Multirate Digital Signal Processing via Sampled-Data H^∞ Optimization

Dissertation

Submitted in partial fulfillment of the requirements for the degree of Doctor of Informatics

Masaaki Nagahara

Department of Applied Analysis and Complex Dynamical Systems

Graduate School of Informatics

Kyoto University

Abstract

In this thesis, we present a new method for designing multirate signal processing and digital communication systems via sampled-data H^∞ control theory. The difference between our method and conventional ones is in the signal spaces. Conventional designs are executed in the discrete-time domain, while our design takes account of both the discrete-time and the continuous-time signals. Namely, our method can take account of the characteristic of the original analog signal and the influence of the A/D and D/A conversion. While the conventional method often indicates that an ideal digital low-pass filter is preferred, we show that the optimal solution need not be an ideal low-pass when the original analog signal is not completely band-limited. This fact can not be recognized only in the discrete-time domain. Moreover, we consider quantization effects. We discuss the stability and the performance of quantized sampled-data control systems. We justify H^∞ control to reduce distortion caused by the quantizer. Then we apply it to differential pulse code modulation. While the conventional Δ modulator is not optimal and besides not stable, our modulator is stable and optimal with respect to the H^∞-norm. We also give an LMI (Linear Matrix Inequality) solution to the optimal H^∞ approximation of IIR (Infinite Impulse Response) filters via FIR (Finite Impulse Response) filters. A comparison with the Nehari shuffle is made with a numerical example, and it is observed that the LMI solution generally performs better. Another numerical study also indicates that there is a trade-off between the pass-band and stop-band approximation characteristics.

Acknowledgments

I would like to express my sincere gratitude to everyone who helped me and contributed in various ways toward completion of this work.

First of all, I am most grateful to my supervisor Professor Yutaka Yamamoto. Since I entered the Graduate School of Kyoto University, he has constantly guided me in every respect of science and technology. Without this knowledge he has endowed me, it is unthinkable that I could proceed this far toward the completion of this thesis. He also taught me fundamental knowledge on sampled-data control and signal processing on which this thesis is based. His strong leadership and ideas have been a constant source of encouragement and have had a definite influence on this work. Without his supervision, this thesis would not exist today.

I would also like to thank Dr. Hisaya Fujioka for his helpful discussions and support in my research, in particular, those on sampled-data control. His warm encouragement was also great help to completing this thesis.

My gratitude should be extended to Dr. Yuji Wakasa for his valuable advice. In particular, his advice on convex optimization was very valuable to my research.

I would thank Mr. Kenji Kashima, Mr. Shinichiro Ashida, and current and past members of the intelligent and control systems laboratory for their academic stimulation and dairy friendship.

Special thanks are addressed toward my parents for their deep understanding and encouragement.

This research was partially supported by Japan Society for the Promotion of Science (JSPS).

Contents

Notation

$\mathbb{Z}_+$: non-negative integers.

$\mathbb{N}$: natural numbers.

$\mathbb{R}$: real numbers.

$\mathbb{R}_+$: non-negative real numbers.

$\mathbb{R}^n$: n-dimensional vector space over $\mathbb{R}$.

$\mathbb{C}$: complex numbers.

l^2: real-valued square summable sequences.

l^∞: real-valued bounded sequences.

$L^2[0,\infty)$ **and** $L^2[0,h)$: Lebesgue spaces consisting of square integrable real functions on $[0,\infty)$ and $[0,h)$, respectively. $L^2[0,\infty)$ may be abbreviated to L^2.

$l^2_{L^2[0,h)}$: square summable sequences whose values are in $L^2[0,h)$.

$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$: transfer function whose realization is $\{A,B,C,D\}$, that is, $\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right](\lambda) := C(\lambda I - A)^{-1}B + D$, where $\lambda := s$ in continuous-time and $\lambda := z$ in discrete-time.

$\mathcal{F}_l(P,K)$: linear fractional transformation of P and K, that is, if $P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$ then $\mathcal{F}_l(P,K) := P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21}$.

$\mathcal{S}_h$: ideal sampler with sampling period h.

$\mathcal{H}_h$: zero order hold with sampling period h.

$\mathbf{L}_h$: lifting for continuous-time signals with sampling period h.

$\mathbb{L}_N$: lifting for discrete-time signals by factor N (discrete-time lifting).

$\mathcal{L}_N(P)$: fast-discretizing and discrete-time lifting of continuous system P, that is, $\mathcal{L}_N(P) := \mathbb{L}_N\mathcal{S}_{h/N}P\mathcal{H}_{h/N}\mathbb{L}_N^{-1}$.

$\uparrow M$: upsampler with upsampling factor M.

$\downarrow M$: downsampler with downsampling factor M.

$r(A)$: maximum absolute value of eigenvalues of matrix A.

A^T: transpose of a matrix (or a vector) A.

Chapter 1

Introduction

This thesis presents a new method in multirate digital signal processing and digital communication systems.

When we execute these design procedures, we must first discretize the original analog signal (e.g., speech, audio or visual image), then the discretized signal is processed in the discrete-time domain (e.g., filtering, compressing, transmitting etc.), and finally we reconstruct an analog signal from the discrete signal.

Conventionally, the design is performed mostly in the discrete-time domain by assuming that the original analog signal is fully band-limited up to the Nyquist frequency. Under this assumption, the sampling theorem gives a method for reconstructing an analog signal from a sampled signal [16, 45].

Theorem 1.1 (Shannon et al.). *Let $f(t)$ be a continuous-time signal ideally band-limited to the range $(-\pi/h, \pi/h)$, that is, its Fourier transform $\hat{f}(\omega)$ is zero outside this interval. Then $f(t)$ can uniquely be recovered from its sampled values $f(nh)$, $n = 0, \pm 1, \pm 2, \ldots$ via the formula*

$$f(t) = \sum_{n=-\infty}^{\infty} f(nh) \frac{\sin \pi(t/h - n)}{\pi(t/h - n)} = \sum_{n=-\infty}^{\infty} f(nh) \mathrm{sinc}(t/h - n). \tag{1.1}$$

Although most of the conventional studies of digital signal processing are based on Theorem 1.1 [9, 32, 33, 35, 45, 46], we encounter two questions in the implementation: the question of D/A and A/D conversions.

We consider the first question. The process (1.1) is a kind of D/A conversion; we convert sampled values $\{f(nh)\}$ to modulated impulses $\{f(nh)\delta(nh)\}$, then filter them by the ideal low-pass filter[1)] [46]. In practice, this conversion is physically impossible, and in reality a zero-order hold followed by a sharp low-pass filter is often used. We should notice that the effect of such a real situation of D/A conversion can never be taken into account only in the discrete-time domain.

The question that we must consider next is the assumption of full band-limitation. The assumption that Theorem 1.1 requires is in reality impossible because no real analog

[1)] The frequency response of this filter is equal to 1 up to the Nyquist frequency ω_N and zero beyond ω_N.

signal is fully band-limited. In order to achieve the assumption, a sharp low-pass filter is attached before sampling. However, such a sharp low-pass characteristic will deteriorate the quality of the analog signal, and moreover even such a filter does not satisfy the assumption. Therefore we have to consider the effect of sampling (i.e., *aliasing*), which can never be captured in the discrete-time domain.

To answer these questions, we must take account of not only discrete-time signals but also continuous-time signals. Therefore, we propose to consider an analog performance to design digital systems by using the modern *sampled-data control theory*. Sampled-data control theory deals with control systems that consist of continuous-time plants to be controlled, discrete-time controllers controlling them, and ideal A/D and D/A converters. The modern theory can take the intersample behavior of sampled-data control systems into account. The key idea is *lifting* [36, 2, 37]. Although sampled-data systems are not time-invariant, the lifting method leads to time-invariant discrete-time models. These models are infinite-dimensional, which can be reduced to equivalent discrete-time finite-dimensional ones [1, 4, 18].

There is another method for sampled-data systems: *fast-sampling/fast-hold* (FSFH) method [19, 43]. The idea is to approximate the continuous-time inputs by step functions of sufficiently small step size and also approximate the continuous-time outputs by taking their samples by sufficiently fast ideal sampler. The approximated system will be a periodically time-varying discrete-time system (finite-dimensional), which can be transformed a time-invariant discrete-time system by using the *discrete-time lifting* [25, 4]

Based on these studies, the H^∞ optimal sampled-data control has been studied [34, 1, 18, 13]. The H^∞ optimality criterion is suitable for the frequency characteristic, which is intuitive for one who designs the system.

In the last few years, several articles have studied digital signal processing via sampled-data control theory. Chen and Francis [5] studied a multirate filter bank design problem with an H^∞ criterion. Although this design was done in the discrete-time domain, they brought the modern H^∞ control theory to the digital signal processing, and thus they threw new light on the subject. The first study of digital signal processing in the sampled-data setting was made by Khargonekar and Yamamoto [21]. They formulated a single-rate signal reconstruction problem by using sampled-data theory. After their study, this method was developed to multirate signal processing [40, 14, 15, 44, 27], digital communications [29, 30] and quantizer design [26].

The purpose of this thesis is to answer the questions mentioned above by sampled-data H^∞ optimal control theory, and to show that this method is effective in designing digital systems.

The organization of this thesis is as follows.

- Chapter 2 surveys sampled-data control theory. We describe the fundamental difficulty in sampled-data systems, and introduce the lifting method, which resolved this difficulty: by using it we can define the frequency response and the H^∞ optimal control. The fast-sampling/fast-hold method for multirate sampled-data control systems is discussed in detail, because we mainly use this method in this thesis.

- Chapter 3 deals with multirate signal processing, in particular, interpolation, decimation and sampling rate conversion. We present a new method for designing these

systems via the sampled-data H^∞ optimization. Design examples shows that our method is superior to the conventional one.

- Chapter 4 presents a new design of digital communication systems. Under signal compression and channel distortion, we design an optimal transmitting/receiving filter by using the sampled-data H^∞ optimization. We show also design examples to indicate that our method is effective in digital communication.

- Chapter 5 investigates issues of quantization. Since quantization is a nonlinear operation, we introduce a linearized model (i.e., additive noise model). By using this model, we discuss stability and performance of a quantized sampled-data control system. Then we apply it to differential pulse code modulation (DPCM) systems. Design examples are shown and our design is superior to the conventional Δ modulation.

- Chapter 6 presents a new method to approximate an IIR filter by an FIR filter, which directly yields an optimal approximation with respect to the H^∞ error norm. We show that this design problem can be reduced to an LMI (Linear Matrix Inequality). We will also make a comparison via a numerical example with an existing method, known as the Nehari shuffle.

- Chapter 7 concludes this thesis with a summary of the results presented and future perspectives.

Chapter 2

Sampled-Data Control Theory

2.1 Sampled-data control systems

A sampled-data control system is a system in which a continuous-time plant is to be controlled by a discrete-time controller. Consider the unity-feedback sampled-data control system shown in Figure 2.1. In this figure, $P(s)$ is a continuous-time plant and $K(z)$ is

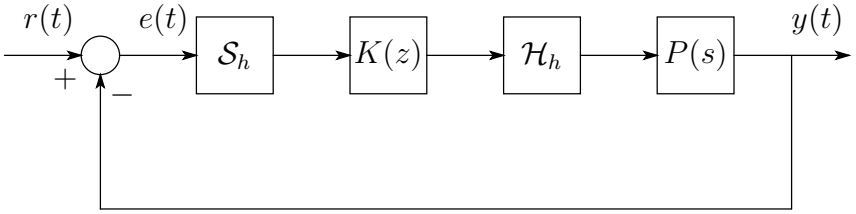

Figure 2.1: Sampled-data control system

a discrete-time controller. In order to include $K(z)$ in this control system, we need an interface. Therefore we introduce the ideal sampler $\mathcal{S}_h$ and the zero-order hold $\mathcal{H}_h$ with sampling time h.

Definition 2.1. *The ideal sampler $\mathcal{S}_h$ and the zero-order hold $\mathcal{H}_h$ are defined as follows:*

$$\begin{aligned} &\mathcal{S}_h : L^2[0,\infty) \ni u \longmapsto v \in l^2, \quad v[k] := u(kh), \\ &\mathcal{H}_h : l^2 \ni v \longmapsto u \in L^2[0,\infty), \quad u(kh+\theta) := H(\theta)v[k], \\ &\qquad k = 0, 1, 2, \dots , \end{aligned}$$

where $H(\cdot)$ is the hold function defined as follows:

$$H(\theta) := \begin{cases} 1, & \theta \in [0, h), \\ 0, & \text{otherwise.} \end{cases} \tag{2.1}$$

In practice, a quantization error occurs in the A/D conversion. We however omit this quantization error here. The quantization effects are discussed in Chapter 5.

The system contains both continuous-time and discrete-time signals and is theoretically regarded as a periodically time-varying system [4]. Since this system is not time-invariant, it is difficult to analyze or design by using such conventional machinery as transfer functions or frequency responses.

2.2 Lifting

Conventionally there are two ways for designing sampled-data systems. One is the following: first design a continuous-time controller in the continuous-time domain, and then discretize the controller. A typical discretization method is the Tustin (or bilinear) transformation [4, 38, 10]. If the sampling period is sufficiently small, the designed system may perform well. However, if the sampling period is not small enough, the performance not only deteriorates, but also the closed-loop system may become unstable.

The other is to approximate the continuous-time signal by a discrete-time one by considering only the signals at sampling points. As a result the sampled-data system becomes a discrete-time time-invariant system. This method preserves the closed-loop stability. However, the output of the system may sometimes induce very large intersample ripples despite a very small sampling period. The reason is that the method ignores the intersample behavior.

Recently, Yamamoto [36, 37] studied the problem of sampled-data control systems. He developed what is now called the *lifting* method, which takes the intersample behavior into account and gives an exact, not approximated, time-invariant discrete-time system for a sampled-data control system.

We begin by defining the lifting operator.

Definition 2.2. *Define the lifting operator* $\mathbf{L}_h$ *by*

$$\mathbf{L}_h : L^2[0,\infty) \longrightarrow l^2_{L^2[0,h)} : \quad \{f(t)\}_{t\in\mathbb{R}_+} \longmapsto \{\widetilde{f}[k](\theta)\}_{k=0}^{\infty}, \quad \theta \in [0,h),$$
$$\widetilde{f}[k](\theta) := f(kh+\theta) \in L^2[0,h).$$

By lifting, continuous-time signals in $L^2[0,\infty)$ will become discrete-time signals whose values are in $L^2[0,h)$, hence the sampled-data system can be rewritten as a time-invariant discrete-time system with infinite-dimensional signal spaces. As a result, we can introduce the concept of transfer functions or the frequency response for sampled-data systems, and hence we can treat sampled-data systems as time-invariant systems without approximation.

Consider the standard sampled-data control system in Figure 2.2, where G is a continuous-time generalized plant, whose state-space equation is given as follows:

$$\dot{x} = Ax + \begin{bmatrix} B_1 & B_2 \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix},$$
$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} x + \begin{bmatrix} D_{11} & D_{12} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}.$$

The signal w is the exogenous input consisting of reference commands, disturbance or sensor noise, while z is the signal to be controlled to have a desirable performance. Note that both of these signals are continuous-time. The system K_d is a digital controller.

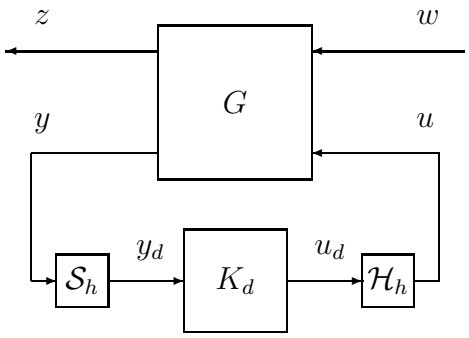

Figure 2.2: Sampled-data control system

The design problem of the sampled-data control system in Figure 2.2 is to obtain the controller K_d which stabilizes the closed-loop system and makes the performance from w to z desirable.

This system is a sampled-data control system, and hence it is a periodically time varying system. However, by lifting the continuous-time signals z and w, and taking $\widetilde{z} := \mathbf{L}_h z$ and $\widetilde{w} := \mathbf{L}_h w$, the sampled-data control system in Figure 2.2 can be converted to a time-invariant discrete-time system shown in Figure 2.3. Namely, the state-space

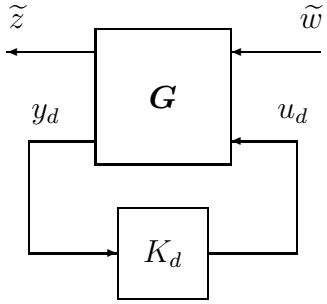

Figure 2.3: Lifted sampled-data control system

equation of the lifted system $\boldsymbol{G}$ is obtained as follows:

$$\begin{bmatrix} x[k+1] \\ \widetilde{z}[k] \\ y_d[k] \end{bmatrix} = \begin{bmatrix} A_d & \mathbf{B}_1 & B_{d2} \\ \mathbf{C}_1 & \mathbf{D}_{11} & \mathbf{D}_{12} \\ C_{d2} & 0 & 0 \end{bmatrix} \begin{bmatrix} x[k] \\ \widetilde{w}[k] \\ u_d[k] \end{bmatrix},$$
$$k = 0, 1, \dots,$$

$$
\begin{aligned}
&A_d := e^{Ah}, \qquad B_{d2} := \int_0^h e^{A(h-\tau)} B_2 d\tau, \qquad C_{d2} := C_2, \\
&\mathbf{B}_1 : L^2[0,h) \longrightarrow \mathbb{R}^n : w \mapsto \int_0^h e^{A(h-\tau)} B_1 w(\tau) d\tau, \\
&\mathbf{C}_1 : \mathbb{R}^n \longrightarrow L^2[0,h) : x \mapsto C_1 e^{A\theta} x, \\
&\mathbf{D}_{11} : L^2[0,h) \longrightarrow L^2[0,h) \\
&\qquad : w \mapsto \int_0^\theta C_1 e^{A(\theta-\tau)} B_1 w(\tau) d\tau + D_{11} w(\theta), \\
&\mathbf{D}_{12} : \mathbb{R}^m \longrightarrow L^2[0,h) \\
&\qquad : u_d \mapsto \int_0^\theta C_1 e^{A(\theta-\tau)} B_2 H(\tau) d\tau u_d + D_{12} H(\theta) u_d, \\
&\qquad \theta \in [0,h).
\end{aligned}
$$

In this equation, n and m are the dimensions of x and u_d, respectively, and $H(\cdot)$ is the hold function defined by (2.1). Note that $\mathbf{B}_1$, $\mathbf{C}_1$, $\mathbf{D}_{11}$ and $\mathbf{D}_{12}$ are operators in infinite-dimensional spaces, while A_d, B_{d2} and C_{d2} are matrices. Therefore the lifted system becomes a discrete-time time-invariant system with infinite-dimensional operators.

2.3 Frequency response and H^∞ optimization

In the previous section, we have shown that sampled-data systems can be represented as time-invariant discrete-time systems. We can then define their transfer function or frequency response. The concept of frequency response for sampled-data systems is introduced by Yamamoto and Khargonekar [41] and its computation is developed, for example, in [12].

Let $\mathcal{T}$ denote the system from w to z in Figure 2.2, and $\widetilde{\mathcal{T}}$ the lifted system of $\mathcal{T}$. Define the state-space equation for $\widetilde{\mathcal{T}}$ as follows:

$$
\begin{aligned}
x_s[k+1] &= Ax_s[k] + \mathbf{B}\widetilde{w}[k], \\
\widetilde{z}[k] &= \mathbf{C}x_s[k] + \mathbf{D}\widetilde{w}[k], \qquad k = 0, 1, 2, \ldots .
\end{aligned}
\tag{2.2}
$$

Note that A is a matrix, while $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are infinite-dimensional operators. We assume that A is a power stable matrix, that is, $A^n \to 0$ as $n \to \infty$.

For the lifted signal $\{\widetilde{f}[k]\}_{k=0}^{\infty}$, define its z *transform* as

$$
\mathcal{Z}[\widetilde{f}](z) := \sum_k^{\infty} \widetilde{f}[k] z^{-k}.
$$

It follows that the *transfer function* $\widetilde{\mathcal{T}}(z)$ of the sampled-data system can be defined as follows:

$$
\widetilde{\mathcal{T}}(z) := \mathbf{D} + \mathbf{C}(zI - A)^{-1}\mathbf{B}, \qquad z \in \mathbb{C}.
$$

The z transform $\widetilde{\mathcal{T}}(z)$ is a linear operator on $L^2[0,h)$ with a complex variable z. By substituting $e^{j\omega h}$ for z in $\widetilde{\mathcal{T}}(z)$, we can define the *frequency response operator*.

Definition 2.3. *Define the frequency response operator of sampled-data system $\mathcal{T}$ by*

$$\widetilde{\mathcal{T}}(e^{j\omega h}) = \mathbf{D} + \mathbf{C}(e^{j\omega h}I - A)^{-1}\mathbf{B} : L^2_{[0,h)} \longrightarrow L^2[0,h), \quad \omega \in \mathbb{R}.$$

The norm of the frequency response operator $\widetilde{\mathcal{T}}(e^{j\omega h})$

$$\|\widetilde{\mathcal{T}}(e^{j\omega h})\| := \sup_{\substack{v \in L^2[0,h) \\ v \neq 0}} \frac{\|\widetilde{T}(e^{j\omega h})v\|_{L^2[0,h)}}{\|v\|_{L^2[0,h)}},$$

is called the *gain* at ω. The H^∞*-norm* of the sampled-data system is then given by

$$\|\widetilde{\mathcal{T}}\|_\infty := \sup_{\omega \in [0, 2\pi/h)} \|\widetilde{T}(e^{j\omega h})\|.$$

The H^∞-norm $\|\widetilde{\mathcal{T}}\|_\infty$ is equivalent to the L^2 induced norm of the sampled-data system (2.2), that is,

$$\|\widetilde{\mathcal{T}}\|_\infty = \|\mathcal{T}\| := \sup_{\substack{w \in L^2[0,\infty) \\ w \neq 0}} \frac{\|\mathcal{T}w\|_{L^2[0,\infty)}}{\|w\|_{L^2[0,\infty)}}.$$

Sampled-data H^∞ control problem is to find a discrete-time controller K_d which stabilizes the closed-loop system and makes the H^∞-norm of the system small. The H^∞ control has the following advantages:

- Since the H^∞-norm is the L^2 induced norm of the sampled-data control system, we can formulate the worst case design.
- Many robustness requirements for the design against the system uncertainty can be described by H^∞-norm constraint.
- We can shape the frequency characteristic with frequency weights, which is intuitive to designers.

Although $\mathcal{T}$ is infinite-dimensional, the H^∞ optimization can be equivalently transformed to that for a finite-dimensional discrete-time system [34, 1, 18, 13, 4]. Note that the obtained finite-dimensional discrete-time optimization problem takes intersample behavior into account.

On the other hand, there is another method for obtaining a finite-dimensional discrete-time system: *fast-sampling/fast-hold approximation* [19, 4]. This method provides not an equivalent but an approximated model, however its computation is easier than that of the method giving equivalent models.

Moreover, with regard to signal reconstruction problem (main theme of this thesis), the method giving equivalent models yields some conservative results. The reason is as follows: we often allow signal reconstruction to have time delays since the system does not have any feedback loop. In other words, a sampled value at a certain time can be reconstructed by using future sampled values, that is, the filter is allowed to be non-causal. However, the method giving equivalent method does not readily apply to this situation, while the fast-sampling/fast-hold method does.

For the reasons mentioned above, we adopt the approximation method in this thesis to consider the problem. In the following section, we discuss this method in detail.

2.4 Fast discretization of multirate sampled-data systems

In this section, we deal with multirate sampled-data control systems by using the fast-sampling/fast-hold method. The fast-sampling/fast-hold technique is a method for approximating the performance of sampled-data systems. The procedure is as follows:

- discretize the continuous-time input by a hold with sampling period h/N,
- discretize the continuous-time output by a sampler with sampling period h/N.

With large $N \in \mathbb{N}$, the discretized signals may be good approximation of the continuous signals.

Figure 2.4 illustrates the procedure. Assume that the controller $\mathcal{K}$ is (M_1, M_2)-periodic

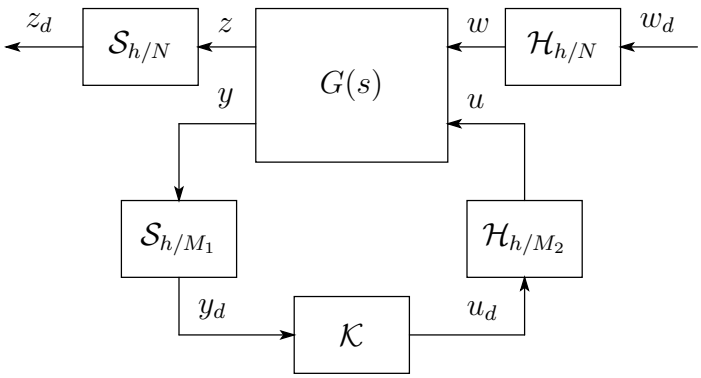

Figure 2.4: Fast-sampling/fast-hold discretization

$(M_1, M_2 \in \mathbb{N})$ [25], that is, $z^{M_2}\mathcal{K}z^{-M_1} = \mathcal{K}$ where z is the unit advance and z^{-1} is the unit time delay.

2.4.1 Discrete-time lifting

By attaching a fast-sampler and a fast-hold as shown in Figure 2.4, the multirate sampled-data system will be converted to a finite-dimensional discrete-time system (we will show this in the next section). However, this system has three sampling periods: h/M_1, h/M_2 and h/N, and hence the system will be time-varying (to put it precisely, periodically time-varying). In order to equivalently convert a multirate system to a single-rate one, the *discrete-time lifting* [20, 25, 4] is useful. The definition is as follows:

Definition 2.4. *Define the discrete-time lifting* $\mathbb{L}_N$ *and its inverse* $\mathbb{L}_N^{-1}$ *by*

$$\mathbb{L}_N : l^2 \longrightarrow l^2 : \{v[0], v[1], \dots\} \mapsto \left\{ \begin{bmatrix} v[0] \\ v[1] \\ \vdots \\ v[N-1] \end{bmatrix}, \begin{bmatrix} v[N] \\ v[N+1] \\ \vdots \\ v[2N-1] \end{bmatrix}, \dots \right\},$$

$$\mathbb{L}_N^{-1} : l^2 \longrightarrow l^2 :$$
$$\left\{ \begin{bmatrix} v_0[0] \\ v_1[0] \\ \vdots \\ v_{N-1}[0] \end{bmatrix}, \begin{bmatrix} v_0[1] \\ v_1[1] \\ \vdots \\ v_{N-1}[1] \end{bmatrix}, \dots \right\} \mapsto \{v_0[0], v_1[0], \dots, v_{N-1}[0], v_0[1], v_1[1], \dots\}.$$

The discrete-time lifting $\mathbb{L}_N$ converts a 1-dimensional signal into an N-dimensional signal and the sampling rate becomes N times slower. This operation makes it possible to equivalently convert multirate systems into single-rate systems, and hence its analysis and design become easier. Note that the discrete-time lifting $\mathbb{L}_N$ is norm-preserving, namely, $\|\mathbb{L}_N v\| = \|v\|$, $v \in l^2$ and so is $\mathbb{L}_N^{-1}$.

2.4.2 Approximating multirate sampled-data systems

By using the discrete-time lifting, the multirate system shown in Figure 2.4 is converted to a single-rate discrete-time system. Take

$$G(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix},$$

where G_{11}, G_{12} and G_{22} are strictly proper and G_{21} is proper. Let their state-space realization be

$$G_{ij}(s) = \left[\begin{array}{c|c} A & B_j \\ \hline C_i & D_{ij} \end{array}\right](s), \quad i, j = 1, 2.$$

Let the system from w_d to z_d in Figure 2.4 be T_{dN}. Then T_{dN} can be rewritten as follows:

$$\begin{aligned} T_{dN} &= \mathcal{F}_l(G_{dN}, \mathcal{K}), \\ T_{dN} &:= \begin{bmatrix} \mathcal{S}_{h/N} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{h/N} & 0 \\ 0 & I \end{bmatrix}. \end{aligned}$$

Then we apply the discrete-time lifting to w_d and z_d as shown in Figure 2.5. Let the system from $\widetilde{w}_d$ to $\widetilde{z}_d$ in Figure 2.5 be $\widetilde{T}_{dN}$. Note that since $\mathbb{L}_N$ and $\mathbb{L}_N^{-1}$ are norm-preserving mappings, we have $\|T_{dN}\| = \|\widetilde{T}_{dN}\|$. Then we rewrite $\widetilde{T}_{dN}$ as follows:

$$\begin{aligned} \widetilde{T}_{dN} &= \mathcal{F}_l(\widehat{G}_{dN}, \mathcal{K}), \\ \widehat{G}_{dN} &:= \begin{bmatrix} \mathbb{L}_N & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathcal{S}_{h/N} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{h/N} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbb{L}_N^{-1} & 0 \\ 0 & I \end{bmatrix}. \end{aligned}$$

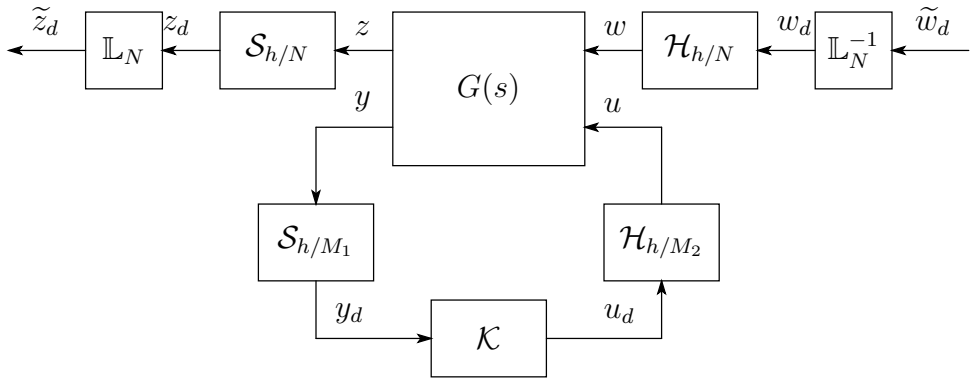

Figure 2.5: Lifted and fast-discretized sampled-data system

Let us turn to the controller $\mathcal{K}$. By the assumption that $\mathcal{K}$ is (M_1, M_2)-periodic, $K_d := \mathbb{L}_{M_2} K \mathbb{L}_{M_1}^{-1}$ is time-invariant [25]. By using this property, we rearrange $\widetilde{G}_{dN}$ as follows:

$$\begin{aligned}
\widetilde{T}_{dN} &= \mathcal{F}_l(\widetilde{G}_{dN}, K_d), \\
\widetilde{G}_{dN} &:= \begin{bmatrix} I & 0 \\ 0 & \mathbb{L}_{M_1}\mathcal{S}_{h/M_1} \end{bmatrix} \widehat{G}_{dN} \begin{bmatrix} I & 0 \\ 0 & \mathcal{H}_{h/M_2}\mathbb{L}_{M_2}^{-1} \end{bmatrix} \\
&= \begin{bmatrix} \mathbb{L}_N\mathcal{S}_{h/N} & 0 \\ 0 & \mathbb{L}_{M_1}\mathcal{S}_{h/M1} \end{bmatrix} \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{h/N}\mathbb{L}_N^{-1} & 0 \\ 0 & \mathcal{H}_{M_2}\mathbb{L}_{M_2}^{-1} \end{bmatrix}.
\end{aligned}$$

Finally, we convert $\widetilde{G}_{dN}$ to a simple time-invariant system by the following proposition.

Proposition 2.1. *Assume $N = kM_1M_2$, $k \in \mathbb{N}$. Then we have the following identities:*

$$\mathbb{L}_{M_1}\mathcal{S}_{h/M_1} = S\mathbb{L}_N\mathcal{S}_{h/N}, \quad \mathcal{H}_{h/M_2}\mathbb{L}_{M_2}^{-1} = \mathcal{H}_{h/N}\mathbb{L}_N^{-1}H, \tag{2.3}$$

where

$$\begin{aligned}
S &:= \left.\begin{bmatrix} p & & \\ & \ddots & \\ & & p \end{bmatrix}\right\} M_1, \quad p := [1, \underbrace{0, \dots, 0}_{kM_2-1}], \\
H &:= \underbrace{\begin{bmatrix} q & & \\ & \ddots & \\ & & q \end{bmatrix}}_{M_2}, \quad q := [\underbrace{1, \dots, 1}_{kM_1}]^T.
\end{aligned} \tag{2.4}$$

This proposition gives us the following equation:

$$\begin{aligned}
\widetilde{G}_{dN} &= \begin{bmatrix} \mathbb{L}_N\mathcal{S}_{h/N} & 0 \\ 0 & S\mathbb{L}_N\mathcal{S}_{h/N} \end{bmatrix} \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{h/N}\mathbb{L}_N^{-1} & 0 \\ 0 & \mathcal{H}_{h/N}\mathbb{L}_N^{-1}H \end{bmatrix} \\
&= \begin{bmatrix} I & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} \mathcal{L}_N(G_{11}) & \mathcal{L}_N(G_{12}) \\ \mathcal{L}_N(G_{21}) & \mathcal{L}_N(G_{22}) \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & H \end{bmatrix},
\end{aligned}$$

where $\mathcal{L}_N(G) := \mathbb{L}_N \mathcal{S}_{h/N} G \mathcal{H}_{h/N} \mathbb{L}_N^{-1}$. Note that $\mathcal{L}_N(G_{ij}) =: G_{d,ij}$ $(i,j = 1,2)$ is a time-invariant discrete-time system [25, 4]. In fact, a state-space realization of $G_{d,ij}$ is given as follows:

$$G_{d,ij} = \left[\begin{array}{c|cccc} A_d^N & A_d^{N-1}B_{d,j} & A_d^{N-2}B_{d,j} & \dots & B_{d,j} \\ \hline C_i & D_{ij} & 0 & \dots & 0 \\ C_iA_d & C_iB_{d,j} & D_{ij} & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ C_iA_d^{N-1} & C_iA_d^{N-2}B_{d,j} & C_iA_d^{N-3}B_{d,j} & \dots & D_{ij} \end{array}\right], \quad i,j = 1,2,$$

where

$$A_d := e^{Ah/N}, \quad B_{d,j} := \int_0^{h/N} e^{At} B_j dt.$$

We show the obtained time-invariant discrete-time system in Figure 2.6.

When N becomes larger, the performance of the discrete-time system $T_{dN} := \mathcal{F}_l(\widetilde{G}_{dN}, K_d)$ converges to that of the original sampled-data system $\mathcal{T} := \mathcal{F}_l(G, \mathcal{H}_{h/M_2}\mathcal{K}\mathcal{S}_{h/M_1})$ [43]. Therefore, if we take sufficiently large N, the error between $\mathcal{T}$ and T_{dN} will small. The error estimate for the fast-sampling factor N is discussed in [39].

We summarize the above discussion as a theorem:

Theorem 2.1. *For the multirate sampled-data control system $\mathcal{T} := \mathcal{F}_l(G, \mathcal{H}_{h/M_2}\mathcal{K}\mathcal{S}_{h/M_1})$, there exists a time-invariant discrete-time system $T_{dN} := \mathcal{F}_l(\widetilde{G}_{dN}, K_d)$ such that*

$$\lim_{N\to\infty} \|T_{dN}\| = \|\mathcal{T}\|.$$

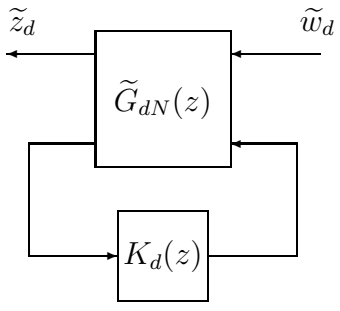

Figure 2.6: FSFH discretized system

Chapter 3

Multirate Signal Processing

3.1 Introduction

Multirate techniques are now very popular in digital signal processing. They are particularly effective in subband coding, and various techniques for economical information saving have been developed [9, 35, 46].

One example is signal decoding in audio/speech processing. For example, in the commercial CD format, the sampling frequency is 44.1 kHz, but one hardly employs the same sampling period in decoding. A popular technique is *interpolation*. The process is as follows: first *upsample* the encoded digital signal (i.e., inserting zeros between two consecutive samples), remove the parasitic *imaging* spectra via a digital low-pass filter, and then convert it back to an analog signal with a hold device and an analog low-pass filter. Imaging is a phenomenon due to zeros inserted by upsampling, and yields high frequency noise.

The chief advantage here is that one can employ a fast hold device, and do not have to use a very sharp analog filter (thereby avoiding much phase distortion induced by a sharp analog filter).

Another example is signal compression. Due to the limitation of the bandwidth of communication channels or of the size of storage devices, one often needs to compress the signals. In signal compression, multirate processing plays a major role. The fundamental operation for signal compression is *decimation*. Decimation is to reduce the sampling rate of a signal; the process is to first filter out the *aliasing* components by using a digital low-pass filter, and then *downsample* the filtered signal. Downsampling is an operation to keep every M-th sample (M is a natural number) and remove in-between samples. The aliasing caused by upsampling is comparative to aliasing in A/D conversion, that is, removing samples causes frequency overlapping.

By combining interpolators and decimators, we can obtain a sampling rate converter. In commercial applications, there are many different sampling rates employed: for example 48kHz for DAT and 44.1kHz for audio CD. The conversion from one sampling rate to the other becomes necessary. In such a process, it is clearly required that the information loss be as little as possible.

The conventional way of doing this is as follows: Suppose we want to convert a signal v with sampling frequency f_1 Hz to another signal u with sampling frequency f_2 Hz.

Suppose also that there exist (coprime) integers L_1 and L_2 such that $f_1 L_1 = f_2 L_2$. We first upsample v by factor L_1, to make the sampling frequency $f_1 L_1$. Suppose that the original signal is perfectly band-limited in the range $|\omega| < f_1/2$. We then introduce a digital filter $H(z)$ to filter out the undesirable imaging component. After this, the obtained signal is downsampled by factor L_2 to become a signal with sampling frequency $f_2 = f_1 L_1 / L_2$ Hz.

In the existing literature, it is a commonly accepted principle that one inserts a very sharp digital low-pass filter after the upsampler or before the downsampler to eliminate the effect of imaging or aliasing components [9, 35, 46]. This is based on the following reasoning: Suppose that the original signal is fully band-limited. Then the imaging (aliasing) components induced by upsampling (downsampler) is not relevant to the original analog signal and hence they must be removed by a low-pass filter. If the original signal is band-limited, the closer is this filter to an ideal filter, the better.

In practice, however, no signals are fully band-limited in a practical range of a pass-band, and they obey only an approximate frequency characteristic. The argument above is thus valid only in an approximate sense. One may rephrase this as a problem of robustness: namely, when the original signals are not fully band-limited but obey only a certain frequency characteristic, how close should the digital filter be to the ideal low-pass filter?

This type of question has been seldom addressed in the signal processing literature until very recently. However, this can be properly placed in the framework of sampled-data control theory, and there are now several investigations that apply the sampled-data control methodology to digital signal processing.

We formulate a multirate digital signal reconstruction problem under the assumption that the original analog signal is subject to a certain frequency characteristic, but not fully band-limited. Under the assumption we will optimize the analog performance with an H^∞ optimality criterion.

This may also be regarded as an optimal D/A converter design. We will show that performance improvement is possible over conventional low-pass filters. It is also seen that the presented method can be used as a new design method for a low-pass filter.

In this chapter, we first introduce the fundamentals of multirate digital signal processing. The conventional idea of its design is also discussed and we point out that there are some drawbacks in its idea. We then present an alternative method of designing multirate systems, that is, sampled-data H^∞ design. The last section provides design examples and we show the advantages of the present method.

3.2 Interpolators and decimators

In this section, we introduce mathematical definitions of interpolators and decimators. An interpolator or a decimator is implemented with upsamplers $\uparrow M$ and downsamplers $\downarrow M$ respectively. We begin by defining the upsampler and the downsampler.

Definition 3.1. *For discrete-time signal* $\{x[k]\}_{k=0}^{\infty}$*, define the upsampler* $\uparrow M$ *and the*

downsampler $\downarrow M$ *by*

$$\uparrow M : \{x[k]\}_{k=0}^{\infty} \longmapsto \{x[0], \underbrace{0, 0, \dots, 0}_{M-1}, x[1], 0, \dots\},$$
$$\downarrow M : \{x[k]\}_{k=0}^{\infty} \longmapsto \{x[0], x[M], x[2M], \dots\}.$$

The upsampling operation is implemented by inserting $M-1$ equidistant zero-valued samples between two consecutive samples of $x[k]$ before the sampling rate is multiplied by the factor M. Figure 3.1 indicates the upsampling operation.

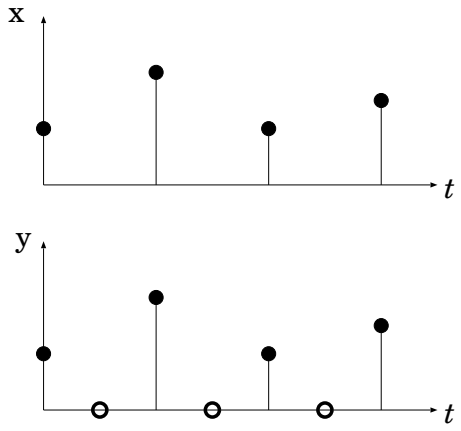

Figure 3.1: Upsampling operation $y = (\uparrow 2)x$

On the other hand, the downsampling operation is implemented by keeping every M-th sample of $x[k]$ and removing in-between samples to generate $y[k]$, then the sampling rate becomes multiplied by $1/M$. This procedure is illustrated in Figure 3.2.

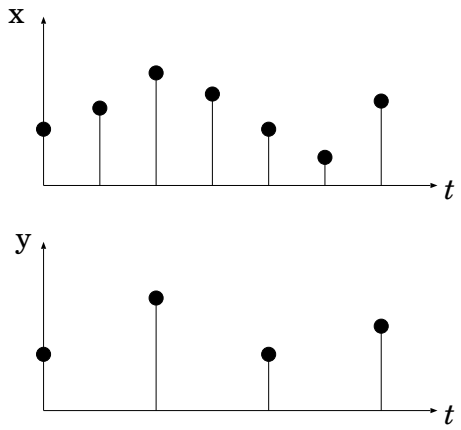

Figure 3.2: Downsampling operation $y = (\downarrow 2)x$

Note that the upsampler is left-invertible (and the downsampler is right-invertible), that is, $(\downarrow M)(\uparrow M) = I$, however $(\uparrow M)(\downarrow M)$ is not the identity. In fact, by $(\uparrow M)(\downarrow M)$

the discrete-time signal $x = \{x[0], x[1], \dots\}$ is converted into

$$(\uparrow M)(\downarrow M)x = \{x[0], \underbrace{0, \dots, 0}_{M-1}, x[M], 0, \dots\} \neq x.$$

To put it differently, downsampling is a lossy data compression, that is, the original signal cannot be perfectly reconstructed from the downsampled signal. On the other hand, we have the duality relation: $(\uparrow M)^* = (\downarrow M)$, $\quad (\downarrow M)^* = (\uparrow M)$, that is, for any signal $x \in l^2$ and $y \in l^2$, we have $\langle(\uparrow M)x, y\rangle = \langle x, (\downarrow M)y\rangle$, $\langle(\downarrow M)x, y\rangle = \langle x, (\uparrow M)y\rangle$, where $\langle\cdot,\cdot\rangle$ denotes the inner product on l^2, that is, $\langle x, y\rangle := \sum_{k=0}^{\infty} y[k]x[k]$.

In addition, an upsampler and a decimator are represented with the discrete-time lifting and its inverse (see Section 2.4):

$$(\uparrow M) = \mathbb{L}_M^{-1} \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^T, \quad (\downarrow M) = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix} \mathbb{L}_M, \tag{3.1}$$

and vice versa:

$$\mathbb{L}_M := (\downarrow M) \begin{bmatrix} 1 & z & \dots & z^{M-1} \end{bmatrix}^T, \quad \mathbb{L}_M^{-1} := \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-M+1} \end{bmatrix} (\uparrow M). \tag{3.2}$$

These relations are used to design interpolators or decimators in Section 3.3 and 3.4.

Having defined the upsampler and the downsampler, we can now explain the interpolator and the decimator.

An interpolator consists of two parts: an upsampler and a digital filter. Figure 3.3 shows the block diagram of an interpolator. First, the upsampler inserts zeros and in-

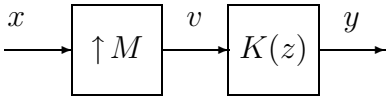

Figure 3.3: Interpolator

creases the sampling rate. Then the filter $K(z)$, called *interpolation filter*, operates on the $M-1$ zero-valued samples inserted by the upsampler $\uparrow M$ to yield nozero values between the original samples, as illustrated in Figure 3.4.

Let us now consider the interpolation in the frequency domain. Assume the sampling period of signal x in Figure 3.3 is 1, that is, the Nyquist frequency is π, and its Fourier transform $X(\omega)$ has characteristic as shown above in Figure 3.5. By upsampling, the Nyquist frequency of the upsampled signal v becomes $2\pi M$, and there occur unwanted frequency imaging components (the shaded portions below in Figure 3.5). In order to remove these imaging components, the frequency response $K(\omega)$ of the interpolation filter must be of low-pass characteristic with cut-off frequency π as shown in Figure 3.5. Therefore, a very sharp low-pass filter close to the ideal one is often used.

On the other hand, a decimator is constructed by using a downsampler and a digital filter, whose block diagram is shown in Figure 3.6. To see the role of the filter $H(z)$, let us look at the frequency domain. Assume the sampling period of signal v in Figure 3.6 is 1

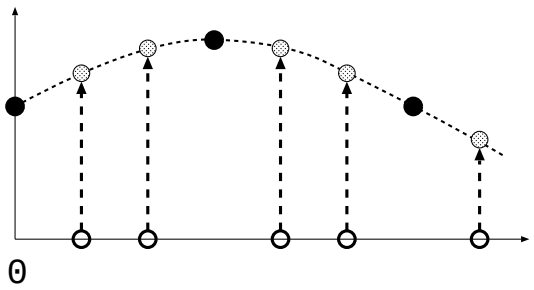

Figure 3.4: Signal interpolation by interpolator

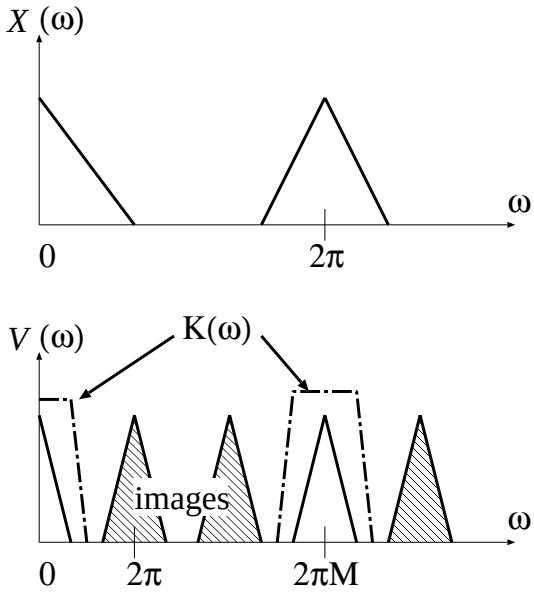

Figure 3.5: Imaging components caused by upsampler

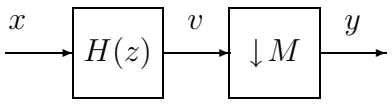

Figure 3.6: Decimator

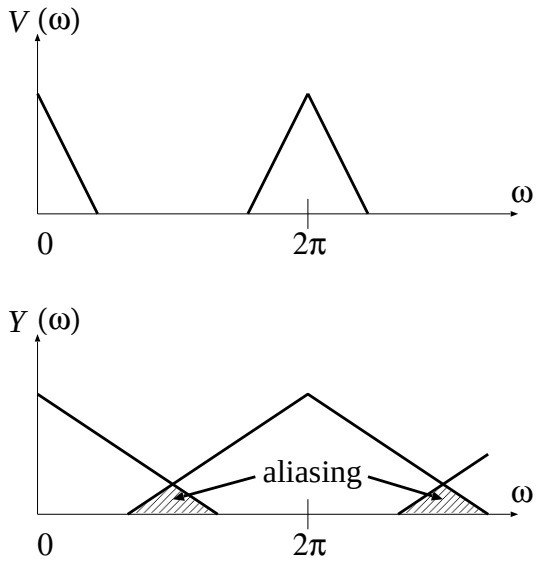

Figure 3.7: Aliasing

(the Nyquist Frequency is π) and its Fourier transform $V(\omega)$ has characteristic as indicated above in Figure 3.7. If $V(\omega)$ is not band-limited to $|\omega| \leq \pi/M$, the spectrum $Y(\omega)$ obtained after downsampling will overlap as shown below in Figure 3.7. This overlapping (the shaded portions below in Figure 3.7) is called *aliasing*.

The filter $H(z)$, called *decimation filter* or *anti-aliasing filter*, is connected before the downsampler to avoid aliases in advance. To eliminate the aliasing completely, the filter $H(z)$ must be the ideal low-pass filter with cut-off frequency π/M. Therefore, similarly to the case of interpolation, a very sharp low-pass filter close to the ideal one is often employed.

As mentioned above, the interpolation filter $K(z)$ or the decimation filter $H(z)$ ideally has the characteristic shown in Figure 3.8. In practice, we cannot realize a filter with

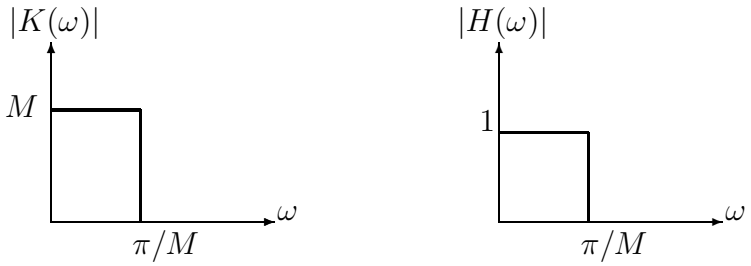

Figure 3.8: Ideal characteristic of interpolation filter $K(z)$ and decimation filter $H(z)$

such a frequency characteristic. Therefore the filter is conventionally designed such that the frequency response approximates that of the ideal filter.

As far as the discrete-time signals are concerned, that is, the spectra of the original analog signals are completely band-limited by the Nyquist frequency, the design may be

correct. However, the real analog signals have the spectra beyond the Nyquist frequency, and hence the ideal characteristic in Figure 3.8 will not be necessarily optimal. In the following sections, we propose an alternative method that takes the analog performance, in particular, the frequency component over the Nyquist frequency, into account.

3.3 Design of interpolators

3.3.1 Problem formulation

We start by formulating a design problem for (sub)optimal interpolators. Consider the block diagram shown in Figure 3.9. The incoming signal w_c first goes through an anti-

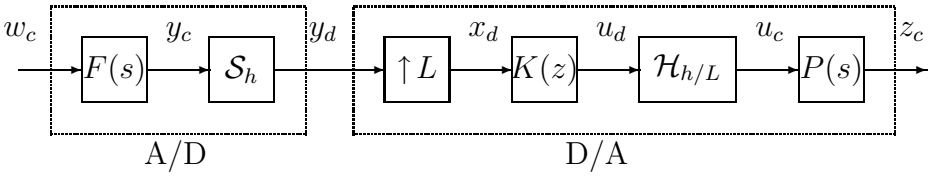

Figure 3.9: Signal reconstruction by interpolator

aliasing filter $F(s)$ and the filtered signal y_c becomes nearly (but not entirely) band-limited. The filter $F(s)$ governs the frequency-domain characteristic of the analog signal y_c. This signal is then sampled by the sampler $\mathcal{S}_h$ to become a discrete-time signal y_d with sampling period h.

To restore y_c we usually let it pass through a digital filter, a hold device and then an analog filter. The present setup however places yet one more step: The discrete-time signal y_d is first upsampled by $\uparrow L$, and becomes another discrete-time signal x_d with sampling period h/L. The discrete-time signal x_d is then processed by a digital filter $K(z)$, becomes a continuous-time signal u_c by going through the zero-order hold $\mathcal{H}_{h/L}$ (that works in sampling period h/L), and then becomes the final signal by passing through an analog filter $P(s)$. An advantage here is that one can use a fast hold device $\mathcal{H}_{h/L}$ thereby making more precise signal restoration possible. The objective here is to design the digital filter $K(z)$ for given $F(s)$, L and $P(s)$.

Figure 3.10 shows the block diagram of the error system for the design. The delay in the upper portion of the diagram corresponds to the fact that we allow a certain amount of time delay for signal reconstruction. Let $\mathcal{T}_I$ denote the input/output operator from w_c to $e_c := z_c(t) - u_c(t - mh)$. Our design problem is as follows:

Problem 3.1. *Given a stable, strictly proper $F(s)$, stable, proper $P(s)$, upsampling factor $L \in \mathbb{N}$, delay step $m \in \mathbb{N}$, sampling period $h > 0$ and an attenuation level $\gamma > 0$, find a digital filter $K(z)$ such that*

$$\|\mathcal{T}_I\| := \sup_{\substack{w_c \in L^2[0,\infty) \\ w_c \neq 0}} \frac{\|\mathcal{T}_I w_c\|_{L^2[0,\infty)}}{\|w_c\|_{L^2[0,\infty)}} < \gamma. \tag{3.3}$$

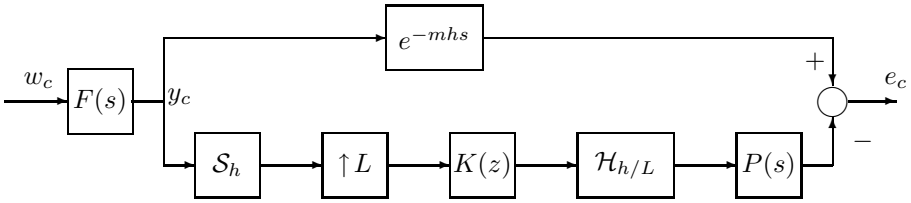

Figure 3.10: Signal reconstruction error system

The norm in (3.3) is an L^2 induced norm, which is equal to the H^∞-norm of the system $\mathcal{T}_I$, so Problem 3.1 is the H^∞ optimization problem.

3.3.2 Reduction to a finite-dimensional problem

A difficulty in Problem 3.1 is that it involves a continuous-time delay, and hence it is an infinite-dimensional problem. Another difficulty is that it contains the upsampler $\uparrow L$ so that it makes the overall system time-varying (to be precise, periodically time-varying).

However we can reduce the problem to a finite-dimensional single-rate one. Let $\{A_F, B_F, C_F, 0\}$ be a realization of $F(s)$, that is, the state equation of $F(s)$:

$$\dot{x}_F = A_F x_F + B_F w_c, \quad y_c = C_F x_F.$$

Theorem 3.1. *For the error system $\mathcal{T}_I$, there exist (finite-dimensional) discrete-time systems $\{T_{I,N} : N = L, 2L, \dots\}$ such that*

$$\lim_{N\to\infty} \|T_{I,N}\| = \|\mathcal{T}_I\|. \tag{3.4}$$

Proof. We first reduce the problem to a single-rate one. Recall the property (3.2) of the discrete-time lifting $\mathbb{L}_L$ and its inverse $\mathbb{L}_L^{-1}$:

$$\mathbb{L}_L := (\downarrow L) \begin{bmatrix} 1 & z & \dots & z^{L-1} \end{bmatrix}^T, \quad \mathbb{L}_L^{-1} := \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-L+1} \end{bmatrix} (\uparrow L).$$

Then $K(z)(\uparrow L)$ can be rewritten as

$$K(z)(\uparrow L) = \mathbb{L}_L^{-1} \widetilde{K}(z), \quad \widetilde{K}(z) := \mathbb{L}_L K(z) \mathbb{L}_L^{-1} \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^T.$$

The lifted system $\widetilde{K}(z)$ is an LTI, single-input/L-output system that satisfies

$$K(z) = \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-L+1} \end{bmatrix} \widetilde{K}(z^L).$$

Using the generalized hold $\widetilde{\mathcal{H}}_h$ defined by

$$\widetilde{\mathcal{H}}_h : l^2 \ni \boldsymbol{v} \mapsto u \in L^2, \quad u(kh + \theta) = \mathbf{H}(\theta)\boldsymbol{v}[k], \quad \theta \in [0, h), \quad k = 0, 1, 2, \dots,$$

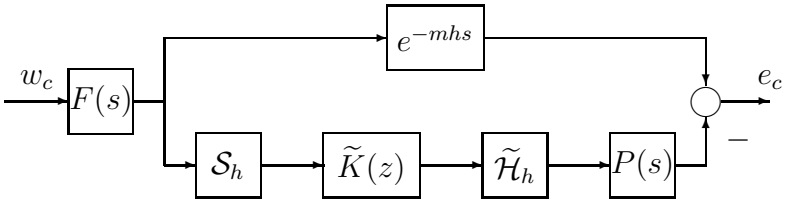

Figure 3.11: Reduced single-rate problem

where $\mathbf{H}(\cdot)$ is the hold function:

$$\mathbf{H}(\theta) := \begin{cases} \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix}, \theta \in [0, h/L), \\ \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \end{bmatrix}, \theta \in [h/L, 2h/L), \\ \qquad\qquad \dots \\ \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \end{bmatrix}, \theta \in [(L-1)h/L, h), \end{cases}$$

we obtain the identity

$$\mathcal{H}_{h/L}\mathbb{L}_L^{-1} = \widetilde{\mathcal{H}}_h.$$

This yields

$$\mathcal{H}_{h/L}K(z)(\uparrow L)\mathcal{S}_h = \widetilde{\mathcal{H}}_h\widetilde{K}(z)\mathcal{S}_h.$$

Hence Figure 3.10 is equivalent to Figure 3.11.

We modify the diagram in Figure 3.11 into the diagram in Figure 3.12, and we in-

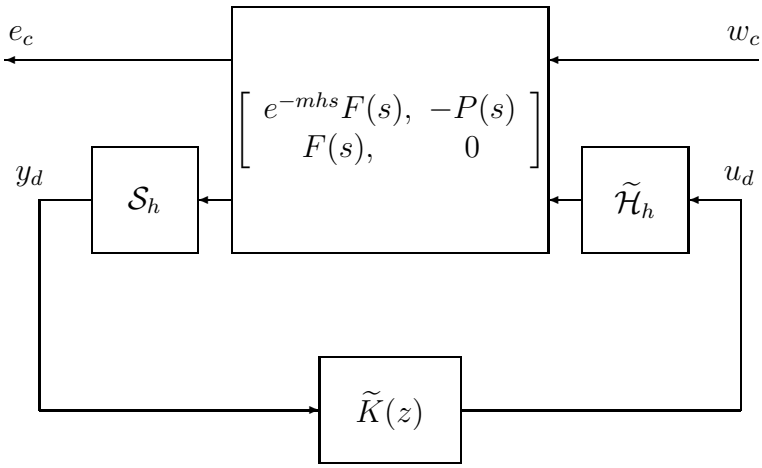

Figure 3.12: Sampled-data system $\mathcal{T}_I$

troduce the fast-sampling/fast-hold approximation [19, 43] in order to obtain a finite-dimensional discrete-time system approximately. Figure 3.13 illustrates the procedure.

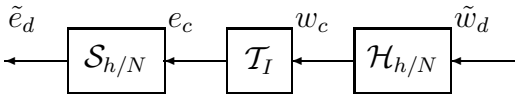

Figure 3.13: Fast-sampling/fast-hold discretization

By the fast-sampling/fast-hold approximation, we obtain the approximated discrete-time system $T_{I,N}$ ($N := Ll,\ l \in \mathbb{N}$),

$$T_{I,N}(z) = z^{-m} G_{I11,N}(z) + G_{I12,N}(z) \widetilde{K}(z) G_{I21,N}(z),$$

where

$$G_{I11,N} := z^{-m} \mathcal{L}_N(F), \quad G_{I12,N} := -\mathcal{L}_N(P) H, \quad G_{I21,N} := S \mathcal{L}_N(F),$$

$$S := [1, \underbrace{0, \ldots, 0}_{N-1}], \quad H := \underbrace{\begin{bmatrix} q & & \\ & \ddots & \\ & & q \end{bmatrix}}_{L}, \quad q := [\underbrace{1, \ldots, 1}_{l}]^T.$$

Figure 3.14 shows the obtained discrete-time system, where $\widetilde{K}(z)$ is an LTI, single-

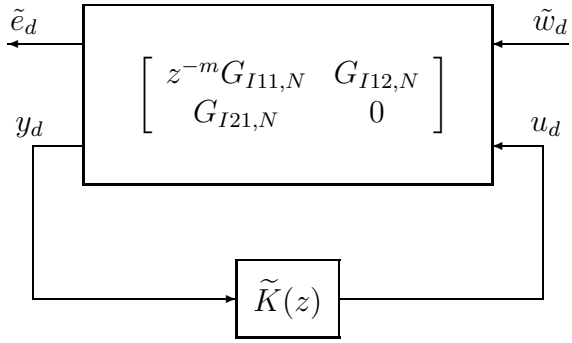

Figure 3.14: Discrete-time system $T_{I,N}$

input/L-output system that satisfies

$$K(z) = \begin{bmatrix} 1 & z^{-1} & \cdots & z^{-L+1} \end{bmatrix} \widetilde{K}(z^L).$$

The convergence of (3.4) is guaranteed in [43]. □

3.4 Design of decimators

3.4.1 Problem formulation

We now formulate a design problem for optimal decimators. While this can be considered dually with interpolators, it is less studied in the literature. Downsampling occurs usually in the filter bank design, and its independent design has received less attention.

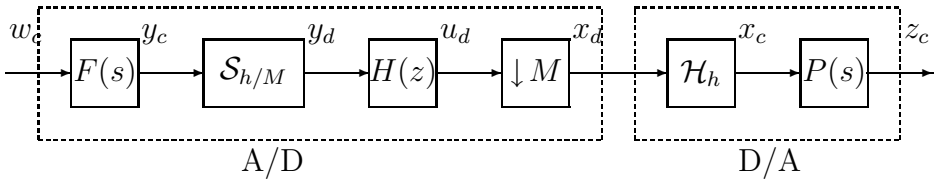

Figure 3.15: Signal reconstruction with decimator

Consider the block diagram Figure 3.15. The incoming signal w_c first goes through an anti-aliasing filter $F(s)$ and the filtered signal y_c becomes nearly (but not entirely) band-limited. This signal is then sampled by $\mathcal{S}_{h/M}$ to become a discrete-time signal y_d with sampling period h/M.

The discrete-time signal y_d is first processed by a digital filter $H(z)$. Then the filtered signal x_d is downsampled by $\downarrow M$, and becomes another discrete-time signal u_d with sampling period h. The discrete-time signal u_d then becomes a continuous-time signal u_c by going through the zero-order hold $\mathcal{H}_h$, and then becomes the final signal by going through an analog filter $P(s)$. The objective here is to design the digital filter $H(z)$ for given $F(s)$, M and $P(s)$.

Figure 3.16 shows the block diagram of the error system for the design. The delay in the upper portion of the diagram corresponds to the fact that we allow a certain amount of time delay for signal reconstruction. Let $\mathcal{T}_D$ denote the input/output operator from w_c

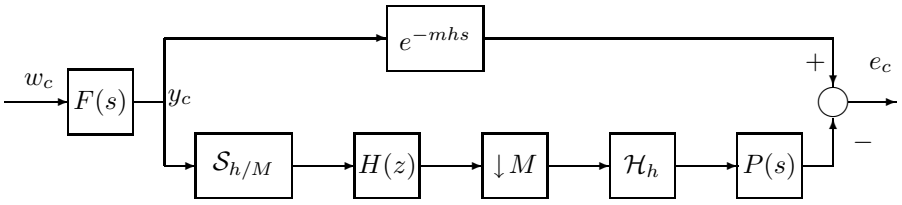

Figure 3.16: Signal reconstruction error system

to e_c in Figure 3.16. Our design problem is then as follows:

Problem 3.2. *Given a stable, strictly proper $F(s)$, stable, proper $P(s)$, downsampling factor $M \in \mathbb{N}$, delay step $m \in \mathbb{N}$, sampling period $h > 0$ and an attenuation level $\gamma > 0$, find a digital filter $H(z)$ such that*

$$\|\mathcal{T}_D\| := \sup_{\substack{w_c \in L^2[0,\infty) \\ w_c \neq 0}} \frac{\|\mathcal{T}_D w_c\|_{L^2[0,\infty)}}{\|w_c\|_{L^2[0,\infty)}} < \gamma. \tag{3.5}$$

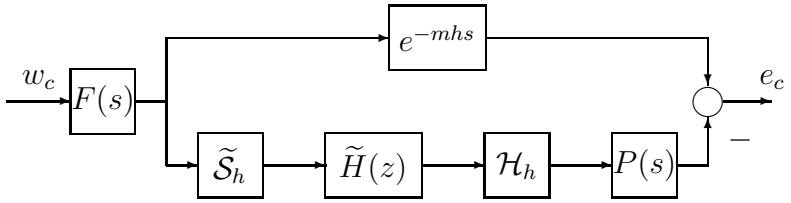

Figure 3.17: Reduced single-rate problem

3.4.2 Reduction to a finite-dimensional problem

Theorem 3.2. *For the error system $\mathcal{T}_D$, there exist (finite-dimensional) discrete-time systems $\{T_{D,N} : N = M, 2M, \dots\}$ such that*

$$\lim_{N\to\infty} \|T_{D,N}\| = \|\mathcal{T}_D\|. \tag{3.6}$$

Proof. Using the discrete-time lifting $\mathbb{L}_M$ we rewrite $(\downarrow M)H(z)$ as

$$(\downarrow M)H(z) = \widetilde{H}(z)\mathbb{L}_M, \quad \widetilde{H}(z) := \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} \mathbb{L}_M H(z) \mathbb{L}_M^{-1},$$

where the system $\widetilde{H}(z)$ is an LTI, M-input/single-output system that satisfies

$$H(z) = \widetilde{H}(z^M) \begin{bmatrix} 1 & z & \dots & z^{M-1} \end{bmatrix}^T.$$

Using the generalized sampler $\widetilde{\mathcal{S}}_h$ defined by

$$\widetilde{\mathcal{S}}_h : L^2 \ni u \longmapsto \boldsymbol{v} \in l^2, \quad \boldsymbol{v}[k] := \begin{bmatrix} u\big(kh\big) \\ u\big(kh + h/M\big) \\ \vdots \\ u\big(kh + (M-1)h/M\big) \end{bmatrix}, \quad k = 0, 1, 2, \dots,$$

we obtain the identity

$$\mathbb{L}_M \mathcal{S}_{h/M} = \widetilde{\mathcal{S}}_h.$$

Hence Figure 3.16 is equivalent to Figure 3.17. As has been mentioned above, this can be reduced to a finite-dimensional discrete-time system.

We modify the block diagram in Figure 3.17 into the block diagram in Figure 3.18. Then by using the fast-sampling/fast-hold method, the sampled-data system in Figure 3.18 is approximated to the following discrete-time system $T_{D,N}$ ($N := Ml$, $l \in \mathbb{N}$),

$$T_{D,N}(z) = z^{-m} G_{D11,N}(z) + G_{D12,N}(z) \widetilde{K}(z) G_{D21,N}(z),$$

where

$$G_{D11,N} := z^{-m}\mathcal{L}_N(F), \quad G_{D12,N} := -\mathcal{L}_N(P)H, \quad G_{D21,N} := S\mathcal{L}_N(F),$$

$$S := \left.\begin{bmatrix} p & & \\ & \ddots & \\ & & p \end{bmatrix}\right\} M, \quad p := [1, \underbrace{0, \dots, 0}_{l-1}], \quad H := [\underbrace{1, \dots, 1}_{N}]^T.$$

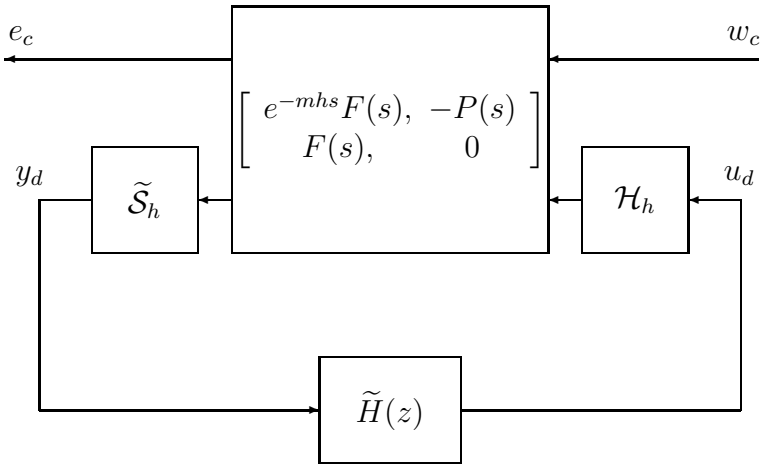

Figure 3.18: Sampled-data system $\mathcal{T}_D$

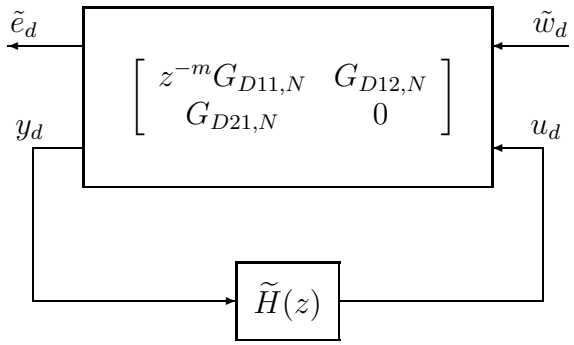

Figure 3.19: Discrete-time system $T_{D,N}$

Figure 3.19 shows the obtained finite-dimensional discrete-time system.

The convergence of (3.6) is guaranteed in [43]. □

Note that the decimation filter

$$H(z) = \widetilde{H}(z^M) \begin{bmatrix} 1 & z & \dots & z^{M-1} \end{bmatrix}^T,$$

may not be causal, thus we adopt the following filter:

$$H(z) = z^{-M} \widetilde{H}(z^M) \begin{bmatrix} 1 & z & \dots & z^{M-1} \end{bmatrix}^T.$$

3.5 Design of sampling rate converters

By combining interpolators and decimators, we can construct a sampling rate converter. Figure 3.20 shows a sampling rate converter, where an interpolation with factor M_1 is followed by a decimation with factor M_2. By this converter, the sampling rate of the input signal is changed by the factor M_1/M_2. In the application of digital audio, the

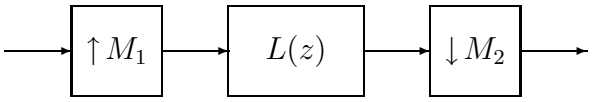

Figure 3.20: Sampling rate converter ($M_1 : M_2$)

conversion from CD signals at 44.1kHz to DAT signals at 48kHz is realized by a converter with the factors $M_1 = 3 \times 7^2$ and $M_2 = 2^5 \times 5$. Conventionally, the digital filter $L(z)$ is designed to be a low-pass filter with the cut-off frequency $\omega = \pi/M, \quad M := \max(M_1, M_2)$ [9, 35]. In this section, we design the filter $L(z)$ that combines the interpolation filter $H(z)$ designed by the method discussed in Section 3.3 and the decimation filter $H(z)$ in Section 3.4. The designed sampling rate converter will be in the form illustrated in Figure 3.21 The block diagrams of the error system for sampled-data H^∞ design of an interpolation

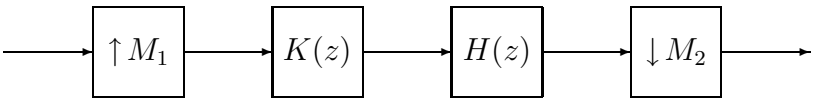

Figure 3.21: Sampling rate converter with interpolator and decimator

filter $K(z)$ and a decimation filter $H(z)$ are shown in Figure 3.22 and in Figure 3.23, respectively. In these diagrams, $h_2 = h \cdot \frac{M_2}{M_1}$ and the analog low-pass filters $F_1(s)$ and $F_2(s)$ have characteristics illustrated in Figure 3.24. The filters F_1 and F_2 take account of the characteristic of the analog input signal, and hence we can design filters $K(z)$ and $H(z)$ that optimize the analog performance using the sampled-data system design method.

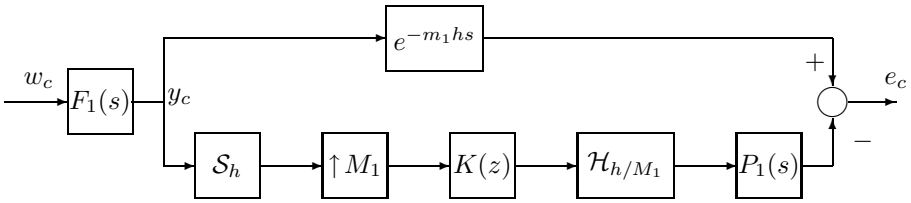

Figure 3.22: Error system for designing interpolator

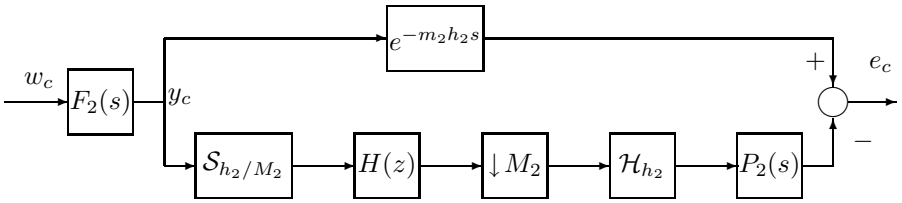

Figure 3.23: Error system for designing decimator

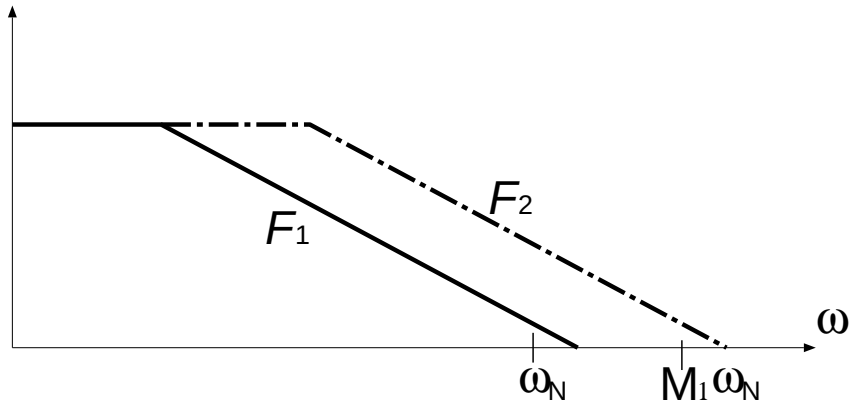

Figure 3.24: Characteristic of F_1 and F_2

The advantage of the design mentioned above is that we can design a converter with large M_1 or M_2. For example, to design a converter for CD/DAT (i.e., $M_1 = 3 \times 7^2$ and $M_2 = 2^5 \times 5$), we first design interpolators with $M = 3, 7$ and decimators with $M = 2, 5$ as shown in Figure 3.25, and then by combining the interpolation filters $K_3(z)$, $K_7(z)$ and

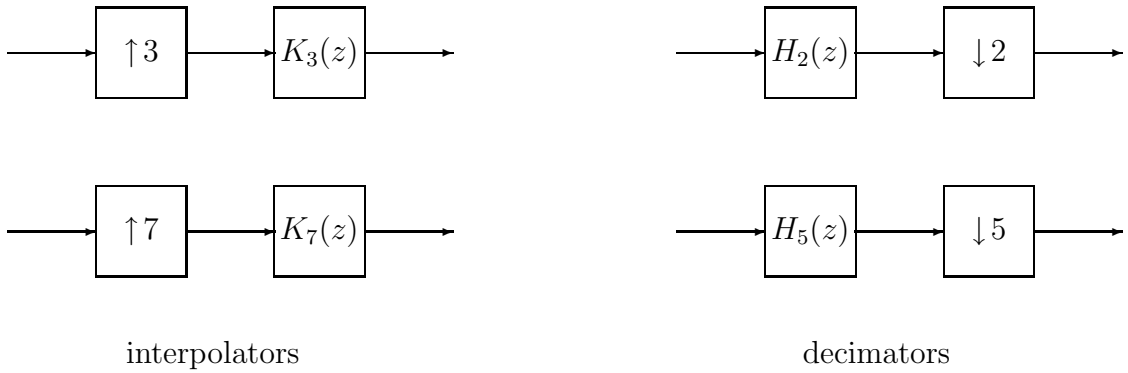

Figure 3.25: Interpolators and decimators for CD/DAT sampling rate conversion

the decimation filters $H_2(z)$, $H_5(z)$, we obtain the sampling rate converter as illustrated in Figure 3.26.

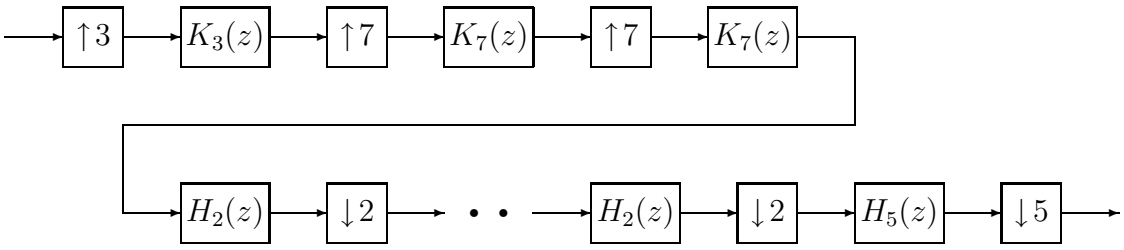

Figure 3.26: Sampling rate converter for CD/DAT

On the other hand, there is a design method for sampling rate converters by periodically time-varying systems [15]. However, the order of the design in the method will be $M_1 \times M_2$, and hence the design of a converter with very large number of M_1 or M_2 such as CD/DAT ($M_1 \times M_2 = 23520$) has much difficulty in numerical computation.

3.6 Design examples

3.6.1 Design of interpolators

In this section, we compare the interpolator designed by the method discussed in Section 3.3 with that designed by Johnston's method [17]. The parameters are as follows: inter-

polation ratio $M = 2$, sampling period $h = 1$ and delay step $m = 2$. The analog filters $F(s)$ and $P(s)$ are

$$F(s) = \frac{1}{(Ts+1)(0.1Ts+1)}, \quad T := 22.05/\pi \approx 7.0187, \quad P(s) = 1.$$

Note that the low-pass filter $F(s)$ has first order attenuation in the frequency range $\omega \in [0.14248, 1.4248]$ [rad/sec], and second order attenuation in the range $\omega > 1.4248$ [rad/sec]. The filter simulates the frequency energy distribution of a typical orchestral music. The Johnston filter is taken to be of order 31.

The frequency responses of the obtained filters are shown in Figure 3.27. The sampled-data design filter is of order 7, which is lower than the Johnston filter. The Johnston filter

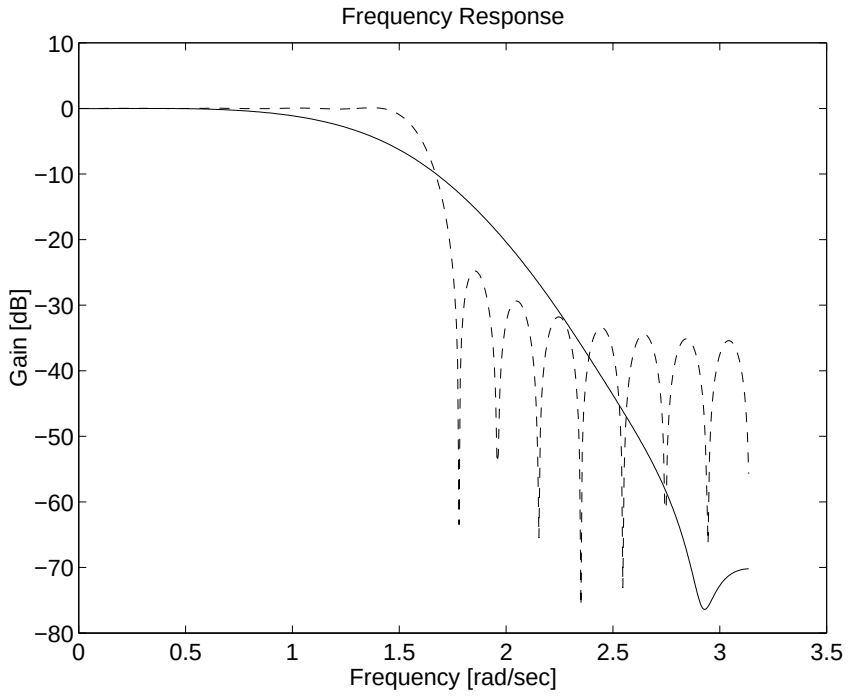

Figure 3.27: Frequency responses of interpolation filters: sampled-data design (solid) and Johnston filter (dash)

shows the sharper decay beyond the cut-off frequency $\omega = \pi/2$, while the filter obtained by the sampled-data design shows a rather slow decay.

On the other hand, the reconstruction error (see Figure 3.10) characteristic in Figure 3.28 exhibits quite an admirable performance in spite of the low-order of the sampled-data design filter. It is almost comparable with 31st order Johnston filter.

While for those frequencies much below the cut-off frequency the gain characteristic of the sampled-data design is not as good as the Johnston filter, the sampled-data designed filter need not be inferior. To see this, let us see the time responses against rectangular waves in Figure 3.29 (sampled-data designed) and in Figure 3.30 (Johnston filter).

The Johnston filter exhibits a very typical Gibbs phenomenon (i.e., we can see ringing caused by the sharp characteristic of the filter), whereas the one by the sampled-data

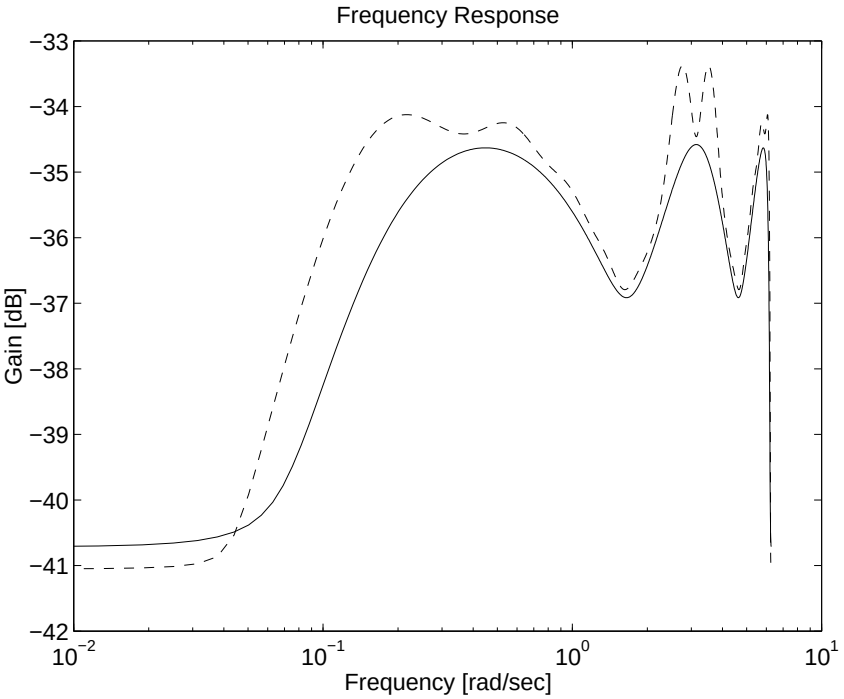

Figure 3.28: Frequency responses of error system: sampled-data design (solid) and Johnston filter design (dash)

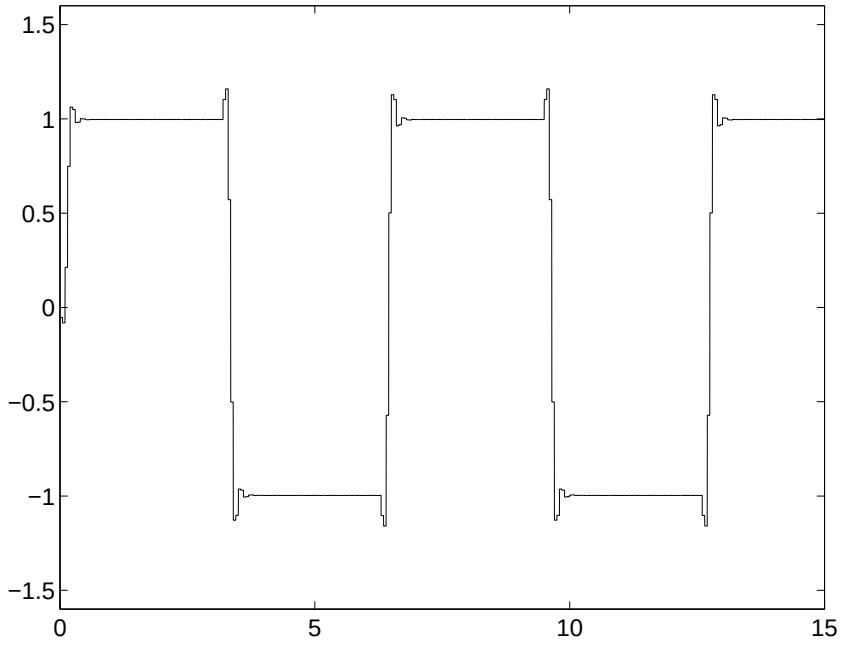

Figure 3.29: Time response (sampled-data design)

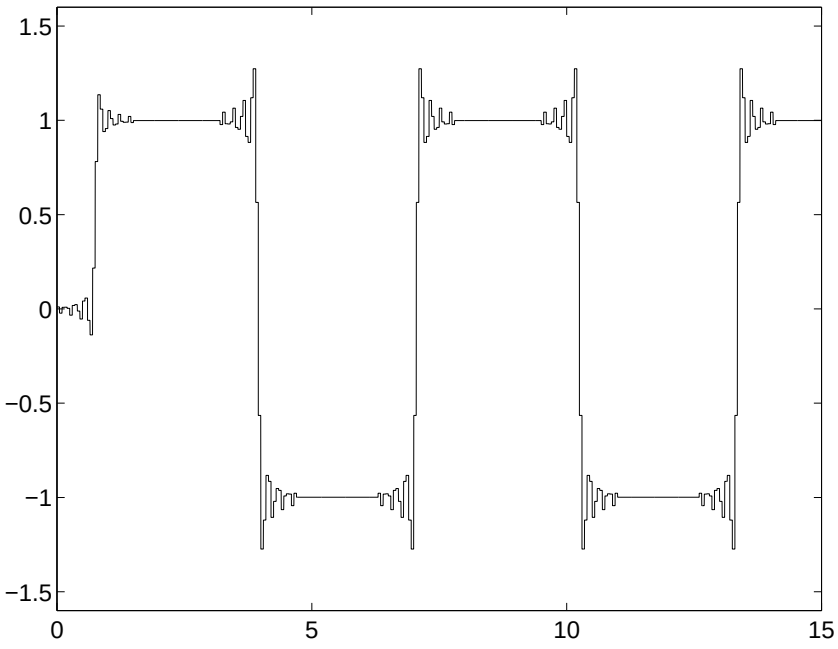

Figure 3.30: Time response (Johnston filter)

design shows a much smaller peak near the edge. We also note that the sampled-data designed filter has nearly linear phase as shown in Figure 3.31.

3.6.2 Design of decimators

We now present an example of the H^∞ design of decimators discussed in Section 3.4. For comparison, we take the Johnston filter of order 31.

Let the decimation ratio $M = 2$ and the other parameters is the same as the interpolator design in the previous section.

Figure 3.32 shows the frequency response of the decimation filters. Note that the filter designed by the sampled-data method is of order 6. The Johnston filter shows the sharper decay beyond the cut-off frequency $\omega = \pi/2$, while the filter by sampled-data design shows a rather slow decay.

Figure 3.33 shows the frequency response of the error system (see Figure 3.16). We can see from the frequency response of the error system that the decimator designed by the sampled-data method exhibits a clear advantage over all frequency range, even though the sampled-data designed filter is of lower order than the Johnston filter. In particular, around the frequency $\omega \approx 4$ [rad/sec], the difference is about 20 dB.

Figure 3.34 and Figure 3.35 shows the time responses against rectangular waves. The sampled-data designed decimator reconstructs the rectangular wave well, while the decimator with the Johnston filter exhibits a large amount of ringing due to the Gibbs phenomenon.

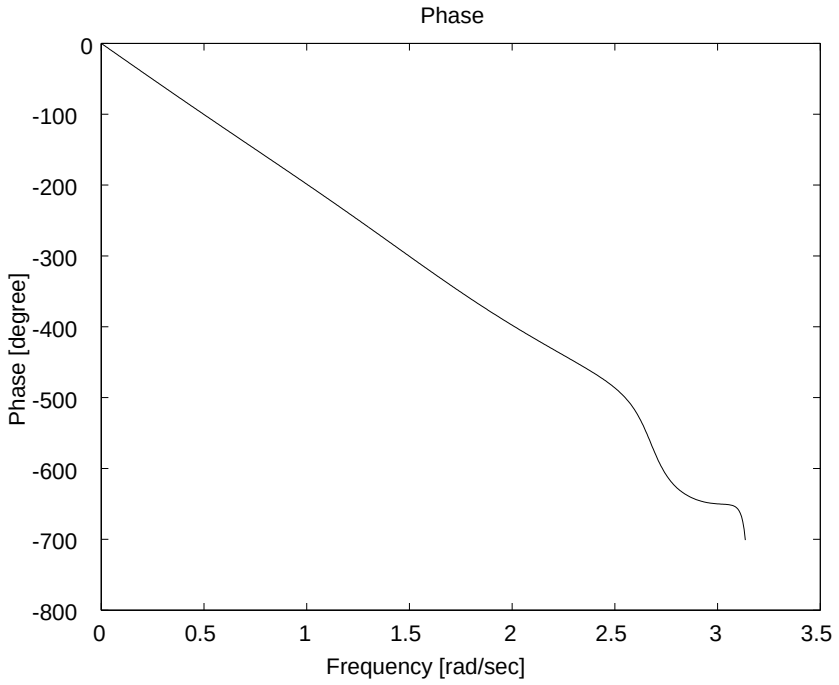

Figure 3.31: Phase response (Sampled-data design)

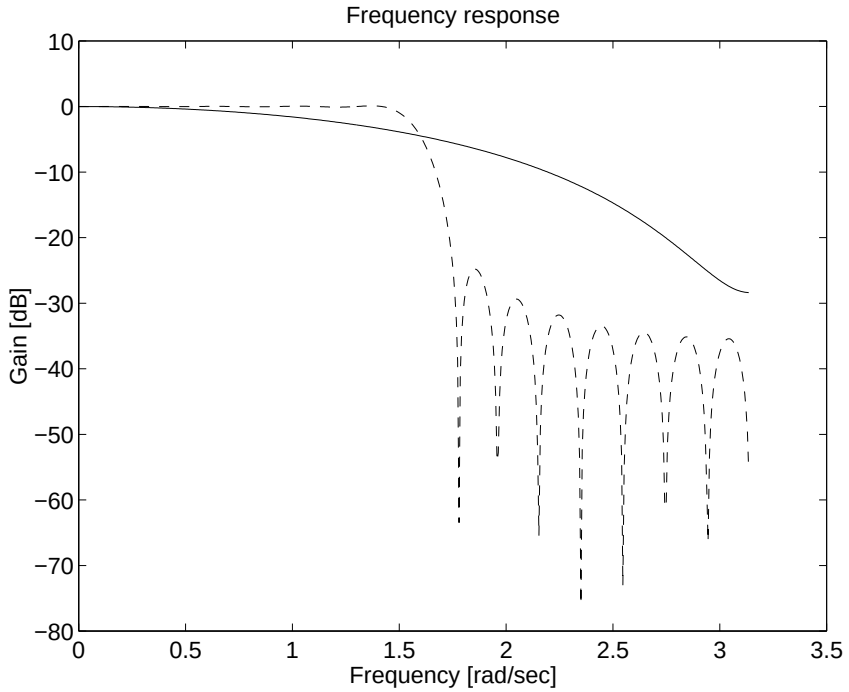

Figure 3.32: Frequency responses of decimation filters: sampled-data design (solid) and Johnston filter (dash)

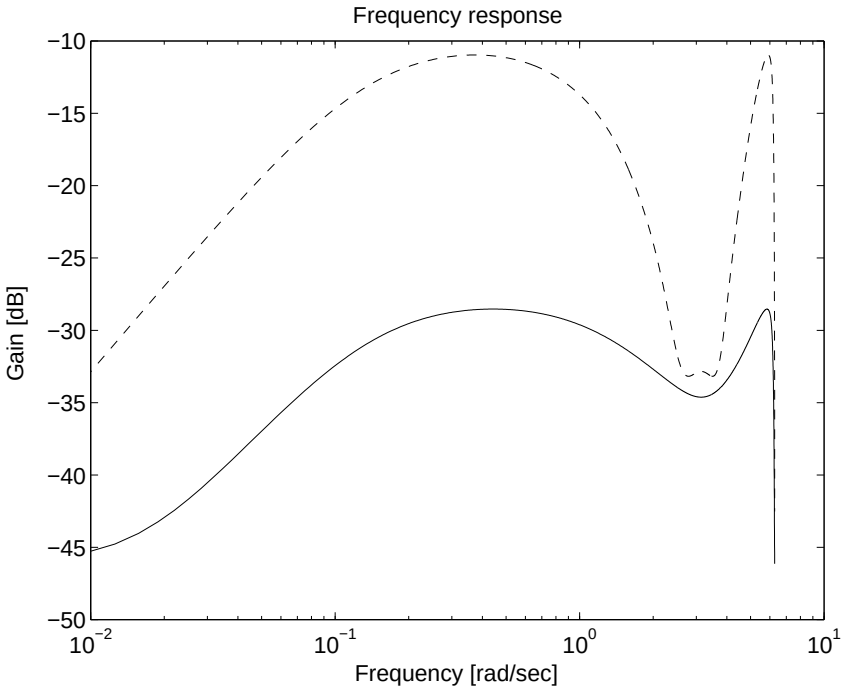

Figure 3.33: Frequency responses of error system: sampled-data design (solid) and Johnston filter design (dash)

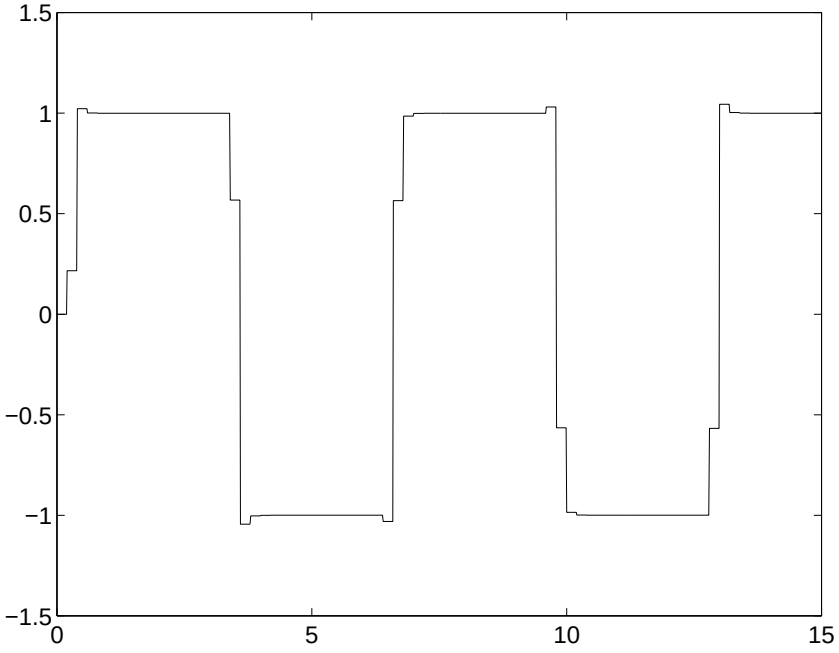

Figure 3.34: Time response (sampled-data design)

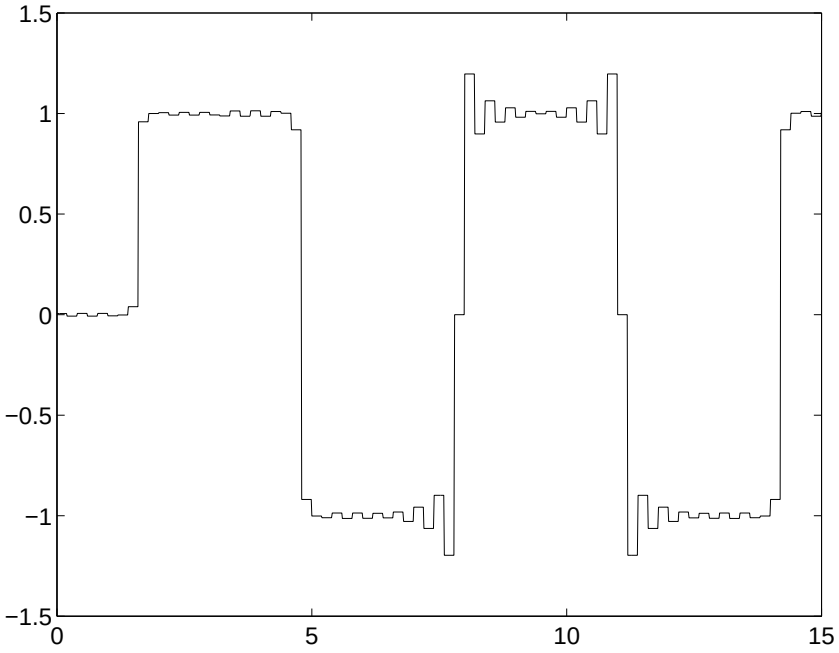

Figure 3.35: Time response (Johnston filter)

3.6.3 Design of sampling rate converters

In this section, we present a design example for the case of changing the sampling period from $h_1 = 1$ to $h_2 = 4/3$. Then we have the sampling rate converter with $M_1 = 3$ and $M_2 = 4$ that are coprime (see Figure 3.20). Let the filter for the interpolator design (see Figure 3.22) be

$$F_1(s) = \frac{1}{(Ts+1)(0.1Ts+1)}, \quad P_1(s) = 1,$$

and that for the decimator design (see Figure 3.23) be

$$F_2(s) = \frac{1}{(T_2 s+1)(0.1T_2 s+1)}, \quad P_2(s) = 1,$$

where $T := 22.05/\pi, \quad T_2 := T/M_1$. The filters simulate the frequency energy distribution of a typical orchestral music, which are observed by FFT analysis of analog records of some orchestral musics.

An approximate design is executed here for $N = M_1 \times 4 = 12$ (interpolator) and $N = M_2 \times 4 = 16$ (decimator). For comparison, we compare it with the equiripple filter obtained by Parks-McClellan method [35, 46] of order 31. Parks-McClellan method is widely used for designing FIR filters [46]. The delay stemps m_1 and m_2 are 2.

The obtained (sub) optimal interpolation filter $K(z)$ is of order 11 and the decimation filter $H(z)$ of order 15. The sampling rate conversion filter $L(z) = H(z)K(z)$ is of order 22[1)].

[1)]The order of $L(z)$ is reduced by the minimal realization method.

Figure 3.36 shows the gain characteristics of these filters. The equiripple filter shows the sharper decay beyond the cut-off frequency ($\pi/4$ [rad/sec]) while the sampled-data design shows a rather mild cut-off characteristic.

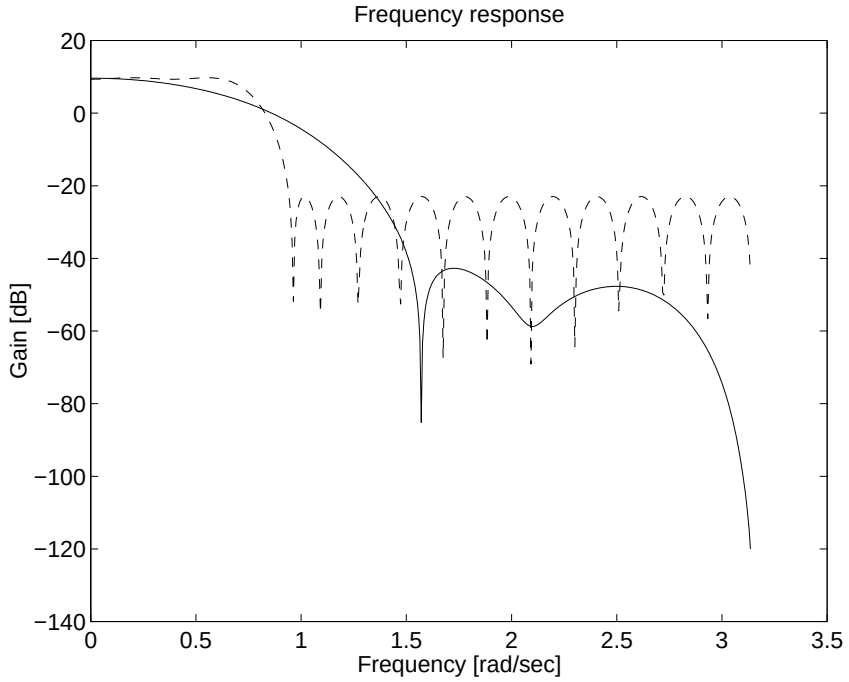

Figure 3.36: Frequency responses of sampling rate conversion filters: sampled-data design $L(z)$ (solid) and quiripple filter $L_e(z)$ (dash)

In spite of these superficial differences, the frequency response of the error system (illustrated in Figure 3.37, where $m := m_1 + m_2$ and $P(s) = P_1(s) = P_2(s) = 1$) of sampling rate converter exhibits quite an admirable performance of the sampled-data design as shown in Figure 3.38.

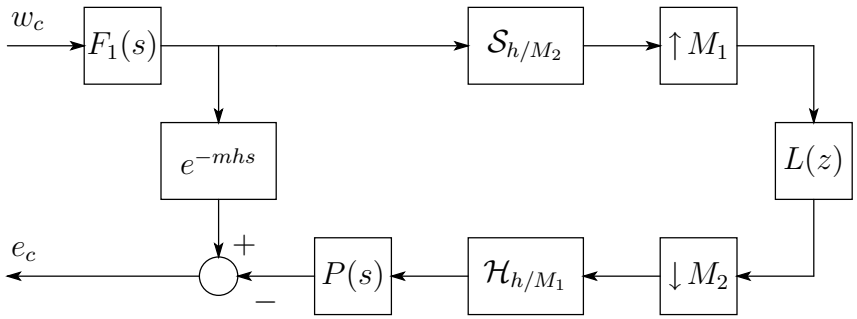

Figure 3.37: Error system of sampling rate converter

It is interesting to observe that the slow decay need not yield an inferior design. To see this, let us see the time responses against a rectangular wave in Figure 3.39 and Figure

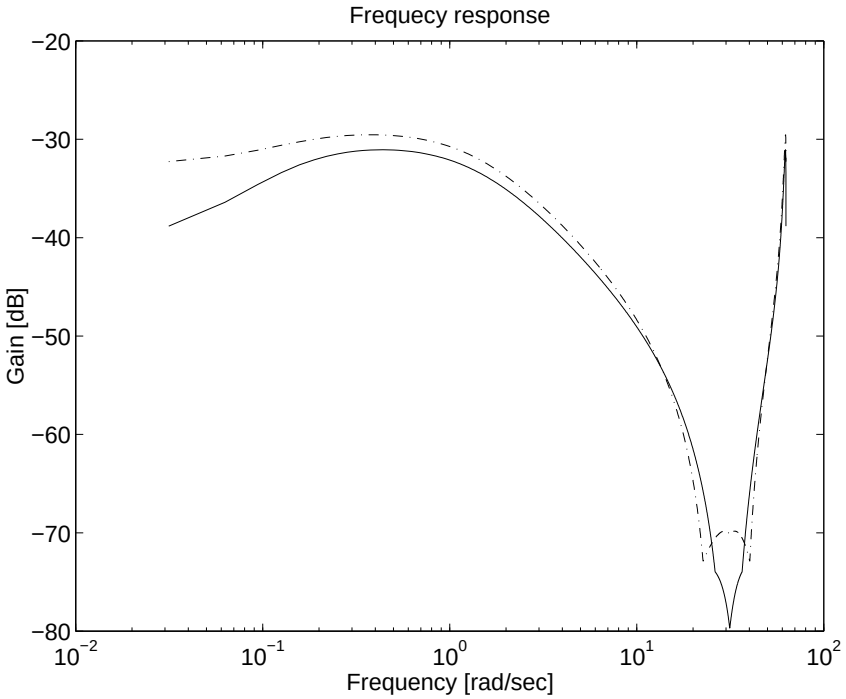

Figure 3.38: Frequency responses of error system: sampled-data design (solid) and equiripple design (dash)

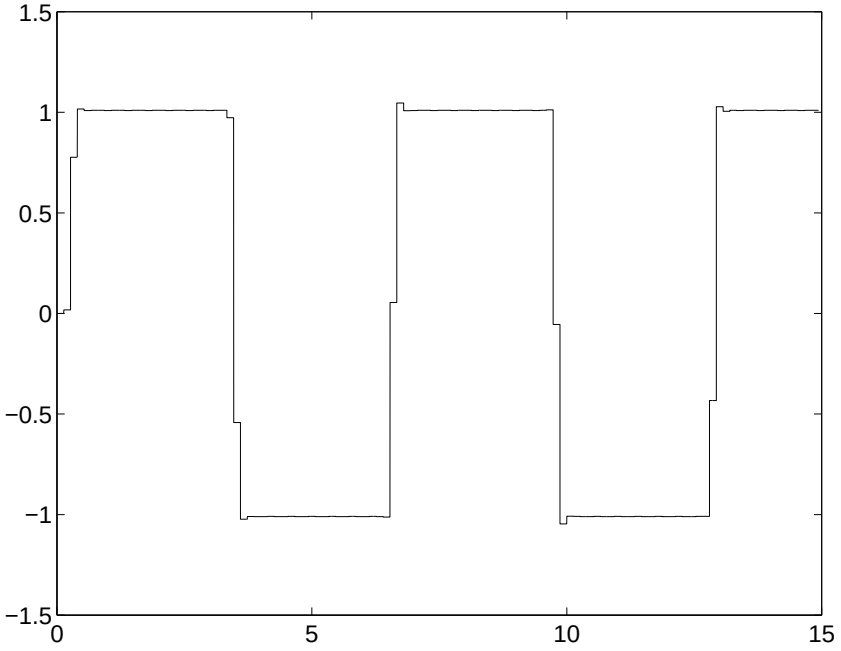

Figure 3.39: Time response (sampled-data design)

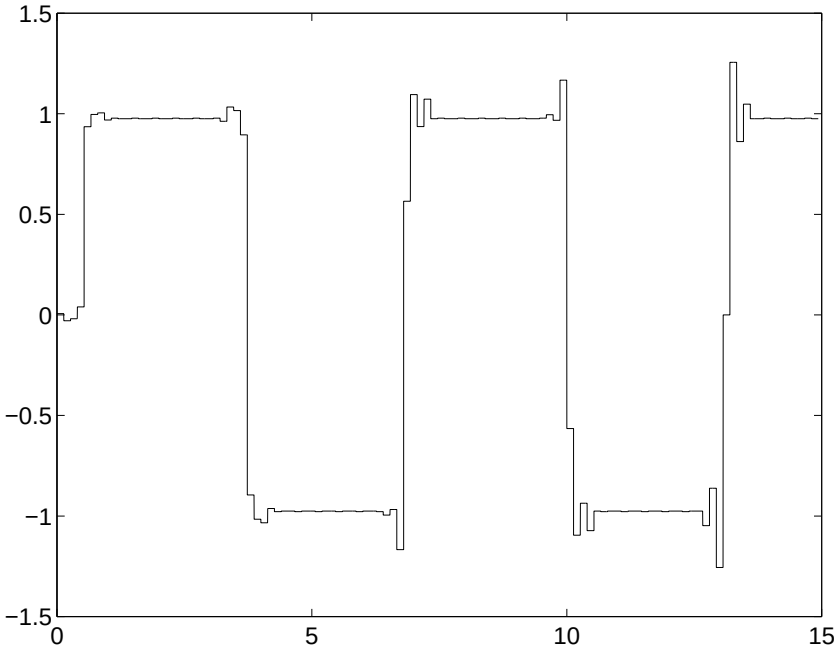

Figure 3.40: Time response (equiripple design)

3.40. The response with the equiripple filter shows a large amount of ringing, whereas that with the filter by the sampled-data design has much less peak around the edge. Note also that $L(z)$ is nearly linear phase up to a certain frequency as shown in Figure 3.41.

3.7 Conclusion

Conventional theories of digital signal processing assert that the ideal filter is the best for interpolation or decimation. However, as we have shown above, a sharp filter characteristic approximating the ideal filter does not necessarily behave well. In particular, such a filter often exhibits a large amount of ringing as illustrated in the previous section. The ringing is due to the Gibbs phenomenon, which is caused by the sharp characteristic of the filter.

On the other hand, our filter shows a slow decay. The reason is that due to the underlying analog characteristic (i.e., $F(s)$), there is an important information content beyond the Nyquist frequency, and such a slow decay is necessary to recover such information.

Moreover, conventional design requires us to give a filter order in advance. The higher the order is, the closer to the ideal characteristic the filter is, and hence filters of a very high order are often used.

In contrast to the conventional design, our method is free from the choice of filter order. Namely, the order of designed filter depends on the order of filter $F(s)$ and $P(s)$, delay step m and upsampling (or downsampling) ratio M. As indicated in the previous section, in spite of the fact that the order of the obtained filter is not very high, the response is better than high-order conventional filters. This fact cannot be recognized without considering the analog performance.

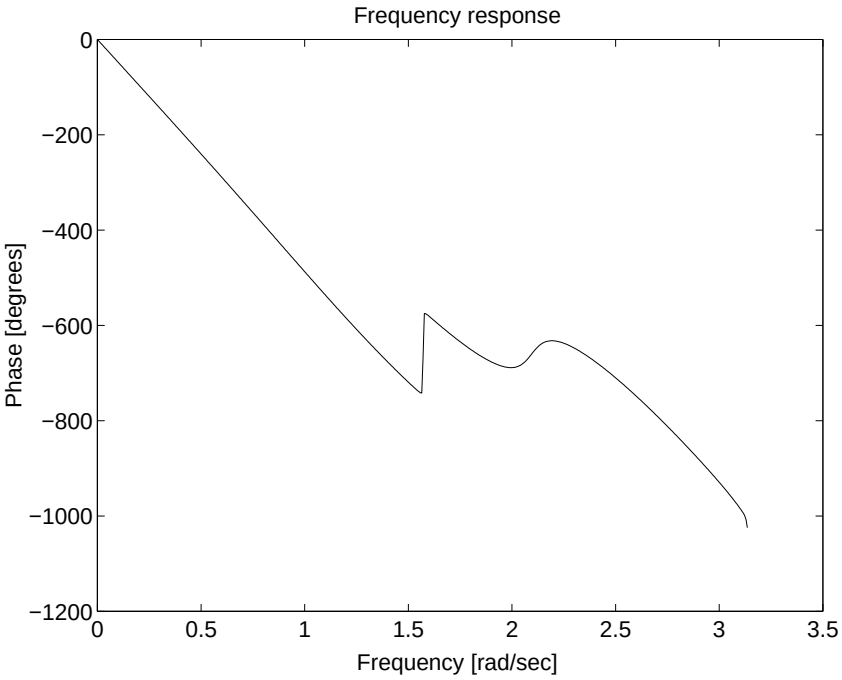

Figure 3.41: Phase plot of $L(z)$

Chapter 4

Application to Communication Systems

4.1 Introduction

The importance of digital communication is ever increasing owing to the rapid growth of the Internet, cellular phones, and so on [32]. In digital communication, especially in pulse amplitude modulation (PAM) or in pulse code modulation (PCM), the analog signal to be transmitted is sampled and becomes a discrete-time signal. In the conventional method, the analog characteristics of the signals are not considered, and hence the total system is regarded as a discrete-time system. Namely, one usually assumes that the original analog signal is band-limited up to the Nyquist frequency.

In [8], a discrete-time H^∞ design of receiving filters or equalizers is introduced. This design is based on the assumption of full band-limitation, but in reality no signals are fully band-limited. Moreover, it is difficult to attenuate both the signal reconstruction error and the distortion caused by the channel only by equalizing after receiving. Therefore an enhancer (or transmitting filter) that amplifies the signal before transmission is often attached in order to increase signal-to-noise ratio [32].

In this chapter, we propose a new design of receiving/transmitting filters by using the sampled-data control theory. Moreover, we introduce the H^∞ method that takes account of a tradeoff between the quality of signal reconstruction and the cost (i.e., the amount of energy of transmitting signals) with an appropriate weighting function. Design examples are presented to illustrate the effectiveness of the proposed method.

4.2 Digital communication systems

Figure 4.1 illustrates a typical digital communication system. In this figure, the source is assumed to be an analog signal (e.g., audio, speech or image). The analog signal will be discretized with an A/D converter, which contains *sampler*, *quantizer* and *encoder* (or *coder*).

Sampling is a discretization in time, while quantization is that in amplitude.

Encoder converts the sampled and quantized signal to a binary valued signal. It

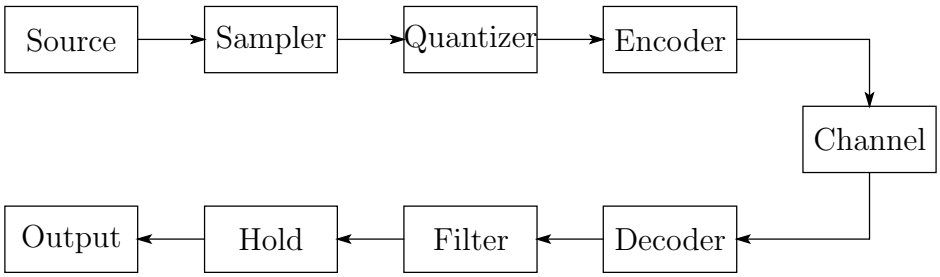

Figure 4.1: Digital communication system

often contains *compressing* and *filtering* for efficiency of signal transmission. For example, *subband coding* is often used for efficient communication. Figure 4.2 illustrates a simple subband coding system. The input signal y is divided into two subband signals[1)]. By

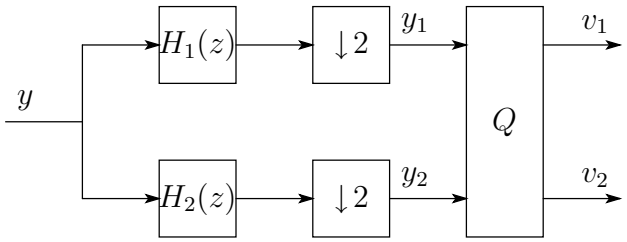

Figure 4.2: Subband coding

taking the filters $H_1(z)$ and $H_2(z)$ appropriately, we can divide the frequency into two subbands, for example, y_1 is a signal with low frequency and y_2 with high frequency. If we need not reconstruct the information in the high frequency range precisely, we can compress the signal by applying fewer bits to the high-frequency signal (i.e. the signal y_2 in Figure 4.2). For simplicity, we apply infinitely many bits to y_1 and no bit to y_2, that is, we assume Q in Figure 4.2 as

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{4.1}$$

This means that we transmit only the signal v_1.

The signal then goes through the communication channel. Since modeling for the chan-

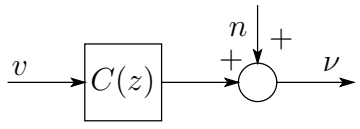

Figure 4.3: Communication channel model

[1)] In real applications, the signal will be subdivided into 16 or 32 signals.

nel is an important and difficult issue, we assume a simple model: a linear time-invariant system and additive noise as shown in Figure 4.3. Note that communication channels are generally time-varying and nonlinear, in particular, in wireless communication, channel modeling is very complicated [32, 33].

Then, the signal distorted by the channel enters the receiver. The receiver decodes the binary signal, and then filters the decoded signal. The filter reduces distortion of the received signal and expands the compressed signal. The process is shown in Figure 4.4. Since we take the encoding (4.1), we similarly assume Q' to be the same as Q, that is,

$$Q' = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}.$$

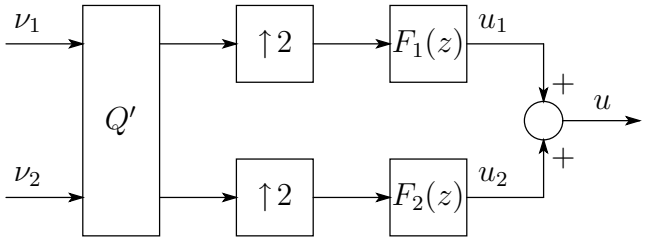

Figure 4.4: Decoding

Then the decoded and filtered signal is converted to an analog signal by a hold device, and finally we obtain a received signal.

The conventional design is executed in the discrete-time domain by assuming that the original analog signal contains no frequency beyond the Nyquist frequency. To satisfy this assumption, we often use a low-pass filter, which deteriorates the quality of communication.

In contrast to this, we introduce the sampled-data H^∞ control theory to take the analog performance into account. We will show that our design is superior to the conventional design in the following sections.

4.3 Design problem formulation

We consider a digital communication system as shown in Figure 4.5. The incoming signal $w_c \in L^2[0, \infty)$ goes through an analog low-pass filter F_c and becomes y_c that is nearly (but not entirely) band limited. The filter F_c governs the frequency-domain characteristic of the analog signal y_c[2)]. The signal y_c is then sampled by the sampler $\mathcal{S}_{h/M}$ to become a discrete-time signal (i.e., PAM signal) y_d with sampling period h/M. Then the signal is compressed by the downsampler with the factor M, and becomes a discrete-time signal with sampling period h. The downsampled signal is then shaped or enhanced by the

[2)] In the conventional design, F_c is considered to be an ideal filter that has the cut-off frequency up to the Nyquist frequency.

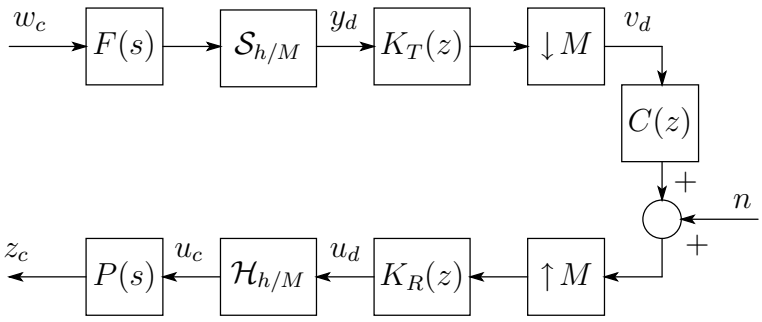

Figure 4.5: Digital communication system

transmitting digital filter K_T to the signal v_d to be transmitted to a communication channel.

The transmitted signal v_d is corrupted by the communication channel C and the additive noise n_d. In PCM communication, n_d is also considered as the noise generated by quantization and coding errors. The received signal goes through the upsampler and receiving digital filter K_R that attempts to attenuate the imaging components caused by the upsampler and the distortion by the channel. Then the signal becomes an analog signal u_c by the hold device $\mathcal{H}_{h/M}$ with sampling period h/M and this analog signal is smoothed by an analog low-pass filter P_c. Finally we have the output signal z_c.

Our objective is to reconstruct the original analog signal y_c by the transmitting filter K_T and the receiving filter K_R against data compression effect caused by the downsampler and the distortion caused by the channel.

We thus consider the block diagram shown in Figure 4.6 that is the signal reconstruction error system for the design. In the diagram the following points are taken into account:

- The time delay e^{-Ls} is introduced because we allow a certain amount of time delay for signal reconstruction.
- The transmitted signal v_d is estimated with a weighting function W_z because the energy or the amplitude of the transmitted signal v_d is usually limited.
- The noise obeys a frequency characteristic W_n.

Our design problem is defined as follows:

Problem 4.1. *Given a stable, strictly proper, continuous-time $F(s)$, stable, proper, continuous-time $P(s)$, stable proper discrete-time weighting functions $W_n(z)$ and $W_z(z)$, stable proper channel model $C(z)$, downsampling factor M, delay time L and a sampling period h, find digital filters $K_T(z)$ and $K_R(z)$ that minimizes*

$$J(K_R, K_T) := \sup_{\substack{w_c \in L^2, n_d \in l^2 \\ \|w_c\|_{L^2} + \|n_d\|_{l^2} \neq 0}} \frac{\|e_c\|_{L^2}^2 + \|z_d\|_{l^2}^2}{\|w_c\|_{L^2}^2 + \|n_d\|_{l^2}^2}. \tag{4.2}$$

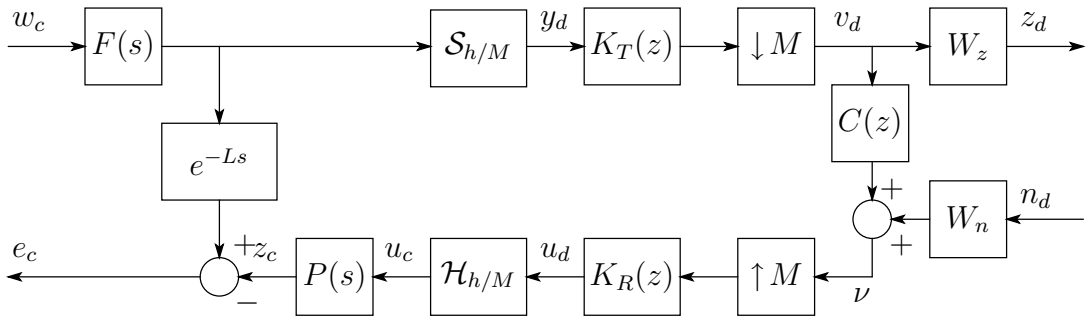

Figure 4.6: Signal reconstruction error system

4.4 Design algorithm

4.4.1 Decomposing design problems

Problem 4.1 is a simultaneous design problem of a transmitting filter and a receiving filter, and it is difficult to solve this problem directly. We thus decompose the design problem into two steps, that is, the design of the receiving filter and that of the transmitting filter.

Obviously the transmitting filter K_T cannot attenuate the additive noise n_d, hence the receiving filter K_R must play this role. Moreover K_R must reconstruct the original signal from the corrupted signal (if K_R did not have to reconstruct, the optimal filter will be clearly $K_R = 0$). Therefore we first design the receiving filter K_R in order to reconstruct the original signal and to attenuate the noise by the block diagram shown in Figure 4.6 with $W_z = 0$ and with $K_T = 1$. We then design the transmitting filter by the block diagram shown in Figure 4.6 with $W_n = 0$ and with K_R that is obtained in the previous design, that is, we consider the channel as $K_R(\uparrow M)C$.

Denote $\mathcal{T}_R$ the system from $[w_c,\ n_d]^T$ to e_c (shown in Figure 4.7), and $\mathcal{T}_T$ the system from w_c to $[e_c,\ z_d]^T$ (shown in Figure 4.8). The design procedure is then as follows:
Step 1 (Design of a receiving filter) Find a receiving filter K_R that minimizes

$$J_1(K_R) := \|\mathcal{T}_R\|^2 := \sup_{\substack{w_c \in L^2, n_d \in l^2 \\ \|w_c\|_{L2} + \|n_d\|_{l^2} \neq 0}} \frac{\|e_c\|_{L^2}^2}{\|w_c\|_{L^2}^2 + \|n_d\|_{l^2}^2}, \tag{4.3}$$

with fixed K_T.
Step 2 (Design of a transmitting filter) Find a transmitting filter K_T that minimizes

$$J_2(K_T) := \|\mathcal{T}_T\|^2 := \sup_{\substack{w_c \in L^2 \\ w_c \neq 0}} \frac{\|e_c\|_{L^2}^2 + \|z_d\|_{l^2}^2}{\|w_c\|_{L^2}^2}, \tag{4.4}$$

with K_R obtained in the previous step. We iterate Step 1 and Step 2 alternately with initial condition $K_T = 1$.

The filter K_T designed in Step 2 cannot have any influence on the performance of the system from n_d to e_c. Therefore, the objective function $J(K_R, K_T)$ in (4.2) monotonically decreases by each step. In fact, we have the following proposition:

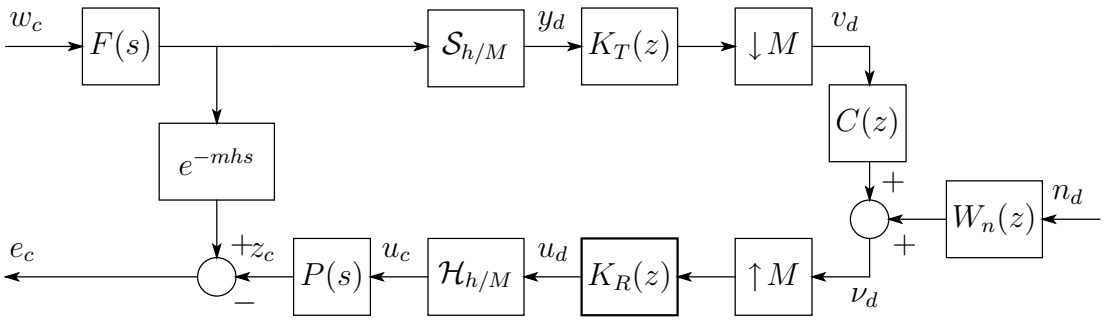

Figure 4.7: Error system $\mathcal{T}_R$ for receiving filter design

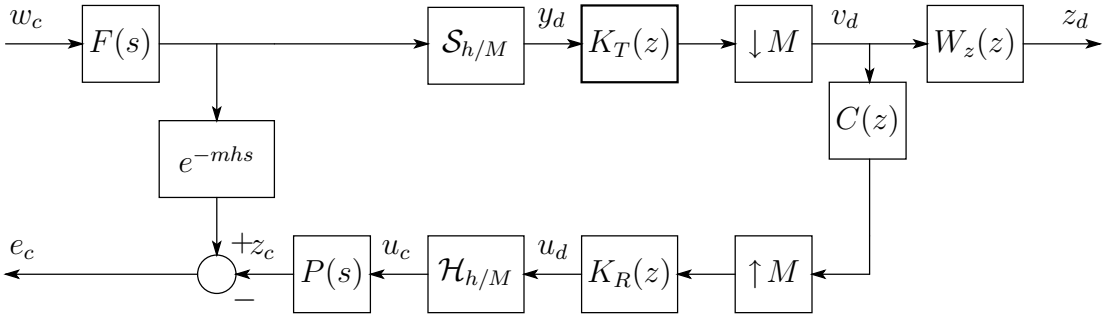

Figure 4.8: Error system $\mathcal{T}_T$ for transmitting filter design

Proposition 4.1. *For any integer $n \geq 1$, the following inequality holds:*

$$J(K_R^{(n-1)}, K_T^{(n-1)}) \geq J(K_R^{(n)}, K_T^{(n)}) \geq J_{\text{opt}},$$

where $K_R^{(n)}$ and $K_T^{(n)}$ are filters obtained at the n-th design step, and define

$$J_{\text{opt}} := \min_{K_R, K_T} J(K_R, K_T). \tag{4.5}$$

Proof. Let $\mathcal{T}_1(K_R, K_T)$ and $\mathcal{T}_2(K_R)$ be the system from w_c to $[e_c, z_d]^T$ and the system from n_d to $[e_c, z_d]^T$, respectively, in Figure 4.6. At the n-th step of K_R design (Step 1), we have

$$K_R^{(n)} = \arg\min_{K_R} J_1(K_R) = \arg\min_{K_R} J(K_R,\ K_T^{(n-1)}),$$

and hence

$$J(K_R^{(n-1)},\ K_T^{(n-1)}) \geq J(K_R^{(n)},\ K_T^{(n-1)}), \tag{4.6}$$

holds. Then at the n-th step of K_T design (Step 2), we have

$$K_T^{(n)} = \arg\min_{K_T} J_2(K_T) = \arg\min_{K_T} \|\mathcal{T}_1(K_R^{(n)},\ K_T)\|.$$

By using this equation, we have

$$\|\mathcal{T}_1(K_R^{(n)},\ K_T^{(n-1)})\| \geq \|\mathcal{T}_1(K_R^{(n)},\ K_T^{(n)})\|.$$

It follows that

$$\left\|\left[\ \mathcal{T}_1(K_R^{(n)},\ K_T^{(n-1)}),\ \ \mathcal{T}_2(K_R^{(n)})\ \right]\right\| \geq \left\|\left[\ \mathcal{T}_1(K_R^{(n)},\ K_T^{(n)}),\ \ \mathcal{T}_2(K_R^{(n)})\ \right]\right\|,$$

holds. This implies

$$J(K_R^{(n)},\ K_T^{(n-1)}) \geq J(K_R^{(n)},\ K_T^{(n)}). \tag{4.7}$$

By (4.6) and (4.7), we have

$$J(K_R^{(n-1)},\ K_T^{(n-1)}) \geq J(K_R^{(n)},\ K_T^{(n)}).$$

Then, by the definition (4.5),

$$J(K_R^{(n)},\ K_T^{(n)}) \geq J_{\rm opt},$$

is obvious. □

4.4.2 Fast-sampling/fast-hold approximation

The design problems (4.3) and (4.4) involve a continuous-time delay component e^{-Ls}, and hence they are infinite-dimensional sampled-data problems. To avoid this difficulty, we employ the fast-sampling/fast-hold approximation method [19, 43]. By the method, our design problems (4.3) and (4.4) are approximated by finite-dimensional discrete-time problems assuming that the delay time L is mh ($m \in \mathbb{N}$).

Theorem 4.1. *Assume that $L = mh$, $m \in \mathbb{N}$. Then,*

1. *for the error system $\mathcal{T}_R$ in Step 1, there exist finite-dimensional discrete-time systems $\{T_{R,N} : N = M, 2M, \dots\}$ such that*

$$\lim_{N\to\infty} \|T_{R,N}\| = \|\mathcal{T}_R\|,$$

2. *for the error system $\mathcal{T}_T$ in Step 2, there exist finite-dimensional discrete-time systems $\{T_{T,N} : N = M, 2M, \dots\}$ such that*

$$\lim_{N\to\infty} \|T_{T,N}\| = \|\mathcal{T}_T\|.$$

Proof. By the fast-sampling/fast-hold method, we approximate continuous-time inputs and outputs to discrete-time ones via the ideal sampler and the zero-order hold that operate in the period h/N (Figure 4.9). Assume $N = Ml, l \in \mathbb{N}$. Then apply the discrete-time lifting $\mathbb{L}_N$ (see Section 2.4) to the discretized input/output signal e_{dN} and w_{dN}, we can obtain the lifted signals

$$\widetilde{e}_{dN} := \mathbb{L}_N(e_{dN}) = \mathbb{L}_N\mathcal{S}_{h/N}e_c, \quad \widetilde{w}_{dN} := \mathbb{L}_N(w_{dN}) = \mathbb{L}_N\mathcal{S}_{h/N}w_c.$$

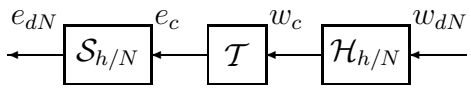

Figure 4.9: Fast-sampling/fast-hold discretization

We next denote $T_{R,N}$ the system from $[\widetilde{w}_{dN}, n_d]^T$ to $\widetilde{e}_{dN}$ and the system $T_{T,N}$ from $\widetilde{w}_{dN}$ to $[\widetilde{e}_{dN}, z_d]^T$, and define their norm

$$\|T_{R,N}\|^2 := \sup_{\substack{w_{dN}, n_d \in l^2 \\ \|w_{dN}\|_{l^2} + \|n_d\|_{l^2} \neq 0}} \frac{\|\widetilde{e}_{dN}\|_{l^2}^2}{\|\widetilde{w}_{dN}\|_{l^2}^2 + \frac{N}{h}\|n_d\|_{l^2}^2},$$

$$\|T_{T,N}\|^2 := \sup_{\substack{w_{dN} \in l^2 \\ w_{dN} \neq 0}} \frac{\|\widetilde{e}_{dN}\|_{l^2}^2 + \frac{N}{h}\|z_d\|_{l^2}^2}{\|\widetilde{w}_{dN}\|_{l^2}^2}.$$

Then we can show $\|T_{R,N}\| \to \|\mathcal{T}_R\|$, $\|T_{T,N}\| \to \|\mathcal{T}_T\|$, as $N \to \infty$ by using the method as shown in [43] under the assumption $L = mh$. □

Once the problems have been reduced to discrete-time problems, they can be solved by standard softwares such as MATLAB. The resulting discrete-time approximant is given by the following:

Theorem 4.2. *The approximated discrete-time systems $T_{R,N}$ and $T_{T,N}$ are given as follows:*

$$T_{R,N} := \mathcal{F}_l\left(G_{R,N},\ \widetilde{K}_R\right), \quad T_{T,N} := \mathcal{F}_l\left(G_{T,N},\ \widetilde{K}_T\right),$$

$$G_{R,N} := \begin{bmatrix} \begin{bmatrix} z^{-m}F_{dN}, & 0 \end{bmatrix}, & -P_{dN}H \\ \begin{bmatrix} C\widetilde{K}_T S F_{dN}, & W_n \end{bmatrix}, & 0 \end{bmatrix},$$

$$G_{T,N} := \begin{bmatrix} \begin{bmatrix} z^{-m}F_{dN} \\ 0 \end{bmatrix}, & \begin{bmatrix} -P_{dN}H\widetilde{K}_R C \\ W_z \end{bmatrix} \\ SF_{dN}, & 0 \end{bmatrix},$$

$$S := \left.\begin{bmatrix} p & & \\ & \ddots & \\ & & p \end{bmatrix}\right\} M, \quad p := [1, \underbrace{0, \ldots, 0}_{k-1}],$$

$$H := \underbrace{\begin{bmatrix} q & & \\ & \ddots & \\ & & q \end{bmatrix}}_{M}, \quad q := [\underbrace{1, \ldots, 1}_{k}]^T,$$

$$F_{dN} := \mathcal{L}_N(F), \quad P_{dN}(z) := \mathcal{L}_N(P).$$

Proof. First, consider the receiving filter design in Figure 4.7. The relation between the

input and the output is as follows:

$$e_c = e^{-mhs}Fw_c - P\mathcal{H}_{h/M}u_d, \tag{4.8}$$
$$\nu_d = C(\downarrow M)K_T\mathcal{S}_{h/M}Fw_c + W_n n_d, \tag{4.9}$$
$$u_d = K_R(\uparrow M)\nu_d. \tag{4.10}$$

Then applying the fast-sampling and the discrete-time lifting $\mathbb{L}_N$ to the input w_c and the output w_c, we have from (4.8)

$$\begin{aligned}\mathbb{L}_N\mathcal{S}_{h/N}e_c &= \mathbb{L}_N\mathcal{S}_{h/N}e^{-mhs}F\mathcal{H}_{h/N}\mathbb{L}_N^{-1}\widetilde{w}_{Nd} - \mathbb{L}_N\mathcal{S}_{h/N}P\mathcal{H}_{h/M}u_d \\ &= z^{-m}\mathcal{L}_N(F)\widetilde{w}_{Nd} - \mathcal{L}_N(P)H\mathbb{L}_M u_d,\end{aligned}$$

and from (4.9)

$$\begin{aligned}\nu_d &= C(\downarrow M)K_T\mathcal{S}_{h/M}F\mathcal{H}_{h/N}\mathbb{L}_N^{-1}\widetilde{w}_{Nd} + W_n n_d \\ &= C(\downarrow M)K_T\mathbb{L}_M^{-1}S\mathcal{L}_N(F)\widetilde{w} + W_n n_d.\end{aligned}$$

In these equations we have used Proposition 2.1. Then by using the relation (3.1), we convert the multirate systems to single rate ones:

$$\mathbb{L}_M u_d = \mathbb{L}_M K_R(\uparrow M)\nu_d = \mathbb{L}_M K_R\mathbb{L}_M^{-1}[1,0,\ldots,0]^T\nu_d = \widetilde{K}_R\nu_d,$$
$$C(\downarrow M)K_T\mathbb{L}_M^{-1} = C[1,0,\ldots,0]\mathbb{L}_M K_T\mathbb{L}_M^{-1} = C\widetilde{K}_T.$$

Consequently, we have

$$\begin{aligned}\widetilde{e}_{dN} &= \begin{bmatrix} z^{-m}F_{dN} & 0 \end{bmatrix}\begin{bmatrix} \widetilde{w}_{Nd} \\ n_d \end{bmatrix} - P_{dN}H\widetilde{u}_d, \\ \nu_d &= \begin{bmatrix} C\widetilde{K}_T S F_{dN} & Wn \end{bmatrix}\begin{bmatrix} \widetilde{w}_{Nd} \\ n_d \end{bmatrix}, \\ \widetilde{u}_d &= \widetilde{K}_R\nu_d,\end{aligned}$$

where $\widetilde{u}_d := \mathbb{L}_M u_d$. It implies that $\widetilde{e}_{dN} = \mathcal{F}_l(G_{R,N}, \widetilde{K}_R)\widetilde{w}_{dN}$.

The proof for $G_{T,N}$ is almost similar to that for $G_{R,N}$ and hence we omit the proof. □

Then our design problems (4.3) and (4.4) are reduced to finite-dimensional discrete-time H^∞ design problems, which are shown in Figure 4.10 and Figure 4.11.

4.5 Design examples

4.5.1 The case of no compression ($M = 1$)

Design for $W_z = 0$

In this section, we present a design example for

$$\begin{aligned}F(s) &:= \frac{1}{10s+1}, \quad P(s) := 1, \quad W_n(z) := 1, \\ C(z) &:= 1 + 0.65z^{-1} - 0.52z^{-2} - 0.2975z^{-3},\end{aligned}$$

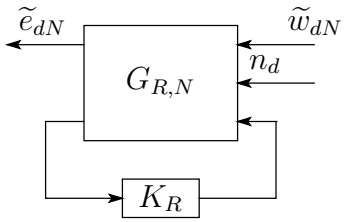

Figure 4.10: Discrete-time system for designing receiving filter K_R

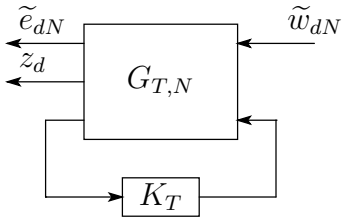

Figure 4.11: Discrete-time system for designing transmitting filter K_T

with sampling period $h = 1$ and time delay $L = mh = 2$. We here consider the case of no compression (i.e., $M = 1$) and no limitation on transmission (i.e., $W_z = 0$). An approximate design is executed here for $N = 8$. For comparison, the discrete-time H^∞ design [8] is also executed.

Figure 4.12 shows the gain responses of the filters, and Figure 4.13 shows the frequency responses of $\mathcal{T}_{ew}$ (the system from the input w_c to the error e_c), and Figure 4.14 shows those of $\mathcal{T}_{zn}$ (the system from the additive noise n_d to the output z_c). Compared with the discrete-time design, the sampled-data one shows better frequency response both in $\mathcal{T}_{ew}$ and in $\mathcal{T}_{zn}$. Moreover, we can say that an equalizer alone cannot attenuate the corruption caused by the channel and the additive noise, that is, we need an appropriate transmitter for transmission.

To explain this fact, we show a simulation of these communication systems. The input signal y_c is a rectangular wave with amplitude 1, and the noise disturbance n_d is a discrete-time sinusoid: $n_d[k] = \sin(2k)$. Figure 4.15 shows the output z_c with the receiving filter and the transmitting filter designed via sampled-data method, and Figure 4.16 shows that with the receiving filter designed in discrete-time (and without any transmitting filter). We see that the former shows much better reconstruction against the noise than the latter.

Design for $W_z(z) \neq 0$

We now consider the design with the estimation of the transmitting signal v_d, that is $W_z(z) \neq 0$.

We observe from Figure 4.12 that the transmitting filter shows high gains around the

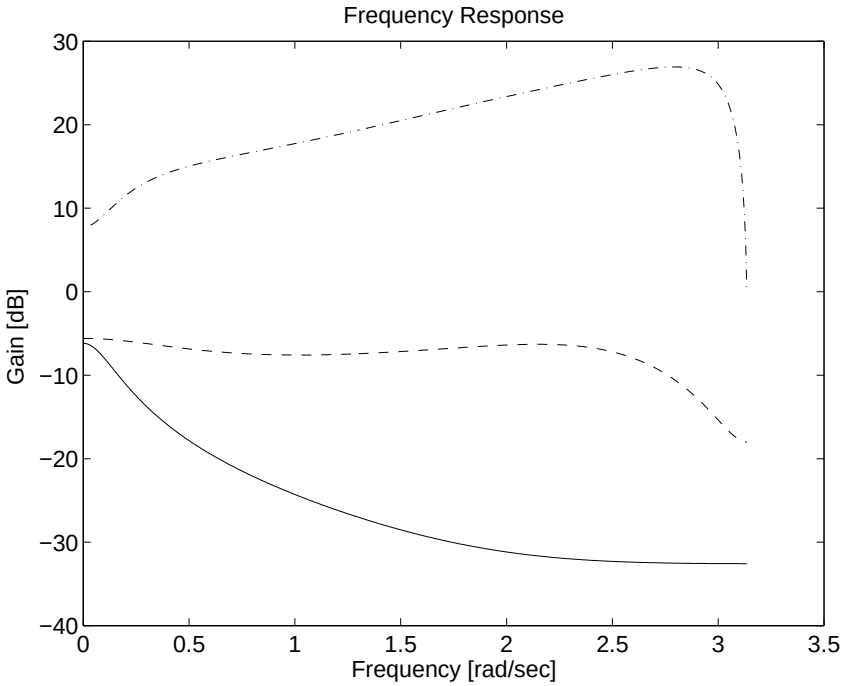

Figure 4.12: Gain responses of filters: sampled-data H^∞ design (transmitting filter: solid, receiving filter: dots) and discrete-time H^∞ design (dash)

Nyquist frequency (i.e. $\omega \approx \pi$), and hence we take

$$W_z(z) = r \cdot \frac{z-1}{z+0.5},$$

as the weighting function of the transmitting signal, where the parameter $r = 0.21$. The other design parameters are the same as the example above.

Figure 4.17 shows the H^∞-norm of $\mathcal{T}_{ew}$ and $\mathcal{T}_{vw}$ (the system from w_c to v_d in Figure 4.6) that varies with $r \in [0, 5]$. We can take account of a trade-off between the error attenuation level and the amount of the transmitting signal with Figure 4.17. For example, we choose $r = 0.21$ in order to attenuate the error less than -26dB.

Figure 4.18 shows the gain responses of transmitting filters designed for $r = 0$ and $r = 0.21$. We can see that the new filter shows better attenuation than the filter designed for $r = 0$ at high frequency.

Figure 4.19 shows the frequency responses of the error system $\mathcal{T}_{ew}$. We see that the attenuation level of $\mathcal{T}_{ew}$ designed for $r = 0.21$ is less than -26dB. Figure 4.20 shows the frequency responses of $\mathcal{T}_{vw}$. We can see that the amount of the transmitting signal is attenuated at high frequency.

4.5.2 Compression effects

In this section, we consider compression effects. First, we take compression with the down-sampling factor $M = 1, 2, 4, 8$. It means that the size of data transmitted is compressed

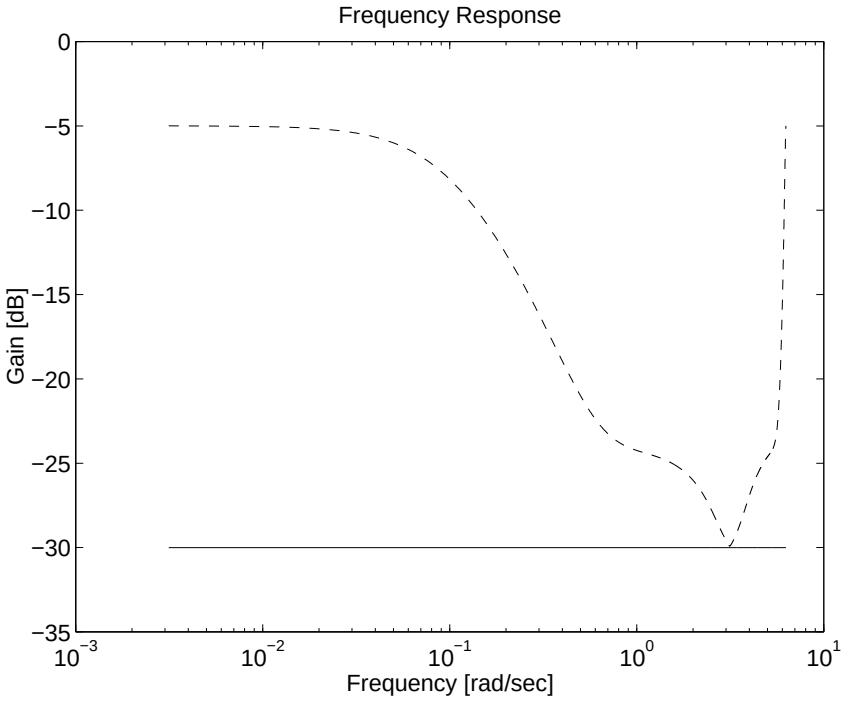

Figure 4.13: Frequency responses of $\mathcal{T}_{ew}$: sampled-data H^∞ design (solid) and discrete-time H^∞ design (dash)

to $1, 1/2, 1/4, 1/8$, respectively. For simplicity, we put $W_z = 0$. The other parameters are the same as the previous example. Denote $\mathcal{T}_{ew}$ the system from w_c to e_c and $\mathcal{T}_{zn}$ the system from n_d to e_c.

We show the frequency responses of $\mathcal{T}_{ew}$ in Figure 4.21 and those of $\mathcal{T}_{zn}$ in Figure 4.22 for $M = 1, 2, 4, 8$. From Figure 4.21, we can see that when we compress the signal to $1/M$, the gain of $\|\mathcal{T}_{ew}\|$ (i.e., the maximum gain) increases about $6 \times M$ dB. In particular, the compression of $M = 2$ is almost comparative with the case of no compression in the low frequency range; the difference is about 0.5dB. Figure 4.22 indicates that the larger the compression ratio M, the smaller $\|T_{zn}\|$.

Next we simulate a transmission with compression ratio $M = 4$. For simplicity, we put $C_d = 1$. The source is a speech signal with the sampling frequency 22.05kHz. The frequency of the downsampled signal transmitted is 5.5125kHz.

Figure 4.23 shows the FFT of the source speech signal. Then we show the FFT of the reconstructed signal u_d (see Figure 4.5) that is designed by the sampled-data H^∞ optimization. For comparison, we take the equiripple filter [9, 35, 46] with the cut-off frequency $\omega_c = 22.05/8 \approx 2.76$kHz. This filter is often used in multirate signal processing. The FFT of the reconstructed signal is shown in Figure 4.25. We can see from Figure 4.25 that the FFT of the signal processed by the equiripple filter is sharply cut at the frequency about 2.76kHz, while that of sampled-data design shows slow decay. The FFT plot of the source signal shown in Figure 4.23 indicates that the frequency of the original signal does not very decay up to 8kHz, and we can say that the response by the sampled-

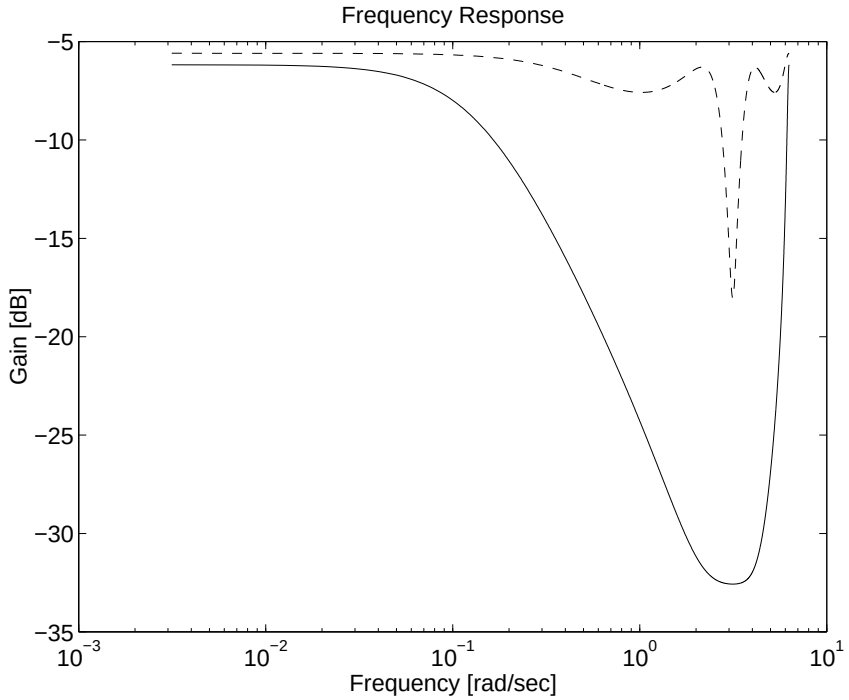

Figure 4.14: Frequency responses of $\mathcal{T}_{zn}$: sampled-data H^∞ design (solid) and discrete-time H^∞ design (dash)

data design is better than that by the conventional design. In fact, the reconstructed voice by equiripple filter sounds blur because of lack of high frequency, while that by the sampled-data design sounds clearer. However, the sound by the sampled-data design has high frequency noise. To reduce the noise, we have to introduce a low-pass filter, whose design needs further discussion.

4.6 Conclusion

In this chapter, we have treated communication systems which contains signal compression. Under distortion by a channel, we have presented a design method of transmitting/receiving filters by using sampled-data H^∞ optimization. By iterating a transmitting filter design and a receiving filter design, we can obtain sub optimal filters. We have shown that the objective function monotonically decreases by the iteration.

In this design, the channel is assumed to be time-invariant. However, the real channel often contains time-varying systems, in particular, in the case of wireless communication. Moreover, the real channel is very complicated and we should notice that the model of the channel always contains a modeling error. To overcome this, we have to choose an adaptive filter. Design for adaptive filters by using sampled-data theory is an important subject for the future.

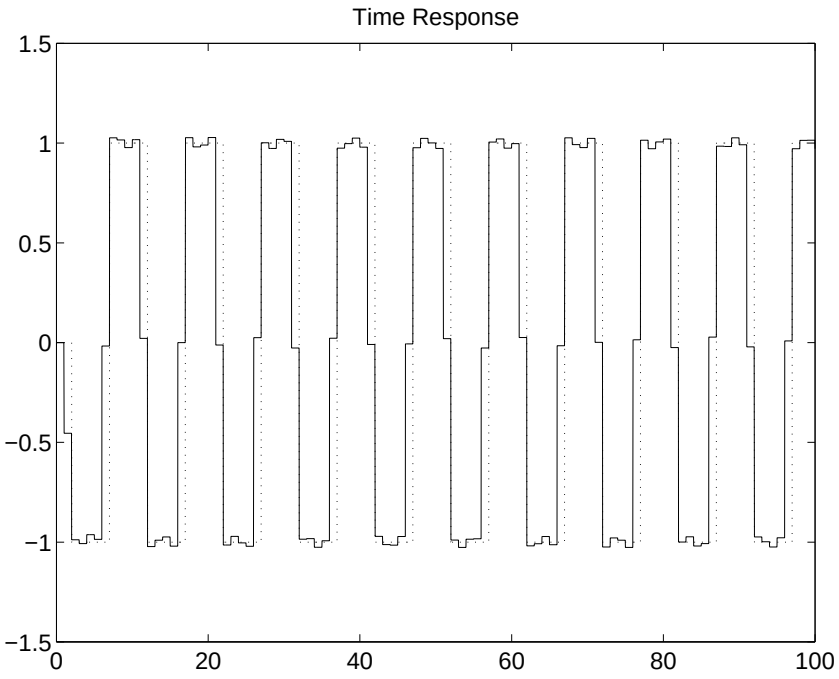

Figure 4.15: Time response with sampled-data design

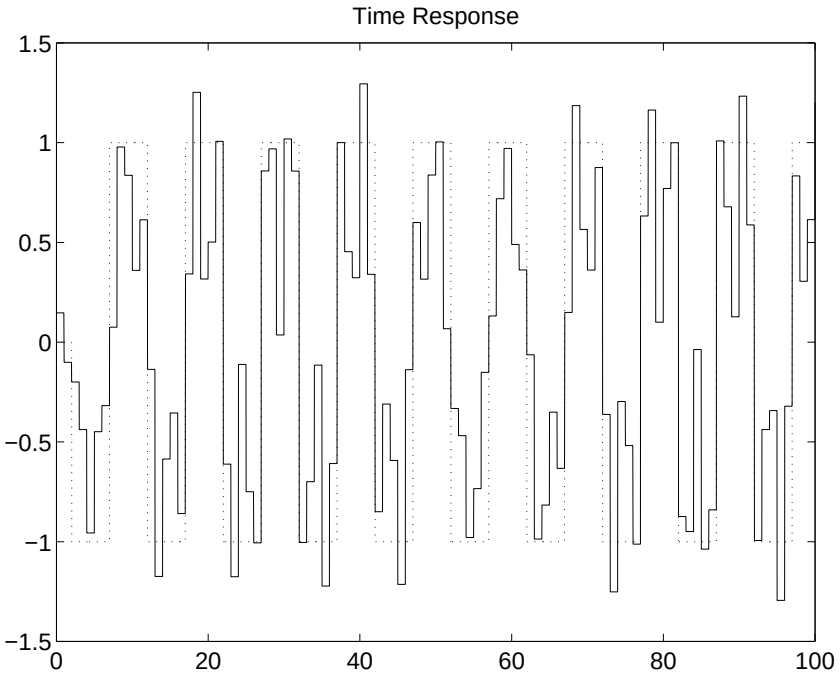

Figure 4.16: Time response with discrete-time design

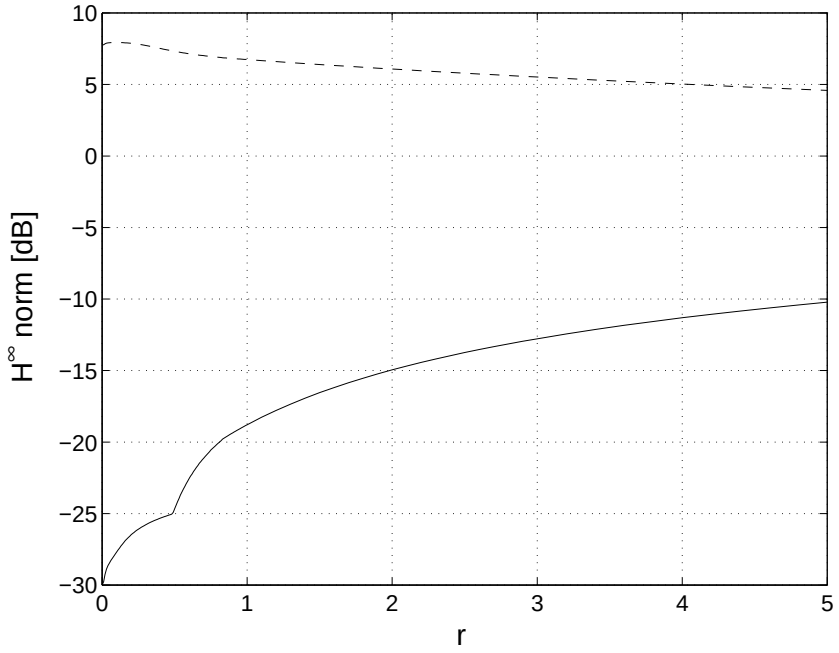

Figure 4.17: Relation between r and $\|T_{ew}\|$ (solid), $\|T_{vw}\|$ (dash)

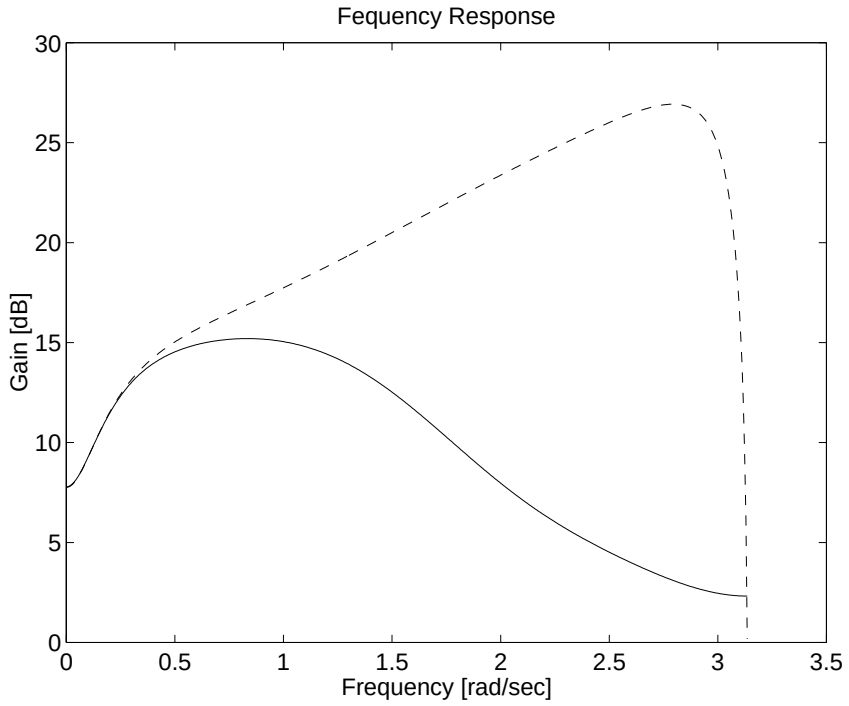

Figure 4.18: Gain responses of transmitting filters designed for $r = 0.21$ (solid) and $r = 0$ (dash)

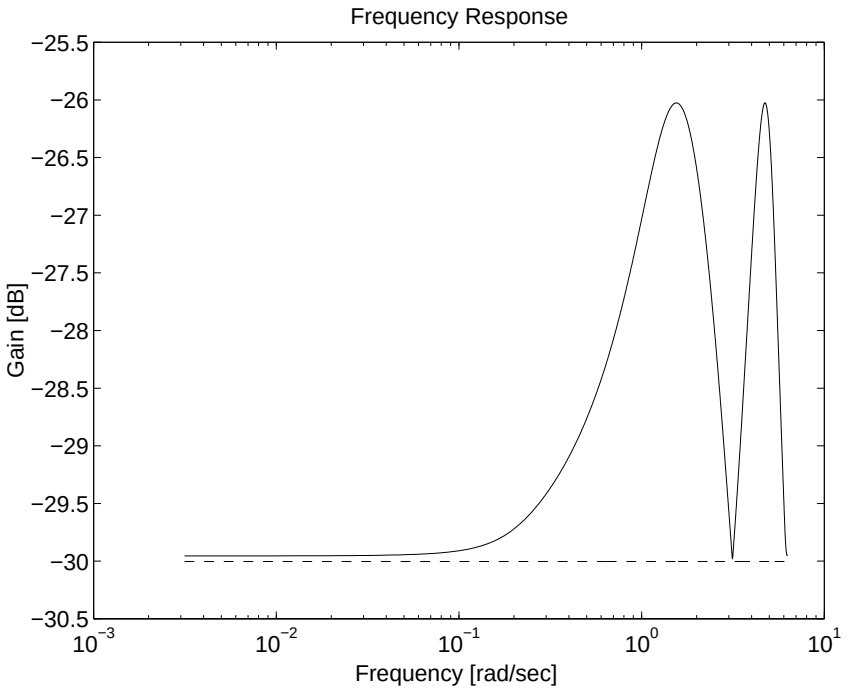

Figure 4.19: Frequency responses of $\mathcal{T}_{ew}$ designed for $r = 0.21$ (solid) and $r = 0$ (dash)

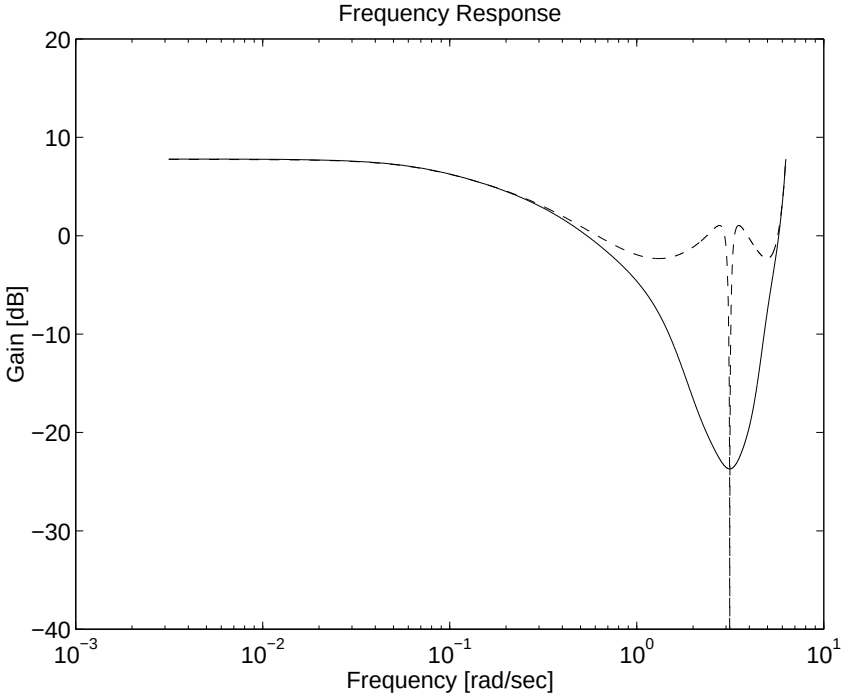

Figure 4.20: Frequency responses of $\mathcal{T}_{vw}$ designed for $r = 0.21$ (solid) and $r = 0$ (dash)

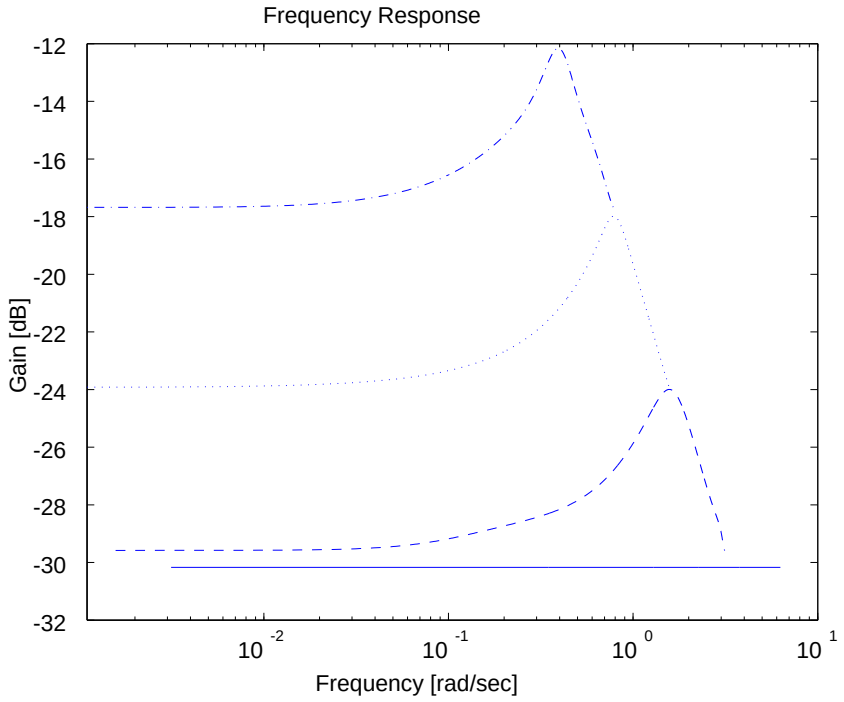

Figure 4.21: Frequency responses of T_{ew}: compression ratio $M = 1$ (solid), $M = 2$ (dash), $M = 4$ (dot) and $M = 8$ (dash-dot)

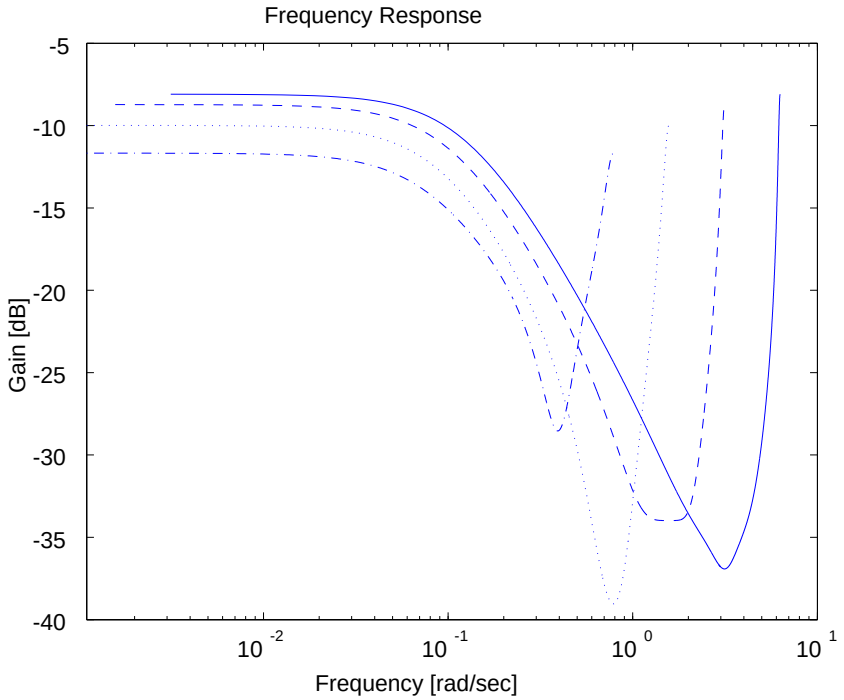

Figure 4.22: Frequency responses of T_{zn}: compression ratio $M = 1$ (solid), $M = 2$ (dash), $M = 4$ (dot) and $M = 8$ (dash-dot)

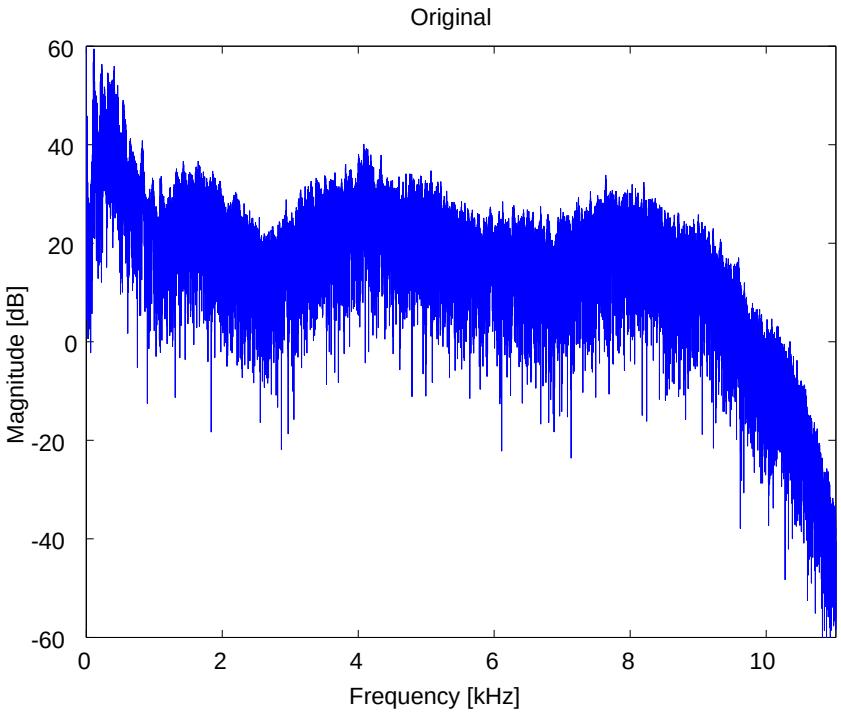

Figure 4.23: FFT of the source

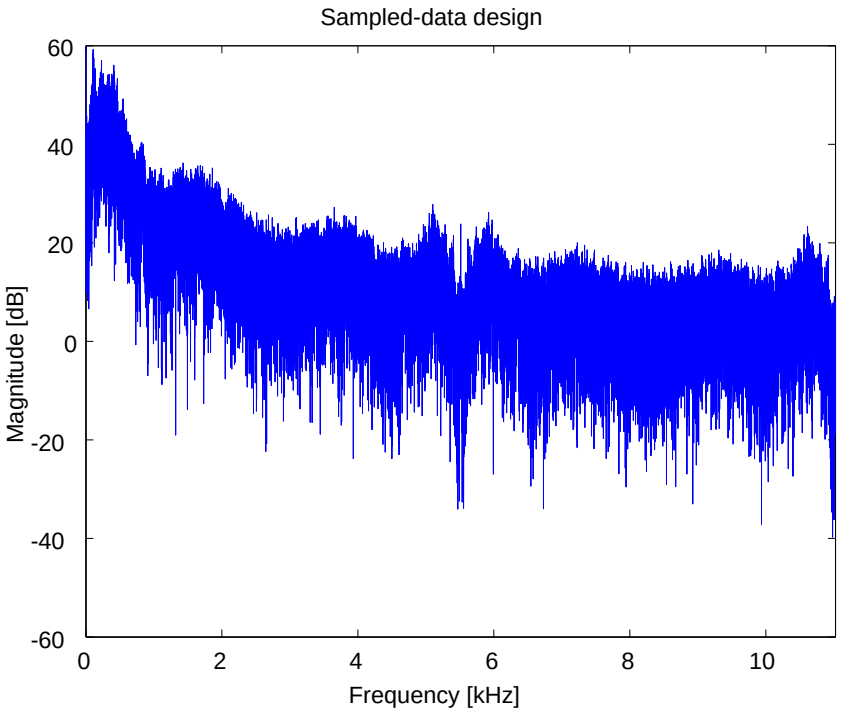

Figure 4.24: FFT of the reconstructed signal (sampled-data design)

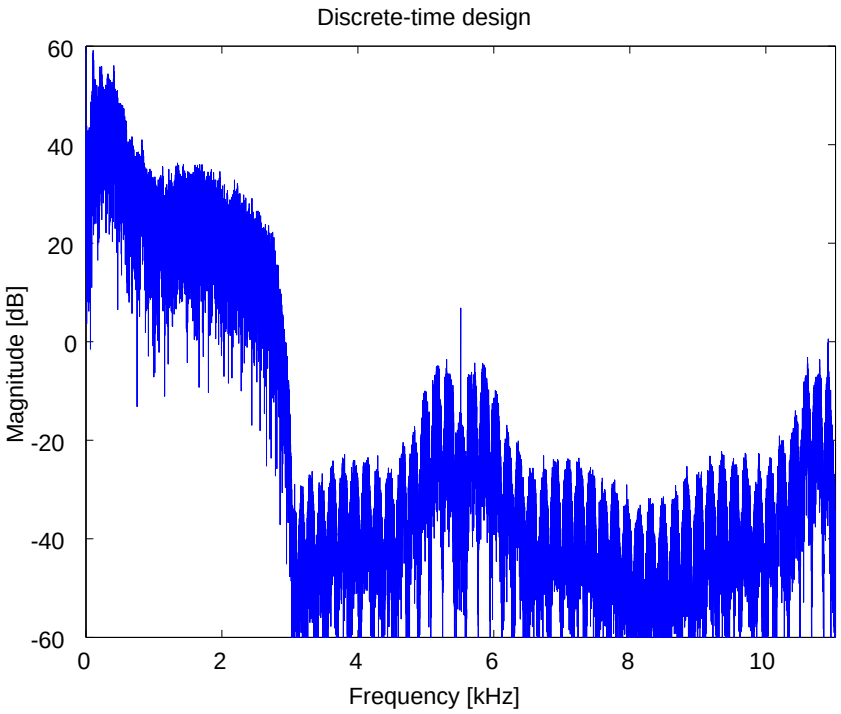

Figure 4.25: FFT of the reconstructed signal (equiripple design)

Chapter 5

Minimization of Quantization Errors

5.1 Introduction

In digital signal processing, digital communications and digital control, analog signals have to be discretized by an A/D converters to become digital signals. In discretization, we have two operations; sampling and quantization. Sampling is discretization in time, and the model is described as a linear one, which is relatively easy to analyze mathematically (e.g., via lifting discussed in Chapter 2). On the other hand, quantization, discretization in amplitude, is a nonlinear operation, and its analysis is much more difficult.

For analyzing such a nonlinearity, an additive noise model has been widely used [10, 6]. The model is then linear and hence we can easily analyze or design a system with quantization, but there has not been much established knowledge on the characteristic of the nonlinear system designed by the linearization method.

In this chapter, we first discuss the stability of quantized sampled-data control systems. Then we investigate how much quantization influences the performance of a sampled-data system. We will show that

- if the linear model is stable, the states of the quantized system are bounded,
- if the linear model has a small L^2 gain (i.e., the H^∞-norm of the system), the quantized system has a small power gain.

It follows that the linearization method is valid for analyzing and designing systems with quantizations.

Next we apply the above results to the design of a quantizer that is called differential pulse code modulation (DPCM) [32]. Although a large number of studies have been made on DPCM systems, little is known on the stability and the performance, which we will discuss by means of linearization.

In this chapter, we measure the performance by the power norm defined as follows:

$$\mathrm{pow}(z)^2 := \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} |z(t)|^2 dt \quad \text{(continuous-time)},$$

$$\mathrm{pow}[z_d]^2 := \lim_{N\to\infty} \frac{1}{2N} \sum_{k=-N}^{N} |z_d[k]|^2 \quad \text{(discrete-time)},$$

and the supremum norm

$$\|z_d\|_\infty := \sup_{k\in\mathbb{Z}_+} |z_d[k]|.$$

5.2 Sampled-data control systems with quantization

5.2.1 Additive noise model for quantizer

We will begin with a sampled-data system with quantization shown in Figure 5.1. In the

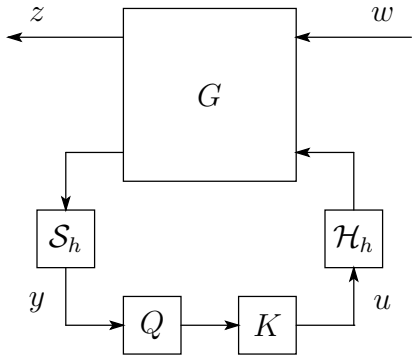

Figure 5.1: Sampled-data control system with quantization

figure, Q is a uniform quantizer with a quantization level Δ, and $K(z)$ is a discrete-time controller. Let $\{A_K, B_K, C_K, D_K\}$ be a realization of the controller $K(z)$, and set

$$G(s) := \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} =: \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & 0 & D_{12} \\ C_2 & 0 & 0 \end{array}\right](s).$$

We use the additive noise model for the uniform quantizer; the quantizer Q is modeled by

$$Qy = y + d, \quad \|d\|_\infty \le \Delta/2.$$

Since we have $\|d\|_\infty = \|y - Qy\|_\infty \le \Delta/2$, the additive noise model covers the input/output relation of the uniform quantization with a belt-like region as shown in Figure 5.2.

5.2.2 Stability of sampled-data systems with quantization

Let us consider the block diagram shown in Figure 5.3 to check the stability of the closed loop system. In the figure, $\mathcal{S}_h G_{22} \mathcal{H}_h K$ is a discrete-time system, of which let $\{\widetilde{A}, \widetilde{B}, \widetilde{C}, \widetilde{D}\}$ be a realization. Note that $\widetilde{A}, \widetilde{B}, \widetilde{C}, \widetilde{D}$ are given by

$$\widetilde{A} := \begin{bmatrix} A_K & 0 \\ B_{2d}C_K & A_d \end{bmatrix}, \quad \widetilde{B} := \begin{bmatrix} B_K \\ B_{2d}D_K \end{bmatrix}, \quad \widetilde{C} := \begin{bmatrix} 0 & C_2 \end{bmatrix}, \quad \widetilde{D} := 0,$$

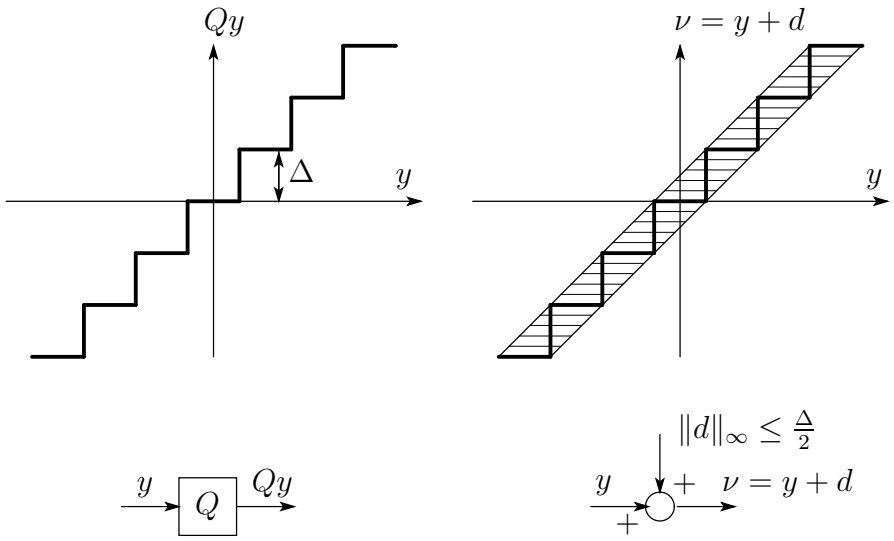

Figure 5.2: Uniform quantization (left) and its additive noise model (right)

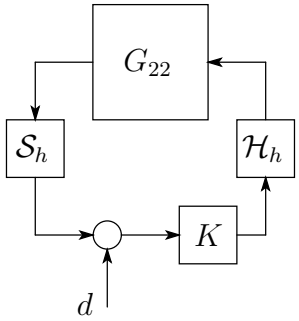

Figure 5.3: Closed loop system

where

$$A_d := e^{Ah}, \quad B_{2d} := \int_0^h e^{At} B_2 dt.$$

Then, we have the state-space representation of the system with the additive noise model shown in Figure 5.3 as follows:

$$x[n+1] = (\widetilde{A} + \widetilde{B}\widetilde{C})x[n] + \widetilde{B}d[n], \quad \|d\|_\infty \le \Delta/2. \tag{5.1}$$

We should notice that without quantization (i.e., $d = 0$) the stability of the feedback system is equal to the stability of the matrix $\widetilde{A} + \widetilde{B}\widetilde{C}$. We have the following theorem of the stability of the system in Figure 5.1.

Theorem 5.1. *Assume that $\widetilde{A} + \widetilde{B}\widetilde{C} =: F \in \mathbb{R}^{n\times n}$ is stable, and let $\gamma > 0$ be a real number such that $r(F) < \gamma < 1$, where $r(F)$ denotes the maximum absolute value of the eigenvalues of F. Then there exists $c > 0$ such that*

$$|x[k] - F^k x_0| \le c\frac{1-\gamma^k}{1-\gamma}\|\widetilde{B}\|\frac{\Delta}{2} =: r_k, \tag{5.2}$$

for all $x_0 := x[0] \in \mathbb{R}^n$ and $k \ge 0$, and

$$\lim_{k\to\infty} |x[k]| \le c\frac{1}{1-\gamma}\|\widetilde{B}\|\frac{\Delta}{2} =: r_\infty, \tag{5.3}$$

for all $x_0 \in \mathbb{R}^n$.

Proof. First of all, let J be the Jordan canonical form of F. That is, for a nonsingular matrix P, we have

$$J = P^{-1}FP = \Lambda + U,$$

where

$$\Lambda := \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_n \end{bmatrix}, \quad U := \begin{bmatrix} 0 & * & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & * \\ 0 & \dots & \dots & \dots & 0 \end{bmatrix}, \quad * = 0 \text{ or } 1,$$

and λ_i $(i = 1, \dots, n)$ denotes the i-th eigenvalue of F. Take $\delta > 0$ such that $r(F) + \delta = \gamma < 1$, and define

$$D := \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & \delta^{-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \delta^{-1+n} \end{bmatrix}.$$

Then we have

$$D^{-1}JD = \Lambda + \delta U.$$

Put $T := DP$. Then for any $x \in \mathbb{R}^n$,

$$|T^{-1}FTx| = |(\Lambda + \delta U)x| \le (r(F) + \delta)|x| = \gamma|x|.$$

It follows that

$$\begin{aligned}
|x[k] - F^k x_0| &\le \sum_{l=1}^{k} |F^{k-l}\widetilde{B}d[l]| \\
&= \sum_{l=1}^{k} |T(T^{-1}FT)^{k-l}T^{-1}\widetilde{B}d[l]| \\
&\le \sum_{l=1}^{k} \|T\| \cdot \gamma^{k-l} \cdot \|T^{-1}\|\|\widetilde{B}\|\frac{\Delta}{2} \\
&= \|T\|\|T^{-1}\|\frac{1-\gamma^k}{1-\gamma}\|\widetilde{B}\|\frac{\Delta}{2}.
\end{aligned}$$

Then put $c := \|T\|\|T^{-1}\|$, we have (5.2). Since F is stable, for any $x_0 \in \mathbb{R}^n$, we have

$$\lim_{k\to\infty} F^k x_0 = 0.$$

Hence by taking $n \to \infty$ for the inequality (5.2), we obtain the second inequality (5.3). □

From this theorem, we conclude that if we stabilize the linearized model, the state of the sampled-data system with quantization shown in Figure 5.1 are bounded, that is, the system is bounded-input bounded-output (BIBO) stable.

Moreover, the set

$$\mathcal{D} := \{x \in \mathbb{R}^n : |x| \le r_\infty\}$$

is a positive invariant set of the system (5.1). In fact, if $x[0] \in \mathcal{D} \in \mathbb{R}^n$, that is, $|x[0]| \le r_\infty$, we have for any $k \ge 1$

$$\begin{aligned}
|x[k]| &\le c\gamma^k|x[0]| + r_k \\
&\le c\gamma^k r_\infty + r_k \\
&= c\left\{\gamma^k\frac{1}{1-\gamma} + \frac{1-\gamma^k}{1-\gamma}\right\}\|\widetilde{B}\|\frac{\Delta}{2} \\
&= c\frac{1}{1-\gamma}\|\widetilde{B}\|\frac{\Delta}{2} \\
&= r_\infty,
\end{aligned}$$

that is, $x[k] \in \mathcal{D}$.

5.2.3 Performance analysis of sampled-data systems with quantization

Generally, quantization deteriorates the performance of control systems, and it is important to know how much quantization affects on the systems. The aim of this section is to analyze the quantization effect by using the additive noise model discussed above. Consider the block diagram shown in Figure 5.4. Although the quantization noise d is

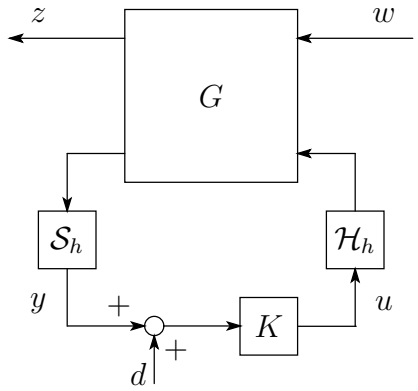

Figure 5.4: Sampled-data control system with additive noise

usually taken as the white noise, we assume the noise d in l^2. The reason is as follows:

- generally the quantization noise is not white; the noise will depend on the input signal of the quantizer,
- we can use the H^∞-norm that has a connection with the worst case analysis.

We denote by $\mathcal{T}_{zw}$ and $\mathcal{T}_{zd}$ the system shown in Figure 5.4 from w to z and from d to z respectively. Assume $\mathcal{T}_{zw}$ and $\mathcal{T}_{zd}$ are stable and

$$\|\mathcal{T}_{zw}\| := \sup_{\substack{w\in L^2\\ w\neq 0}} \frac{\|\mathcal{T}_{zw}w\|_{L^2}}{\|w\|_{L^2}} = \gamma_1, \quad \|\mathcal{T}_{zd}\| = \sup_{\substack{d\in l^2\\ d\neq 0}} \frac{\|\mathcal{T}_{zd}d\|_{L^2}}{\|d\|_{l^2}} = \gamma_2.$$

We will now discuss the performance of the system in Figure 5.1.

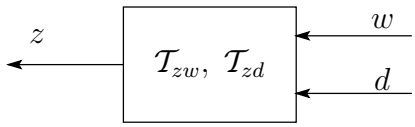

Figure 5.5: Additive noise model for sampled-data system with quantization

Lemma 5.1. *Let $\mathcal{T}$ be a stable sampled-data system with discrete-time inputs and continuous-time outputs. We have the following inequality for any discrete-time, power signal u:*

$$\mathrm{pow}(\mathcal{T}u) \le \|\mathcal{T}\|\,\mathrm{pow}[u].$$

Proof. First of all, define the power norm of lifted signal $\widetilde{y} = \mathbf{L}_h y$ as follows:

$$\mathrm{pow}\{\widetilde{y}\}^2 := \lim_{N\to\infty} \frac{1}{2N} \sum_{k=-N}^{N} \|\widetilde{y}[k]\|^2_{L^2[0,h)}, \quad \widetilde{y}[k] \in L^2[0,h).$$

Note that for y and $\widetilde{y} = \mathbf{L}_h y$, we have

$$\mathrm{pow}(y) = \mathrm{pow}\{\widetilde{y}\}. \tag{5.4}$$

For a lifted power signal $\widetilde{y}$ (i.e., $\mathrm{pow}\{\widetilde{y}\} < \infty$), define the autocorrelation function

$$R_y[k] := \lim_{N\to\infty} \frac{1}{2N} \sum_{n=-N}^{N} \langle \widetilde{y}[n], \widetilde{y}[n+k] \rangle,$$

where $\langle \cdot, \cdot \rangle$ is the inner product on $L^2_{[0,h)}$, that is, $\langle x, y \rangle := \int_0^h y(t)^T x(t) dt$. Let S_y denote the (discrete) Fourier transform of R_y:

$$S_y(e^{j\omega h}) := \sum_{k=-\infty}^{\infty} R_y[k] e^{-j\omega k h}.$$

Similarly, for a discrete-time u, define R_u and S_u as

$$\begin{aligned} R_u[k] &:= \lim_{N\to\infty} \frac{1}{2N} \sum_{n=-N}^{N} u[n+k]^T u[n], \\ S_u(e^{j\omega h}) &:= \sum_{k=-\infty}^{\infty} R_u[k] e^{-j\omega k h}. \end{aligned}$$

Then we have

$$\begin{aligned} \mathrm{pow}\{\widetilde{y}\}^2 = R_y[0] &= \frac{h}{2\pi} \int_0^{\frac{2\pi}{h}} S_y(e^{j\omega h}) d\omega, \\ \mathrm{pow}[u]^2 = R_u[0] &= \frac{h}{2\pi} \int_0^{\frac{2\pi}{h}} S_u(e^{j\omega h}) d\omega. \end{aligned} \tag{5.5}$$

For the sampled-data system $\mathcal{T}$, we denote by $\widetilde{\mathcal{T}}(e^{j\omega h})$ the frequency response [41] of the system $\mathcal{T}$. Let u and y be the input and output of $\mathcal{T}$ respectively, that is, $y = \mathcal{T} u$. Then we obtain

$$S_y(e^{j\omega h}) = \widetilde{\mathcal{T}}(e^{j\omega h}) \widetilde{\mathcal{T}^*}(e^{j\omega h}) S_u(e^{j\omega h}) = \|\widetilde{\mathcal{T}}(e^{j\omega h})\|^2 S_u(e^{j\omega h}), \tag{5.6}$$

where $\mathcal{T}^*$ is the dual system [41] of $\mathcal{T}$. The equation (5.6) can be proven by the same

method as the continuous-time version [7]. By using (5.4), (5.5) and (5.6), we have

$$\begin{aligned}\mathrm{pow}(y)^2 = \mathrm{pow}\{\widetilde{y}\}^2 &= \frac{h}{2\pi}\int_0^{\frac{2\pi}{h}} S_y(e^{j\omega h})d\omega \\ &= \frac{h}{2\pi}\int_0^{\frac{2\pi}{h}} \|\widetilde{T}(e^{j\omega h})\|^2 S_u(e^{j\omega h})d\omega \\ &\le \sup_{\omega\in(0,2\pi/h)} \|\widetilde{T}(e^{j\omega h})\|^2 \frac{h}{2\pi}\int_0^{\frac{2\pi}{h}} S_u(e^{j\omega h})d\omega \\ &= \|\widetilde{\mathcal{T}}\|_\infty^2 \,\mathrm{pow}[u]^2 = \|\mathcal{T}\|^2\,\mathrm{pow}[u]^2.\end{aligned}$$

□

Using Lemma 5.1, we have the following theorem.

Theorem 5.2. *For any input $w \in L^2$ and $d \in l^\infty$ of the sampled-data system $[\mathcal{T}_{zw}, \mathcal{T}_{zd}]$ defined above, the output z satisfies*

$$\mathrm{pow}(z) \le \frac{\gamma_2\Delta}{2}.$$

Proof. Since $\mathcal{T}_{zw}w \in L^2$, we have $\mathrm{pow}(\mathcal{T}_{zw}w) = 0$. Generally, if $\|d\|_\infty \le 1$ then $\mathrm{pow}[d] \le 1$, and hence

$$\sup_{\|d\|_\infty\le 1} \mathrm{pow}(\mathcal{T}_{zd}d) \le \sup_{\mathrm{pow}[d]\le 1} \mathrm{pow}(\mathcal{T}_{zd}d).$$

From Lemma 5.1 we have

$$\sup_{\mathrm{pow}[d]\le 1} \mathrm{pow}(\mathcal{T}_{zd}d) \le \|\mathcal{T}_{zd}\|.$$

Therefore

$$\begin{aligned}\mathrm{pow}(z) &= \mathrm{pow}(\mathcal{T}_{zw}w + \mathcal{T}_{zd}d) \\ &\le \mathrm{pow}(\mathcal{T}_{zw}w) + \mathrm{pow}(\mathcal{T}_{zd}d) \\ &= \mathrm{pow}(T_{zd}d) \le \|T_{zd}\|\|d\|_\infty \le \frac{\gamma_2\Delta}{2}.\end{aligned}$$

□

The theorem leads us to the conclusion that if we take the H^∞ design to attenuate $\|T_{zd}\|_\infty$, the quantization has a small effect on the output z with respect to the power, and hence the H^∞ design will be valid.

5.3 Differential pulse code modulation

5.3.1 Differential pulse code modulation

Differential pulse code modulation (DPCM) is a quantization system, which is used, for example, in the telephone communication systems. Figure 5.6 shows a DPCM system. The encoder quantizes the error $e = r - u$, where u is a prediction of the input signal r. If the filter $K_1(z)$ well predicts r from the quantizer output $\hat{e} := Qe$, the error e will be smaller than the input r, and hence fewer bits are required to quantize the signal. Assume that the quantizer Q is a uniform quantizer with a quantization level Δ.

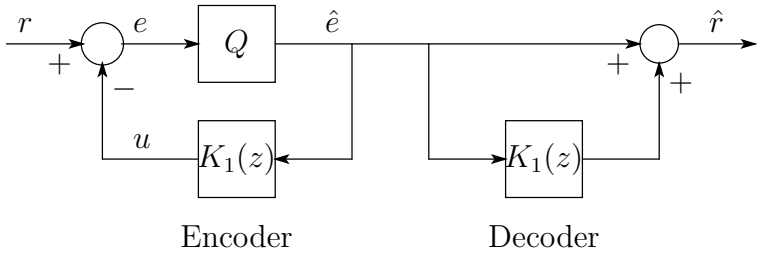

Figure 5.6: DPCM system

The signal $\hat{e}$ is transmitted into a communication channel or stored in digital media, and then the decoder reconstructs the original signal r and makes the output $\hat{r}$. Note that the filter in the decoder is the same as that in the encoder, and we have the following error estimate

$$\|\hat{r} - r\|_\infty = \|\hat{e} - e\|_\infty \le \frac{\Delta}{2}.$$

That is to say, the reconstruction error is less than $\Delta/2$. Therefore, DPCM can transmit data with fewer bits.

However, the model does not take account of the channel noise that adds the transmitted signal $\hat{e}$, and hence the estimate is not valid. For example, in the Δ modulation, the filter K_1 is the adder $K_1(z) = 1/(z-1)$. Since this filter is unstable, the channel noise will be amplified in the decoder.

Therefore, we use a model taking the channel noise into account for designing the encoder and the decoder.

5.3.2 Problem formulation

The block diagram of our DPCM system to be designed is illustrated in Figure 5.7. In the figure, n is the channel noise. Our purpose is to attenuate the prediction error e to be quantized and the reconstruction error. For this purpose, we design the filter $K_1(z)$ and $K_2(z)$ in Figure 5.7 by using the sampled-data H^∞ optimization method.

The block diagram of the error system for designing these filters is shown is Figure 5.8. We first take the quantization error caused by Q for the additive noise d. Then, assume that the analog signal to be quantized has frequency characteristic $W(s)$, and introduce a

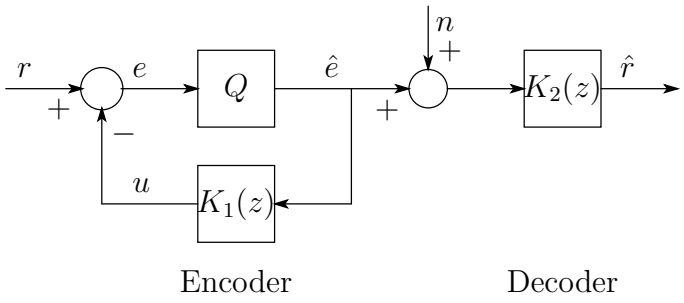

Figure 5.7: DPCM system with channel noise

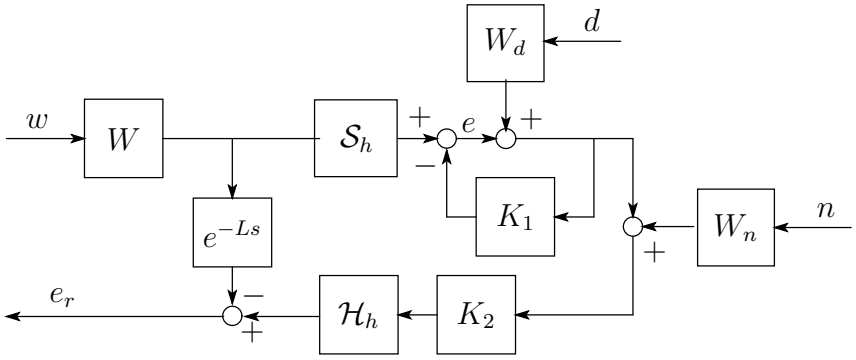

Figure 5.8: Error system for designing filters

time delay e^{-Ls}. This delay time L is the time that $K_1(z)$ and $K_2(z)$ will take to process signals. In the error system, let $\mathcal{T}_1$ be the system from $[w, d]^T$ to e, and $\mathcal{T}_2$ be the system from $[w, d, n]^T$ to e_r. Then the design problem is formulated as follows:

Problem 5.1. *Given an analog filter $W(s)$, time delay L and a sampling period h, find filters $K_1(z)$ and $K_2(z)$ that minimize*

$$\|\mathcal{T}_1\|^2 := \sup_{\substack{w\in L^2, d\in l^2 \\ \|w\|_{L^2}+\|d\|_{l^2}\neq 0}} \frac{\|e\|_{l^2}^2}{\|w\|_{L^2}^2 + \|d\|_{l^2}^2},$$

$$\|\mathcal{T}_2\|^2 := \sup_{\substack{w\in L^2, d, n\in l^2 \\ \|w\|_{L^2}+\|d\|_{l^2}+\|n\|_{l^2}\neq 0}} \frac{\|e_r\|_{l^2}^2}{\|w\|_{L^2}^2 + \|d\|_{l^2}^2 + \|n\|_{l^2}^2},$$

respectively.

This is a sampled-data H^∞ optimization problem and assuming $L = mh$ ($m \in \mathbb{N}$), the solution can be obtained by using the fast-sampling/fast-hold method discussed in Chapter 2.

Theorem 5.3. *Assume that $L = mh$, $m \in \mathbb{N}$. Then, for the sampled-data systems $\mathcal{T}_1$ and $\mathcal{T}_2$, there exist finite-dimensional discrete-time systems $\{T_{1,N} : N = 1, 2, \dots\}$ and*

$\{T_{2,N} : N = 1, 2, \dots\}$ *such that*

$$\lim_{N\to\infty} \|T_{1,N}\| = \|\mathcal{T}_1\|,$$
$$\lim_{N\to\infty} \|T_{2,N}\| = \|\mathcal{T}_2\|.$$

The proof of this theorem is almost the same as the proof of Theorem 5 in Chapter 4. The approximated discrete-time systems $T_{1,N}$ and $T_{2,N}$ are as follows:

$$T_{1,N} = \mathcal{F}_l(G_{1,N}, K_1), \quad G_{1,N} := \begin{bmatrix} \left[\widetilde{W}_N, 0\right] & -1 \\ \left[\widetilde{W}_N, W_d\right] & -1 \end{bmatrix},$$
$$T_{2,N} = \mathcal{F}_l(G_{2,N}, K_2), \quad G_{2,N} := \begin{bmatrix} [z^{-m}, 0, 0] & -1 \\ \left[S_d\widetilde{W}_N, S_dW_d, W_n\right] & 0 \end{bmatrix},$$
$$\widetilde{W}_N := [1, \underbrace{0, \dots, 0}_{N-1}]\mathcal{L}_N(W), \quad S_d := (1 + K_1)^{-1}.$$

5.4 Design example

In this section, we present a design example of DPCM. The design parameters are as follows: the sampling period $h = 1$, the reconstruction delay $L = 2$, the weighting functions are

$$W_d = 0.5, \quad W_n(z) = 0.1 \times \frac{0.01753z^2 - 0.03506z + 0.01753}{z^2 + 0.572z + 0.3147},$$

where $W_n(z)$ is a Chebyshev type I high-pass filter [9, 35, 46], and the analog filter $W(s)$ is

$$W(s) = \frac{1}{(10s + 1)^2}.$$

For comparison, we take the Δ modulation, that is, $K_1(z) = 1/(z - 1)$ and $K_2(z) = 1 + K_1(z) = z/(z - 1)$.

Figure 5.9 shows the frequency responses of the system $\mathcal{T}_1$. We can see that the system $\mathcal{T}_1$ designed by H^∞ optimization has lower gain than that of the Δ modulation over the whole frequency, in particular, in the high frequency range, our system shows better attenuation.

Figure 5.10 shows the frequency responses of the system $\mathcal{T}_2$. Since the Δ modulation is not stable (i.e., the system has a pole at $z = 1$), the frequency response of $\mathcal{T}_2$ is indefinite. Therefore in Figure 5.10, we take a decoder with the decoding filter $K_2 = 1+K_1$ where K_1 is the H^∞ (sub) optimal filter of the encoder, and compare it with the optimal decoding filter designed by sampled-data H^∞ optimization. We can see that the proposed system attenuates the gain over the whole frequency. It follows that the channel noise on the received signal will be attenuated more than the conventional system.

Then we show a simulation for the obtained DPCM systems. The parameters are as follows: the quantization level $\Delta = 0.125$, the input $r = \sin(\frac{\pi}{10}t)$, the channel noise

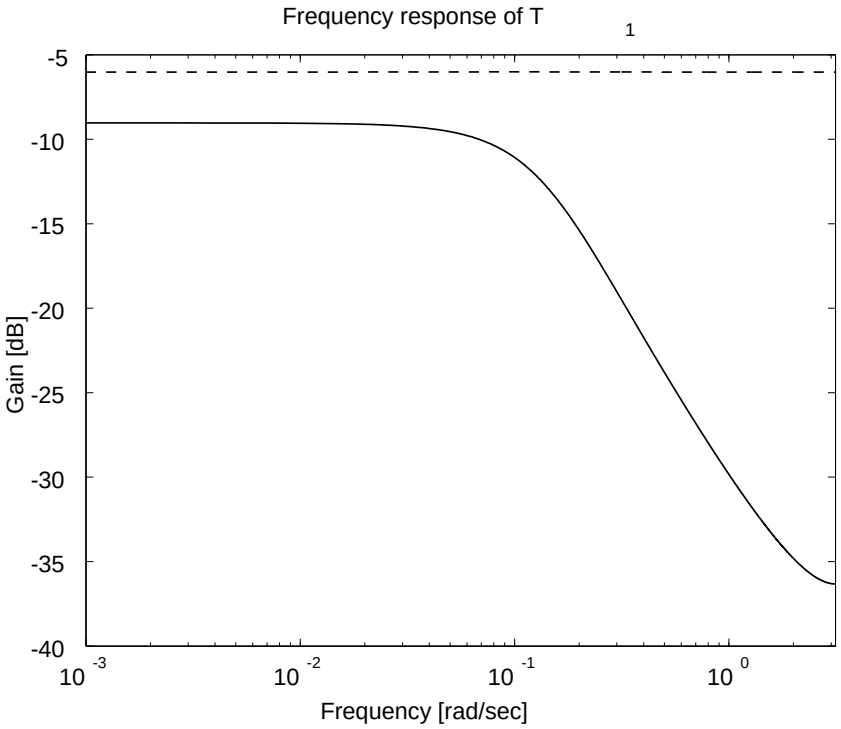

Figure 5.9: Frequency responses of $\mathcal{T}_1$: proposed (solid) and conventional (dash)

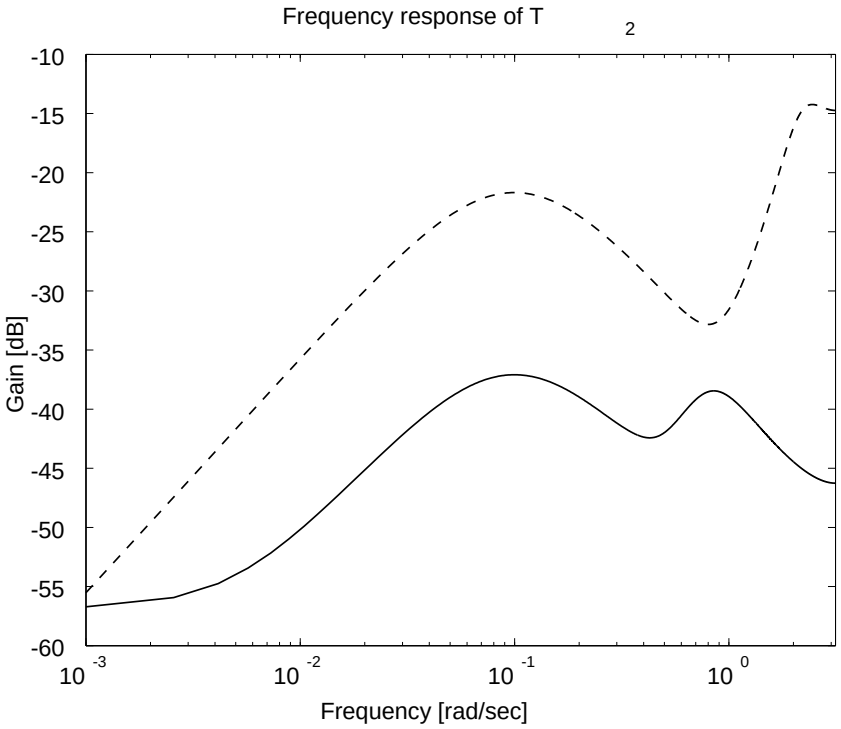

Figure 5.10: Frequency responses of $\mathcal{T}_2$: proposed (solid) and conventional (dash)

$n = 0.1\sin(2t)$ and the sampling period $h = 1$. Figure 5.11 shows the time response of the sampled-data designed DPCM system, and Figure 5.12 shows that of the conventional Δ modulation system. We can see that the proposed system attenuates the channel noise considerably better than the conventional system.

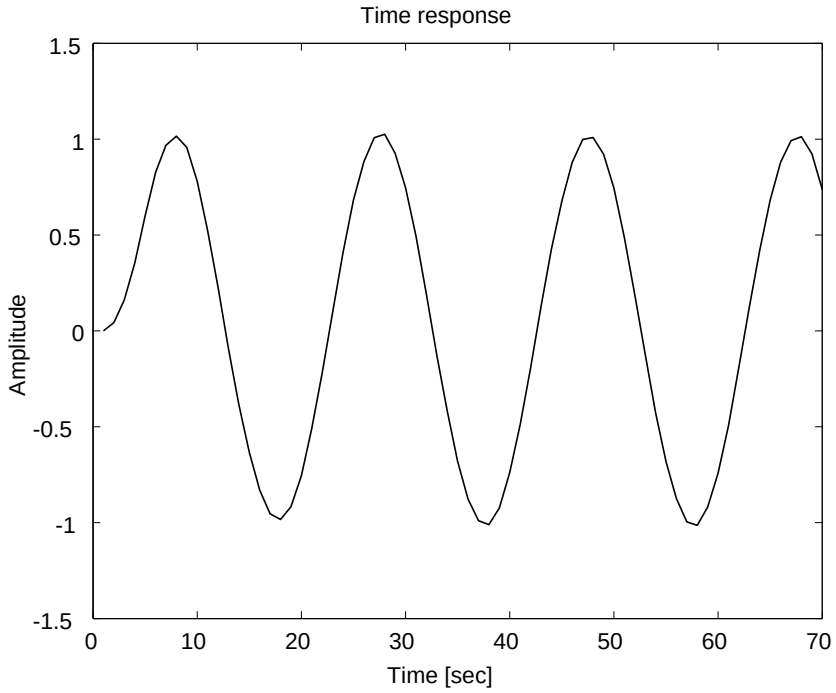

Figure 5.11: Time response of sampled-data designed system

5.5 Conclusion

In this chapter, we have discussed the stability and the performance of quantized sampled-data control systems. We have adopted the additive noise model, and by using it we have shown the BIBO stability and the performance of quantized systems. Moreover, we have proposed a new method for designing DPCM systems. Since the conventional Δ modulation system is not stable, a channel noise can be amplified at the encoder, while our system will attenuate the channel noise.

However, there remains an issue. We have not dealt with saturation in a quantizer. The additive noise model can be applied effectively in the case that the noise is small, but with saturation, the noise can be enormous. In this case, we have to treat the quantizer as a nonlinear system. We consider that hybrid system theory, in particular, switching system theory may be applicable to that case.

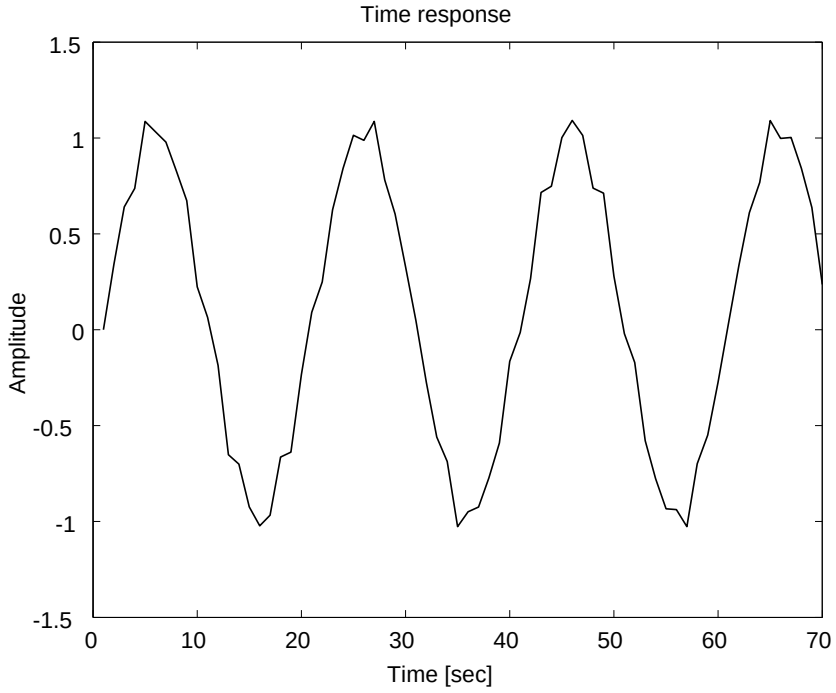

Figure 5.12: Time response of conventional Δ modulation system

Chapter 6

Optimal FIR Approximation

6.1 Introduction

According to the method we have discussed in the previous chapters, the filter we obtain is an IIR (Infinite Impulse Response) filter:

$$F(z) = \frac{\sum_{k=0}^{M} a_k z^{-k}}{1 + \sum_{k=1}^{N} b_k z^{-k}}. \tag{6.1}$$

The structure of an IIR filter is shown in Figure 6.1.

In practice, FIR (Finite Impulse Response) filters are often preferred to IIR ones. They have finitely many nonzero Markov parameters:

$$F(z) = \sum_{k=0}^{M} a_k z^{-k}. \tag{6.2}$$

The structure of an FIR filter is shown in Figure 6.2.

The reasons why FIR filters are often preferred to IIR filters are as follows [46]:

- FIR filters are intrinsically stable; the stability issue is a non-issue.
- They can easily realize various features that are not possible or are difficult to achieve with IIR filters, e.g., linear phase property.
- They can be free from certain problems in implementation, for example, limit cycles, attributed to quantization and the existence of a feedback loop in IIR filters.

On the other hand, a design process may have to start with an IIR filter for variety of reasons. For example, we have a large number of continuous-time filters available, and a digital filter may be obtained by discretizing one of them. It is then desired that such an IIR filter be approximated by an FIR filter. An easy way of doing this is to just truncate the Markov parameters of the IIR filter at a desired number of steps. This may however lead to either a very high-dimensional filter with good approximation, or a lower dimensional filter with an unsatisfactory approximation, depending on the truncation point.

The following problem is thus very natural and of importance:

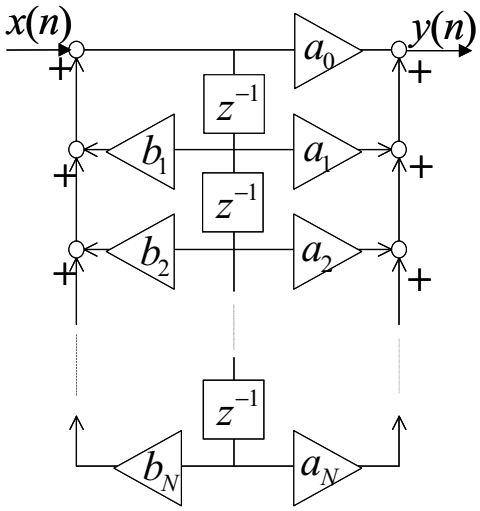

Figure 6.1: IIR filter

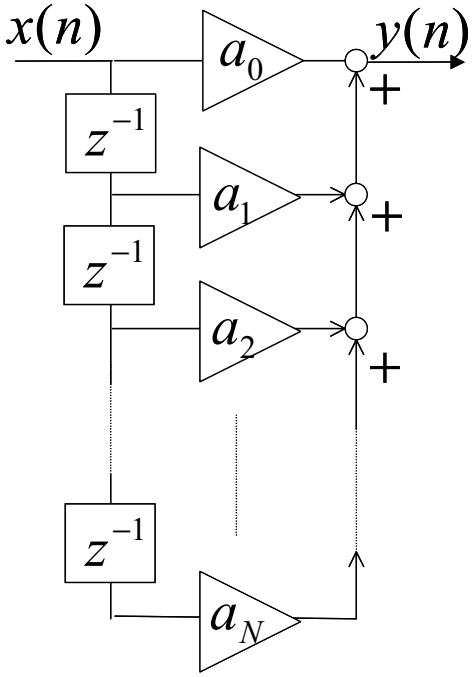

Figure 6.2: FIR filter

Problem 6.1. *Given an IIR filter $K(z)$ and a positive integer N, find an optimal FIR approximant $K_f(z)$ that has order N and approximates $K(z)$ with respect to a certain performance measure.*

There is a very elegant method called the *Nehari shuffle*, proposed by Kootsookos et al. [22, 23]. Its basic idea may be described as follows: For a given IIR filter G and a desired degree $r-1$ that an approximating FIR filter should assume, one first truncate the impulse response of G to the first r steps. This is a mere truncation, and it may induce a large error. One then takes its residual G_1, and suitably shifting and taking the mirror image, one can reduce this to the situation of the Nehari extension (approximation) problem. This will induce a truncation in the second step. By taking the residual further, this process can be continued, and the approximation can be improved in each step. (Details may be found in [31]. An advantage here is that this procedure gives rise to certain a priori and a posteriori error bounds. On the other hand, it does not necessarily give an optimal approximation with respect to the H^∞-norm.

In contrast to the Nehari shuffle, we here propose a method that directly deals with (sub)optimal approximants with respect to the H^∞ error norm. It is shown that

- the design problem is reducible to a Linear Matrix Inequality(LMI) [3]; and
- the obtained filter can be made close to be optimal by an iterative procedure.

A comparison with the Nehari shuffle is made for the Chebyshev filter of order 8, which has been studied in detail in [23].

6.2 FIR approximation problem

Consider the block diagram Figure 6.3. $K(z)$ is a given (rational and stable) IIR filter,

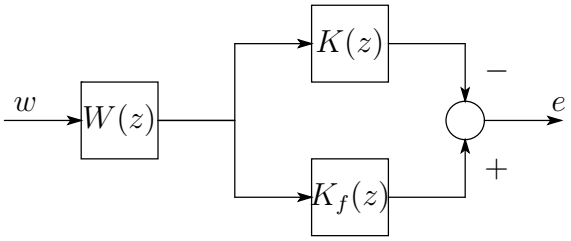

Figure 6.3: Error system

$W(z)$ is a proper and rational weighting function, and $K_f(z)$ is an FIR filter of order N. Denote by $T_{ew}(z)$ the transfer function from w to e in Figure 6.3. The objective here is to find $K_f(z)$ that makes the H^∞ error norm less than a prespecified bound $\gamma > 0$, that is,

$$\|T_{ew}\|_\infty := \sup_{\substack{w\in l^2 \\ w\neq 0}} \frac{\|T_{ew}w\|_2}{\|w\|_2} < \gamma.$$

Introduce state space realizations

$$W(z) := C_W(zI - A_W)^{-1}B_W + D_W,$$
$$K(z) := C_K(zI - A_K)^{-1}B_K + D_K,$$

and

$$K_f(z) = \sum_{k=0}^{N} a_k z^{-k} = C_f(\alpha)(zI - A_f)^{-1}B_f + D_f(\alpha),$$
$$C_f(\alpha) = \begin{bmatrix} a_N & a_{N-1} & \dots & a_1 \end{bmatrix}, \quad D_f(\alpha) = a_0,$$
$$\alpha = \begin{bmatrix} a_N & a_{N-1} & \dots & a_0 \end{bmatrix},$$

where $\alpha = \begin{bmatrix} a_N & a_{N-1} & \dots & a_0 \end{bmatrix}$ denote the Markov parameters of the filter $K_f(z)$ to be designed. The matrices A_f and B_f are defined as follows:

$$A_f = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ \vdots & & & \ddots & 1 \\ 0 & \cdots & \cdots & \cdots & 0 \end{bmatrix}, \quad B_f = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix},$$

and they contain just zeros and ones.

A realization of T_{ew} is given as follows:

$$T_{ew}(z) =: C(\alpha)(zI - A)^{-1}B + D(\alpha),$$
$$A = \begin{bmatrix} A_W & 0 & 0 \\ B_K C_W & A_K & 0 \\ B_f C_W & 0 & A_f \end{bmatrix}, \quad B = \begin{bmatrix} B_W \\ B_K D_W \\ B_f D_W \end{bmatrix},$$
$$C(\alpha) = \begin{bmatrix} (D_f(\alpha) - D_K)C_W & -C_K & C_f(\alpha) \end{bmatrix},$$
$$D(\alpha) = \begin{bmatrix} (D_K + D_f(\alpha))D_W \end{bmatrix}.$$

The important asset here is that the design parameter α appears only in the C and D matrices linearly, and the underlying structure is of the one-block type. Hence the overall transfer operator is linear in α, and the design problem of choosing α to minimize the H_∞-norm can be expected to become a linear matrix inequality. In fact, the bounded real lemma [3] readily yields the following:

Theorem 6.1. *$\|T_{ew}\|_\infty < \gamma$ if and only if there exists $P > 0$ such that*

$$\begin{bmatrix} A^T P A - P & A^T P B & C(\alpha)^T \\ B^T P A & -\gamma I + B^T P B & D(\alpha)^T \\ C(\alpha) & D(\alpha) & -\gamma I \end{bmatrix} < 0. \tag{6.3}$$

Proof. By the bounded real lemma [3], $\|T_{ew}\|_\infty < \gamma$ is equivalent to the condition that there exists a matrix $\tilde{P} > 0$ such that

$$Q^T \begin{bmatrix} \tilde{P} & 0 \\ 0 & I \end{bmatrix} Q < \begin{bmatrix} \tilde{P} & 0 \\ 0 & \gamma^2 I \end{bmatrix}, \tag{6.4}$$

where

$$Q := \left[\begin{array}{cc} A & B \\ C(\alpha) & D(\alpha) \end{array} \right].$$

Although the inequality (6.4) is not affine in α, it can be converted to an affine one by the Schur complement [3]:

$$\left[\begin{array}{cc} \Phi_{11} & \Phi_{12} \\ \Phi_{12}^T & \Phi_{22} \end{array} \right] < 0,$$

is equivalent to $\Phi_{22} < 0$ and $\Phi_{11} < \Phi_{12}\Phi_{22}^{-1}\Phi_{12}^T$. By dividing the inequality (6.4) by $\gamma > 0$, we get

$$\left[\begin{array}{cc} A^T\gamma^{-1}\tilde{P}A - \gamma^{-1}\tilde{P} & A^T\gamma^{-1}\tilde{P}B \\ B^T\gamma^{-1}\tilde{P}A & B^T\gamma^{-1}\tilde{P}B - \gamma I \end{array} \right] < \left[\begin{array}{c} C^T \\ D^T \end{array} \right] (-\gamma^{-1} I) \left[\begin{array}{cc} C & D \end{array} \right].$$

Then by using the Sure complement for

$$\begin{aligned} \Phi_{11} &:= \left[\begin{array}{cc} A^TPA - P & A^TPB \\ B^TPA & B^TPB - \gamma I \end{array} \right]^T, \\ \Phi_{22} &:= -\gamma I, \\ \Phi_{12} &:= \left[\begin{array}{cc} C & D \end{array} \right]^T, \end{aligned}$$

where $P := \gamma^{-1}\tilde{P} > 0$, we get the inequality (6.3). □

The obtained condition is an LMI in α, and can be effectively solved by standard MATLAB routines [11].

6.3 Numerical example

6.3.1 Comparison of H^∞ design via LMI and the Nehari shuffle

Take the following Chebyshev filter of order 8

$$\begin{aligned} K(z) &= 10^{-3} \\ &\times \frac{0.04705z^8 + 0.3764z^7 + 1.317z^6 + 2.635z^5 + 3.294z^4 + 2.635z^3 + 1.317z^2 + 0.3764z + 0.04705}{z^8 - 4.953z^7 + 11.71z^6 - 16.95z^5 + 16.29z^4 - 10.58z^3 + 4.552z^2 - 1.161z + 0.1369} \end{aligned}$$

as a target filter to be approximated. This has been studied thoroughly by Kootsookos and Bitmead [23] for the Nehari shuffle, and is suitable for comparison with the present method. For simplicity, we confine ourselves to approximations by FIR filters with 32 tap coefficients (of order 31).

The design depends crucially on the choice of the weight $W(z)$. One natural choice ([31]) would be to take $W(z)$ to be equal to $K^{-1}(z)$ (or some variant of it having the same gain on the imaginary axis, since K is not minimum phase). This is relative error approximation, where (approximately) dB and phase errors are weighted uniformly with

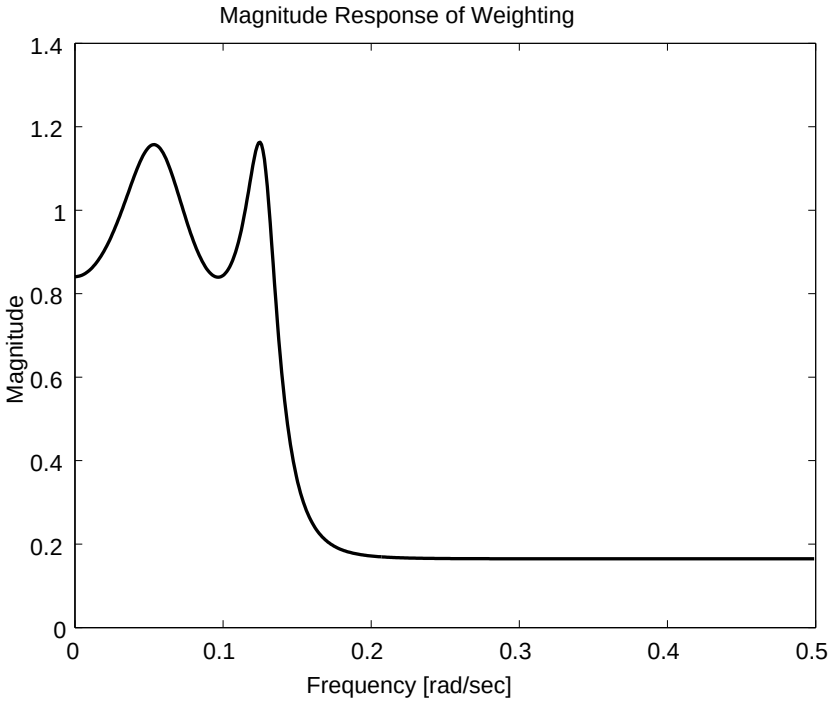

Figure 6.4: Inverse of the weighting function

frequency. Since the optimal overall error in Figure 6.3 will become all-pass, this will have the effect of attenuating the stop-band error with the weight of $K^{-1}(z)$ (which is very large) while maintaining reasonable pass-band characteristic. Unfortunately, however, due to the very small gain of $K(z)$ in the stop-band, this will make the solution of the approximation problem Figure 6.3 numerically hard. Neither the Nehari shuffle nor the LMI method gave a satisfactory result in this case. Hence one should sacrifice the stop-band attenuation to obtain a reasonable $W(z)$. There is also a trade-off, empirically observed, between the stop-band attenuation and the pass-band ripples.

Kootsookos and Bitmead [23] thus employed the weight as depicted in Figure 6.4. To be precise, the frequency response shown here is the inverse of the para-Hermitian conjugate of the weight function. The reason for taking the para-Hermitian conjugate is that the Nehari shuffle makes use of causal approximation for anti-causal transfer function, so that we must reciprocate the poles and zeros. Then by taking the inverse, the weight attenuates the stop-band by the inverse of its gain and approximately shapes the pass-band as it is in the pass band. On the other hand, for the FIR approximation as in Figure 6.3, we simply take the inverse of this weight, since we do not need to make the weight anti-stable.

The gain responses of obtained FIR filters based on the Nehari shuffle and Theorem 6.1 are given in Figure 6.5. Figure 6.6 shows their phase plots.

We see that the gain of the H^∞ approximant shows smaller pass-band ripples and better stop-band attenuation than those by the Nehari shuffle. The phase characteristics of these are about the same up to the edge of the transition band.

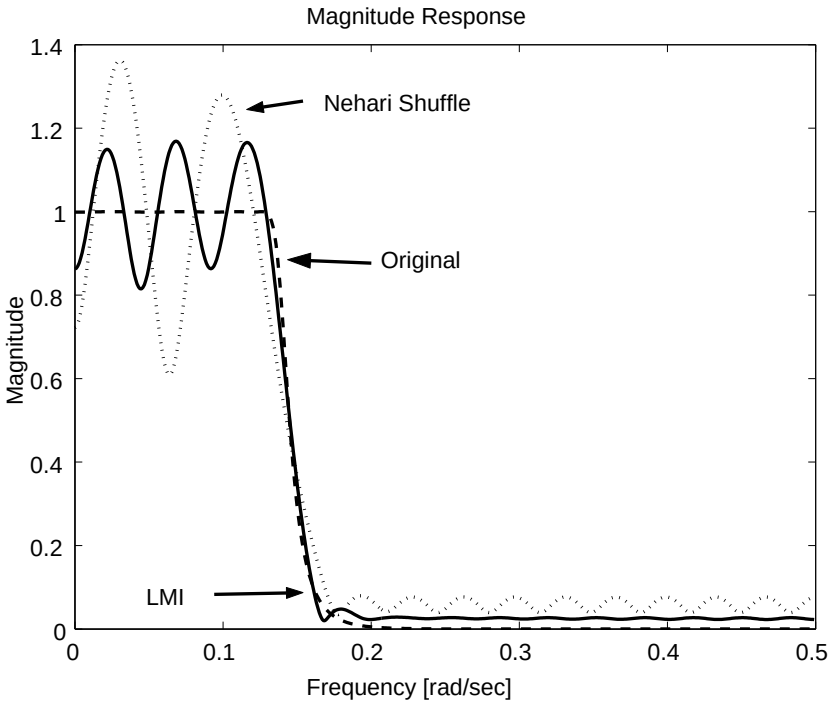

Figure 6.5: Gain responses of FIR approximants with weight function in Figure 6.4: H^∞ via LMI (solid), Nehari shuffle (dots) and original IIR Chebyshev filter (dash)

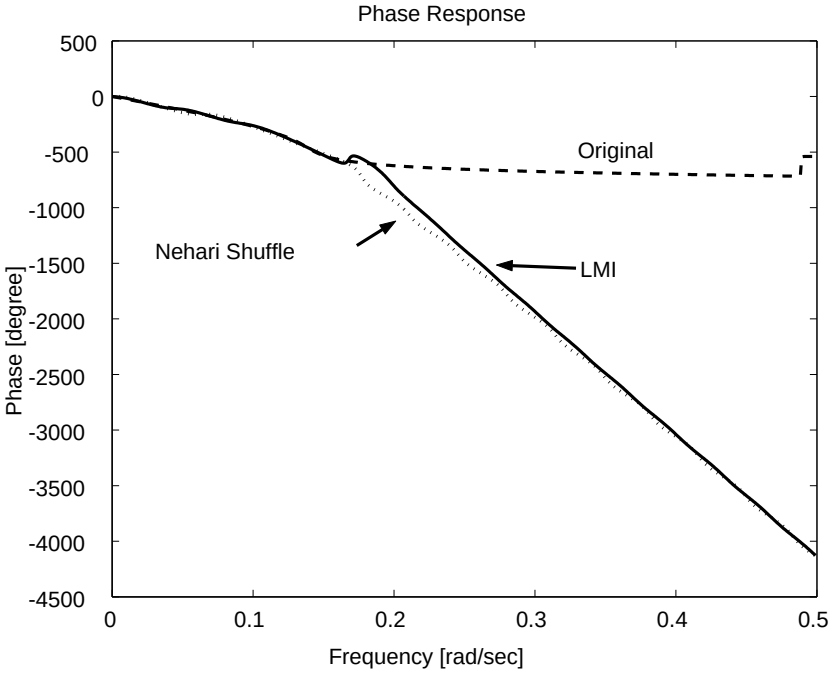

Figure 6.6: Phase plots of FIR approximants with weight function in Figure 6.4: H^∞ via LMI (solid), Nehari shuffle (dots) and Chebyshev (dash)

Figure 6.7 shows the error magnitude responses. The design by the LMI method has the advantage of 5–7 dB over the one by the Nehari shuffle.

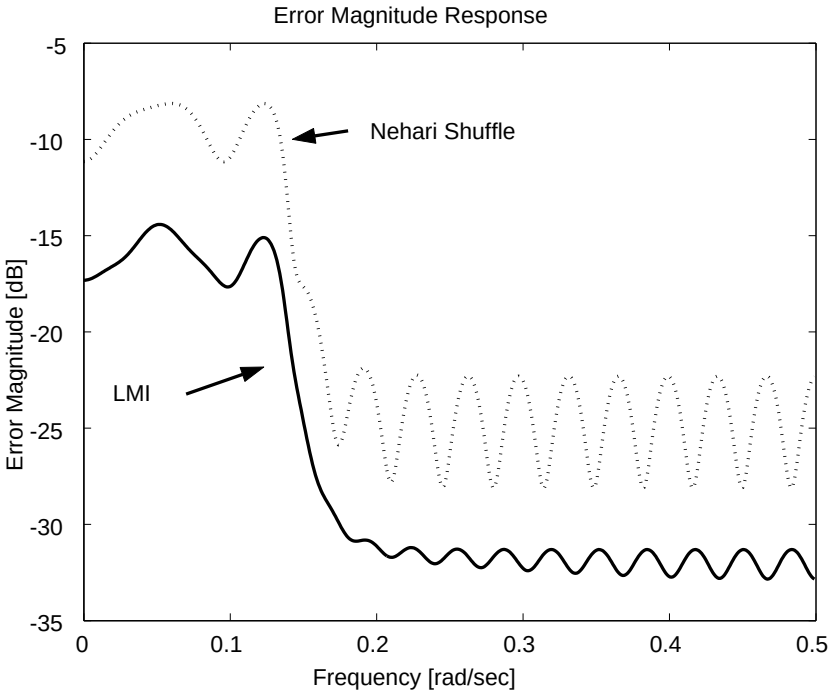

Figure 6.7: Gain of the error $K - K_f$: H^∞ design via LMI (solid); Nehari shuffle (dots)

6.3.2 Trade-off between pass-band and stop-band characteristics

The design in the previous subsection depends crucially on the weighting function. It is desirable to obtain smaller pass-band ripples while maintaining reasonable stop-band attenuation. In this section we attempt to see how the choice of a weighting function affects the overall approximation.

We consider the following three weighting functions:

$$W_1 = \frac{0.7661z^2 - 1.305z + 0.675}{z^2 - 1.735z + 0.9289},$$

$$W_2 = \frac{0.2831z^4 - 0.5515z^3 + 0.5416z^2 - 0.2708z + 0.05882}{z^4 - 2.865z^3 + 3.6z^2 - 2.268z + 0.6056},$$

$$W_3 = 10^{-3} \times \frac{14.44z^7 - 7.838z^6 + 19.02z^5 - 4.448 + 6.697z^3 - 0.1857z^2 + 0.5287z + 0.01134}{z^7 - 4.229z^6 + 8.561z^5 - 10.43z^4 + 8.172z^3 - 4.089z^2 + 1.206z - 0.1613}.$$

Table 6.1: H^∞ and H^2 error norm

	W_1	W_2	W_3
H^∞ error norm	0.0954	0.1838	0.2627
H^2 error norm	0.0713	0.0840	0.1333

These functions W_1, W_2, W_3 are, respectively, obtained as the 2nd, 4th, 7th-order Hankel norm approximations [31] of the IIR Chebyshev filter to be approximated. The weight W_2 is the same as that used in the previous section. Their magnitude frequency responses are shown in Figure 6.8.

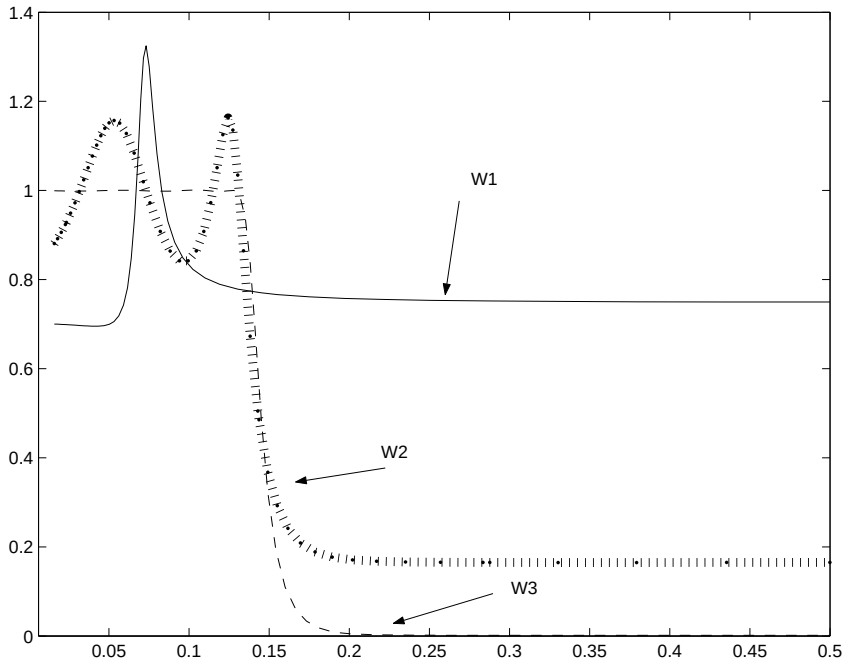

Figure 6.8: Gain of inverse of weighting functions

Figure 6.9 and Figure 6.10 show the resulting gain and phase responses of the FIR filters designed with respective weighting functions. Figure 6.9 in particular shows that there is a clear trade-off between the magnitude of the pass-band ripples and the stop-band attenuation. That is, if we attempt to decrease the stop-band error, we must sacrifice the pass-band characteristic (i.e., larger ripples), and vice versa.

Figure 6.11 shows the error magnitude responses. Table 1 also shows the H^∞ and H^2 error norms. Interestingly, the design via W_1 exhibits the best overall approximation in both performance measures, although its stop-band attenuation is not as good as those by W_2 and W_3. Note also that the one by W_3 approximates the phase characteristic of the original filter up to the edge of the stop-band as Figure 6.10 shows.

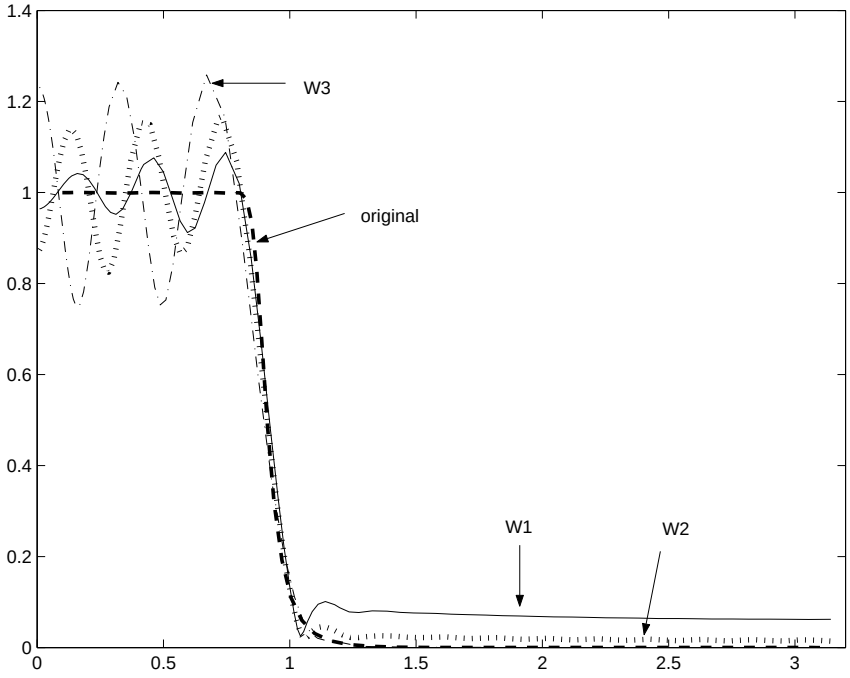

Figure 6.9: Gain responses of FIR filters via LMI

6.4 Conclusion

We have given an LMI solution to the optimal H^∞ approximation of IIR filters via FIR filters. A comparison with the Nehari shuffle is made with a numerical example, and it is observed that the LMI solution generally performs better. Another numerical study also indicates that there is a trade-off between the pass-band and stop-band approximation characteristics.

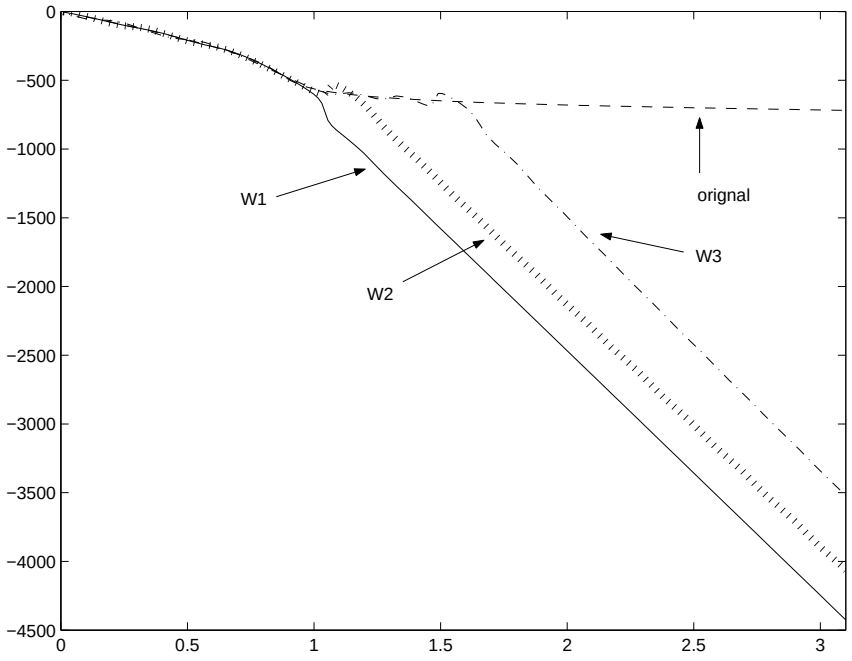

Figure 6.10: Phase responses of FIR filters via LMI

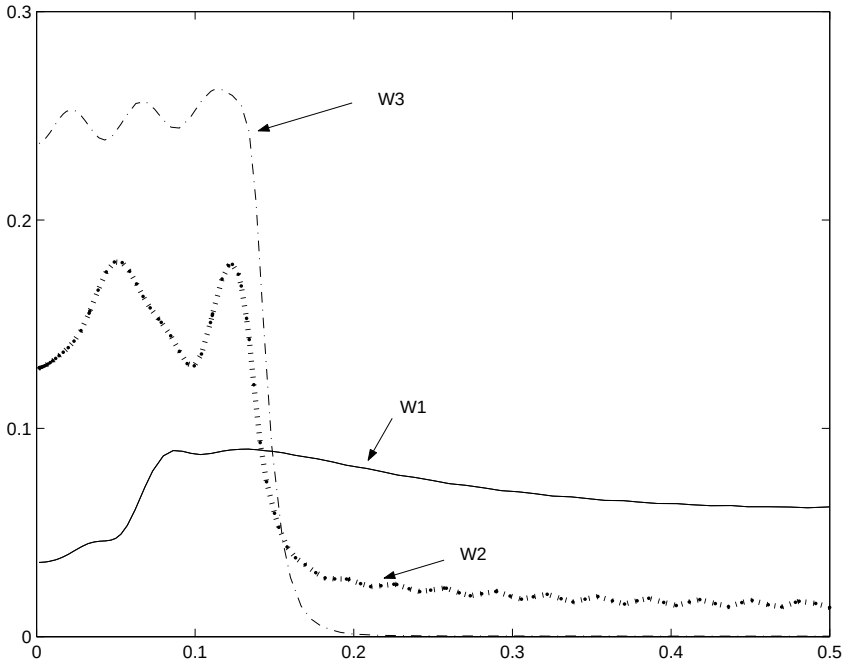

Figure 6.11: Gain responses of the error $K_f - K$

Chapter 7

Conclusion

In this thesis, we proposed a new design method for multirate digital signal processing and digital communication systems. Conventionally, they are designed under the assumption that the original analog signal is fully band-limited, while our method takes the analog characteristic into account that goes often beyond the Nyquist frequency, and optimizes the analog performance via the sampled-data H^∞ optimization.

In Chapter 3, we have presented a sampled-data design for multirate signal processors, in particular, interpolation, decimation and sampling rate conversion. Conventionally, a filter used in these systems is designed to have a sharp characteristic which approximates the ideal filter. However, as shown by design examples, such a sharp filter is not necessarily optimal for reconstruction. This fact will not be recognized without taking the analog signal into account.

In Chapter 4, we have treated communication systems which contains signal compression. Under distortions by a channel, we have presented a design method of a transmitting filter and a receiving filter by using sampled-data H^∞ optimization. By iterating a transmitting filter design and a receiving filter design, we can obtain sub-optimal filters. We have shown that the objective function monotonically decreases by the iteration.

In Chapter 5, we have investigated the stability and the performance of quantized sampled-data control systems. By using an additive noise model (linearized model) for the quantization, we have shown that if the linearized model is stable, the states of the quantized system are bounded, and that if the linearized model has small L^2 gain, the quantized system has small power gain. Then we have applied the results to DPCM design. We have pointed out that the conventional Δ modulation is not stable and a channel noise will be amplified at the decoder. Therefore we have proposed a design of the decoder and the encoder; the decoder reduces the quantization noise, while the encoder reduces the channel noise.

In Chapter 6, We have given an LMI solution to the optimal H^∞ approximation of IIR filters via FIR filters. A comparison with the Nehari shuffle is made with a numerical example, and it is observed that the LMI solution generally performs better. Another numerical study also indicates that there is a trade-off between the pass-band and stop-band approximation characteristics.

We conclude by giving future directions of research as follows:

- We have treated one-dimensional signals (e.g., audio/speech) and we believe that the

present method can be effectively extended to image processing, which is a future direction of our research. In particular, the popular format JPEG or MPEG is a multirate filter bank system, which can be designed by using the method discussed in Chapter 3.

- We have discussed a design of communication systems, where the channel is time-invariant. However, the real channel often contains time-varying systems, in particular, in the case of wireless communication. Moreover, the real channel is very complicated and we should notice that the model of the channel always contains a modeling error. To overcome this, we have to choose an adaptive filter. Design for adaptive filters by using sampled-data theory is an important subject for the future.

Bibliography

[1] B. Bamieh and J. B. Pearson: A general framework for linear periodic systems with application to H_∞ sampled-data control, *IEEE Trans. Autom. Control*, **AC-37**, pp. 418–435, 1992.

[2] B. Bamieh, J. B. Peason, B. A. Francis and A. Tannenbaum: A lifting technique for linear periodic systems with applications to sampled-data control systems, *Syst. Control Lett.*, **vol 17**, pp. 79–88, 1991.

[3] S. Boyd, L. E. Ghaoui, E. Feron and V. Balakrishnan: *Linear Matrix Inequalities in Systems and Control Theory*, SIAM, 1994.

[4] T. Chen and B. A. Francis: *Optimal Sampled-Data Control Systems*, Springer, 1995.

[5] T. Chen and B. A. Francis: Design of multirate filter banks by $\mathcal{H}^\infty$ optimization, *IEEE Trans. Signal Processing*, **SP-43**, pp. 2822–2830, 1995.

[6] D. F. Delchamps: Stabilizing a linear system with quantized state feedback, *IEEE Trans. Autom. Control*, **AC-35**, pp. 916–924, 1990.

[7] J. C. Doyle, B. A. Francis and A. R. Tannenbaum: *Feedback Control Theory,* Maxwell Macmillan, 1992.

[8] A. T. Erdogan, B. Hassibi and T. Kailath: On linear H^∞ equalization of communication channels, *IEEE Trans. Signal Processing*, **SP-48**, No. 11 pp. 3227–3232, 2000.

[9] N. J. Fliege: *Multirate Digital Signal Processing*, Wiley, 1994.

[10] G. F. Franklin, J. D. Powell, M. Workman: *Digital Control of Dynamic Systems 3rd Ed.*, Addison Wesley, 1998.

[11] P. Gahinet, A. Nemirovski, A. J. Laub and M. Chilali: *LMI Control Toolbox*, the Math Works Inc., 1995.

[12] S. Hara, H. Fujioka, P. P. Khargonekar and Y. Yamamoto: Computational aspects of gain frequency response for sampled-date systems, *Proc. of 34th Conf. on Decision and Control*, pp. 1784–1789, 1995.

[13] Y. Hayakawa, S. Hara and Y. Yamamoto: H_∞ type problem for sampled-data control systems — a solution via minimum energy characterization, *IEEE Trans. Autom. Control*, **AC-39**, pp. 2278–2284, 1994.

[14] H. Ishii and Y. Yamamoto: Sampled-data H^∞ and H^2/H^∞ design of multirate D/A converter, *Systems, Information and Control*, vol. 11, No. 10, pp. 586–593 1998. (in Japanese)

[15] H. Ishii, Y. Yamamoto and B. A. Francis: Sample-rate conversion via sampled-data H^∞ control, *Proc. of 38th Conf. on Decision and Control*, pp. 3440–3445, 1999.

[16] A. J. Jerri: The Shannon sampling theorem - its various extension and applications: a tutorial review, *Proc. IEEE*, vol. 65, pp.1565-1596, 1977.

[17] J. D. Johnston: A filter family designed for use in quadrature mirror filter banks, *Proc. of IEEE International Conf. on Acoustics, Speech and Signal Processing*, pp. 291–294, 1980.

[18] P. T. Kabamba and S. Hara: Worst case analysis and design of sampled data control systems, *IEEE Trans. Autom. Control*, **AC-38**, pp. 1337–1357, 1993.

[19] J. P. Keller and B. D. O. Anderson: A new approach to the discretization of continuous-time controllers, *IEEE Trans. Autom. Control*, **AC-37**, pp. 214–223, 1992.

[20] P. P. Khargonekar, K. Poolla and A. Tannenbaum: Robust control of linear time-invariant plants using periodic compensation, *IEEE Trans. Autom. Control,* **AC-30** pp. 1088–1096, 1985.

[21] P. P. Khargonekar and Y. Yamamoto: Delayed signal reconstruction using sampled-data control, *Proc. of 35th Conf. on Decision and Control*, pp. 1259–1263, 1996.

[22] P. J. Kootsookos, R. B. Bitmead and M. Green: The Nehari shuffle: FIR(q) filter design with guaranteed error bounds, *IEEE Trans. Signal Processing*, **SP-40**, pp. 1876–1883, 1992.

[23] P. J. Kootsookos and R. B. Bitmead: The Nehari shuffle and minimax FIR filter design, in *Control and Dynamic Systems, Academic Press*, **64**, pp. 239–298, 1994.

[24] S. Mallat: *A Wavelet Tour of Signal Processing*, Academic Press, 1998.

[25] D. G. Meyer: A new class of shift-varying operators, their shift-invariant equivalents, and multi-rate digital systems, *IEEE Trans. Autom. Control,* **AC-35,** pp. 429–433, 1990.

[26] M. Nagahara, S. Ashida and Y. Yamamoto: An analysis for sampled-data systems with quantization and quantizer design, *SICE 2nd Annual Conference on Control Systems,* pp. 175–178, 2002. (in Japanese)

[27] M. Nagahara and Y. Yamamoto: A new design for sample-rate converters, *Proc. of 39th Conf. on Decision and Control*, pp. 4296–4301, 2000.

[28] M. Nagahara and Y. Yamamoto: Sampled-data H^∞ design of interpolators, *Systems, Information and Control*, **vol. 14, No. 10**, pp. 483–489, 2001. (in Japanese)

[29] M. Nagahara and Y. Yamamoto: Design for digital communication systems via sampled data H^∞ control, *IFAC Workshop on Periodic Control Systems*, pp. 211–216, 2001.

[30] M. Nagahara and Y. Yamamoto: Sampled-data H^∞ design for digital communication systems, *Systems, Information and Control*, **vol. 16, No. 1**, pp. 38–43, 2003. (in Japanese)

[31] G. Obinata and B. D. O. Anderson: *Model Reduction for Control System Design*, Springer-Verlag, 2001.

[32] J. G. Proakis: *Digital Communications,* McGraw Hill, 1989.

[33] J. G. Proakis and M. Salehi: *Communication Systems Engineering,* Prentice Hall, 1994.

[34] H. T. Toivonen: Sampled-data control of continuous-time systems with an $\mathcal{H}_\infty$ optimality criterion, *Automatica*, **vol. 28**, pp. 45–54, 1992.

[35] P. P. Vidyanathan: *Multirate Systems and Filter Banks*, Prentice Hall, 1993.

[36] Y. Yamamoto: New approach to sampled-data systems: a function space method, *Proc. of 29th Conf. on Decision and Control*, pp. 1881–1887, 1990.

[37] Y. Yamamoto: A function space approach to sampled-data control systems and tracking problems, *IEEE Trans. Autom. Control,* **AC-39** pp. 703–712, 1994.

[38] Y. Yamamoto: Digital Control, *Encyclopedia of Electrical and Electronics Engineering,* John-Wiley, J. Webster Ed., **5**, pp. 445–457, 1999.

[39] Y. Yamamoto, B. D. O. Anderson and M. Nagahara: Approximating sampled-data systems with applications to digital redesign, *Proc. of 41th Conf. on Decision and Control*, pp. 3724–3729, 2002.

[40] Y. Yamamoto, H. Fujioka and P. P. Khargonekar: Signal reconstruction via sampled-data control with multirate filter banks, *Proc. of 36th Conf. on Decision and Control*, pp. 3395–3400, 1997.

[41] Y. Yamamoto and P. P. Khargonekar: Frequency Response of sampled-data systems, *IEEE Trans. Autom. Control*, **AC-41** pp. 166–176, 1996.

[42] Y. Yamamoto and P. P. Khargonekar: From sampled-data control to signal processing, in *Learning, Control and Hybrid Systems*, Springer Lecture Notes in Control and Information Sciences, vol. 241, pp. 108–126, 1998.

[43] Y. Yamamoto, A. G. Madievski and B. D. O. Anderson: Approximation of frequency response for sampled-data control systems, *Automatica*, **vol. 35**, pp. 729-734, 1999.

[44] Y. Yamamoto, M. Nagahara and H. Fujioka: Multirate signal reconstruction and filter design via sampled-data H^∞ control, *MTNS2000*, 2000.

[45] A. I. Zayed: *Advances in Shannon's Sampling Theory,* Boca Raton, CRC Press, 1993.

[46] G. Zelniker and F. J. Taylor: *Advanced Digital Signal Processing: Theory and Applications*, Marcel Dekker, 1994.