton iterations. Because of the higher smoothness of case 2, the safety factor could be increased for case 2. Indeed, for $\epsilon = 0.6$TOL, the cases need 1847 and 1667 Newton iterations, respectively.

An important question is whether the new designed controllers also have a better performance for a real circuit simulator. Therefore, in the next three cases a real circuit is simulated, while a variable integration order is used [5]. In case 1 the default stepsize controller of the simulator is used. In the other cases, the stepsize controllers are based on digital linear control theory having the properties $p_A = 1$ and $p_F = p_R = 0$. For all three cases, the safety factor

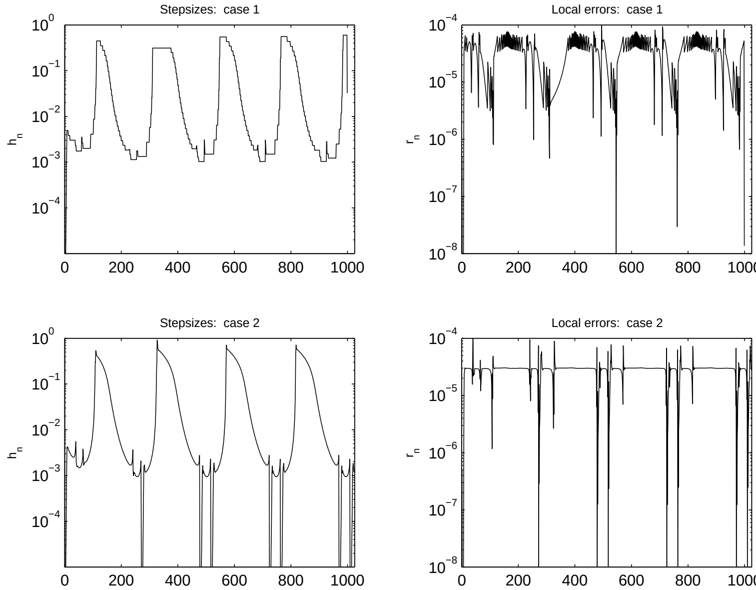


Fig. 2. Stepsize and error sequences for the two tested controllers.

is variable. The smoothness of the stepsize and error sequences is quantified by means of the number $s(x) = \sqrt{\sum_{m=1}^{N}(x_m - x_{m-1})^2 / \|x\|_2}$. Table 1 shows the results of these three cases. Note that for the cases 2 and 3 the smoothness of the results is improved, while the computational work is about the same. Furthermore the performance is even better (8%) than for case 1.

Table 1. Numerical results for perf_mos7_qubic_6953 ($p_A = 1$,$p_F = p_R = 0$).

Case	# stepsizes	# rejections	# Newton iterations	$s(\hat{r})$	$s(h)$
1	6465	947	43232	0.85	0.58
2	6934	777	40234	0.79	0.48
3	6423	714	39619	0.74	0.85

5 Conclusions

It has been tried to derive a linear model for the behaviour of the local error. For onestep methods this is less complex, because then the local error only depends on the last stepsize. But because circuit simulators use the multistep BDF-methods, also the application for BDF-methods has been studied. In that case, a linearized linear model can be derived, which is only correct for small variations of the stepsizes.

From the experiments it seems not always attractive to use higher order adaptive controllers. However, filtering appears to be attractive because it reduces the high-frequent noise, which makes the behaviour of the stepsizes and the errors much smoother.

Because the described method is only developed for a fixed order of integration, the theoretical results for a variable integration order are not known yet. Clearly the local error also depends on the integration order and this affects the process model. It seems not possible to describe this behaviour by means of a linear model. Despite this application in the variable integration order case works satisfactory.

To deal with the trade-off between the smoothness and the speed, optimal control could be applied. In this case, a cost function should be defined which is dependent on the stepsize sequence and the error sequence.

References

1. Gustafsson, K.: Control Theoretic Techniques for Stepsize Selection in Implicit Runge-Kutta Methods. ACM TOMS **20** 496–517, (1994)
2. Kuo, B.C.: Digital Control Systems. Holt-Saunders International Editions, New York (1980)
3. Sjö, A.: Analysis of computational algorithms for linear multistep methods. PhD Thesis, Lund University, Lund (1999)
4. Söderlind, G.: Digital filters in adaptive time-stepping. ACM Tr. on Math.Softw., **Vol.V, No.N** 1–24 (2002)
5. Verhoeven, A.: Automatic control for adaptive time stepping in electrical circuit simulation. MSc Thesis, Technische Universiteit Eindhoven, Eindhoven (2004), Technical Note TN-2004/00033, Philips Research Laboratories, Eindhoven (2004)
6. Verhoeven, A., Beelen, T.G.J., M.L.J. Hautus, Maten, E.J.W. ter: Digital linear control theory applied to automatic stepsize control in electrical circuit simulation. ECMI Proceedings, (2004)
7. Zhuang, M., Mathis, W.: Research on Stepsize Control in the BDF Method for Solving Differential-Algebraic Equations. IEEE Proceedings (ISCAS94) 229–232, (1994)