Digital Filters in Adaptive Time-Stepping

GUSTAF SÖDERLIND
Lund University

Adaptive time-stepping based on linear digital control theory has several advantages: the algorithms can be analyzed in terms of stability and adaptivity, and they can be designed to produce smoother stepsize sequences resulting in significantly improved regularity and computational stability. Here, we extend this approach by viewing the closed-loop transfer map $H_{\hat{\varphi}} : \log \hat{\varphi} \mapsto \log h$ as a digital filter, processing the signal $\log \hat{\varphi}$ (the principal error function) in the frequency domain, in order to produce a smooth stepsize sequence $\log h$. The theory covers all previously considered control structures and offers new possibilities to construct stepsize selection algorithms in the asymptotic stepsize-error regime. Without incurring extra computational costs, the controllers can be designed for special purposes such as higher order of adaptivity (for smooth ODE problems) or a stronger ability to suppress high-frequency error components (nonsmooth problems, stochastic ODEs). Simulations verify the controllers' ability to produce stepsize sequences resulting in improved regularity and computational stability.

Categories and Subject Descriptors: G.1.7 [**Numerical Analysis**]: Ordinary Differential Equations—*initial value problems*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Adaptivity, algorithm analysis, control theory, digital filters, error control, mathematical software, stepsize control

1. INTRODUCTION

This article will develop new strategies for adaptive stepsize selection using linear digital control theory. Although it might at first appear unfamiliar to the numerical analyst, digital control (see, e.g., Åström and Wittenmark [1990]) is based on our common, classical theories of linear difference equations, difference operators and stability, making extensive use of the discrete Laplace transform (the z transform). Bearing this in mind, basic control theory is readily accessible also to the numerical analyst. A survey of control theoretic adaptive stepsize selection, introducing the pertinent terminology and techniques,

The research was in part funded by the Swedish Research Council for Engineering Sciences under contract TFR 222/98-74.
Author's address: Numerical Analysis, Centre for Mathematical Sciences, Lund University, Box 118, SE-221 00 Lund, Sweden; email: Gustaf.Soderlind@na.lu.se.

is found in Söderlind [2002], which also develops the analysis of the established PI stepsize controllers, [Gustafsson 1991, 1994]. The reader is assumed to be acquainted with that background.

We shall assume that the stepsizes used in the numerical adaptive solution of an initial value ODE or DAE problem are such that the local error estimator's dependence on the stepsize is accurately described by the *asymptotic model*

$$\hat{r}_n = \hat{\varphi}_n h_n^k, \tag{1}$$

where $\hat{\varphi}_n$ is the norm of the principal error function. *No further assumptions about the computational process will be made.* For convenience, the recursion indexing in (1) departs from Gustafsson [1991, 1994] and Söderlind [2002] in order to eliminate a trivial common factor in the z transforms that represent the system.

The elementary stepsize selection algorithm commonly used in locally adaptive time-stepping [Gear 1971, p. 156] is

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{1/k} h_n, \tag{2}$$

where $\varepsilon = \theta \cdot \text{TOL}$, $\theta < 1$ is a suitable safety factor, and TOL is the local error tolerance; if the local error estimate $\hat{r}_n$ exceeds TOL, the step will be rejected and recomputed with a reduced stepsize. If the order of convergence of the time-stepping method is p, then one takes the power $k = p + 1$ for an error-per-step (EPS) control, and $k = p$ for an error-per-unit-step (EPUS) control; the choice is in no way crucial to the theory that will be developed below. This elementary control is typically implemented with limiters and discontinuities that, when less judiciously employed, make the order of adaptivity equal 0, see the survey [Söderlind 2002]. Such heuristic schemes will not be treated in this article. Instead, we aim for a rigorous analysis based on linear control and filter theory.

The behavior of the recursion (2) can be analyzed in terms of the theory of linear difference equations by taking logarithms; one then obtains

$$\log h_{n+1} = \log h_n + \frac{1}{k}(\log \varepsilon - \log \hat{r}_n). \tag{3}$$

This is a first order adaptive, purely integrating deadbeat controller [Söderlind 2002]. It has been thoroughly analyzed for one-step (Runge–Kutta) methods and is known to have several shortcomings such as an oscillatory behavior when the stepsize is limited by numerical stability, as well as a "nervous" and nonsmooth response to error estimates contaminated by numerical noise. A number of alternative approaches have therefore been studied in the literature, see Gustafsson et al. [1988], Gustafsson [1991, 1994], Hall [1985, 1986], Hall and Higham [1988], Higham and Hall [1990], Söderlind [2002], Watts [1984], and Zonneveld [1964]. In particular, there is an extensive theory for PI (proportional–integral) controlled time-stepping [Gustafsson et al. 1988; Gustafsson 1991; Hairer et al. 1993; Hall 1985; Söderlind 2002], including error estimation [de Swart and Söderlind 1997], parameterization and synchronization with other types of logic and support algorithms in software [Gustafsson and Söderlind 1997], as well as pseudocode descriptions of the controllers to facilitate a simple implementation [Gustafsson 1991, 1994; de Swart 1997].

We shall develop a fully general control structure for locally adaptive time-stepping, using digital filter theory. The controller can in principle be employed at *no extra computational expense*. Controller parameters are selected to attenuate high frequency contents ("noise") in $\{\log \hat{\varphi}_n\}$, with the aim to provide highly regular stepsize sequences for nonsmooth problems such as stochastic differential equations. The theory is however equally important for implicit methods for ODEs and DAEs, where irregularities are often incurred by, for example, remaining Newton iteration errors. The efficiency gain is in terms of qualitative improvement and increased computational stability.

2. TRANSFER FUNCTIONS, FREQUENCY RESPONSE AND DIGITAL FILTERS

Let $\log \hat{r}$, $\log h$ and $\log \hat{\varphi}$, respectively (i.e., without subscripts), denote the sequences $\{\log \hat{r}_n\}$, $\{\log h_n\}$ and $\{\log \hat{\varphi}_n\}$. Further, let q denote the forward shift operator. The difference equation (3) is then written $(q-1)\log h = k^{-1}(\log \varepsilon - \log \hat{r})$, corresponding to the control law

$$\log h = \frac{1}{k}\,\frac{1}{q-1}(\log \varepsilon - \log \hat{r}) = C(q)\cdot(\log \varepsilon - \log \hat{r}), \tag{4}$$

where $C(q)$ is the *control transfer function*, which for the elementary controller (2) is given by

$$C(q) = \frac{1}{k}\,\frac{1}{q-1}\,. \tag{5}$$

As $\Delta = q - 1$ is the forward difference operator, $1/(q-1)$ is a summation operator—the discrete analogue of an integral operator—hence, the name *integral control*.

The asymptotic stepsize—error relation (1) is written as $\log \hat{r} = G(q)\log h + \log \hat{\varphi}$, where $G(q) = k$ is the *process transfer function*. The asymptotic model is therefore static with a constant gain k.

The interaction of process and controller is described by the linear system

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

$$\log h = C(q)\cdot(\log \varepsilon - \log \hat{r}). \tag{7}$$

Solving for $\log \hat{r}$ and $\log h$, using the asymptotic process model $G(q) = k$ but leaving the choice of $C(q)$ open, we obtain the *closed loop dynamics* [Söderlind 2002],

$$\log \hat{r} = R_{\varepsilon}(q)\log \varepsilon + R_{\hat{\varphi}}(q)\log \hat{\varphi} \tag{8}$$

$$\log h = H_{\varepsilon}(q)\log \varepsilon + H_{\hat{\varphi}}(q)\log \hat{\varphi}. \tag{9}$$

This expresses how the two *inputs*, the setpoint $\log \varepsilon$ and the disturbance $\log \hat{\varphi}$, influence the two *outputs*, the error estimate $\log \hat{r}$ and the stepsize $\log h$, when the process/controller interaction has been taken into account, see Figure 1. Note that $\log h$ is the *internal* means of adaptivity, or the control, making the error adapt to $\log \varepsilon$, which is the *external* means of adaptivity.

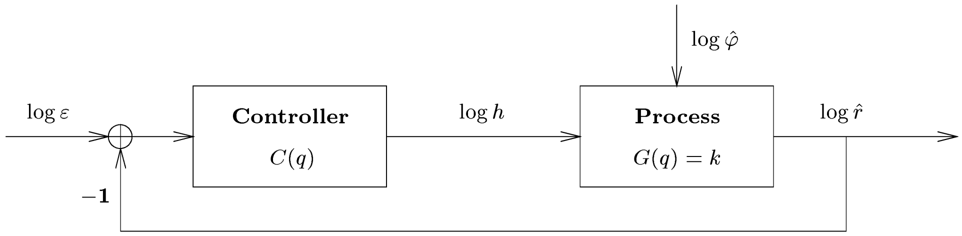

Fig. 1. *Adaptive stepsize selection viewed as a feedback control system*. The process consists of the discretization method which takes a given stepsize $\log h$ as input and produces an error estimate output $\log \hat{r} = G(q) \log h + \log \hat{\varphi}$, where the external, additive disturbance $\log \hat{\varphi}$ accounts for the properties of the ODE. The error estimate is fed back with reversed phase and added to $\log \varepsilon$ to compare actual and desired error levels. This control error is mapped by the controller to the next stepsize $\log h$, through $\log h = C(q) \cdot (\log \varepsilon - \log \hat{r})$. The entire closed loop system has two inputs, the setpoint $\log \varepsilon$ and the disturbance $\log \hat{\varphi}$. It has two outputs, the error $\log \hat{r}$ and the internal control $\log h$. They are related to the inputs through the closed loop transfer functions.

As $\log \varepsilon$ is constant, we may here for convenience but without loss of generality put $\log \varepsilon = 0$, but several formulas below will for clarity still include ε. We are then left with the *stepsize transfer map* $H_{\hat{\varphi}}(q) : \log \hat{\varphi} \mapsto \log h$ and the *error transfer map* $R_{\hat{\varphi}}(q) : \log \hat{\varphi} \mapsto \log \hat{r}$, given by

$$H_{\hat{\varphi}}(q) = -\frac{C(q)}{1 + k \cdot C(q)}; \qquad R_{\hat{\varphi}}(q) = \frac{1}{1 + k \cdot C(q)}, \tag{10}$$

where $C(q)$ remains to be chosen. The error transfer map can be viewed both as a map from $\log \hat{\varphi}$ to the error $\log \hat{r}$ *and* to the control error $\log \varepsilon - \log \hat{r}$, as these quantities only differ by a constant. As we have taken $\log \varepsilon = 0$, these become identical (up to a sign), and the controller's objective, which is to make the control error small, can be studied directly from the behavior of $R_{\hat{\varphi}}(q)$.

In our context, a *digital filter* is a discrete-time dynamical system. Here, we shall interpret $H_{\hat{\varphi}}(q)$ and $R_{\hat{\varphi}}(q)$ as digital filters, implying that the stepsize sequence $\log h$ is considered to be obtained through digital signal processing of the external disturbance $\log \hat{\varphi}$. The filter properties are determined by the poles and zeros of these transfer maps, and will be analyzed in the frequency domain.

If we consider the stepsize transfer map, then (9) and (10) imply that the closed loop dynamical system is described by the difference equation

$$(1 + k \cdot C(q)) \log h = -C(q) \log \hat{\varphi}, \tag{11}$$

which depends on the actual choice of controller dynamics $C(q)$. The operators $C(q)$, $H_{\hat{\varphi}}(q)$ and $R_{\hat{\varphi}}(q)$ are rational functions of q, and in each case the numerator and denominator are assumed to have no common factor. However, $H_{\hat{\varphi}}(q)$ and $R_{\hat{\varphi}}(q)$ have the same denominator.

Definition 2.1. The *order of dynamics* p_D of the closed loop system equals the degree of the denominator of $H_{\hat{\varphi}}(q)$.

The poles of the transfer functions (the roots of the characteristic equation) determine the stability of the closed-loop system. The system is called *stable* if all poles of $H_{\hat\varphi}(q)$ are located strictly inside the unit circle. The *homogeneous solutions* of (11) are further supposed to be smooth and decay reasonably fast. If these necessary conditions are met, the next criterion is to make sure that the *particular solutions* of (11) can be shown to have an improved smoothness compared to the forcing term $\log\hat\varphi$; this is the filter design problem.

The *spectral properties* of the transfer map $H_{\hat\varphi}(q)$ have a significant effect on the smoothness of stepsize sequences. A bounded input signal $\log\hat\varphi$ may be represented by a linear combination of "periodic" data sequences $\{\cos\omega n\}$ with frequencies $\omega \in [0,\pi]$; constant functions correspond to $\omega = 0$, and the Nyquist frequency $\omega = \pi$ corresponds to the oscillation $(-1)^n$, which is the highest frequency that can be resolved according to the sampling theorem. To simplify the analysis, one considers complex data sequences $\log\hat\varphi = \{e^{i\omega n}\}$, one frequency $\omega \in [0,\pi]$ at a time [Söderlind 2002]. As $H_{\hat\varphi}(q) : \log\hat\varphi \mapsto \log h$ is a linear map, the output $\log h$ has the same spectral content as $\log\hat\varphi$. Hence, $\log h = A(\omega)\{e^{i\omega n}\}$. Disregarding *phase*, the *amplitude* $|A(\omega)|$ reveals whether the frequency ω is amplified or attenuated. From (11), we obtain

$$(1 + k\cdot C(e^{i\omega}))A(\omega)e^{i\omega n} = -C(e^{i\omega})e^{i\omega n}. \tag{12}$$

and it follows that $|A(\omega)| = |H_{\hat\varphi}(e^{i\omega})|$.

Definition 2.2. The *error frequency response* and *scaled stepsize frequency response* are defined by $|R_{\hat\varphi}(e^{i\omega})|$ and $|kH_{\hat\varphi}(e^{i\omega})|$, respectively, for $\omega \in [0,\pi]$.

The scaling factor k is a normalization that makes $|kH_{\hat\varphi}(1)| = 1$, irrespective of the actual method order. Frequency responses will be plotted in log-log diagrams (Bode diagrams), and measured in the ISO unit of decibel (dB), that is, in terms of $20\log_{10}|kH_{\hat\varphi}(e^{i\omega})|$ and $20\log_{10}|R_{\hat\varphi}(e^{i\omega})|$, respectively.

Now, for the elementary controller (5), the transfer functions are

$$H_{\hat\varphi}(q) = -\frac{1}{kq}\,; \quad R_{\hat\varphi}(q) = \frac{q-1}{q}\,. \tag{13}$$

Here we make three observations: First, the pole is located at the origin, showing that the closed loop is stable. Second, as the elementary controller's scaled stepsize frequency response $|kH_{\hat\varphi}(e^{i\omega})| \equiv 1$ is independent of ω, it has no regularizing effect on the stepsize sequence. Third, $R_{\hat\varphi}(1) = 0$, which demonstrates that the controller is at least *first-order adaptive*, a notion we define as follows:

Definition 2.3. Let $R_{\hat\varphi}(q)$ have all its poles strictly inside the unit circle. If the error transfer function satisfies $|R_{\hat\varphi}(q)| = O(|q-1|^{p_A})$ as $q \to 1$, the controller's *order of adaptivity* is p_A.

This order notion can be expressed in the time domain in terms of polynomials: let $\log\hat\varphi = \{P(n)\}$ be a polynomial sampled at integer points. As $R_{\hat\varphi}(q)$ contains the difference operator $(q-1)^{p_A} = \Delta^{p_A}$, it will annihilate all polynomials of degree $p_A - 1$. Hence, $\log\varepsilon - \log\hat r_n \to 0$ at a rate determined by the magnitude of the poles: *the local error is adapted to the tolerance*. But the notion can also be expressed in the frequency domain: if $\log\hat\varphi = \{e^{i\omega n}\}$, then, since $\Delta^{p_A}\{e^{i\omega n}\} = (e^{i\omega} - 1)^{p_A}\{e^{i\omega n}\}$, we have $\log\varepsilon - \log\hat r = O(\omega^{p_A})$ as $\omega \to 0$, if homogeneous solutions have decayed. Thus, the error frequency response of a stable system is $|R_{\hat\varphi}(e^{i\omega})| = O(\omega^{p_A})$ as $\omega \to 0$ if and only if the order of adaptivity is p_A, and this order is revealed by the slope of the error frequency response graph [Söderlind 2002].

Apart from the error transfer function's zero at $q = 1$, it is possible to regularize the stepsize sequence $\log h = H_{\hat\varphi}(q) \log \hat\varphi$ by making sure that $H_{\hat\varphi}(q) = 0$ at $q = e^{i\pi} = -1$; this will annihilate the frequency $\omega = \pi$. Thus, $\log h$ will not contain the oscillatory sequence $\{(-1)^n\}$ even if it is present in $\log \hat\varphi$. Other high frequencies will be suppressed as well. Therefore, by placing a zero of $H_{\hat\varphi}(q)$ at a suitable location on the unit circle, signal transmission of that particular frequency is blocked. Here, we limit ourselves to $q = -1$ and introduce a simple notion of filter order.

Definition 2.4. Let $H_{\hat\varphi}(q)$ have all its poles strictly inside the unit circle. If the stepsize transfer function satisfies $|H_{\hat\varphi}(q)| = O(|q + 1|^{p_F})$ as $q \to -1$, the stepsize *filter order at* $q = -1$ is p_F.

In control theory, it is well known that a controller $C(q)$ must contain the operator $1/(q - 1)$, known as "integral action," in order to have $p_A \geq 1$, see also Söderlind [2002]. From this basic requirement, we can construct a general controller.

Definition 2.5. The *general control map for adaptive time-stepping* is represented by the rational function

$$C(q) = \frac{P(q)}{(q-1)Q(q)}, \tag{14}$$

where the polynomials P and Q are relatively prime and $P(1) \neq 0$. Further, $\deg(Q) = \deg(P) = p_D - 1$, where p_D is the order of the closed loop dynamics.

The general controller's stepsize recursion $\log h = C(q) \cdot (\log \varepsilon - \log \hat r)$ now corresponds to the difference equation

$$(q-1)Q(q) \log h = P(q) \cdot (\log \varepsilon - \log \hat r). \tag{15}$$

As $\log \hat r_n$ depends on $\log h_n$, the degree of P must not exceed the degree of Q, or the stepsize recursion would become implicit. Thus, the general controller is completely parameterized by introducing the polynomials

$$P(q) = \sum_{j=1}^{p_D} \beta_j q^{p_D - j}; \quad Q(q) = q^{p_D - 1} + \sum_{j=2}^{p_D} \alpha_j q^{p_D - j}. \tag{16}$$

We shall especially consider third-order dynamics, in which case we have

$$P(q) = \beta_1 q^2 + \beta_2 q + \beta_3; \quad Q(q) = q^2 + \alpha_2 q + \alpha_3. \tag{17}$$

Controllers with $p_D = 2$ are naturally embedded within the class of $p_D = 3$ controllers. If one starts from (16) or puts $\alpha_3 = \beta_3 = 0$ in (17) is immaterial; a common factor of q may be eliminated from (14)–(15) as this pole-zero cancellation does not affect the dynamics.

Table I. Control Structures that have been Used for Adaptive Time-Stepping. Included, Free Controller Parameters are Marked '×' and the Maximum Orders for Each Structure is Given

Parameters					Orders			Type
$k\beta_1$	$k\beta_2$	$k\beta_3$	α_2	α_3	p_D	p_A	p_F	
1					1	1	–	elementary control [Gear 1971]
×					1	1	convol.	I control [Söderlind 2002]
×	×				2	1	≤ 1	PI control*
2	−1		−1		2	2	0	PC deadbeat†
×	×		−1		2	2	0	predictive control [Söderlind 2002]
×	×	×			3	1	≤ 2	PID control
×	×	×	−1		3	2	≤ 1	predictive PID
×	×		×		2	≤ 2	≤ 1	general filter
×	×	×	×	×	3	≤ 3	≤ 2	general filter

*See Gustafsson et al. [1988], Gustafsson [1991], Hall [1985], and Söderlind [2002].
†See Gustafsson [1994], Watts [1984], and Zonneveld [1964] .

Inserting the operators $P(q)$ and $Q(q)$ into (15), we find the stepsize recursion

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{\beta_1} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{\beta_2} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{\beta_3} \left(\frac{h_n}{h_{n-1}}\right)^{-\alpha_2} \left(\frac{h_{n-1}}{h_{n-2}}\right)^{-\alpha_3} h_n. \qquad (18)$$

This structure covers all linear controllers with $p_D \le 3$, and provides a full set of five parameters for the design of the *stepsize* and *error filters*

$$-kH_{\hat{\varphi}}(q) = \frac{kP(q)}{(q-1)Q(q) + kP(q)}; \quad R_{\hat{\varphi}}(q) = \frac{(q-1)Q(q)}{(q-1)Q(q) + kP(q)}, \qquad (19)$$

which are obtained by inserting (14) into (10). The actual controller parameterization is then a byproduct of the filter design, as $C(q) = -H_{\hat{\varphi}}(q)/R_{\hat{\varphi}}(q)$.

Controllers will be categorized by the labeling Hp_D, p_A, p_F, to indicate the orders of dynamics p_D and adaptivity p_A, as well as the filter order p_F at $q = -1$. For example, the elementary controller is in the $H110$ category and PI controllers are in $H210$ and $H211$, but there are also other controllers in these categories. Deadbeat controllers are identified by a subscript 0, like in H_0110 for the elementary controller. Finally, the letter R replaces H, like in R_0321, to indicate that the filter is applied to the error sequence $\log \hat{r}$ instead of to the stepsize sequence $\log h$. Table I indicates where in the literature various subclasses of controllers have been considered and gives their main properties.

Some general properties of the filter pair (19) should be noted. When $p_D = 3$, $H_{\hat{\varphi}}(q)$ has two zeros and three poles. As we have five parameters at our disposal, we have, in principle, full control of the stepsize filter. But *there is a complementarity between $H_{\hat{\varphi}}(q)$ and $R_{\hat{\varphi}}(q)$*. For example, if for some q^* we have $P(q^*) = 0$, then $R_{\hat{\varphi}}(q^*) = 1$, see (19). Conversely, if $Q(q^*) = 0$, then $-kH_{\hat{\varphi}}(q^*) = 1$. For the frequency responses in particular, this implies that, if $|kH_{\hat{\varphi}}(e^{i\omega^*})| = 0$, then $|R_{\hat{\varphi}}(e^{i\omega^*})| = 1$, and vice-versa. Thus, for example, a $(-1)^n$ oscillation in $\log \hat{\varphi}$ cannot simultaneously be removed from the sequences $\log h$ and $\log \hat{r}$. In view of (1), the effects of a nonsmooth $\log \hat{\varphi}$ must naturally be accommodated by either $\log h$ or $\log \hat{r}$: choosing a constant stepsize implies that $\log \hat{r}$ accommodates the full variation of $\log \hat{\varphi}$. Conversely, if it were possible to choose $\log h_n = (\log \varepsilon - \log \hat{\varphi}_n)/k$, the error estimate would have been constant, $\log \hat{r} = \log \varepsilon$. The filter design problem is to find a compromise that keeps $\log \hat{r} \approx \log \varepsilon$ while $\log h$ remains smooth.

We shall develop new controllers of class $H211$, $H312$ and $H321$, and show both theoretically and in simulations that the proposed controllers have a strong ability to suppress noise in $\log \hat{\varphi}$. A single implementation of filter/controller could be employed, while still allowing particular problem classes to use special controllers. We also discuss a factorization of the controller that makes it possible to separate the filter characteristic from the basic, integral control action; this is of particular interest as it enables the control error $\log \varepsilon - \log \hat{r}$ to stay closer to zero.

3. ORDER CONDITIONS

Order conditions are given below for controllers of dynamic order $p_D \leq 3$. From (19), it follows that $Q(q)$ determines the order of adaptivity p_A. Similarly, the stepsize filter order p_F is determined by $P(q)$. This subdivision makes it possible to apply different filter design objectives to the control structure (18).

3.1 Adaptivity Order Conditions

The order of adaptivity is increased by placing extra zeros of $R_{\hat{\varphi}}(q)$ at $q = +1$. The adaptivity order conditions are

$$p_A = 2 \Leftrightarrow \alpha_2 + \alpha_3 = -1, \tag{20}$$

$$p_A = 3 \Leftrightarrow \alpha_2 = -2; \quad \alpha_3 = 1. \tag{21}$$

If $p_D = 2$, then $p_A \leq 2$ as $\alpha_3 = 0$. For $p_D = 3$, (21) implies (20). If the order of adaptivity is p_A, then the *difference operator* $(q-1)^{p_A - 1}$ is a factor of $Q(q)$.

3.2 Stepsize Low-Pass Filter Order Conditions

For nonsmooth problems, a controller providing some stepsize regularization may be required. *Stepsize low-pass filters* remove high-frequency content and let low-frequency content pass through. They are obtained by placing one zero (or more) of $H_{\hat{\varphi}}(q)$ at $q = -1$. The stepsize filter order conditions at $q = -1$ are

$$p_F = 1 \Leftrightarrow \beta_1 - \beta_2 + \beta_3 = 0, \tag{22}$$

$$p_F = 2 \Leftrightarrow \beta_1 = \beta_2/2 = \beta_3. \tag{23}$$

If $p_D = 2$, then $\beta_3 = 0$ and $p_F \leq 1$. For $p_D = 3$, (23) implies (22). A filter order p_F implies that the *averaging operator* $(q+1)^{p_F}$ is a factor of $P(q)$. Thus, (23) corresponds to repeated averaging, a classical technique for regularizing noisy data.

3.3 Error Low-Pass Filter Order Conditions

A low-pass filter may be used in a similar way to regularize the error sequence $\log \hat{r}$, by placing one zero (or more) of $R_{\hat{\varphi}}(q)$ at $q = -1$. Thus, the error filter order conditions at $q = -1$ are

$$p_R = 1 \Leftrightarrow \alpha_2 - \alpha_3 = 1, \tag{24}$$

$$p_R = 2 \Leftrightarrow \alpha_2 = 2; \quad \alpha_3 = 1, \tag{25}$$

where the subscript R indicates error filtering. Again, (25) implies (24). As these conditions use the same parameters as (20)–(21), we must give up some order of adaptivity to filter $\log \hat{r}$. Moreover, as it is impossible to simultaneously remove the same frequency from the stepsize and error sequences (complementarity), we would also have to give up stepsize low-pass filtering altogether; recall that $|R_{\hat{\varphi}}(-1)| = 0$ implies $|kH_{\hat{\varphi}}(-1)| = 1$.

4. DEADBEAT CONTROLLERS AND HIGH-FREQUENCY EMPHASIS

We shall first derive the simplest controllers that generalize the elementary controller $h_{n+1} = (\varepsilon/\hat{r}_n)^{1/k} h_n$, which is known as a *deadbeat* controller as its poles are located at the origin. Deadbeat can be achieved for all orders p_A with the controller (14); the characteristic equation, of degree p_D, is $(q-1)Q(q) + kP(q) = 0$, or

$$q^3 + (k\beta_1 + \alpha_2 - 1)q^2 + (k\beta_2 - \alpha_2 + \alpha_3)q + k\beta_3 - \alpha_3 = 0. \tag{26}$$

Out of the controller's $2p_D - 1$ parameters, see (17), the $p_D - 1$ coefficients α_i are specified by the adaptivity order conditions (21). The remaining p_D paramaters $k\beta_i$ can be used to place the p_D poles at any prescribed locations.

For $p_D = 1$, there is a single parameter $k\beta_1$, and the choice $k\beta_1 = 1$ puts the pole at the origin; this defines the elementary controller, labeled H_0110. For $p_D = 2$, we have second order adaptivity if $\alpha_2 = -1$, and we must then take $k\beta_1 = 2$ and $k\beta_2 = -1$ to achieve deadbeat control; this is the H_0220 predictive controller suggested in Gustafsson [1994], Watts [1984], and Zonneveld [1964] and analyzed in Gustafsson [1994]. Finally, for $p_D = 3$, the adaptivity condition (21) imposes $\alpha_2 = -2$ and $\alpha_3 = 1$; this leads to $k\beta_1 = -k\beta_2 = 3$ and $k\beta_3 = 1$, or the third order adaptive H_0330 controller

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{3/k} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{-3/k} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{1/k} \left(\frac{h_n}{h_{n-1}}\right)^{2} \left(\frac{h_{n-1}}{h_{n-2}}\right)^{-1} h_n. \tag{27}$$

With their poles at the origin, deadbeat controllers have the best possible intrinsic stability. But Figure 2 reveals that *deadbeat controllers put an undesirable emphasis on high frequencies* that makes both $\log h$ and $\log \hat{r}$ rougher than $\log \hat{\varphi}$. They are therefore suitable only for very smooth problems, and also put stringent demands on how supporting algorithms, such as equation solvers, are implemented.

Still within deadbeat designs, to reduce the high-frequency emphasis, we may use either a stepsize low-pass filter or an error low-pass filter. Let us for $p_D = 2$ require $p_F = 1$, that is, we impose the filter condition (22). The two remaining parameters are used to place the roots of (26) at the origin. This implies $k\beta_1 = k\beta_2 = \alpha_2 = 1/2$ and leads to the unique H_0211 controller

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{1/(2k)} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{1/(2k)} \left(\frac{h_n}{h_{n-1}}\right)^{-1/2} h_n. \tag{28}$$

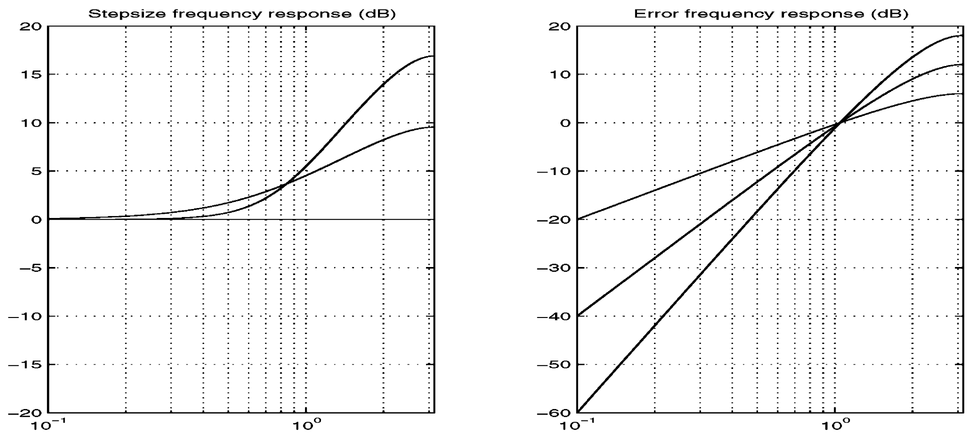

Fig. 2. *H_0110, H_0220 and H_0330 deadbeat controllers.* Scaled stepsize frequency response $20\log_{10}|kH_{\hat\varphi}(e^{i\omega})|$ (left) and error frequency response $20\log_{10}|R_{\hat\varphi}(e^{i\omega})|$ (right) is shown for $\omega \in [0.1, \pi]$. As $|R_{\hat\varphi}(e^{i\omega})| = O(\omega^{p_A})$, we observe the characteristic 20, 40 and 60 dB/decade slopes at low frequencies for adaptivity orders $p_A = 1, 2, 3$, respectively (right). Consequently, as p_A is increased, low-frequency components in $\log\varepsilon - \log\hat r$ are strongly suppressed, although at the cost of high frequency emphasis: $|R_{\hat\varphi}(e^{i\pi})|$ increases from +6 dB to +12 dB and +18 dB. High-frequency content in $\log h$ (left) *also* increases, however, by +10 dB and +17 dB for H_0220 and H_0330, respectively. The controllers are therefore suitable only for smooth problems, where $\log\hat\varphi$ has a negligible high-frequency content.

Table II. Overview of the Nine Unique, Maximum Order Deadbeat Controllers with $p_D \le 3$

$k\beta_1$	$k\beta_2$	$k\beta_3$	α_2	α_3	p_D	p_A	p_F	p_R	Designation
1					1	1	–	–	H_0110
2	−1		−1		2	2	0	–	H_0220
1/2	1/2		1/2		2	1	1	–	H_0211
0	1		1		2	1	–	1	R_0211
3	−3	1	−2	1	3	3	0	–	H_0330
5/4	1/2	−3/4	−1/4	−3/4	3	2	1	–	H_0321
1	1	−1	0	−1	3	2	–	1	R_0321
1/4	1/2	1/4	3/4	1/4	3	1	2	–	H_0312
−1	1	1	2	1	3	1	–	2	R_0312

For $p_D \le 3$, there are exactly nine structurally different deadbeat controllers maximizing adaptivity and/or filter order by various combinations of the order conditions of Section 3. Table II describes their structure and parameterization.

5. FILTER DESIGN

The main objective for constructing stepsize/error filters is to overcome the deadbeat controller's high frequency emphasis and generate smoother stepsize and error sequences. An offending high frequency content in $\log\hat\varphi$ can be reduced in both $\log h$ and $\log\hat r$, at the cost of an increased low frequency content in the control error $\log\varepsilon - \log\hat r$. Examples are the H_0211 and H_0312 compared to the elementary H_0110, all first-order adaptive, see Figure 3. Moreover, the comparison of H_0321 with a similar nondeadbeat $H321$ controller shows that properly designed "noise shaping" can *simultaneously* further reduce the high

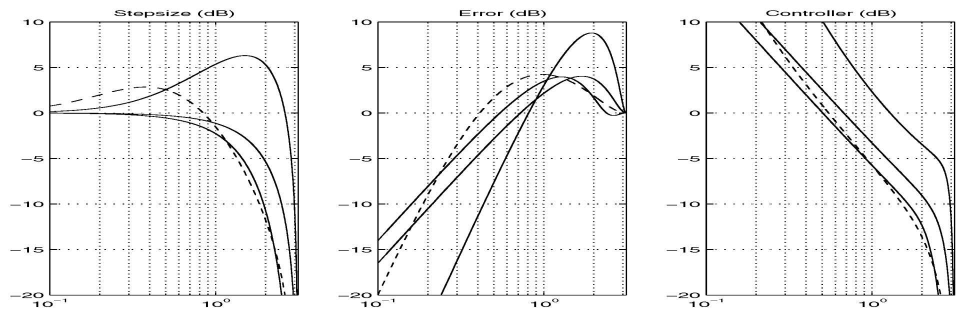

Fig. 3. *Controllers with stepsize low-pass filters.* Stepsize (left), error (center) and controller frequency response $20\log_{10}|C(e^{i\omega})|$ (right) for $\omega \in [0.1, \pi]$. In each diagram the deadbeat controllers (solid lines) intersect the -10 dB level in the following order from left to right: H_0312, H_0211, H_0321. For the first two, the low-pass filters significantly reduce high frequency emphasis (left, center). In H_0321, however, amplification for $\omega \in (1, 2)$ is considerable and only top frequencies are attenuated. For H_0211 and H_0312, the controller response (right), shows distinct -20 dB/decade slopes up to $\omega = 2$, demonstrating first order integral action. The second order integral action of H_0321 would, however, only be seen below $\omega = 0.3$. The dashed line is the non-deadbeat $H321$ controller. Compared to H_0321, it shows that frequency emphasis can be reshaped. High-frequency content is significantly reduced both in stepsize and error.

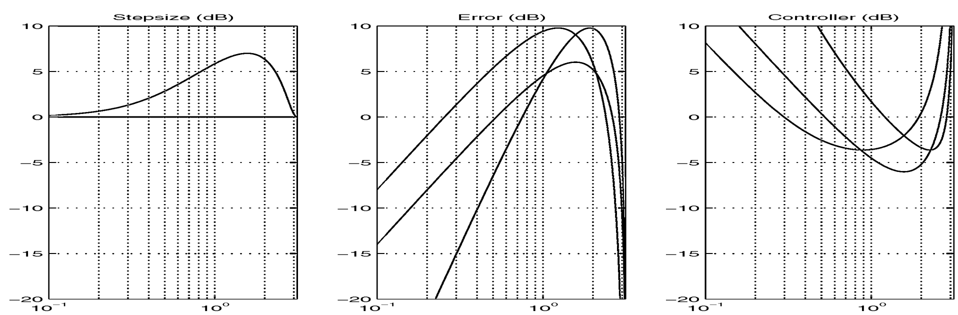

Fig. 4. *Deadbeat controllers with error low-pass filters.* Stepsize frequency response (left) of R_0211 is independent of ω and responses of R_0321 and R_0312 coalesce. The error (center) and controller frequency responses (right) intersect the 0 dB level in the left-to-right order: R_0312, R_0211, R_0321. Error low-pass filtering is effective only at top frequencies (center) and the considerable amplification for $\omega \in (0.5, 2)$ shows no substantial improvement over the deadbeat controllers in Figure 2. The controller now has a pole instead of a zero at $\omega = \pi$.

frequency content in $\log h$ *and* $\log \hat{r}$, although (1) always holds. The price is an increased low-frequency error.

A comparison of stepsize low-pass filtering vs. error low-pass filtering indicates that the former is preferable (see Figures 3 and 4); the controller's frequency response then has a zero rather than a pole at $\omega = \pi$. Thus, the starting point for designing good filters is to consider modifications to H_0211, H_0312 and H_0321.

Parameter choice is anything but arbitrary. Specific filter characteristics are found only in certain (affine) parameter subspaces, defined by order conditions. Stable filters (closed loops) are located only in a bounded subset, the stability region, of each subspace. However, only part of the stability region corresponds to acceptable closed-loop dynamics. Pole placement, frequency responses and time domain simulations together determine the final parameterization; different filters can be constructed for different classes of method/problem combinations, in particular for smooth and nonsmooth problems. ODE, DAE and SDE solvers can benefit from using dedicated classes of filters; in some cases, a smooth $\log h$ is more important than having $|\log \varepsilon - \log \hat{r}_n|$ small at all times.

5.1 First-Order Dynamics

For $p_D = 1$, we have the $H110$ controller $h_{n+1} = (\varepsilon/\hat{r}_n)^{\beta_1} h_n$ with stepsize filter $H_{\hat{\varphi}}(q) = -\beta_1/(q-1+k\beta_1)$ and error filter $R_{\hat{\varphi}}(q) = (q-1)/(q-1+k\beta_1)$. As the pole is $q = 1 - k\beta_1$, the closed loop is stable if $k\beta_1 \in (0,2)$, and $k\beta_1 = 1$ turns it into the deadbeat H_0110. A *reduced* integral gain, $k\beta_1 \in (0,1)$, gives slower dynamics and smoother stepsizes. The map $\log h = H_{\hat{\varphi}}(q) \log \hat{\varphi}$ then implies the difference equation $(q-1+k\beta_1)\log h = -\beta_1 \log \hat{\varphi}$, with solution

$$\log h_n = (1-k\beta_1)^n \log h_0 - \beta_1 \sum_{m=1}^{n} (1-k\beta_1)^{n-m} \log \hat{\varphi}_{m-1}. \tag{29}$$

When $k\beta_1 \in (0,1)$ this is known as "exponential forgetting" or a *convolution filter*, see Söderlind [2002] and Table I, with a limited ability to attenuate high-frequency contents in $\log \hat{\varphi}$. The smaller one chooses $k\beta_1$, the smoother is the stepsize sequence. However, the homogeneous solution also decays slower, see (29), so a compromise is necessary. For smooth problems, $k\beta_1 \in (0.7, 1)$ is likely to work fine, but if $\log \hat{\varphi}$ has a significant high-frequency content, then $k\beta_1 \in (0.3, 0.5)$, offers an improved attenuation of high frequencies. Plots of stepsize and error frequency responses are found in Söderlind [2002].

A similar convolution filter expression can also be obtained for the error,

$$\log \hat{r}_n = (1-k\beta_1)^n \log r_0 + \sum_{m=1}^{n} (1-k\beta_1)^{n-m} (\log \hat{\varphi}_m - \log \hat{\varphi}_{m-1}), \tag{30}$$

showing a regularization of the difference $\{\log \hat{\varphi}_n - \log \hat{\varphi}_{n-1}\}$ if $k\beta_1 \in (0,1)$. Note, however, that if $k\beta_1 = 1$, then $\log \hat{r}_n = \log \hat{\varphi}_n - \log \hat{\varphi}_{n-1}$. Hence, if $\log \hat{\varphi}$ contains the oscillation $(-1)^n$, then $\log \hat{r}_n$ could be twice as large as $\log \hat{\varphi}_n$. As a factor of 2 corresponds to $+6$ dB, this explains the magnitude of the error frequency response at $\omega = \pi$ for the elementary H_0110 controller, see Figure 2.

5.2 Second-Order Dynamics

For $p_D = 2$, the controller is $h_{n+1} = (\varepsilon/\hat{r}_n)^{\beta_1} (\varepsilon/\hat{r}_{n-1})^{\beta_2} (h_n/h_{n-1})^{-\alpha_2} h_n$. This structure offers a wide range of possibilities, covering all PI and predictive controllers [Gustafsson 1991, 1994; Söderlind 2002]. PI controllers are first order adaptive and generally belong to $H210$ [Gustafsson 1991], but some belong to $H211$, provided a negative proportional gain is acceptable. The H_0220 is fully covered in Gustafsson [1994] and Söderlind [2002]. But the free parameter α_2 implies that new $H211$ controllers can be constructed.

In order to obtain a smoother behavior than that of H_0211, the overall control gain must be reduced. Given the first-order filter condition, the pole locations

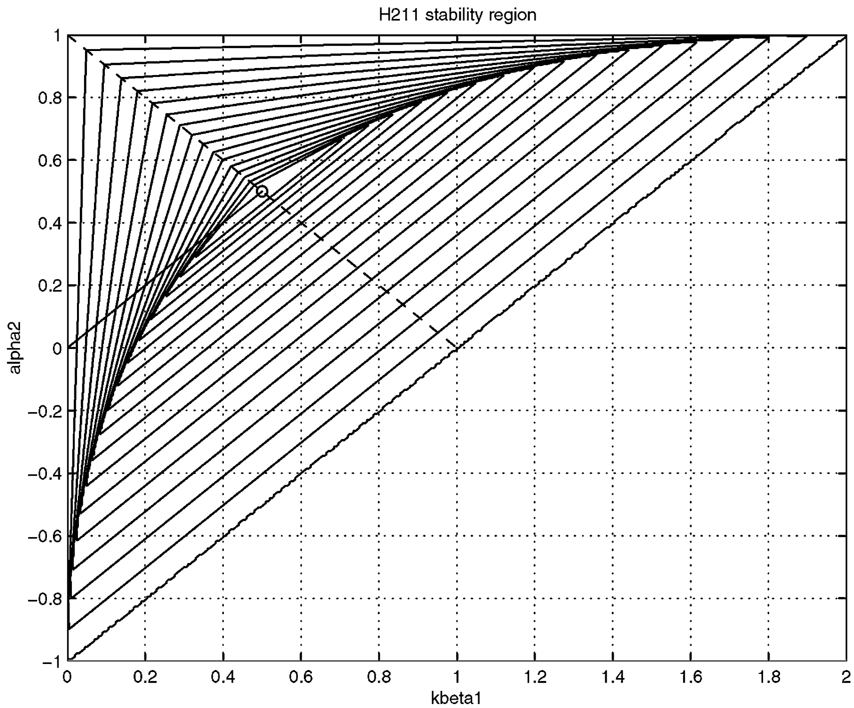

Fig. 5. $H211$ *closed loop stability region* in the $(k\beta_1, \alpha_2)$ parameter plane. Level curves enclose stable controllers with closed loop poles of maximum magnitude $1, 0.95, 0.9$, etc. Poles are real above the caustic and complex conjugate below it; the dominating pole has positive real part below the dashed line and negative above it. Controllers with a desirable behavior are found above the caustic and below the dashed line. The $H211b$ family is located on the solid line starting at the deadbeat H_0211, marked '∘' at $(1/2, 1/2)$. Further, the class $H221$ is empty, since $p_A = 2$ requires $\alpha_2 = -1$, which forces $k\beta_1 = k\beta_2 = 0$.

are determined by $(k\beta_1, \alpha_2)$; in Figure 5, the stability region is plotted. $H211$ controllers with well-located closed-loop poles and good frequency responses are given by the one-parameter family $H211b$, defined by

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{1/(bk)} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{1/(bk)} \left(\frac{h_n}{h_{n-1}}\right)^{-1/b} h_n. \tag{31}$$

The closed-loop poles are $0, 1 - 2/b$, that is, one pole has moved out from the origin. Stability then requires $b \in (1, \infty)$, although $b \geq 2$ is needed to prevent the nonzero pole from being negative and causing an oscillatory closed-loop impulse response. But b can be varied significantly while the overall control behavior largely remains qualitatively intact, and one may in practice choose $b \in [2, 8]$, with larger values offering increased smoothness. A value of $b = 4$ is recommended, see Figure 6. An important consequence of the wide parameter range is *robustness*: if the value of k is wrong because of order reduction or similar phenomena, the dynamics of the controller will not change dramatically.

The class $H221$ of second-order adaptive, first order stepsize low-pass filtering controllers is empty. This is easily seen both from the stability

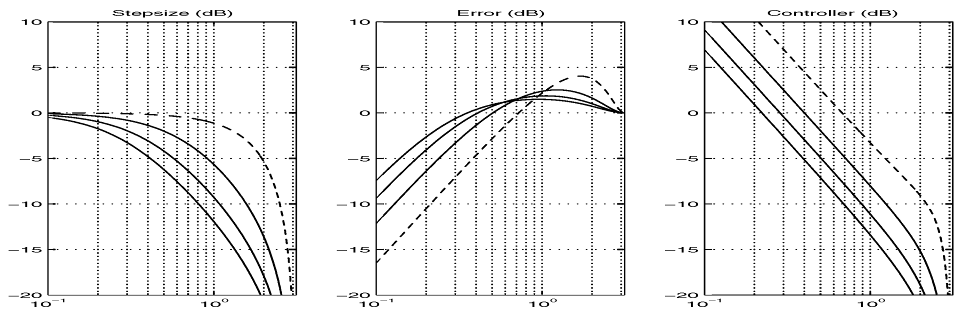

Fig. 6. *One-parameter family of $H211b$ controllers.* Stepsize (left), error (center) and controller frequency response (right) for $H211b$ controllers with $b = 2, 4, 6, 8$. The dashed line indicates deadbeat H_0211 for which $b = 2$. Increasing values of b, corresponding to a lowered overall integral gain (right), increases stepsize and error smoothness, but also increases low-frequency control errors.

region (Figure 5) or from the characteristic equation by applying the Schur criterion.

5.3 Third-Order Dynamics

In order to construct controllers of class $H312$, with enhanced regularity compared with the deadbeat H_0312, we shall move some poles out of the origin. Leaving (at least) one pole $q = 0$, the stability region can be studied in the $(k\beta_1, \alpha_2)$ plane, see Figure 7, and a construction similar to that of the $H211b$ family is possible. Thus, we define the $H312b$ family by

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{1/(bk)} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{2/(bk)} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{1/(bk)} \left(\frac{h_n}{h_{n-1}}\right)^{-3/b} \left(\frac{h_{n-1}}{h_{n-2}}\right)^{-1/b} h_n.$$

Just like the $H211b$, the $H312b$ family is located on a straight line segment, connecting the deadbeat controller to the origin in the stability region. The closed loop poles are $0, 0, 1 - 4/b$; stability therefore requires $b \in (2, \infty)$, but preventing oscillations requires $b \geq 4$. In practice one may choose $b \in [4, 16]$, with larger values offering increased smoothness. The value $b = 8$ is recommended, and frequency responses become very similar to those of Figure 6, except with the high frequency attenuation in the stepsize sequence doubled due to $p_F = 2$; the controller therefore offers even higher regularity.

The construction of $H321$ controllers follows similar lines. To increase regularity, the overall control gain must be reduced. We prescribe the order conditions for $p_A = 2$ and $p_F = 1$ and place the poles at $q = 1/3, 1/2, 2/3$. This leads to the parameterization

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{1/(3k)} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{1/(18k)} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{-5/(18k)} \left(\frac{h_n}{h_{n-1}}\right)^{5/6} \left(\frac{h_{n-1}}{h_{n-2}}\right)^{1/6} h_n.$$

Its frequency responses are shown with a dashed line in Figure 3, and show that a significant redistribution of the frequency content has been achieved compared to H_0321. As a result, this $H321$ controller offers improved smoothness.

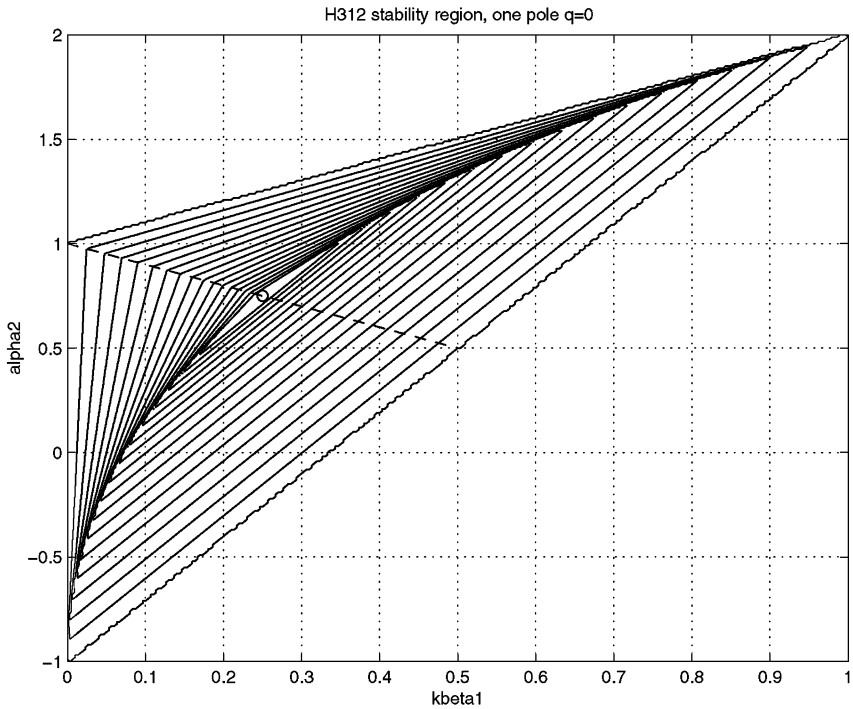

Fig. 7. *$H312$ closed loop stability region* in the $(k\beta_1, \alpha_2)$ parameter plane for $k\beta_3 = \alpha_3$, that is, at least one pole is $q = 0$, see (26). The deadbeat controller H_0312 is marked '∘' at $(1/4, 3/4)$. Level curves, caustic line and dashed separatrix have the same interpretation as in Figure 5.

For $p_D = 3$, higher order controllers than $H312$, $H321$ and $H330$ cannot be constructed as the classes $H322$ and $H331$ are empty.

6. PID CONTROL

Within the general control structure, one finds the best-known and most frequently used controllers. A standard type is PID control, and we investigate controllers of first- and second-order adaptivity.

6.1 First-Order Adaptivity

A discrete PID controller has third order dynamics. Its structure is defined by

$$C^{\mathrm{PID}}(q) = q^{-1}\left(k_{\mathrm{I}}\,\frac{q}{q-1} + k_{\mathrm{P}} + k_{\mathrm{D}}\,\frac{q-1}{q}\right), \tag{32}$$

where $k_{\mathrm{I}}, k_{\mathrm{P}}, k_{\mathrm{D}}$ are the integral, proportional and derivative gains, respectively. The first term is recognized from (5) as the "integral" part. In addition to this, the PID controller has a proportional part k_{P} and a "derivative" part; the latter is recognized by the backward difference operator $\nabla = (q-1)/q$. The PI controller is obtained as the special case $k_{\mathrm{D}} = 0$, in which case the order of dynamics is two. The control map $\log h = C^{\mathrm{PID}}(q) \cdot (\log \varepsilon - \log \hat{r})$ now implies the difference equation

$$\Delta \log h = (k_{\mathrm{I}} + k_{\mathrm{P}}\nabla + k_{\mathrm{D}}\nabla^2) \cdot (\log \varepsilon - \log \hat{r}), \tag{33}$$

which is equivalent to the recursion $\log h_{n+1} - \log h_n = k_{\mathrm{I}}(\log \varepsilon - \log \hat{r}_n) - k_{\mathrm{P}}(\log \hat{r}_n - \log \hat{r}_{n-1}) - k_{\mathrm{D}}(\log \hat{r}_n - 2\log \hat{r}_{n-1} + \log \hat{r}_{n-2})$. Hence, the general PID controller for adaptive time-stepping can be written

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{k_{\mathrm{I}}+k_{\mathrm{P}}+k_{\mathrm{D}}} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{-(k_{\mathrm{P}}+2k_{\mathrm{D}})} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{k_{\mathrm{D}}} h_n. \tag{34}$$

A PID controller is therefore a special case of (18) with $\alpha_2 = \alpha_3 = 0$. All properties of the controller, in particular its filter characteristics, are determined by the parameters $(kk_{\mathrm{I}}, kk_{\mathrm{P}}, kk_{\mathrm{D}})$. They are related to the $k\beta_i$ through the involutive parameter transformation

$$\begin{pmatrix} k_{\mathrm{I}} \\ k_{\mathrm{P}} \\ k_{\mathrm{D}} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ & -1 & -2 \\ & & 1 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{pmatrix}. \tag{35}$$

For the error transfer function, we find

$$R_{\hat{\varphi}}(q) = \frac{q^2(q-1)}{q^3 - (1 - kk_{\mathrm{I}} - kk_{\mathrm{P}} - kk_{\mathrm{D}})q^2 - (kk_{\mathrm{P}} + 2kk_{\mathrm{D}})q + kk_{\mathrm{D}}}, \tag{36}$$

and note that, provided that $kk_{\mathrm{I}} \neq 0$, its numerator contains the forward difference operator $\Delta = q - 1$. From the definition of adaptivity order, it then follows that every stable PID controller with $kk_{\mathrm{I}} > 0$ is first order adaptive. Hence, integral action is necessary in order to have first order adaptivity. By (35), this condition is equivalent to $\beta_1 + \beta_2 + \beta_3 > 0$, which implies the (fully general) condition $P(1) \neq 0$ already imposed on $C(q)$ in Definition 2.5.

By (22) and (35), $H311$ PID controllers are given by the two-parameter family

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{3k_{\mathrm{I}}/4+k_{\mathrm{P}}/2} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{k_{\mathrm{I}}/2} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{-(k_{\mathrm{I}}/4+k_{\mathrm{P}}/2)} h_n, \tag{37}$$

while $H312$ PID controllers are given by the one-parameter family

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{k_{\mathrm{I}}/4} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{k_{\mathrm{I}}/2} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{k_{\mathrm{I}}/4} h_n. \tag{38}$$

Good parameterizations are found by studying the stability region in the $(kk_{\mathrm{I}}, kk_{\mathrm{P}})$ plane, see the left diagram of Figure 8. For the $H312$ PID controller, the closed loop is stable for $kk_{\mathrm{I}} \in (0, 4/3)$. Useful controllers are however found in a much smaller interval near $kk_{\mathrm{I}} = 0.2$, and we recommend the particular value $kk_{\mathrm{I}} = 2/9$, which effectively minimizes the magnitude of the poles. Due to its second order stepsize filter, this controller has a high ability to quench $(-1)^n$ oscillations, as demonstrated in Figure 9. The repeated averaging is clearly recognized in the coefficients of (38), and in view of (2) the naive parameter choice would be $kk_{\mathrm{I}} = 1$. But the poles would then be complex and exceed 0.9 in magnitude. A simple time domain simulation would quickly put such a controller

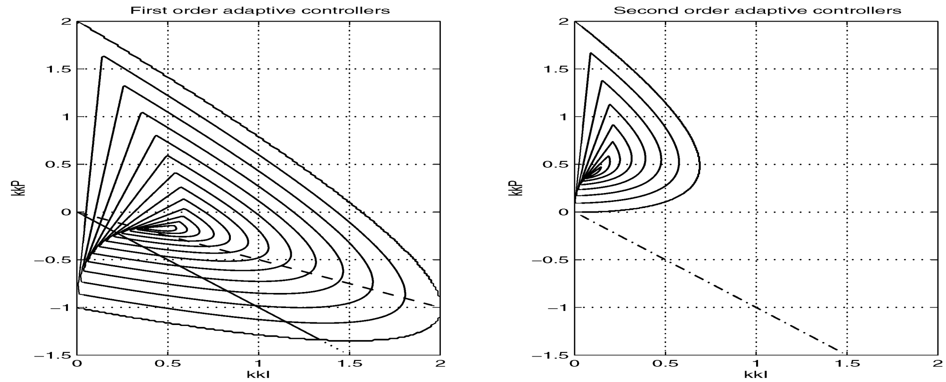

Fig. 8. *Closed loop stability regions* in the $(kk_{\mathrm{I}}, kk_{\mathrm{P}})$ plane for first order adaptive $H311$ PID (left) and second-order adaptive $H321$ predictive PID controllers (right). Level curves enclose stable controllers with closed-loop poles of maximum magnitude $1, 0.95, 0.9, \ldots$. $H211$ controllers (left) that are PI controllers (i.e., $kk_{\mathrm{D}} = 0$) are found on the dashed straight line. Stable $H312$ PID controllers (left) are found on the solid straight line corresponding to $p_F = 2$. Stable $H322$ controllers do not exist, however (right), as the $p_F = 2$ line (dash-dot) does not intersect the stability region. For closed-loop stability, as well as for adaptivity, $kk_{\mathrm{I}} > 0$ is always necessary.

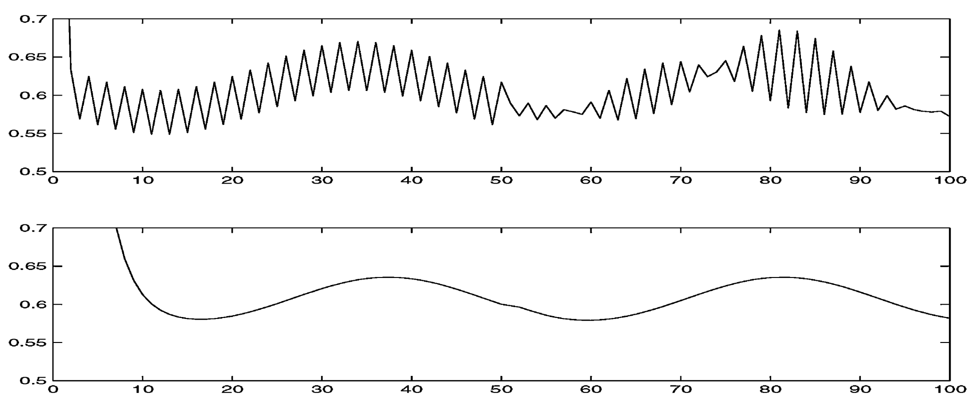

Fig. 9. *Closed loop stepsize sequences* show a 100-step simulation when $\log \hat{\varphi}$ consists of an initial jump and a smooth sinusoidal component onto which a $(-1)^n$ oscillation has been superimposed. From step 50 onwards, this oscillation is amplitude modulated by a quasi-periodic signal. The top graph shows stepsize output from elementary deadbeat control (2); lower graph shows output from an $H312$ PID controller with $kk_{\mathrm{I}} = 2/9$: the second-order filter has completely removed $(-1)^n$ oscillations, but the sudden onset of amplitude modulation causes a barely visible kink after step 50. The controller also shows a slight phase lag.

out of practical use, and the idea of using repeated averaging would fall in disrepute. The less conventional starting point of digital filter theory is necessary to find a proper parameterization.

Among $H311$ PID controllers, there are also $H211$ PI controllers, see Figure 8. The choice $kk_{\mathrm{I}} = 1/3$, $kk_{\mathrm{P}} = -1/6$ produces nearly minimal poles located at $q = 1/2$ and $q = 1/3$. The corresponding parameterization $k\beta_1 = k\beta_2 = 1/6$ is located just above the caustic line in Figure 5. This controller's stepsize low-pass filtering is slightly stronger than for the $H211b$ with $b = 4$.

6.2 Second-Order Adaptivity

Second order adaptivity requires a prediction of the evolution of $\{\log \varphi_n\}$. Predictive controllers based on a PI control structure were considered in Gustafsson [1994] and reviewed in Söderlind [2002]. Here, we extend that approach to predictive controllers based on a PID control structure and introduce

$$C^{\mathrm{PC}}(q) = \frac{1}{q-1}\left(k_{\mathrm{I}}\,\frac{q}{q-1} + k_{\mathrm{P}} + k_{\mathrm{D}}\,\frac{q-1}{q}\right), \tag{39}$$

with the necessary double integral action included. This leads to the difference equation $\log h = C^{\mathrm{PC}}(q) \cdot (\log \varepsilon - \log \hat{r})$, corresponding to the stepsize recursion

$$\Delta(\nabla \log h) = (k_{\mathrm{I}} + k_{\mathrm{P}}\nabla + k_{\mathrm{D}}\nabla^2) \cdot (\log \varepsilon - \log \hat{r}), \tag{40}$$

which is a PID controller for the *stepsize ratio* $\nabla \log h_n = \log(h_n/h_{n-1})$, cf. (33). The predictive PID controller can therefore be written

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{k_{\mathrm{I}}+k_{\mathrm{P}}+k_{\mathrm{D}}} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{-(k_{\mathrm{P}}+2k_{\mathrm{D}})} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{k_{\mathrm{D}}} \frac{h_n}{h_{n-1}}\, h_n. \tag{41}$$

For convenience, we keep the parameter notation unchanged although there are major differences compared to the conventional PID controller; there is now a double integral, a single integral, and a proportional part, but no derivative part. The predictive PID controller still has third-order dynamics. As a counterpart to the case of PID control, one finds that the error transfer function contains the second order difference operator $\Delta^2 = (q-1)^2$, provided that $kk_{\mathrm{I}} \neq 1$. Thus, every stable predicitive PID controller with $kk_{\mathrm{I}} > 0$ is second order adaptive.

As the free parameters enter (34) and (41) in exactly the same way, conditions for first and second order filters at $q = -1$ remain unchanged. The stability region in the $(kk_{\mathrm{I}}, kk_{\mathrm{P}})$ plane is shown in the right diagram of Figure 8. We note in particular that the only intersection between the stability region and the $p_F = 2$ line is $kk_{\mathrm{I}} = kk_{\mathrm{P}} = 0$, that is, for predictive PID controllers, stable second-order filters at $q = -1$ do not exist; the class $H322$ is empty. Only first-order stable filters can be found. The $H321$ predictive PID controllers form a two-parameter family

$$h_{n+1} = \left(\frac{\varepsilon}{\hat{r}_n}\right)^{3k_{\mathrm{I}}/4+k_{\mathrm{P}}/2} \left(\frac{\varepsilon}{\hat{r}_{n-1}}\right)^{k_{\mathrm{I}}/2} \left(\frac{\varepsilon}{\hat{r}_{n-2}}\right)^{-(k_{\mathrm{I}}/4+k_{\mathrm{P}}/2)} \frac{h_n}{h_{n-1}}\, h_n. \tag{42}$$

A good parameterization is $(kk_{\mathrm{I}}, kk_{\mathrm{P}}) = (0.1, 0.45)$, but as the maximum magnitude of the poles is 0.7325, the response of this controller is somewhat slower than of the $H321$ controller of Section 5.3; the frequency responses are however very similar.

7. TIME DOMAIN SIMULATIONS

Time domain simulations are important as a complement to the theoretical investigations of a controller's properties. In particular, it is necessary to verify that the special properties are also observed in practice. In order to compare controllers, it is also necessary to arrange *reproducible simulations* that provide different controllers with *exactly the same input or computational situation*. (This is not possible if the controllers are tested inside an ODE solver.)

The time domain simulations have therefore been arranged as follows: The external disturbance sequence $\log \hat{\varphi}$ is modeled as composed of a deterministic part, the *signal*, and a superimposed, additive *noise*. The signal is assumed to be a continuous function $\log \psi(t)$, but the noise is modeled as "events" related only to the step number. The noise consists of a $N(0, 1)$ sequence, a sequence of random numbers with rectangular distribution in $[-1, 1]$, normalized to have unit standard deviation, and finally a $(-1)^n$ oscillation. These three noise components are added in the proportions $4 : 2 : 1$ to create the noise sequence $\{\log \nu_n\}$. A number of such sequences have been recorded and some have also been processed by a convolution filter to change their spectral properties.

In this way, a large variety of $\log \hat{\varphi}$ sequences have been generated, for selected amplitudes A, as $\log \hat{\varphi}_n = \log \psi(t_n) + A \cdot \log \nu_n$ for $t_n = t_{n-1} + h_n$, where $\{h_n\}$ *is the actual stepsize output generated by the individual controllers*. It is then possible to discern whether different controllers proceed through the time-stepping at different rates. If necessary, this construction also makes it possible to generate identical data sequences by sampling the signal $\log \psi(t)$ at exactly the same points even when the controllers generate different stepsize sequences.

Naturally, only a few simulations can be reported here, and we have chosen to focus on comparative tests of the controllers, in particular as regards enhanced stepsize sequence smoothness, whether the price in terms of increased low-frequency control errors is acceptable, and whether the different controllers on average use equally large stepsizes. For this purpose, we have chosen to use only one signal, which, at times, forces fairly quick stepsize changes, for all tests. We have also selected a single noise sequence, but its amplitude varies depending on the class of controllers and their ability to regularize the stepsize sequence. Startup strategies and stepsize rejections have not been included; the latter is particularly important as we want to study the control errors and how close to the setpoint the different controllers are able to stay.

The stepsize is plotted as a function of the step number; the graphs end prematurely as the simulation only covers the necessary number of steps to reach from $t = 0$ to $t = 55$. This verifies that the different controllers are equally competitive in terms of average stepsize.

The graphs shown in Figures 10–12 confirm that controllers based on stepsize low-pass filters have a significant noise suppression and regularizes the stepsize sequence. The price is an increased control error, in particular as regards low-frequency content. This implies that it may be necessary to use different values of the safety factor θ in the setpoint $\varepsilon = \theta \cdot \text{TOL}$ to prevent frequent step rejections in practical use. As the required head-room in practice depends on the noise level as well, we consider this factor to be part of the controller choice, when a special class of problems such as, for example, stochastic ODEs is approached.

The controllers with strong low-pass filtering show larger control errors. Even with a pure signal without noise, the high-order filters will exhibit a setpoint deviation incurred by the signal alone. This indicates that for a smooth,

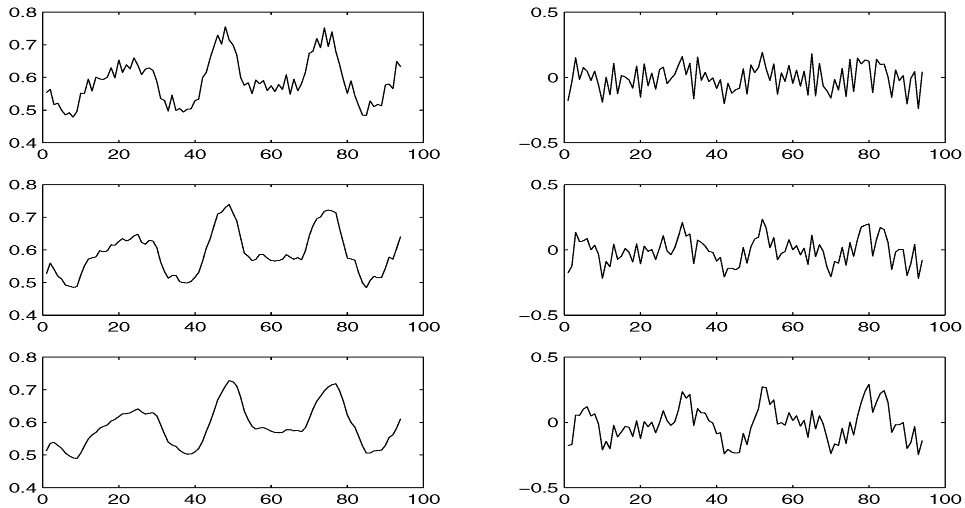

Fig. 10. *Time domain simulation with first-order adaptive controllers.* Stepsize outputs h (left) and error sequences $\log(\hat{r}/\varepsilon)$ (right) are plotted vs. step number for $t \in [0, 55]$, when $\log\hat{\varphi}$ is a smooth, varying signal with additive noise. All controllers take 94 steps to reach $t = 55$. From top to bottom: elementary H_0110 deadbeat; H_0211 deadbeat; $H211b$ with $b = 4$. The successively improved stepsize smoothing is evident. The error is also smoother but low frequencies have an increased amplitude. As predicted by error frequency response graphs, the $H211b$ has the largest error deviations from the setpoint $\log\varepsilon$ (right).

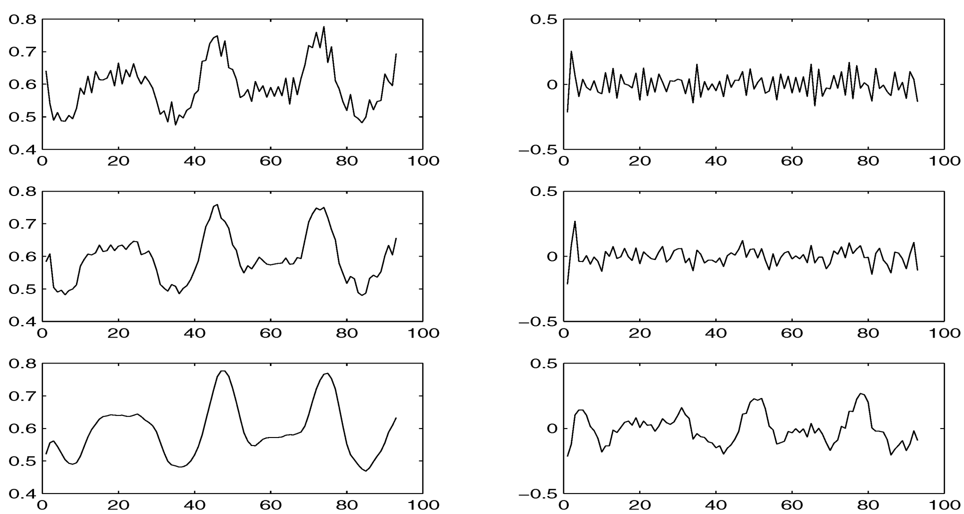

Fig. 11. *Time domain simulation with second-order adaptive controllers.* Stepsize outputs h (left) and error sequences $\log(\hat{r}/\varepsilon)$ (right) vs. step number for $t \in [0, 55]$, when $\log\hat{\varphi}$ is a smooth, varying signal with additive noise. All controllers take 93 steps to reach $t = 55$. From top to bottom: H_0220 deadbeat; H_0321 deadbeat; new $H321$ controller with poles at $q = 1/3, 1/2, 2/3$. The setup is identical to that in Figure 10, except that the amplitude of the input noise sequence has been reduced by a factor of 2 as H_0220 has roughly twice the noise amplification of H_0110. For the deadbeat controllers the second order adaptivity implies that control errors are kept close to the setpoint, but increasing low-pass filtering increases control errors, and the nondeadbeat $H321$ has +15 dB (a factor of 6) more low-frequency control error than H_0321, see also Figure 3.

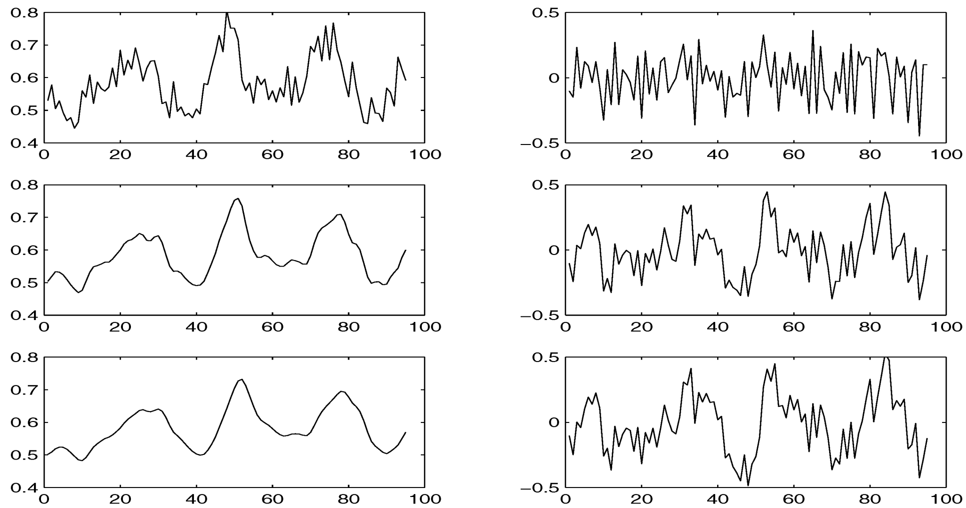

Fig. 12. *Time domain simulation with first-order adaptive controllers*. Stepsize outputs h (left) and error sequences $\log(\hat{r}/\varepsilon)$ (right) vs. step number for $t \in [0, 55]$, when $\log \hat{\varphi}$ is a smooth, varying signal with additive noise. All controllers take 95 steps to reach $t = 55$. From top to bottom: elementary $H_0 110$ deadbeat; $H312b$ with $b = 8$; $H312$ PID control with $kk_{\mathrm{I}} = 2/9$. The setup is the same as in the previous experiments but noise amplitude has been doubled as the two lower controllers have second order stepsize low-pass filters. The control error is fairly large as the strongly filtering controllers also smooth the signal's turns and corners. Even if the spectral content of the control errors is entirely different compared with $H_0 110$, the peak-to-peak control error amplitude is still fairly moderate in simulations employing stepsize low-pass filtering.

noise-free problem, one should at most use moderate low-pass filtering. For substantial noise levels, however, the objective of adaptive time-stepping is to extract signal trends, and strong filtering may be required.

8. ERROR FILTERING AND STEP REJECTION

Although the peak-to-peak amplitude of low-frequency control errors displays a fairly moderate increase, a reorganization of the order in which filtering and control is applied may have a significant impact on the decision of whether a step can be accepted or not. In particular, this concerns the choice of the safety factor θ in the setpoint $\varepsilon = \theta \cdot \text{TOL}$. We have also seen that stepsize low-pass filtering precludes error low-pass filtering, but the error sequence can nevertheless be affected. In the former case, we may use control error filtering, and in the latter error sequence filtering.

8.1 Control Error Filtering

By using the factorization

$$C(q) = \frac{1}{q-1}\,\frac{P(q)}{Q(q)} \tag{43}$$

of the general controller we can split its action into a filter part $P(q)/Q(q)$ and a single integral control action $1/(q-1)$. We then write the control in the form

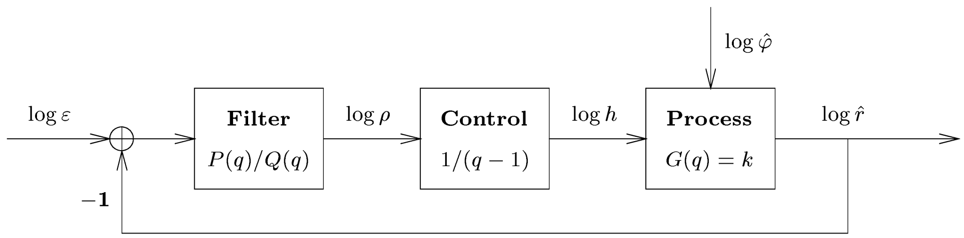

Fig. 13. *Control error filtering*. The controller $C(q) = P(q)/Q(q)/(q-1)$ is split into a filter $P(q)/Q(q)$ and a single integrating controller $1/(q-1)$. The filter is applied to the control error before the summation operator corrects the stepsize $\log h$. The filtered control error $\log\rho$ is significantly smaller and smoother than $\log\varepsilon - \log\hat{r}$. Step rejection is then based on the controller's suggested stepsize change (i.e., the stepsize ratio ρ) rather than on control error magnitude. Overall stepsize, error and filter characteristics are unaffected.

$h_{n+1} = \rho_n h_n$ with

$$\rho_n = (\varepsilon/\hat{r}_n)^{\beta_1}\, (\varepsilon/\hat{r}_{n-1})^{\beta_2}\, (\varepsilon/\hat{r}_{n-2})^{\beta_3}\, \rho_{n-1}^{-\alpha_2}\, \rho_{n-2}^{-\alpha_3}, \tag{44}$$

where $\log\rho$ is considered to be the *filtered control error*, see Figure 13. Although (44) is equivalent to (18), the difference is that by using $\log\rho$ to test whether the step should be rejected or not, it is possible to significantly reduce the risk of rejections caused by high-frequency content in the error; due to the filtering, $\log\rho$ is considerably smoother and smaller than $\log\varepsilon - \log\hat{r}$ as high-frequency noise has been removed, see the time domain simulation in Figure 14. The test for rejection then becomes a matter of whether a proposed stepsize change ρ_n, given the method order k, can be considered normal. This may be preferable to basing a rejection decision on a noise contaminated error—recall that if stepsize filtering is employed, the error $\log\hat{r}$ accommodates the major part of the noise.

8.2 Error Sequence Filtering

Another reason to modify the rejection criterion is to better reflect the propagation of global errors. For nonstiff error components, the simplest global error propagation model is $e_{n+1} = (1 + h_n\mu)e_n + r_n$, or, for constant steps,

$$qe = (1 + h\mu)e + r \Rightarrow e = \frac{r}{q - (1 + h\mu)}\,. \tag{45}$$

For $|h\mu| \ll 1$ or if $h\mu$ is small and negative, but not negligible, the recursion acts as a convolution filter that attenuates high frequency noise in the error; frequency response is similar to a first order integrating controller's response, which is just the inverse (the negative) of the H_0110 error frequency response, see the right graph in Figure 2. Hence, lower frequencies dominate the global error (which is unbounded for $\omega = 0$ when $h\mu = 0$), and the higher the frequency, the stronger is the attenuation in the map $r \mapsto e$. This also holds for stiff computations if an L-stable method is used. In other words, *high frequency content in the local errors can be expected to have a relatively minor effect on the global error*, and a step-rejection decision should rather be based on an error from

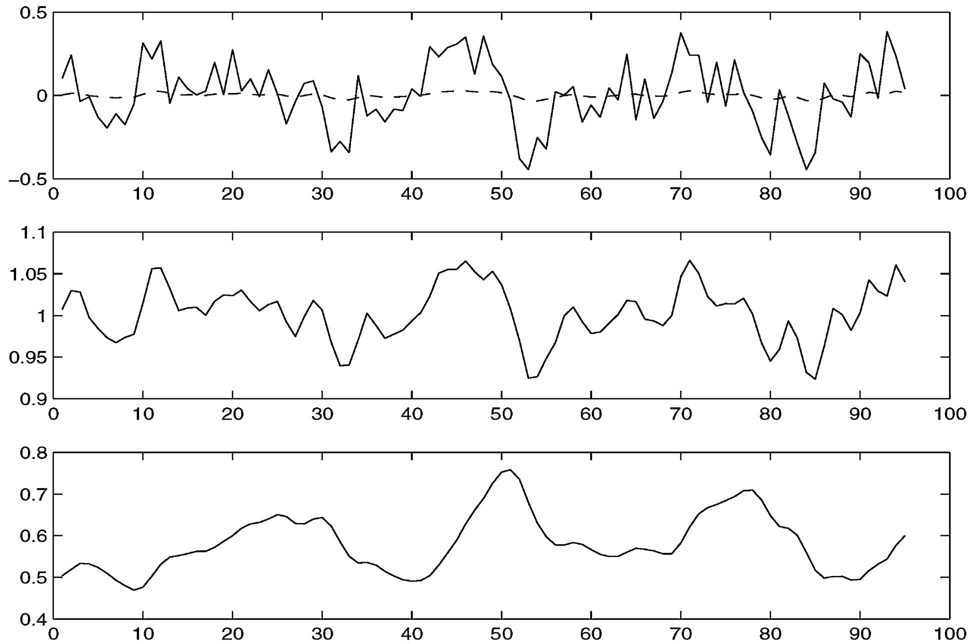

Fig. 14. *Control error filtering.* This time domain simulation, using $H312b$ with $b = 8$ and $k = 4$, is identical to that in the middle graph of Figure 12. The top graph shows unfiltered control error sequence $\log(\varepsilon/\hat{r})$ (solid) and filtered control error sequence $\log\rho$ (dashed) vs. step number. The middle graph shows stepsize ratios ρ. In spite of considerable noise and the strong correlation with $\log(\varepsilon/\hat{r})$, stepsize changes rarely exceed $\pm 5\%$. The lower graph shows stepsize output.

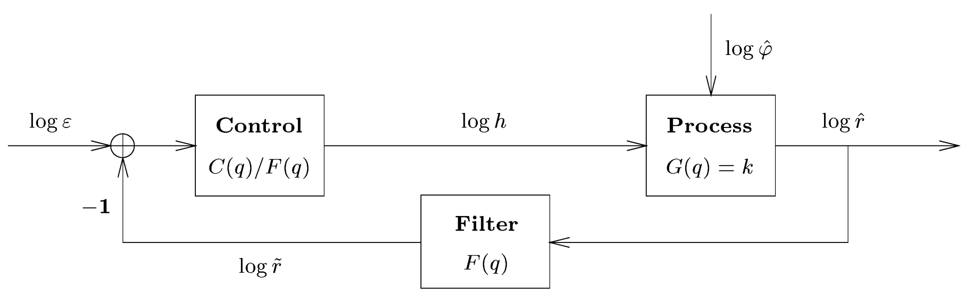

Fig. 15. *Error sequence filtering*. The controller $C(q)$ is split into a filter $F(q)$ with $F(1) = 1$ and the remaining controller $C(q)/F(q)$, by putting the averaging part of the stepsize transfer function into $F(q)$, for example, $(q+1)/(2q)$ for the $H321$ controller. The filter is applied to the error estimate, producing $\log\tilde{r} = F(q)\log\hat{r}$ before correcting the stepsize $\log h$. The filtered control error $\log\varepsilon - \log\tilde{r}$, on which step rejection is based, is smoother than $\log\varepsilon - \log\hat{r}$ without affecting overall stepsize, error and filter characteristics.

which high-frequency content has been removed. It is therefore worthwhile to consider *error sequence filtering* as shown in Figure 15, as an alternative to the straightforward implementation of filter-based controllers. Error sequence filtering has a rather small effect apart from removing top frequencies from $\log\tilde{r}$, see Figure 16. This implies that it can be considered to be a standard way of implementing filter based controllers.

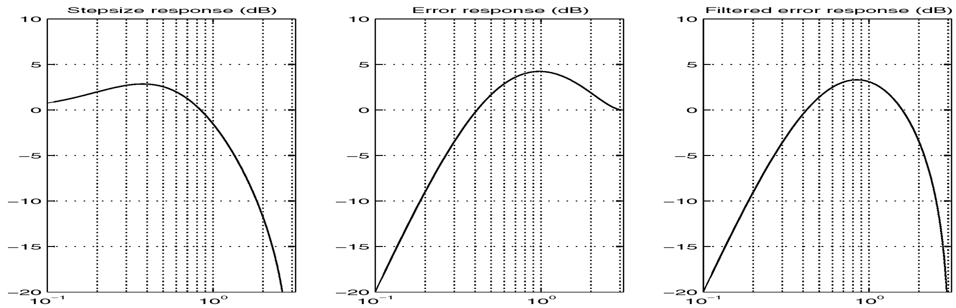

Fig. 16. *Error sequence filtering in $H321$ controller.* Stepsize (left), error (center) and filtered error frequency response (right) for $H321$, with $F(q) = (q+1)/(2q)$, show that the filtered error $\log \tilde{r}$ is similar to $\log \hat{r}$ except that top frequencies are removed to create a smoother control error.

Table III. Recommended Controllers with Stepsize Low-Pass Filters and their Problem Classes

$k\beta_1$	$k\beta_2$	$k\beta_3$	α_2	α_3	Class	Problem type
1/2	1/2		1/2		H_0211	smooth to medium
$1/b$	$1/b$		$1/b$		$H211b$	medium to nonsmooth
1/6	1/6				$H211$ PI	medium to nonsmooth
1/4	1/2	1/4	3/4	1/4	H_0312	medium
$1/b$	$2/b$	$1/b$	$3/b$	$1/b$	$H312b$	nonsmooth
1/18	1/9	1/18			$H312$ PID	nonsmooth
5/4	1/2	−3/4	−1/4	−3/4	H_0321	smooth
1/3	1/18	−5/18	−5/6	−1/6	$H321$	medium

9. CONCLUSIONS

The single assumption on the computational process is that the stepsize—error relation of a time discretization method is accurately described by the asymptotic model $\hat{r}_n = \hat{\varphi}_n h_n^k$. Using elementary digital filter theory, the article has shown how to construct stepsize and error filter associated with a general control structure that covers all linear controllers of third-order dynamics.

Order conditions for adaptivity, stepsize and error low-pass filtering are given, and the proper parameterization is studied with respect to stability, frequency response, regularization and time domain simulations. The simulations verify that the controllers with stepsize low-pass filters generate much smoother stepsize sequences. The controllers are simple, and do not incur extra computational costs, neither in themselves nor in their control performance, as they all use stepsizes of the same average magnitude.

Table III presents a number of new controllers based on this theory, and the classes of problems for which they can be expected to do well. The terms "smooth," "medium" and "nonsmooth" are used in a relative sense to indicate which new controller to select if stepsize sequences are nonsmooth or if control errors appear too large. A more precise definition of problem properties would require a detailed study of noise power spectra.

For a full implementation of the above controllers, several things need to be considered. First, as they are third-order dynamical systems, the process cannot start with back data missing. This implies that a *starting procedure* is needed, just like for PI controllers [Gustafsson 1991]. Another need for such a procedure is after repeated rejected steps; the asymptotic model might then no longer hold, and the present state of the controller is of little value, implying that back data will have to be discarded. In addition, at large discontinuous input, the controllers may react with large transients. This implies that *safety nets* in terms of logic are needed; this should also take care of situations where drastic stepsize reductions are called for, before the controller can be restarted. A startup should consist of a purely integrating controller, say of gain $k\beta_1 = 0.7$, which is reduced on the following steps until sufficient data are available for the $p_D > 1$ controllers to run on their own.

Second, for increased robustness it is common to employ *limiters* that prevent divide by zero as well as unrestrained stepsize increases or decreases. This is a nonlinearity which can be designed without discontinuities and so that the normal control action is not disturbed. It still incurs a change in the controller's state, which may have to be compensated by *anti windup* [Åström and Wittenmark 1990], to preserve the controller's ability to control the process.

Finally, there is the possibility of implementing the controller using error sequence or control error sequence filtering. In all, a controller for full use in an ODE/DAE/SDE solver is a separate piece of software that should be carefully analyzed and implemented, but individual implementations of the different controllers are not necessary. A general implementation follows the same lines as those indicated by the pseudo codes in Gustafsson [1991, 1994], but details and aspects of implementation will be studied and evaluated elsewhere.

ACKNOWLEDGMENTS

The author is grateful to Prof. John Butcher for arranging an extended visit to the University of Auckland where this article was written. The author would also like to thank the referees for useful comments on the presentation.

REFERENCES

ÅSTRÖM, K. J. AND WITTENMARK, B. 1990. *Computer-Controlled Systems. Theory and design.* 2nd ed. Prentice-Hall, Englewood Cliffs, N.J.

DE SWART, J. J. B. 1997. Parallel software for implicit differential equations. Ph.D. dissertation, CWI, Amsterdam, The Netherlands.

DE SWART, J. J. B. AND SÖDERLIND, G. 1997. On the construction of error estimators for implicit Runge–Kutta methods. *J. Comp. Appl. Math. 86*, 347–358.

GEAR, C. W. 1971. *Numerical Initial Value Problems in Ordinary Differential Equations.* Prentice-Hall, Englewood Cliffs.

GUSTAFSSON, K. 1991. Control theoretic techniques for stepsize selection in explicit Runge–Kutta methods. *ACM Trans. Math. Softw. 17*, 533–554.

GUSTAFSSON, K. 1994. Control theoretic techniques for stepsize selection in implicit Runge–Kutta methods. *ACM Trans. Math. Softw. 20*, 496–517.

GUSTAFSSON, K., LUNDH, M., AND SÖDERLIND, G. 1988. A PI stepsize control for the numerical solution of ordinary differential equations. *BIT 28*, 270–287.

GUSTAFSSON, K. AND SÖDERLIND, G. 1997. Control strategies for the iterative solution of nonlinear equations in ODE solvers. *SIAM J. Sci. Comput. 18*, 23–40.

HAIRER, E., NØRSETT, S. P., AND WANNER, G. 1993. *Solving Ordinary Differential Equations. I: Nonstiff Problems*, 2nd revised edition. Springer-Verlag, Berlin, Germany.

HAIRER, E. AND WANNER, G. 1996. *Solving Ordinary Differential Equations II: Stiff and Differential-algebraic Problems*, 2nd revised edition. Springer-Verlag, Berlin, Germany.

HALL, G. 1985. Equilibrium states of Runge–Kutta schemes. *ACM Trans. Math. Softw. 11*, 289–301.

HALL, G. 1986. Equilibrium states of Runge–Kutta schemes, part II. *ACM Trans. Math. Softw. 12*, 183–192.

HALL, G. AND HIGHAM, D. 1988. Analysis of stepsize selection schemes for Runge–Kutta codes. *IMA J. Num. Anal. 8*, 305–310.

HIGHAM, D. AND HALL, G. 1990. Embedded Runge–Kutta formulae with stable equilibrium states. *J. Comput. Appl. Math. 29*, 25–33.

SÖDERLIND, G. 2002. Automatic control and adaptive time-stepping. *Numer. Alg. 31*, 281–310.

WATTS, H. A. 1984. Step size control in ordinary differential equation solvers. *Trans. Soc. Comput. Sim. 1*, 15–25.

ZONNEVELD, J. A. 1964. Automatic numerical integration. Ph.D. dissertation. Math. Centre Tracts 8. CWI, Amsterdam, The Netherlands.

Received March 2001; revised April 2002 and September 2002; accepted September 2002