ELSEVIER

Signal Processing 81 (2001) 1865–1877

SIGNAL PROCESSING

www.elsevier.com/locate/sigpro

High-performance conversions between continuous- and discrete-time systems

Qing-Guo Wang[a,*], Qiang Bi[b], Xue-Ping Yang[a]

[a] *Department of Electrical & Computer Engineering, National University of Singapore, 10 Kent Ridge Crescent, 119260 Singapore*

[b] *Research and Development Center, Seagate Technology International, Singapore Science Park, 118249 Singapore*

Received 18 May 2000; received in revised form 15 January 2001

Abstract

In this paper, a new algorithm for converting an S-domain model to a Z-domain model is proposed. The resultant Z-domain model is much closer to the original S-domain model than that obtained from conventional conversion schemes such as Tustin approximation with or without frequency pre-warping, zero-pole mapping equivalent, and Triangle hold equivalent. The proposed algorithm adopts the standard recursive least squares approach, and it has the following advantages. First, the computation is trivial and can be easily implemented online. Second, it can handle both parametric and non-parametric models, while conventional conversion methods can only handle parametric models. Third, it is very flexible and applicable to Z- to S-domain model conversion, sampling rate selection, re-sampling, time delay system conversion, and model order selection. The superior performance of the method is demonstrated with several simulation examples in this paper.

Keywords: Discretization; Discrete systems; Sampling rate; Modelling; Optimization

1. Introduction

In computer and micro-processor applications such as computer control and digital signal processing (DSP), signals are often manipulated in digital form in order to implement complex algorithms and achieve higher performance. Conversion of a model between continuous S-domain and discrete Z-domain is often encountered in these applications. For example, normally, a filter is designed in a continuous domain and to implement the filter in digital form, converting the S-domain filter into a Z-domain is unavoidable. In digital model-based control implementation, a Z-domain model is also needed. The conversion accuracy is critical and directly affects the final performance of the system.

Quite a number of approaches of converting a model between S- and Z-domain have been reported. Commonly used methods are Tustin approximation with or without frequency pre-warping, zero-pole

* Corresponding author. Tel.: +65-874-2282; fax: +65-779-1103.
E-mail address: elewqg@nus.edu.sg (Q.-G. Wang).

0165-1684/01/$ - see front matter
PII: S0165-1684(01)00065-2

mapping equivalent, Triangle hold equivalent, etc. [5,7,10]. Tustin (or Bilinear) approximation is one of the most popular methods, and is given in [17]. This method uses the approximation

$$z = e^{sT_s} \approx \frac{1 + sT_s/2}{1 - sT_s/2} \tag{1}$$

to relate S- and Z-domain transfer functions, where T_s is the sampling interval of the discrete system.

These common conventional methods are straightforward and easy to use. The frequency response (e.g., bode plot) of the equivalent digital system obtained by these methods is quite close to that of the original continuous system when the sampling frequency is sufficiently high in regarding the bandwidth of the continuous system [11]. Serious performance degradation may be observed when the sampling frequency is low. In real life, the sampling frequency of a digital system is often limited by a micro-processor bandwidth, costs, etc. The resultant performance of a low sampling frequency digital system may not be guaranteed if the conventional conversion methods are used [9]. Moreover, these methods can only perform a conversion between a parametric model in the S-domain or in the Z-domain. For a non-parametric model, such as bode plot data measured by a Dynamic Signal Analyzer (made by HP for example) using sine sweep, which is common in industry, these methods cannot be used directly. Usually, the non-parametric model will be converted into an S-domain parametric model and then these methods will be applied to get Z-domain model for digital implementation. Accuracy may be lost during this conversion. Besides, when a system contains delay, then the conversion will be quite complex using classical methods.

Recently, some special approaches to the solution were presented in [12–14,18]. And, optimization based discretization has been widely studied [2,3,6,9,16]. However, the algorithms proposed in these papers are either too complicated or state-space based which are only applicable for delay-free cases. Thus, the conventional methods are still widely used in the industry. This paper uses a new frequency domain recursive least square (RLS) based method to solve the model conversion problem. The objective of this method is to obtain the discrete transfer function (DTF) so that its frequency response fits that of the continuous transfer function (CTF) to be converted. A DTF model can be obtained either from a CTF or directly from the system frequency response such as bode plot data. It produces a very good result even when sampling frequency is low. The method can easily take pure delay into account. It is also applicable to converting discrete-time models into continuous time models, and resampling a discrete-time model to another sampling frequency, which is sometimes needed in a digital design. Simulation results show that the accuracy of the conversion is much better than these conventional methods.

This paper is organized as follows. The proposed method is presented in Section 2. Simulation results for converting CTF into DTF, and some extension applications are shown in Sections 3 and 4, respectively. Finally, some concluding remarks are drawn in Section 5.

2. The proposed method

Consider a CTF:

$$H_c(s) = \frac{b_n s^n + b_{n-1}s^{n-1} + \cdots + b_1 s + b_0}{s^n + a_{n-1}s^{n-1} + \cdots + a_1 s + a_0}, \tag{2}$$

where a_i and b_i are real coefficients. Suppose that an equivalent discrete transfer function (DTF) $H_d(z)$ is given by

$$H_d(z) = \frac{\beta_m z^m + \beta_{m-1}z^{m-1} + \cdots + \beta_1 z + \beta_0}{z^m + \alpha_{m-1}z^{m-1} + \cdots + \alpha_1 z + \alpha_0}. \tag{3}$$

The present work is to propose a new conversion algorithm, which can yield the best DTF approximation to the original CTF regardless of CTF's order and characteristics. The proposed idea is to match the frequency response of $H_d(z)$ to $H_c(s)$ with the general transformation from S- to Z-domain, i.e.,

$$z = e^{T_s s}. \tag{4}$$

The objective can be realized by minimizing the standard loss function

$$\min_{H_d} J \triangleq \min_{H_d} \sum_{i=1}^{M} |H_d(e^{jT_s\omega_i}) - H_c(j\omega_i)|^2, \tag{5}$$

where the interval $[\omega_1, \omega_M]$ defines the frequency range of interest. Eq. (5) falls into the framework of transfer function identification in a frequency domain, in which H_c is the known frequency response, whereas H_d is a parametric transfer function to be identified. For this identification, a number of methods are available [8], and RLS algorithm is suitable for our application and is briefly described as follows.

Substituting Eqs. (2) and (3) into Eq. (5) with $s = j\omega_i$ and $z = e^{jT_s\omega_i}$ gives

$$J^{(k)} \triangleq \sum_{i=1}^{M} |\bar{W}_i^{(k)} \{[\beta_m^{(k)}(e^{jT_s\omega_i})^m + \beta_{m-1}^{(k)}(e^{jT_s\omega_i})^{m-1} + \cdots + \beta_0^{(k)}]$$
$$-H_c(j\omega_i)[(e^{jT_s\omega_i})^m + \alpha_{m-1}^{(k)}(e^{jT_s\omega_i})^{m-1} + \cdots + \alpha_0^{(k)}]\}|^2, \tag{6}$$

where

$$\bar{W}_i^{(k)} \triangleq \frac{W(e^{jT_s\omega_i})}{(e^{jT_s\omega_i})^m + \alpha_{m-1}^{(k-1)}(e^{jT_s\omega_i})^{m-1} + \cdots + \alpha_0^{(k-1)}} \tag{7}$$

acts as a weighting function for the standard least squares problem in braces. Different weighting functions should be employed in the various methods cited in [8], and in this paper, $\bar{W}_i^{(k)}$ is chosen as $1/[(e^{jT_s\omega_i})^m + \alpha_{m-1}^{(k-1)}(e^{jT_s\omega_i})^{m-1} + \cdots + \alpha_0^{(k-1)}]$ [15]. We re-arrange Eq. (6) to yield

$$J^{(k)} \triangleq \sum_{i=1}^{M} |\eta_i^{(k)} - \phi_i^{(k)T}\theta^{(k)}|^2,$$

where

$$\eta_i^{(k)} = \frac{-H_c(j\omega_i)(e^{jT_s\omega_i})^m}{(e^{jT_s\omega_i})^m + \alpha_{m-1}^{(k-1)}(e^{jT_s\omega_i})^{m-1} + \cdots + \alpha_0^{(k-1)}},$$

$$\theta^{(k)} = [\alpha_{m-1}^{(k)} \cdots \alpha_0^{(k)} \beta_m^{(k)} \beta_{m-1}^{(k)} \cdots \beta_1^{(k)} \beta_0^{(k)}]^T,$$

$$\phi_i^{(k)} = \frac{[H_c(j\omega_i)(e^{jT_s\omega_i})^{m-1} \cdots H_c(j\omega_i) - (e^{jT_s\omega_i})^m - (e^{jT_s\omega_i})^{m-1} \cdots - (e^{jT_s\omega_i}) - 1]^T}{(e^{jT_s\omega_i})^m + \alpha_{m-1}^{(k-1)}(e^{jT_s\omega_i})^{m-1} + \cdots + \alpha_0^{(k-1)}}.$$

Then, we have the RLS algorithm [1,4] for $\theta^{(k)}$ as

$$\theta^{(k,i)} = \theta^{(k,i-1)} + K^{(k,i)}\epsilon^{(k,i)}, \quad i = 1, 2, \ldots, M, \tag{8}$$

where

$$K^{(k,i)} = P^{(k,i-1)}\phi_i^{(k)}(I + \phi_i^{(k)T}P^{(k,i-1)}\phi_i^{(k)})^{-1}, \tag{9}$$

$$P^{(k,i)} = (I - K^{(k,i)}\phi_i^{(k)T})P^{(k,i-1)}, \tag{10}$$

$$\epsilon^{(k,i)} = \eta_i^{(k)} - \phi_i^{(k)T}\theta^{(k,i-1)}. \tag{11}$$

After the above RLS in Eqs. (8)–(11) has been completed, the resultant parameter vector $\theta^{(k)} = \theta^{(k,M)}$ is used to update $\bar{W}_i^{(k)}$:

$$\bar{W}_i^{(k+1)} = \frac{1}{(\mathrm{e}^{\mathrm{j}T_s\omega_i})^m + \alpha_{m-1}^{(k)}(\mathrm{e}^{\mathrm{j}T_s\omega_i})^{m-1} + \cdots + \alpha_0^{(k)}} \tag{12}$$

and Eqs. (8)–(11) are repeated to calculate $\theta^{(k+1)}$. On convergence, the resultant parameter vector will form one solution to Eq. (5).

The frequency range in the optimal fitting plays an important role. In general, our studies suggest that the frequency range $[\omega_1 \ \cdots \ \omega_M]$ be chosen as $[\frac{1}{100}\omega_b, \omega_b]$ with the step of $(\frac{1}{100} \sim \frac{1}{10})\omega_b$, where ω_b is the bandwidth of the S-domain transfer function. In this range, RLS yields satisfactory fitting results in the frequency domain.

The maintenance of stability is a crucial issue in frequency domain based identification. In our study, we found that the RLS method with zero initial parameter value might result in unstable $H_d(z)$, especially for high-order models, even though $H_c(s)$ is stable. Since Eq. (5) is a non-linear problem, the RLS algorithm may get different local optimal solutions, if it starts with different initial points. Among these solutions, only the stable $H_d(z)$ models are required. If we set a stable model as the initial value in the RLS algorithm, it is likely to get a stable approximate as its convergent solution. Thus, in this paper, for a parametric $H_c(s)$, we set the initial model as the Bilinear equivalent via Eq. (1) to $H_c(s)$, for it has been proven that the Bilinear rule maintains stability during continuous and discrete conversions [5]. Our extensive simulations show that this technique works very well.

The proposed method is also applicable to inverse conversion, i.e., from DTF to CTF, provided that the loss function Eq. (5) is replaced by

$$\min_{H_c} J \triangleq \min_{H_c} \sum_{i=1}^{M} |H_c(\mathrm{j}\omega_i) - H_d(\mathrm{e}^{\mathrm{j}T_s\omega_i})|^2. \tag{13}$$

Moreover, the method can be applied to transfer a non-parametric S-domain model such as a bode plot data to Z-domain data.

3. Examples

In this section, the primary application of the proposed method, that is, to convert CTF into DTF with both T_s and m fixed, will be illustrated with some typical plants which are chosen from [5 or 19]. The desired sampling multiple ($\triangleq \omega_s/\omega_b$) for a reasonably smooth time response is chosen [5] to be

$$6 \lesssim \frac{\omega_s}{\omega_b} \lesssim 40. \tag{14}$$

Once a discrete equivalent is found, the following criterion should be used to validate the solution:

$$E = \max_{\omega} \left| \frac{H_d(\mathrm{e}^{\mathrm{j}T_s\omega}) - H_c(\mathrm{j}\omega)}{H_c(\mathrm{j}\omega)} \right|. \tag{15}$$

For a comparative study, the results of the proposed conversion method are compared with the commonly used methods, Tustin, zero-pole mapping transformations or Triangle equivalent.

Example 1. Consider the continuous filter transfer function

$$H_c(s) = \frac{1}{s^3 + 2s^2 + 2s + 1}$$

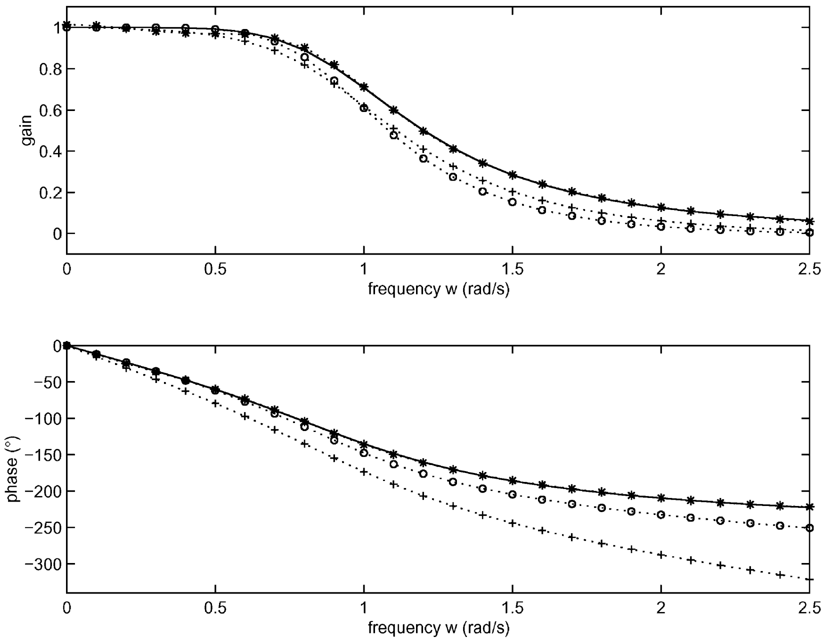


Fig. 1. Digital equivalents to third-order lowpass filter (—— CTF, $\cdots * \cdots$ proposed method, $\cdots \mathrm{o} \cdots$ Tustin, $\cdots + \cdots$ P–Z mapping).

with unity pass bandwidth [5, pp. 141]. For $T_s = 1$ s, i.e., $\omega_s/\omega_b = 2\pi > 6$, as considered in [5], Tustin transformation gives

$$H_d(z) = \frac{0.0476z^3 + 0.1429z^2 + 0.1429z + 0.0476}{z^3 - 1.1900z^2 + 0.7143z - 0.1429}$$

and the zero-pole mapping transformation gives

$$H_d(z) = \frac{0.0920(z+1)^2}{(z - 0.3679)(z^2 - 0.7859z + 0.3679)}.$$

For $m = 3$, the proposed method yields

$$H_d(z) = \frac{0.0080z^3 + 0.2257z^2 + 0.1381z - 0.0072}{z^3 - 1.1583z^2 + 0.6597z - 0.1367}$$

with $E = 0.26\%$. In Fig. 1, the responses of the proposed method, Tustin ($E = 24.67\%$) and zero-pole mapping ($E = 62.84\%$) transformations are compared. We notice in particular that the proposed method yields excellent response. Moreover, Fig. 2 shows that in the reasonable range of ω_s in Eq. (14), the proposed method exhibits a consistent superior approximation performance than Tustin and zero-pole mapping schemes.

Example 2. Consider a double integrator system which is often encountered in a hard disk driver control [5, pp. 152],

$$H_c(s) = \frac{1}{s^2}.$$

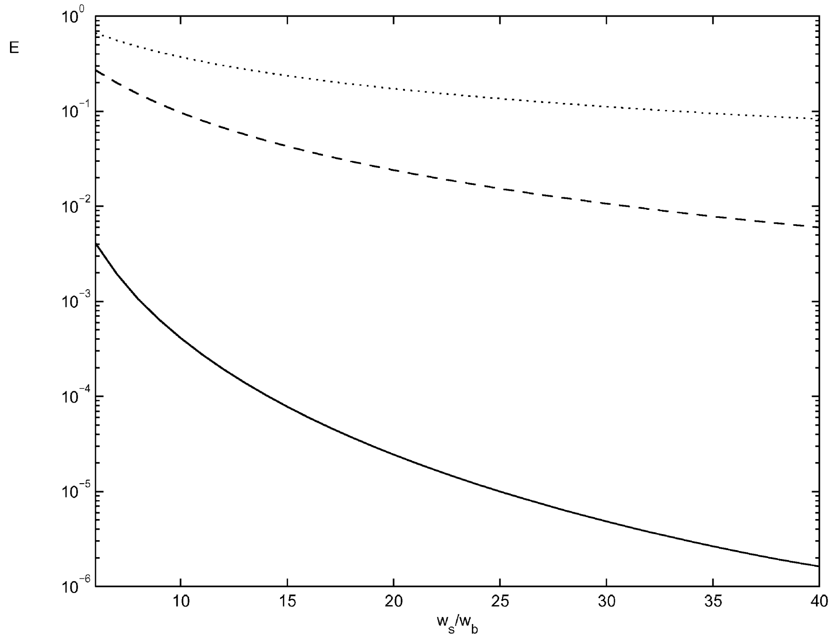


Fig. 2. Approximation error E versus sample rate ω_s (—— proposed method, – · – · – Tustin, ········· P–Z mapping).

Assume that the closed loop bandwidth is $\omega_b = 1$ rad/s. Let $T_s = 1$ s, i.e., $\omega_s/\omega_b = 2\pi > 6$, which fulfills the requirement in Eq. (14). The conversion results derived by Tustin transformation, zero-pole mapping transformation, and the proposed method are tabulated in Table 1.

Example 3. Consider the system with time delay

$$H_c = \frac{s-1}{s^2+4s+5} e^{-0.35s}, \tag{16}$$

with $T_s = 0.1$ s (Control System Toolbox User's Guide, MathWorks, ver. 4.2, 1998, *c2d* function). Here, the input delay in $H_c(s)$ amounts to 3.5 times the sampling period of 0.1 s. Accordingly, the discretized model $H_d(z)$ inherits an input delay of three sampling periods, and the residual half-period delay is factored into the coefficients of $H_d(z)$ by the discretization algorithm. However, the Tustin and matched pole/zero methods are accurate only for delays that are integer multiples of the sampling period [19]. From the conversion results tabulated in Table 1, it can be seen that the proposed design is also superior to Triangle approximation, which is preferred to be used for models with delays among the conventional conversion schemes.

4. Extensions

Besides the primary use of the proposed method, there are some other cases that can be benefited.

4.1. Optimal sampling time

If the user specifies T_s and m, then the proposed RLS algorithm will generate the parameters of $H_d(z)$ as shown in the previous section. On the other hand, if the user has no preference for T_s, our algorithm

Table 1
Summary of simulation results[a]

Example		Results					
		Tustin		Zero-pole mapping		Proposed algorithm	
CTF/DTF	T_s(s)	DTF/CTF	E (%)	DTF/CTF	E (%)	DTF/CTF	E (%)
$\frac{1}{s^3+2s^2+2s+1}$	1	$\frac{0.0476z^3+0.1429z^2+0.1429z+0.0476}{z^3-1.1900z^2+0.7143z-0.1429}$	24.67	$\frac{0.0920(z+1)^2}{(z-0.3679)(z^2-0.7859z+0.3679)}$	62.84	$\frac{0.0080z^3+0.2257z^2+0.1381z-0.0072}{z^3-1.1583z^2+0.6597z-0.1367}$ **	0.26
$\frac{1}{s^2}$	1	$\frac{0.2500z^2+0.5000z+0.2500}{z^2-2.0000z+1.0000}$	16.23	$\frac{0.7941(z+1)}{(z-1)^2}$	79.83	$\frac{0.0916z^2+0.8209z+0.0916}{z^2-2.0000z+1.0000}$	4.42
$\frac{s-1}{s^2+4s+5}e^{-0.35s}$	0.1	$z^{-3}\frac{0.0392z^2-0.0041z-0.0433}{z^2-1.6290z+0.6701}$	29.88	$z^{-3}\frac{0.0115z^3+0.0456z^2-0.0562z-0.0091}{z^3-1.6290z^2+0.6703z}$ *	5.12	$z^{-3}\frac{0.0042z^2+0.0684z-0.0807}{z^2-1.6333z+0.6740}$	0.50
$\frac{1}{s^2+0.02s+1}$	0.1	$\frac{0.0025z^2+0.0050z+0.0025}{z^2-1.9880z+0.9980}$	5.84	$\frac{0.0050z+0.0050}{z^2-1.9880z+0.9980}$	12.03	$\frac{0.0008z^2+0.0083z+0.0008}{z^2-1.9880z+0.9980}$	0.002
$\frac{s^2+0.02s+1}{(s+1)^3}$	0.1	$\frac{0.0433z^3-0.0428z^2-0.0429z+0.0433}{z^3-2.7143z^2+2.4558z-0.7406}$	5.91	$\frac{0.0863z^2-0.1716z+0.0862}{z^3-2.7145z^2+2.4562z-0.7408}$	12.66	$\frac{0.0440z^3-0.0435z^2-0.0437z+0.0440}{z^3-2.7097z^2+2.4466z-0.7361}$	0.1
$\frac{0.1}{s(s+0.1)}$	2	$\frac{0.0909z^2+0.1818z+0.0909}{z^2-1.8180z+0.8182}$	58.70	$\frac{0.4549(z+1)}{(z-1)(z-0.8187)}$	166.88	$\frac{0.0517z^2+0.2987z+0.0399}{z^2-1.8550z+0.8305}$	32.27
	1	$\frac{0.0238z^2+0.0476z+0.0238}{z^2-1.9050z+0.9048}$	16.18	$\frac{0.0762(z+1)}{(z-1)(z-0.9048)}$	81.59	$\frac{0.0090z^2+0.0783z+0.0081}{z^2-1.9049z+0.9049}$	3.00
						$\frac{4.9698z-3.8562}{z+0.1137}$	3.21
$\frac{s+1}{0.1s+1}$	0.25	$\frac{5.0000z-3.8890}{z+0.1111}$	5.60	$\frac{4.1497(z-0.7788)}{z-0.0821}$	14.83	$\frac{5.4842z^2-1.7873z-1.9617}{z^2+0.8632z-0.1015}$	0.01
$\frac{z-0.7}{z-0.5}$	0.05	—	—	$\frac{z-0.8243}{z-0.7071}$	1.88	$\frac{1.0572z-0.8858}{z-0.7143}$	0.003
**	1	$\frac{0.0080s^3+0.0788s^2+0.5099s+0.9860}{s^3+1.9890s^2+1.9880s+0.9862}$	15.64	—	—	$\frac{0.0001s^3-0.0003s^2+0.0005s+0.9991}{s^3+1.9992s^2+1.9993s+0.9993}$	0.56

[a] *Note*: $*$ denotes that the equivalent is generated by Triangle approximation. $**$ denotes the DTF generated by the proposed method for Example 1.

starts with a small sample rate under $m = n$. Based on the sampling theorem: the sample rate must be at least twice the required closed-loop bandwidth, ω_b, of the system, that is, $\omega_s/\omega_b > 2$ [5]. Thus, for an optimal sampling time tuning, we start the design procedure from $\omega_s^0 = \pi\omega_b$, where ω_b can be the bandwidth of a transfer function or closed-loop bandwidth, some specific frequency point which needs a close mapping, or the largest frequency in fitting, i.e., $\omega_b = \max\{\omega_i\}$. If the approximation accuracy is satisfactory, the design is completed; otherwise, we will increase the sample rate with a step of $\pi\omega_b$, i.e.,

$$\omega_s^k = \omega_s^{k-1} + \pi\omega_b \tag{17}$$

till Eq. (18)

$$E = \max_{\omega} \left| \frac{H_d(e^{jT_s\omega}) - H_c(j\omega)}{H_c(j\omega)} \right| \leqslant \varepsilon \tag{18}$$

holds true, where k represents the kth iteration, and ε is the fitting error threshold. ε is specified according to the desired degree of performance, or accuracy of the CTF approximation to the DTF. Usually ε may be set as 3%.

Example 4. Consider a second-order CTF [5, pp. 160]

$$H_c(s) = \frac{0.1}{s(s+0.1)}$$

and the desired closed-loop bandwidth is assumed to be 1 rad/s. We start the conversion operation from $\omega_s^0 = \pi\omega_b = \pi$, i.e., $T_s = 2$ s. The proposed method produces a DTF

$$H_d(z) = \frac{0.0517z^2 + 0.2987z + 0.0399}{z^2 - 1.8550z + 0.8305} \tag{19}$$

with an approximation error $E = 32.27\%$, which cannot fulfill the threshold of 3%, and the frequency response is very poor, as shown in Fig. 3(a), but it is still better than the results of Tustin and zero-pole mapping schemes. Then, ω_s is adjusted to $\omega_s^1 = \omega_s^0 + \pi\omega_b = \pi + \pi \times 1 = 2\pi$ according to the proposed tuning rule Eq. (17). The new ω_s results in

$$H_d(z) = \frac{0.0090z^2 + 0.0783z + 0.0081}{z^2 - 1.9049z + 0.9049}$$

by the proposed method. Fig. 3(b) shows that at $T_s = 1$ s the proposed design can generate a DTF approximating the existing CTF closely, while the results obtained from Tustin and zero-pole mapping transformations do not have good performance.

4.2. Optimal model order

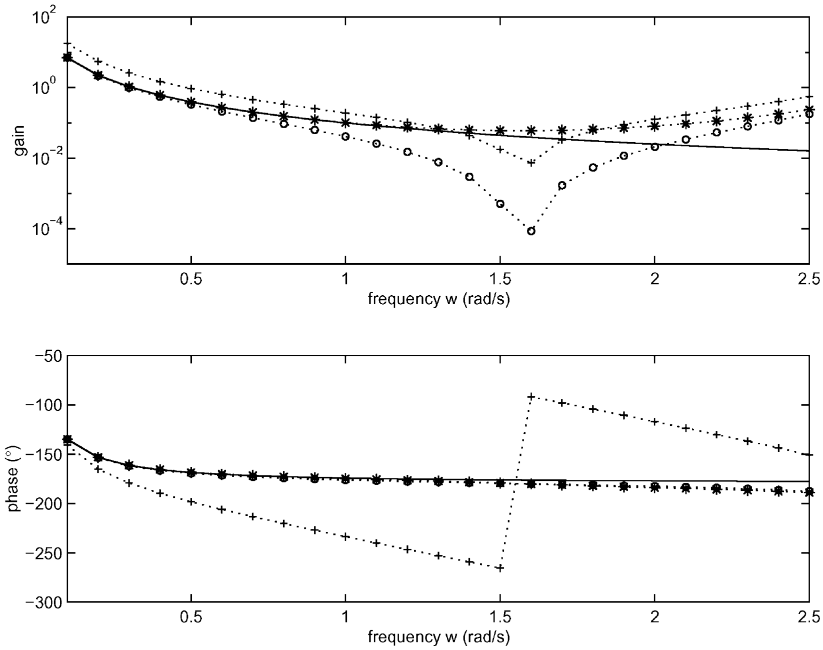
If the sample rate is limited by the hardware, one may gradually increase the complexity of $H_d(z)$ such that the simplest approximation $H_d(z)$ is attained with the guaranteed accuracy to $H_c(s)$ Eq. (18), i.e., the approximation performance may be improved at the expense of an increasing complexity.

Example 5. Consider a lead network

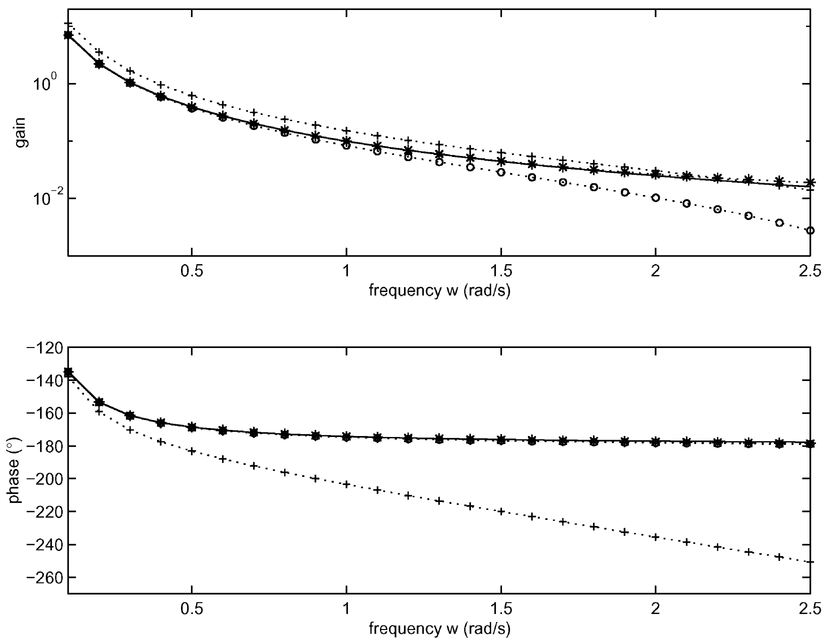
$$H_c(s) = \frac{s+1}{0.1s+1}$$

with $T_s = 0.25$ s, and a close mapping is required at $\omega_b = 3$ rad/s [5, pp. 156]. When $m = 1$, the proposed method generates

$$H_d(z) = \frac{4.9698z - 3.8562}{z + 0.1137}$$



(a) $T_s = 2$ sec



(b) $T_s = 1$ sec

Fig. 3. Tuning of sampling time (—— CTF, $\cdots * \cdots$ proposed method, $\cdots \mathrm{o} \cdots$ Tustin, $\cdots + \cdots$ P–Z mapping).

with $E = 3.21\%$, which cannot fulfill the accuracy threshold. However, as shown in Fig. 4(a), the proposed design is still better than the results obtained from Tustin and zero-pole mapping transformations. On increasing m to $m = 2$, we can get a satisfactory result by using the proposed method

$$H_{\mathrm{d}}(z) = \frac{5.4842z^2 - 1.7873z - 1.9617}{z^2 + 0.8632z - 0.1015}$$

with $E = 0.01\%$. Fig. 4(b) shows that by increasing the order of digital equivalent, we can get a much better performance. However, with zero initial parameters, the second-order DTF obtained is

$$H_{\mathrm{d}}(z) = \frac{5.6125z^2 - 0.4610z - 3.0618}{z^2 + 1.1987z - 0.0915}$$

which is unstable.

4.3. Resampling DTF

For this application, the proposed method does not need the new sampling period T to be an integer multiple of the original sampling period. And it is a direct method to obtain a new DTF by fitting the two equivalents in frequency domain, while in conventional schemes, the resampling assumes zero-order holder on the inputs and is equivalent to consecutive DTF to CTF and CTF to DTF conversions (Control System Toolbox User's Guide, MathWorks, Version 4.2, 1998, *d2d* function).

Example 6. Consider the zero-pole-gain model

$$H_{\mathrm{d}}(z) = \frac{z - 0.7}{z - 0.5}$$

with sample time $T_{\mathrm{s}} = 0.1$ s and resample this model at $T_{\mathrm{s}} = 0.05$ s (Control System Toolbox User's Guide, MathWorks, Version 4.2, 1998, *d2d* function). Thus, zero-pole mapping method, products

$$H_{\mathrm{d}}(z) = \frac{z - 0.8243}{z - 0.7071}$$

with $E = 1.88\%$, while the proposed method generates

$$H_{\mathrm{d}}(z) = \frac{1.0572z - 0.8858}{z - 0.7143}$$

with $E = 0.0026\%$. From Fig. 5, it can be easily seen that the proposed method shows superior performance.

4.4. Conversion of DTF into CTF

In the inverse operation, Eq. (18) should be written as

$$E = \max_{\omega} \left| \frac{H_{\mathrm{d}}(\mathrm{e}^{\mathrm{j}T_{\mathrm{s}}\omega}) - H_{\mathrm{c}}(\mathrm{j}\omega)}{H_{\mathrm{d}}(\mathrm{e}^{\mathrm{j}T_{\mathrm{s}}\omega})} \right| \leqslant \varepsilon \tag{20}$$

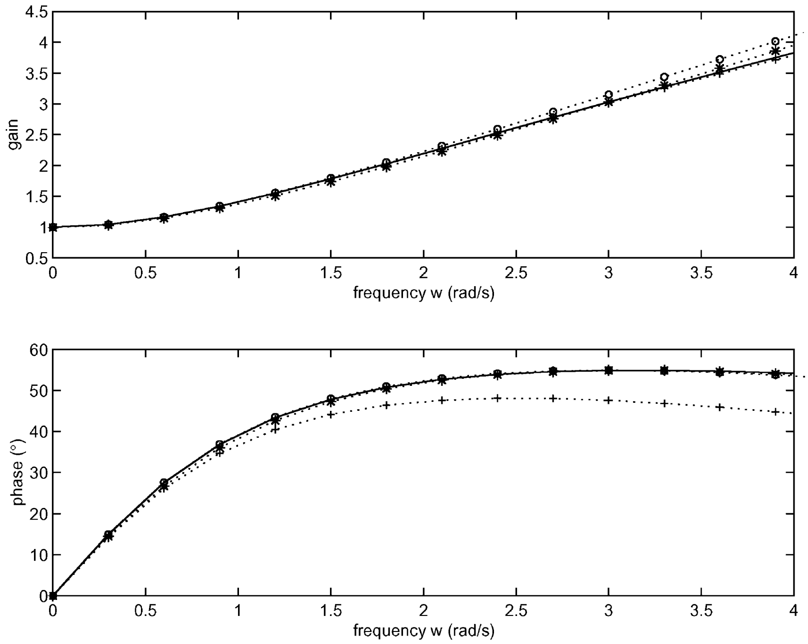
to validate the solution. Here, our method also shows high approximation accuracy.

Example 7. In Example 1, we obtained a DTF,

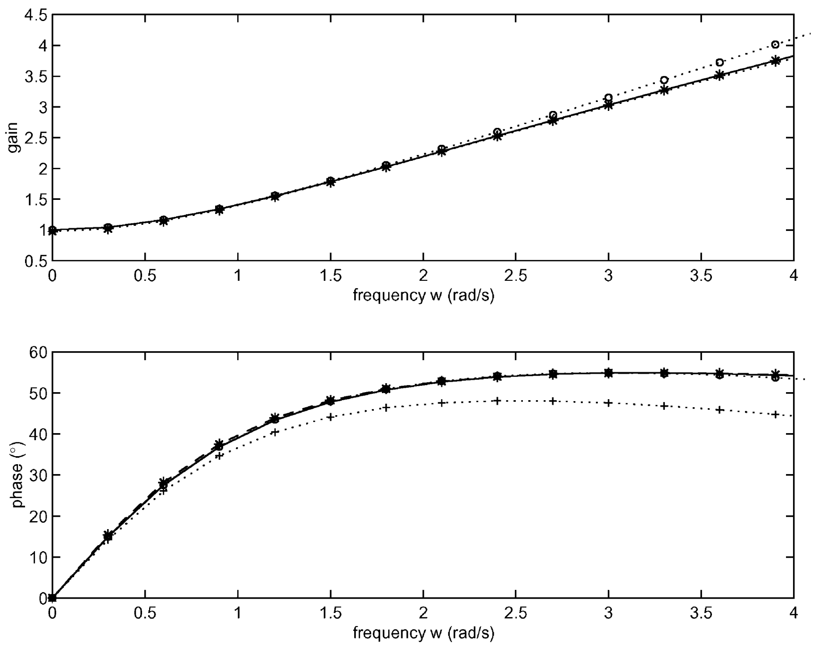
$$H_{\mathrm{d}}(z) = \frac{0.0080z^3 + 0.2257z^2 + 0.1381z - 0.0072}{z^3 - 1.1583z^2 + 0.6597z - 0.1367}$$

with $T_{\mathrm{s}} = 1$ s. To get back the original CTF, Tustin transformation products

$$H_{\mathrm{c}}(s) = \frac{0.0080s^3 + 0.0788s^2 + 0.5099s + 0.9860}{s^3 + 1.9890s^2 + 1.9880s + 0.9862}$$



(a) $m = 1$



(b) $m = 2$

Fig. 4. Tuning of model order (—— CTF, $\cdots * \cdots$ proposed method, $\cdots \text{o} \cdots$ Tustin, $\cdots + \cdots$ P–Z mapping).

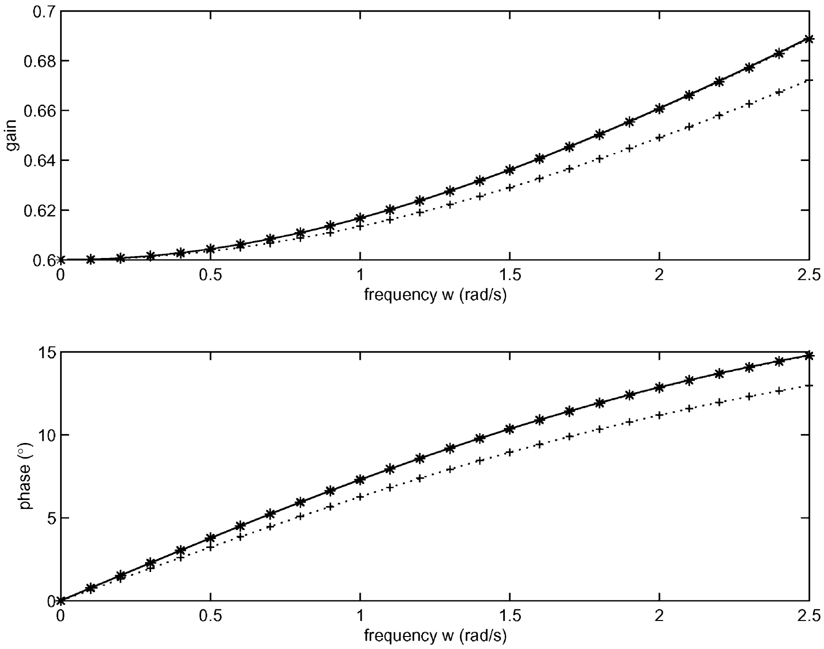


Fig. 5. Resampling digital equivalents (—— original DTF, $\cdots * \cdots$ proposed method, $\cdots + \cdots$ P–Z mapping).

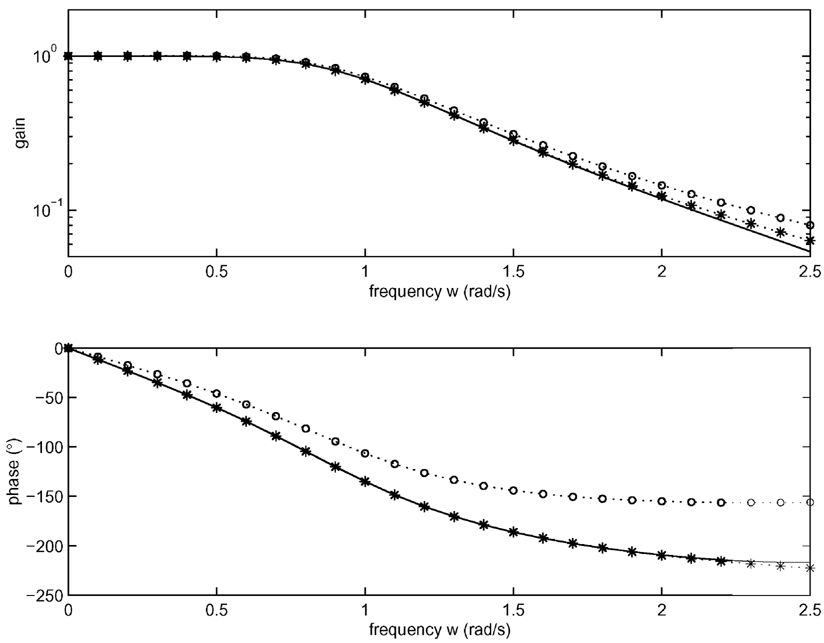


Fig. 6. Continuous equivalents to third-order DTF (—— DTF, $\cdots * \cdots$ proposed method, $\cdots \mathrm{o} \cdots$ Tustin).

with $E = 15.46\%$, while when $m = 3$, for the proposed method generates

$$H_c(s) = \frac{0.0001s^3 - 0.0003s^2 + 0.0005s + 0.9991}{s^3 + 1.9992s^2 + 1.9993s + 0.9993}.$$

In Fig. 6, the frequency responses of the resultant CTFs are compared, and the proposed method shows superior performance.

5. Conclusion

In this paper, an accurate algorithm for converting an S-domain model into a Z-domain model is proposed. It results in a much better model compared to the conventional conversion schemes such as Tustin approximation with or without frequency pre-warping, zero-pole mapping equivalent, Triangle hold equivalent, etc. The source model can be either parametric or non-parametric. A technique based on the Bilinear transformation is proposed which can generally yield a stable target model from the source stable parametric model. It also can be applied to Z- to S-domain model conversion, system re-sampling, time delay system conversion, and sampling rate selection. The method is thus very useful in digital signal processing and digital control.

References

[1] K.J. Åström, B. Wittenmark, Adaptive Control, Addison-Wesley, Reading, MA, 1995.

[2] T. Chen, B.A. Francis, H_2-optimal sampled-data control, IEEE Trans. Automat. Control 36 (4) (1991) 387–397.

[3] T. Chen, B.A. Francis, Optimal Sampled-Data Control Systems, Springer, New York, 1995.

[4] P. Eykhoff, System Identification-Parameter and State Estimation, Wiley, New York, 1974.

[5] G.F. Franklin, J.D. Powell, M.L. Workman, Digital Control of Dynamic Systems, Addison-Wesley, Reading, MA, 1990.

[6] J.P. Keller, B.D.O. Anderson, A new approach to the discretization of continuous-time controllers, IEEE Trans. Automat. Control 37 (1992) 214–223.

[7] B.C. Kuo, Digital Control Systems, Holt, Rinehart and Winston, New York, 1980.

[8] R. Pintelon, P. Guillaume, Y. Rolain, J. Schoukens, H. Van Hamme, Parametric identification of transfer functions in the frequency domain-a survey, IEEE Trans. Automat. Control 39 (11) (1994) 2245–2260.

[9] N. Rafee, T. Chen, O. Malik, A technique for optimal digital redesign of analog controllers, IEEE Trans. Control Systems Technol. 5 (1) (1997) 89–99.

[10] K.S. Rattan, Digitalization of existing continuous control system, IEEE Trans. Automat. Control AC-29 (3) (1984) 282–285.

[11] K.S. Rattan, Compensating for computational delay in digital equivalent of continuous control systems, IEEE Trans. Automat. Control 34 (8) (1989) 895–899.

[12] D. Raviv, E.W. Djaja, Technique for enhancing the performance of discretized controllers, IEEE Control Systems Mag. 19 (3) (1999) 52–57.

[13] Y.N. Rosenvasser, B.P. Lampe, Parametric transfer functions for sampled-data systems with time-delayed controllers, Proceedings of the IEEE Conference on Decision and Control, San Diego, CA, Vol. 2, December 1997, pp. 1609–1614.

[14] Y.N. Rosenvasser, K.Y. Polyakov, B.P. Lampe, Application of Laplace transformation for digital redesign of continuous control systems, IEEE Trans. Automat. Control 44 (4) (1999) 883–886.

[15] C.K. Sanathanan, J. Koerner, Transfer function synthesis as a ratio of two complex polynomials, IEEE Trans. Automat. Control 8 (1963) 56–58.

[16] L.S. Shieh, J.L. Zhang, J.W. Sunkel, A new approach to the digital redesign of continuous-time controllers, Control Theory Adv. Technol. 8 (1992) 37–57.

[17] D. Tabak, Digitalization of control systems, Comput. Aided Des. 3 (2) (1971) 13–18.

[18] J. Tornero, Y.P. Gu, M. Tomizuka, Analysis of multi-rate discrete equivalent of continuous controller, Proceedings of the American Control Conference, San Diego, CA, June 1999, pp. 2759–2763.

[19] Control System Toolbox User's Guide, Version 4.2, The MathWorks Inc, Natick, MA, 1998.