Graz University of Technology

Doctoral Thesis

Low Complexity Correction Structures for Time-Varying Systems

Michael Soudan

Signal Processing and Speech Communication Laboratory
Graz University of Technology, Austria

First Examiner:
Univ.-Prof. Dr. Gernot Kubin
Graz University of Technology, Austria

Second Examiner:
Prof. Dr. Peter Händel
KTH Royal Institute of Technology, Sweden

Graz, September 2011

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Graz, am ....

(Unterschrift)

Kurzfassung

Zeitvariante Systeme treten in vielen Bereichen der Signalverarbeitung auf. Sofern das zeitvariante Verhalten des Systems unerwünscht ist, erzeugt das System ein verfälschtes Ausgangssignal. Dedizierte zeitvariante Systeme können mit dem ursprünglichen System kaskadiert werden, um die Auswirkung des unerwünschten zeitvarianten Verhaltens auf das Ausgangssignal zu korrigieren. In Anwendungen, in denen mit hoher Genauigkeit korrigiert werden soll, stellt der Rechenaufwand für den Entwurf und Einsatz von flexiblen digitalen Korrektursystemem eine Herausforderung dar. Dieser Rechenaufwand ist im Besonderen problematisch, wenn das Korrektursystem zur Laufzeit für jeden Zeitpunkt neu entworfen werden muss. In dieser Arbeit werden Methoden mit geringer Komplexität für den Entwurf von linearen zeitvarianten Korrekturfiltern vorgestellt. Diese Korrekturfilter können lineare zeitvariante Systeme sowohl vor- als auch nachbearbeiten. Um den Rechenaufwand zu verringern, wurde ein Algorithmus mit geringer Komplexität für die Methode der kleinsten Quadrate entwickelt, der Vorwissen für den Enwurf nutzt und sowohl für die Vor- als auch für die Nachverarbeitung verwendet werden kann. Weiterhin wurden Strukturen zur Korrektur von schwach zeitvarianten linearen Systeme vorgestellt. Der Einsaz dieser Stukturen ermöglicht eine starke Verringerung des Rechenaufwands, der für einen neuen Entwurf des Korrektursystems notwendig ist, jedoch zu einer größeren Komplexität der Korrekturstruktur führt. Zuletzt wird die Erweiterung des zuvor genannten Prinzips auf schwach zeitvariante Volterra Systeme untersucht.

Abstract

Time-varying systems are encountered in various fields of engineering. If the time-varying behavior of a system is undesired, it produces a distorted output signal. Dedicated time-varying systems can be cascaded with the original system to correct the impact of the undesired time-varying behavior on the output signal. In applications where a high reconstruction accuracy is important, the computational cost of designing and employing flexible digital correction systems remains challenging. In particular, the computational load becomes a major challenge if the digital correction system needs to be redesigned for each time instant. In this thesis, low complexity correction methods for the design of linear time-varying correction filters are presented. These filters can be applied to postcorrect or precorrect linear time-varying systems. In order to mitigate the computational complexity of the filter design, a low complexity filter design algorithm for the least-squares norm is derived which exploits prior knowledge for the design and can be applied to postcorrect or precorrect linear time-varying systems. Furthermore, correction structures for weakly time-varying linear systems are presented. Employing these structures provides a significant reduction of the computational effort, that is required to redesign the correction system, at the expense of an increased complexity of the correction structure. Finally, the extension of the latter concept to time-varying Volterra systems is explored.

Acknowledgement

Several people made this work possible. First of all, I would like to thank Dr. Vogel for his unconditional support and helpful suggestions. Furthermore, I would like to express my sincere gratitude to Professor Kubin who supported me throughout my research with helpful discussions and comments. I would also like to thank all the members of the SPSC lab for our interesting discussions.

Michael Soudan
Graz, September 2011

Contents

List of Figures

List of Tables

List of Abbreviations

ADC	Analog-to-Digital Converter
BIBO	Bounded Input Bounded Output
CTFT	Continuous-Time Fourier Transform
DAC	Digital-to-Analog Converter
DTFS	Discrete-Time Fourier Series
DTFT	Discrete-Time Fourier Transform
FIR	Finite Impulse Response
SFDR	Spurious-Free Dynamic Range
SH	Sample-and-Hold
SNR	Signal-to-Noise Ratio
TIADC	Time-Interleaved Analog-to-Digital Converter
ZOH	Zero-Order Hold

Symbols and Operators

The list of symbols is not complete. Any additional symbols employed in this thesis will be introduced in the context of the chapter.

ZOH System Model

Ω	Continuous-time angular frequency
T	Sampling period
$\tilde{A}_n(j\Omega)$	CTFT of a time-varying ZOH
$H_{\mathbf{id}}(j\Omega)$	CTFT of an ideal low-pass filter
$\tilde{a}_n(t)$	Time-varying impulse response of a ZOH
$h_{\mathbf{id}}(t)$	Impulse response of an ideal low-pass filter
$\hat{A}_n(e^{j\omega})$	DTFT of a time-varying ZOH
ω	Discrete-time angular frequency
λ	Spectral parameter of the Farrow structure
Δ_n	Discrete-time time-varying jitter term

Filter Design Algorithm

$\mathbf{g}_n^K$	Postcorrection filter coefficient column vector of length K
$\mathbf{d}_n$	Column vector comprising the desired frequency response for time instant n
$\mathbf{W}$	Modulation Matrix
$\mathbf{M}_n^K$	Design matrix at time instant n
$\mathbf{M}_n^{K-1}$	Lower order design matrix at time instant n
$\left(\mathbf{M}_n^K\right)^\dagger$	Generalized inverse of the filter design problem at time instant n
$\left(\mathbf{M}_n^{K-1}\right)^\dagger$	Lower order generalized inverse of the filter design problem at time instant n

Volterra Systems

$H_n\{.\}$	Volterra system of infinite order
$H_n^{\langle P \rangle}\{.\}$	P^{th}-order truncation of the Volterra system $H_n\{.\}$
$H_{p,n}\{.\}$	p^{th}-order operator of the Volterra system $H_n\{.\}$
$y_p[n]$	Intermediate signal of the p^{th}-order operator

1 Introduction

Linear systems are encountered in various technical areas, e.g., in the field of mixed-signal processing, automatic control and communications theory to characterize components in the signal processing chain [1–5]. Although time-invariant systems are employed in many technical areas for their superior properties, specific applications do require to take time-varying behavior into account, e.g., in wireless communication systems [5, 6]. In this thesis, time-varying systems are investigated whose impact on the signal is undesired, and the compensation of this impact is required to ensure a desired signal quality [7–9]. Correction schemes can be found in different mixed signal applications, e.g., in terms of the equalization of linear time-varying communication channels [10] and the digital enhancement of time-varying analog circuits [11] such as time-interleaved analog-to-digital converters (TIADCs) [9, 12]. In the first application, the equalization of communication channels benefits from the robustness of digital modulation against noise which promoted the success of wireless communication systems [13]. As analog systems lack this inherent robustness, the digital enhancement of analog systems proves to be demanding since even a small mismatch of the corrected system from an ideally corrected system results in a considerable degradation of the performance [14, 15]. This problem is aggravated by an undesired time-varying behavior of the analog system as it reduces the spurious-free dynamic range (SFDR) of the system's output signal [16].

On the system level, the interaction of a time-varying digital or analog system and a dedicated time-varying correction filter can be represented as a cascade of two time-varying systems [8]. In this context, the correction of the undesired system is performed by designing the correction filter, i.e., the cascade system exhibits the desired characteristic, e.g., an all-pass behavior in the frequency band of interest [9]. As the technological progress of analog building blocks proceeds less rapidly than the progress of digital circuitry [11], the digital enhancement of analog systems becomes increasingly attractive. Furthermore, digitally assisted circuits are a promising approach to improve the performance of analog

systems and to follow the trend to replace analog with digital circuitry as typified in the software-defined radio concept [17]. Thus, the correction of undesired linear systems plays an important role in modern communication systems alleviating the performance penalty introduced by non-ideal analog circuits. Moreover, a low complexity implementation of the correction scheme is important for the overall system's power consumption and integration in an existing system.

1.1 Scope of the Work

The objective of this work is to develop low complexity structures which facilitate the correction of time-varying systems. Different figures of merit are used to evaluate these structures with respect to their ubiquitous application, reconstruction performance, implementation complexity and design complexity. The first figure of merit regards the application of the developed structures in mixed-signal scenarios and considers possible constraints which have to be imposed on the time-varying system for a given structure. Additionally, the ability of structures to precorrect and postcorrect a time-varying system is considered. The reconstruction performance of the correction structure refers to the ability to suppress undesired signal components that are induced by the time-varying system. Furthermore, the computational complexity of these structures is considered in terms of their implementation complexity, which is mainly determined by the number of multipliers and adders of the filters comprised in a correction structure. Moreover, the design complexity evaluates the computational effort required to redesign the correction structure with respect to the time-varying behavior of the system that is to be corrected.

Different correction structures and methods for their design and adaption are developed in this thesis and their performance is assessed. To this end, simulation environments are developed in Matlab© to evaluate the aforementioned figures of merit. Filter designs are performed by utilizing the Matlab software CVX for solving convex optimization problems [18] unless stated otherwise. Furthermore, a fixed point implementation framework is developed and utilized to investigate the stability of a proposed recursive filter design algorithm.

1.2 Outline of the Thesis and Main Contributions

System Models In Chapter 2, mathematical representations of time-varying systems are presented. To this end, the concepts of time-varying filters, polynomial impulse response filters, modulator banks, filter banks and the requirements for their equivalence are introduced. These representations provide a foundation for the following

chapters of this thesis. Thus, correction schemes can be derived based on the most suitable framework for a given problem, and the most favorable implementation of the developed correction scheme can be chosen based on its implementation and design complexity.

Correction of Linear Time-varying Systems In Chapter 3, a framework for the design of time-varying filters of finite impulse response (FIR) type is presented. This framework treats both the precorrection and postcorrection scenario and yields the same cost function as the postcorrection method proposed in [19]. The presented correction methods provide a powerful and versatile approach to correct non-periodically time-varying systems by performing a dedicated filter design for each time instant. The filter order and the filter design norm can be adapted for each individual filter design in order to meet the application's requirements. To alleviate the computational burden introduced by the filter design process, a low complexity filter design algorithm for the least-squares norm is derived which can be applied to postcorrect and precorrect linear time-varying systems, respectively. Moreover, the proposed filter design algorithm is analyzed in terms of its stability and computational complexity. To allow a comparison of the presented algorithm, a reference algorithm is presented whose computational complexity is identical to the complexity of the algorithms presented in [9] and [20]. In this comparison, the proposed algorithm proves to be more computationally efficient than the algorithms presented in [9], [20] and [21]. Furthermore, a system model for the precorrection of a digital-to-analog converter employing non-uniform zero-order-hold (ZOH) signals is introduced which complements the postcorrection of analog-to-digital converters in [22]. Moreover, the presented application example extends the precorrection of non-uniformly time-varying ZOH signals from the periodic case as presented in [23] to the non-periodic case. The work that is presented in this chapter is related to the following publications:

- M. Soudan, and C. Vogel, *On the Correction of Linear Time-varying Systems by Means of Time-varying FIR Filters*, Proceedings of the 54th Midwest Symposium on Circuits and Systems, MWSCAS, August 2011 [24].
- M. Soudan, and C. Vogel, *Low Complexity Least Squares Filter Design for the Correction of Linear Time-varying Systems*, Proceedings of the 20th European Conference on Circuit Theory and Design, ECCTD, August 2011 [25].

Correction Structures for Weakly Time-Varying Systems In Chapter 4, structures for the correction of weakly time-varying systems are derived. The utilized structures

are constituted by a cascade of correction stages which gradually reduce the error induced by the weakly time-varying system. Furthermore, a condition for weakly time-varying systems is presented which can be easily evaluated in real-time applications. The presented structures can be applied in weakly time-varying scenarios, where the design complexity of the filter design based correction scheme, as presented in Chapter 3, can not be afforded. The proposed precorrection structure complements postcorrection structures that have been presented in [26–29]. By establishing the commutativity of the precorrection structure and the undesired linear system, results regarding reconstruction performance, conditions for convergence and computational complexity, as presented in [26–29], can directly be applied to the proposed precorrection structure. Additionally, the viability of the proposed structures is demonstrated via the precorrection of non-uniform ZOH signals by an explicit filter design of FIR correction filters. Moreover, a modified Farrow filter structure is developed which can be redesigned by tuning a few time-varying multiplies only. Applying this Farrow filter in the precorrection structure, a further reduction of the design complexity is achieved. The resulting precorrection structure complements the mixed-signal postcorrection scenarios in [30, 31]. The work that is presented in this chapter is related to the following publications:

- M. Soudan, and C. Vogel, *Correction Structures for Linear Weakly Time-Varying Systems*, Transactions on Circuits and Systems, Resubmitted after Minor Revision, 2011.
- M. Soudan, and C. Vogel, *On the Correction of Linear Time-varying Systems by Means of Time-varying FIR Filters*, Proceedings of the 54th Midwest Symposium on Circuits and Systems, MWSCAS, August 2011 [24].

Correction of Weakly Time-Varying Volterra Systems In Chapter 5, the extension of the principles as presented in Chapter 4 to nonlinear systems is explored as an outlook for future research. To this end, the definition of the P^{th}-order inverse is extended to time-varying Volterra systems to provide a reference correction scheme. To alleviate the design complexity of the time-varying P^{th}-order inverse, structures based on fixed point iteration methods are proposed to correct Volterra systems with weakly time-varying linear components. A discussion of continuative research concludes this chapter.

Conclusion In Chapter 6, a summary of the presented work is given.

2

System Models for Time-Varying Systems

2.1 Introduction

Time-varying linear systems can be divided into non-periodically and periodically time-varying systems, where the latter class of systems exhibits periodically recurring behavior [32]. Non-periodically time-varying systems have been applied in various areas, e.g. for the parameter estimation of autoregressive processes, in the equalization of multichannel FIR systems [33, 34] and to characterize analog systems exhibiting time-varying behavior, i.e., TIADCs [1, 16, 35]. Deterministic and statistical mathematical models have been proposed in the literature to represent the behavior of time-varying systems [3–5, 26]. In this chapter, the focus will be on a deterministic description of time-varying FIR filters, polynomial impulse response FIR filters, modulator banks and filter bank representations of M-periodically time-varying systems [19, 26, 36]. Furthermore, the requirements for the equivalence of these representations will be investigated.

2.2 Time-Varying FIR Filter

The output $y[n]$ of a time-varying system can be represented as the convolution of the input signal $x[n]$ with the filter's time-varying finite impulse response $h_n[k]$ at a given time instant n [19], i.e.,

$$y[n] = \sum_{k=-\infty}^{\infty} h_n[k]x[n-k]. \tag{2.1}$$

A block diagram illustrating the signal flow for the time-varying FIR filter is illustrated in Fig. 2.1. In a scenario, where the impulse response $h_n[k]$ recurs with a period M, i.e., $h_n[l] = h_{n+M}[l]$, the respective time-varying filter will be referred to as *M-periodically time-varying filter*.

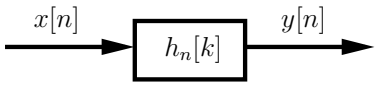


Figure 2.1 Time-varying FIR filter.

A polynomial expansion of the impulse response of a linear time-varying system can be performed in terms of the spectral parameter λ as first proposed in [37]. To this end, each coefficient of the overall time-varying filter structure can be represented in terms of a polynomial in λ, i.e.,

$$\hat{h}[k,\lambda] = \sum_{l=0}^{L} b_l[k]\ \lambda^l, \tag{2.2}$$

where $b_l[k]$ indicate the time-invariant coefficients of the subfilter with index l. Depending on the given application, the coefficients of these $L+1$ subfilters could be either real-valued as for the approximation of variable delay elements as presented in [37] or complex-valued, e.g., for the processing of bandpass signals or in array signal processing applications [38]. In both cases, a time-varying behavior of the filter can be achieved without performing a redesign of the time-invariant subfilters by updating the parameter λ in (2.2) in accordance with a time-varying model parameter. Therefore, for model parameters with periodically recurring values, a periodically time-varying filter can be obtained. The overall Farrow structure is depicted in Fig. 2.2.

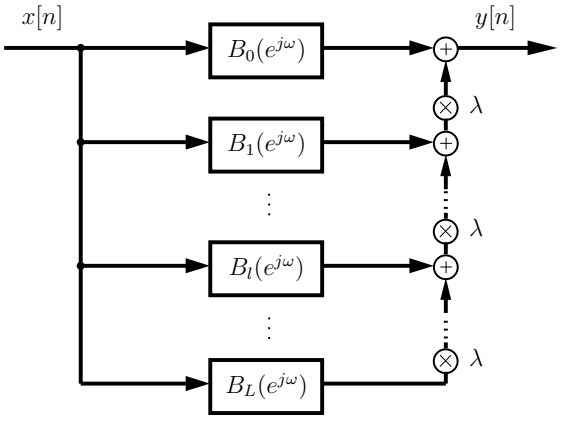


Figure 2.2 Farrow structure comprising $L+1$ time-invariant FIR filters.

2.3 Modulator Banks

A viable form of implementing M-periodically time-varying systems are modulator banks [26]. As the periodically time-varying system's impulse response $h_n[l]$ is periodic in l, it may be represented in terms of the discrete-time Fourier series (DTFS) pair [26]

$$\breve{h}_k[l] = \frac{1}{M} \sum_{m=0}^{M-1} h_m[l] e^{-jkm\frac{2\pi}{M}} \tag{2.3}$$

and

$$h_m[l] = \sum_{k=0}^{M-1} \breve{h}_k[l] e^{jkm\frac{2\pi}{M}}. \tag{2.4}$$

Using (2.4) to rewrite the time-varying filtering operation in (2.1), we obtain

$$y[n] = \sum_{l=-\infty}^{\infty} \sum_{k=0}^{M-1} \left(\breve{h}_k[l] x[n-l] \right) e^{jkn\frac{2\pi}{M}}. \tag{2.5}$$

Taking the discrete-time Fourier transform of (2.5), the frequency response of the time-varying filter output is given as [26]

$$Y(e^{j\omega}) = \sum_{k=0}^{M-1} \breve{H}_k \left(e^{j\left(\omega - k\frac{2\pi}{M}\right)} \right) X \left(e^{j\left(\omega - k\frac{2\pi}{M}\right)} \right), \tag{2.6}$$

where shifted images of the frequency response of the input signal $X(e^{j\omega})$ and the filter $\breve{H}_k(e^{j\omega})$ are multiplied and added to yield the spectrum of the output signal of the modulator bank $Y(e^{j\omega})$. In order to synthesize an equivalent implementation of (2.6), the input signal is processed by the time-invariant filter characterized by $\breve{H}_k(e^{j\omega})$, and the resulting signal is modulated via a complex exponential series $e^{jkn\frac{2\pi}{M}}$. Repeating this processing for all occurring values of k and summing the individual outcomes, a modulator bank implementation as shown in Fig. 2.3 can be derived [26].

2.4 Filter Banks

2.4.1 Preliminary

Filter banks constitute a type of multirate system, where the input signal is passed through a set of time-invariant analysis filters, the sampling rate of the system is altered consecutively, and the resulting signals are processed by a set of time-invariant synthesis filter before they are summed up in the final step. The alteration of the sampling rate can be performed by two basic building blocks of multirate systems. The first building block

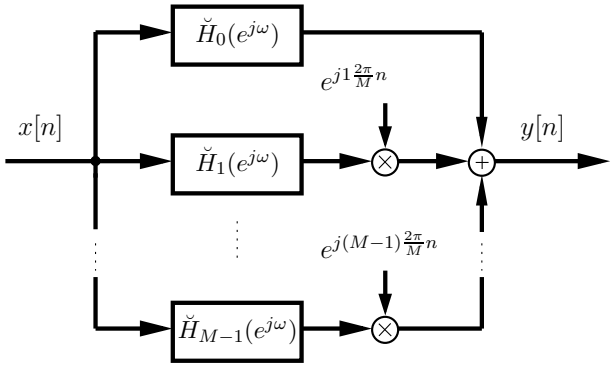


Figure 2.3 Modulator bank model of a time-varying filter employing M time-invariant filters [26].

is a downsampler and utilized to decrease the sampling rate of the system by a factor of Q, i.e., the resulting frequency is $\tilde{\omega}_s = \frac{\omega_s}{Q}$. The second block increases the sampling rate by a factor of P, i.e., the resulting frequency results in $\tilde{\omega}_s = \omega_s P$, and is referred to as an upsampler. A configuration of these two blocks is depicted in Fig. 2.4, where the frequency response of the downsampled signal $v_q[n]$ can be calculated as [39]

$$V_q(e^{j\omega}) = \frac{1}{Q}\sum_{k=0}^{Q-1} U_q(e^{j\left(\frac{\omega-k2\pi}{Q}\right)}). \tag{2.7}$$

The subsequent upsampling block increases the angular frequency by a factor of P, i.e., $W_q(e^{j\omega}) = V_q(e^{j\omega P})$, and the frequency response of the final output signal $w_q[n]$ results in

$$W_q(e^{j\omega}) = \frac{1}{Q}\sum_{k=0}^{Q-1} U_q(e^{j\left(\frac{\omega P}{Q}-\frac{k2\pi}{Q}\right)}). \tag{2.8}$$

If the employed downsampling factor, upsampling factor and number of utilized signal

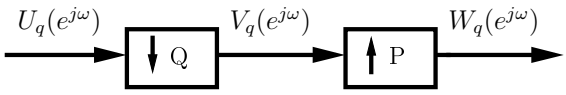


Figure 2.4 Downsampling and upsampling blocks altering the sampling rate of the input signal.

branches in the filter bank have an identical value, the resulting structure is called a maximally decimated filter bank. For a M-channel maximally decimated filter bank, the discrete-time Fourier transform (DTFT) of the output signal of the upsampling block results in [39]

$$W_m(e^{j\omega}) = \frac{1}{M}\sum_{k=0}^{M-1} U_m(e^{j\left(\omega-k\frac{2\pi}{M}\right)}) \tag{2.9}$$

with $0 \leq m \leq M-1$. This type of filter bank is of interest as it allows for the description of M-periodically time-varying filters by means of multirate theory.

2.4.2 Connection to Time-varying Filter

The connection between periodically time-varying filters and filter banks can be found in [19, 39]. The following derivation shows the relationship between modulator banks and filter banks. To this end, the DTFT of the time-varying filter output, as presented in (2.6), can be reformulated based on the M-periodic property of the time-varying system which is valid both in the frequency and in the time domain [26]. We introduce an equivalent frequency domain representation of the DTFS pair in (2.3) and (2.4) by calculating their respective DTFTs which results in

$$\breve{H}_k(e^{j\omega}) = \frac{1}{M}\sum_{m=0}^{M-1} H_m(e^{j\omega})e^{-jkm\frac{2\pi}{M}} \tag{2.10}$$

and

$$H_m(e^{j\omega}) = \sum_{k=0}^{M-1} \breve{H}_k(e^{j\omega})e^{jkm\frac{2\pi}{M}}. \tag{2.11}$$

Rewriting (2.6) using (2.10), we obtain

$$Y(e^{j\omega}) = \frac{1}{M}\sum_{m=0}^{M-1}\sum_{k=0}^{M-1} X(e^{j(\omega-k\frac{2\pi}{M})})H_m(e^{j(\omega-k\frac{2\pi}{M})})e^{-jkm\frac{2\pi}{M}}. \tag{2.12}$$

Expanding (2.12) with the term $e^{j\omega m}e^{-j\omega m}$, the DTFT of the output signal becomes

$$Y(e^{j\omega}) = \frac{1}{M}\sum_{m=0}^{M-1}\sum_{k=0}^{M-1} X(e^{j(\omega-k\frac{2\pi}{M})})H_m(e^{j(\omega-k\frac{2\pi}{M})})e^{j(\omega-k\frac{2\pi}{M})m}\ e^{-j\omega m}. \tag{2.13}$$

The relation in 2.13 can be reformulated, i.e., the output depends on the downsampled and upsampled signals $W_m(e^{j\omega})$ by

$$Y(e^{j\omega}) = \sum_{m=0}^{M-1} W_m(e^{j\omega})e^{-j\omega m}, \tag{2.14}$$

where the output of the upsampling blocks $W_m(e^{j\omega})$ is given as

$$W_m(e^{j\omega}) = \frac{1}{M}\sum_{k=0}^{M-1} X(e^{j(\omega-k\frac{2\pi}{M})})H_m(e^{j(\omega-k\frac{2\pi}{M})})e^{j(\omega-k\frac{2\pi}{M})m}. \tag{2.15}$$

In order to derive a multirate implementation, the relation in (2.15) is compared with (2.9), which results in the identity

$$U_m(e^{j\omega}) = X(e^{j\omega})H_m(e^{j\omega})e^{j\omega m}. \tag{2.16}$$

Inspection of (2.16) reveals that the input signal $u_m[n]$ of each downsampling block needs to be preprocessed by the cascade of the time-invariant filters $H_m(e^{j\omega})$ and $e^{j\omega m}$. The M cascades of these filters constitute the analysis part of the maximally decimated filter bank. The synthesis part of the filter bank is described by the relation in (2.14), where the output of the upsamplers are postprocessed by all-pass filters. A multirate implementation of the overall M-channel maximally decimated filter bank is shown in Fig. 2.5 [19].

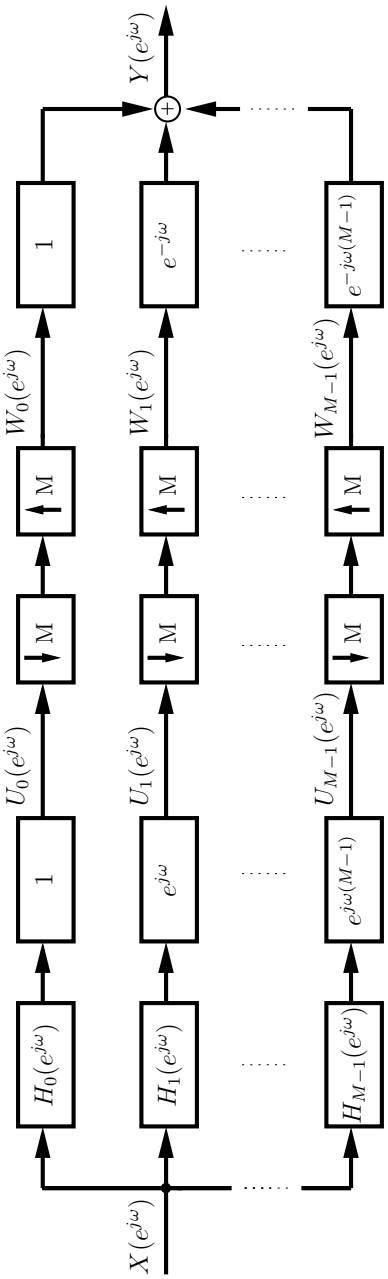


Figure 2.5 Multirate model of an M-periodic time-varying filter.

2.5 Concluding Remarks

The concepts of time-varying filters, polynomial impulse response filters, modulator banks, filter banks and the requirements for their equivalence have been presented in this chapter. These representations provide the means for implementing a periodically time-varying system choosing the most favorable form of representation in terms of its design complexity in a given application. Different form of implementation could also be employed in order to reduce the implementation complexity of the chosen representation, e.g., the Farrow structure implementation employing signed power-of-two coefficients [40] or the polyphase implementation of filter banks [39].

3

Correction of Linear Time-Varying Systems

3.1 Introduction

Linear systems can be classified in time-invariant and time-varying systems, where the latter class offers a more general representation. Time-varying systems can be further divided into non-periodically and periodically time-varying systems, where the second type of system can be used to characterize the recurring behavior of physical devices [1, 41]. If the impact of a linear system on the signal is undesired, an additional linear correction filter may be introduced in order to precorrect or postcorrect the undesired system's output. In this case, two types of complexity can be identified. The *implementation complexity* is determined by the computational complexity of the correction filter, whereas the *design complexity* refers to the computational effort required to compute the coefficients of the correction filter.

A means of performing the correction of linear systems are adaptive filters [42]. In this scheme, a desired signal is required to determine the error of the adaptation process [7]. This error signal is in turn processed by an algorithm which adapts the coefficients of the correction filter. Various algorithms with different design complexities and convergence behaviors are available to perform the adaptation of the correction filter, e.g., the least-squares or the recursive least-squares algorithm [42]. An adaptive system is illustrated in Fig. 3.1, where the adaptive correction filter is indicated by the equalizer block which postcorrects the undesired linear system.

In the absence of a desired signal, a different scheme can be employed which is based on the explicit design of the correction filter as illustrated in Fig. 3.2 for the postcorrection case. In this filter design based correction scheme, a dedicated functional block calculates the coefficients of the correction filter by performing a filter design process. This functional block computes the filter coefficients based on a characterization of the undesired linear system. The respective characteristics of the undesired linear system could be its impulse

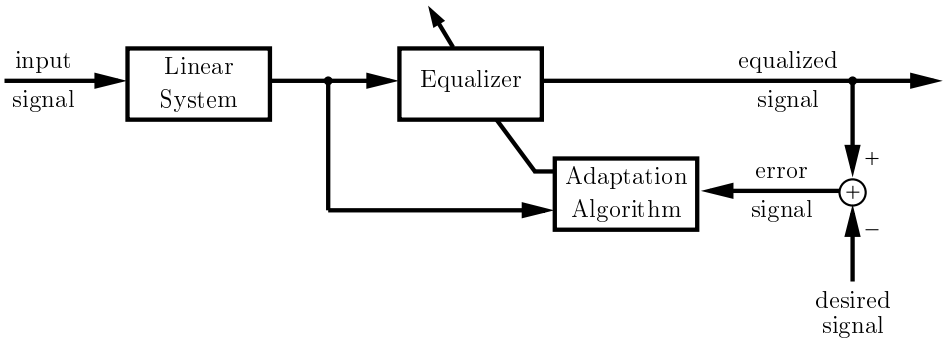


Figure 3.1 Adaptive equalizer performing a postcorrection of a linear system [42].

response, DTFT or model parameters, e.g., the cut-off frequency of a low-pass filter. The characteristics of the undesired system are either provided by an additional functional block performing a blind identification [43] or are simply known a-priori, e.g., via an offline calibration process [21]. For time-invariant systems, various techniques exist to calculate the correction filter as the inverse filter of the undesired system, either exactly, e.g., for minimum-phase systems [44], or via an approximation according to a specified error criterion [9, 45, 46]. Schemes for the correction of periodically time-varying systems have been proposed in [47] and [36] which alleviate the impact of non-uniform sampling employing multirate theory. A structure for precorrecting 2-periodically non-uniform ZOH signals of digital-to-analog converters (DACs) has been proposed in [48], and the digital compensation of in-band spurious tones was presented in [23] for the M-periodic case.

In [9], a method has been presented which can be used for the design of FIR correction filters postcorrecting non-periodically time-varying systems. This is achieved by performing a dedicated filter design of the correction filter for each time instant according to a specified error norm.

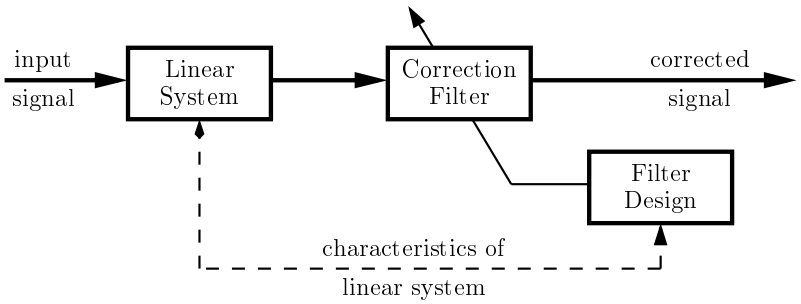


Figure 3.2 Correction scheme performing a postcorrection of a linear system based on an explicit filter design process.

3.1.1 Problem Statement

The filter design based correction scheme exhibits a high design complexity and generates a high computational load since matrix inversions have to be performed or optimization problems need to be solved in the filter design process [9, 21, 45]. For time-invariant systems and periodically time-varying systems, this computational load might be acceptable as the filter design can be performed offline. The filters compensating periodically time-varying systems can be updated during operation employing sets of offline calculated coefficients [21, 49]. In non-periodically time-varying scenarios, e.g., scenarios where the linear system continuously changes its characteristics during operation, the correction filter has to be continuously redesigned to meet the desired compensation performance. A method for the design of correction filters has been presented in [22] for the postcorrection scenario only. An example of how the performance of a time-varying system, which is enhanced by an offline designed correction system, may deteriorate due to drift can be found in [21].

3.1.2 Contributions

In this chapter, methods for designing time-varying FIR filters are presented that can precorrect or postcorrect linear non-periodically time-varying systems. Moreover, a low complexity algorithm for the least-squares design of these time-varying FIR filters is proposed that can be applied to the precorrection and postcorrection scenario, respectively. The contribution are listed in more detail as

a) Two methods for the design of time-varying FIR correction filters are presented. The first proposed precorrection method can be applied to enhance the performance of non-periodically time-varying systems. The second presented postcorrection method complements the precorrection case and provides the same cost function as given in [19], however, its derivation is different and presented for the sake of completeness.

b) An algorithm for the least-squares design of a time-varying correction filter is proposed. Employing this algorithm, each coefficient vector of the time-varying filter of length K can be obtained without resorting to computationally expensive matrix operations in the filter design which would result in a complexity of $\mathcal{O}(RK^2)$, with $R \geq K$. Instead, the filter design is determined by utilizing results from the previous filter design which achieves a complexity of $\mathcal{O}(RK)$ and significantly reduces the computational complexity of existing algorithms, i.e., [9, 20, 21].

c) A continuous-time system model of a DAC employing non-uniform ZOH signals is introduced. To facilitate the digital precorrection, a discrete-time system model of the ZOH is devised, and a time-varying filter representation is introduced. The presented

scheme complements the postcorrection of analog-to-digital converters (ADCs) presented in [22]. Furthermore, the presented application example extends the precorrection of non-uniformly time-varying ZOH signals from the periodic case as presented in [23] to the non-periodic case.

3.2 Precorrection

In this section, a method for the design of a correction filter $h_n[k]$ preprocessing an undesired time-varying system $g_n[k]$, as illustrated in Fig. 3.3, is presented. To this end, the output of the correction filter $y[n]$ is described as the convolution of the input signal $x[n]$ with its time-varying impulse response $h_n[k]$, i.e.

$$y[n] = \sum_{k=-\infty}^{\infty} h_n[k]x[n-k]. \tag{3.1}$$

Consecutively, the precorrected signal $y[n]$ is processed by the undesired time-varying system $g_n[k]$ resulting in

$$\hat{x}[n] = \sum_{k=-\infty}^{\infty} g_n[k]y[n-k]. \tag{3.2}$$

Rewriting (3.2) with (3.1), the reconstructed output of the overall system $\hat{x}[n]$ may be represented as

$$\hat{x}[n] = \sum_{k=-\infty}^{\infty} f_n[k]x[n-k] \tag{3.3}$$

where $f_n[k]$ is the impulse response of the two cascaded time-varying systems given as [7]

$$f_n[k] = \sum_{l=-\infty}^{\infty} g_n[l]h_{n-l}[k-l]. \tag{3.4}$$

A block diagram illustrating the signal flow for the cascade of the two time-varying systems is depicted in Fig. 3.3. Calculating the DTFT of (3.4) results in a representation of the

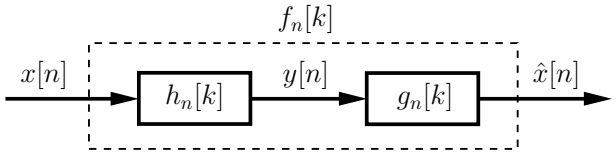


Figure 3.3 Block diagram illustrating the signal flow for a cascade of two time-varying systems.

cascaded system that can be used for the design of a time-varying postcorrection filter

as has been shown in [19]. For the precorrection case, a more complicated derivation is required. The DTFT of the cascaded system has to be expressed as a function of the impulse response of the precorrection filter and the DTFT of the subsequent time-varying system. In order to allow for the calculation of the DTFT of the cascaded system and the second time-varying system, the index controlling the time-varying behavior of the precorrection filter has to depend only on the discrete time n. To ensure this property prior to the calculation of the Fourier transform, the impulse response of the cascaded system in (3.4) has to be rewritten using the transform pair

$$g_n[k] = \hat{g}_{n-k}[k] \tag{3.5}$$

$$\hat{g}_n[k] = g_{n+k}[k] \tag{3.6}$$

as

$$f_n[k] = \sum_{l=-\infty}^{\infty} \hat{g}_{n-l}[l] h_{n-l}[k-l]. \tag{3.7}$$

By substituting the indexes $p = k - l$ and $l = k - p$, we get

$$f_n[k] = \sum_{p=-\infty}^{\infty} h_{n-k+p}[p] \hat{g}_{n-k+p}[k-p]. \tag{3.8}$$

Shifting the time index of $f_n[k]$ to $f_{n+k}[k]$ yields

$$f_{n+k}[k] = \sum_{p=-\infty}^{\infty} h_{n+p}[p] \hat{g}_{n+p}[k-p] \tag{3.9}$$

and applying (3.6) to the terms $f_{n+k}[k]$ and $h_{n+p}[p]$ in (3.9), results in

$$\hat{f}_n[k] = \sum_{p=-\infty}^{\infty} \hat{h}_n[p] \hat{g}_{n+p}[k-p]. \tag{3.10}$$

By calculating the DTFT of (3.10), we obtain the frequency response of the cascaded system for each time instant n as

$$\hat{F}_n(e^{j\omega}) = \sum_{p=-\infty}^{\infty} \hat{h}_n[p] \hat{G}_{n+p}(e^{j\omega}) e^{-j\omega p}. \tag{3.11}$$

The relationship in (3.11) exhibits favorable properties. First, to characterize $\hat{F}_n(e^{j\omega})$ at the time instant n, the impulse response $\hat{h}_n[p]$ can be treated as being time-invariant which simplifies the design of the correction filter. Second, for the design of an FIR filter with K coefficients, $K-1$ states of $\hat{G}_n(e^{j\omega})$ are sufficient to determine $\hat{F}_n(e^{j\omega})$. In order

to yield a cost function, an error function is defined as the deviation of $\hat{F}_n(e^{j\omega})$ from a desired frequency response $\hat{D}_n(e^{j\omega})$ as

$$\hat{E}_n(e^{j\omega}) = \hat{F}_n(e^{j\omega}) - \hat{D}_n(e^{j\omega}) \tag{3.12}$$

which is in turn used to formulate a filter design problem similar as in the time-invariant case [19, 45] as

$$\min ||\hat{E}_n(e^{j\omega})||_{n_e} \quad \text{for } \omega \in \omega_D \tag{3.13}$$

for a design domain ω_D and error norm n_e. By solving the optimization problem in (3.13) [45], an FIR filter design of $\hat{h}_n[k]$ for the time instant n is performed. Applying the relation in (3.5) to the resulting design, we obtain $h_n[k]$.

3.3 Postcorrection

In order to yield a cost function for the design of the postcorrection filter $g_n[l]$, we calculate the DTFT of (3.4) as [19]

$$F_n(e^{j\omega}) = \sum_{l=-\infty}^{\infty} g_n[l] H_{n-l}(e^{j\omega}) e^{-j\omega l}. \tag{3.14}$$

which complements the filter design for the precorrection scenario in (3.11). As in the precorrection case, (3.14) does not rely on a periodically time-varying behavior of the undesired time-varying system. Furthermore, $F_n(e^{j\omega})$ can be characterized for each time instant n by only resorting to present or past states of $H_n(e^{j\omega})$. Analogously to the precorrection case, the cost function is given as [19]

$$E_n(e^{j\omega}) = F_n(e^{j\omega}) - D_n(e^{j\omega}), \tag{3.15}$$

and the filter design is obtained by minimizing

$$\min ||E_n(e^{j\omega})||_{n_e} \quad \text{for } \omega \in \omega_D \tag{3.16}$$

for a design domain ω_D and error norm n_e. Employing the designed postcorrection filter in (3.14), the cascade of the undesired time-varying system and of the correction filter approximates $D_n(e^{j\omega})$. If it is desired to compensate the impact of $h_n[l]$, then $D_n(e^{j\omega})$ can be chosen time-invariant as an all-pass filter with a group delay that equals the sum of the delay of the undesired time-varying system and of the correction filter. However, the presented relations are not limited to this time-invariant case as the designer may select $D_n(e^{j\omega})$ to be time-varying and thus an arbitrary time-varying behavior of the cascaded system could be generated.

3.4 Low Complexity Least-Squares Filter Design

3.4.1 Problem Statement

Before introducing a new low complexity filter design algorithm, an algorithm for the least-squares design of a time-varying correction filter $g_n[k]$ is presented. This algorithm provides an alternative derivation of the relation presented in [9] for the least-squares error norm. The obtained algorithm will serve as a means of comparison with the proposed low complexity filter design algorithm. This algorithm will also be referred to as the *reference algorithm*. Without loss of generality only the postcorrection case is considered in the following. The application of the presented theory to the precorrection case is straightforward and changes which have to be performed to the presented framework will be outlined in the mixed-signal application example in Section 3.5.

The discrete-time Fourier transform of (3.4) can be represented as

$$F_n(e^{j\omega}) = \sum_{l=0}^{K-1} M_{n,l}(e^{j\omega})\ g_n[l] \tag{3.17}$$

with

$$M_{n,l}(e^{j\omega}) = e^{-j\omega l} H_{n-l}(e^{j\omega}) \tag{3.18}$$

which provides an alternative formulation of the design equation given in (3.14). In order to allow for a least-squares design of the correction filter, the cost function [46]

$$J(g_n[0], \ldots, g_n[K-1]) = \int_{\omega \in \omega_{\mathbf{D}}} |E_n(e^{j\omega})|^2 d\omega \tag{3.19}$$

is introduced with $\omega_{\mathbf{D}}$ specifying the design domain of the filter design. In order to determine the filter coefficient vector $\mathbf{g}_n^K = [g_n[0], g_n[1], \ldots, g_n[K-1]]^T$ with K coefficients, the cost function in (3.19) is evaluated on a discrete set of R equally spaced frequency points. This set is represented by the complex column vector $\mathbf{w} = [e^{-j\omega_0}, e^{-j\omega_1}, \ldots, e^{-j\omega_{R-1}}]^T$, which spans the design domain of the filter design in (3.19). As a design rule, the parameter R, which determines the number of elements in $\mathbf{w}$, can be obtained as $R = 15(K-1)$ [45]. The cost function of the least-squares filter design can be expressed as the matrix equation [46]

$$J(\mathbf{g}_n^K) = \left(\left(\mathbf{g}_n^K\right)^* \left(\mathbf{M}_n^K\right)^* - (\mathbf{d}_n)^* \right) \left(\mathbf{M}_n^K \mathbf{g}_n^K - \mathbf{d}_n \right) \tag{3.20}$$

where the operator $(.)^*$ indicates the conjugate transpose, and the column vector $\mathbf{d}_n$ is constituted by the desired discretized frequency response

$$\mathbf{d}_n = \begin{bmatrix} D_n(e^{j\omega_0}) \\ D_n(e^{j\omega_1}) \\ \vdots \\ D_n(e^{j\omega_{R-1}}) \end{bmatrix}. \tag{3.21}$$

Furthermore, the time-varying R-by-K matrix $\mathbf{M}_n^K$ in (3.20) is composed of K column vectors, i.e.,

$$\mathbf{M}_n^K = [\mathbf{m}_{n,0},\ \mathbf{m}_{n,1},\ \ldots\ ,\ \mathbf{m}_{n,K-1}]. \tag{3.22}$$

The column vectors constituting $\mathbf{M}_n^K$ in (3.22) are given as

$$\mathbf{m}_{n,l} = \mathbf{W}^l\ \mathbf{h}_{n-l} \tag{3.23}$$

with

$$\mathbf{h}_{n-l} = \begin{bmatrix} H_{n-l}(e^{j\omega_0}) \\ H_{n-l}(e^{j\omega_1}) \\ \vdots \\ H_{n-l}(e^{j\omega_{R-1}}) \end{bmatrix} \tag{3.24}$$

and

$$\mathbf{W}^l = \begin{bmatrix} e^{-j\omega_0 l} & 0 & \ldots & 0 \\ 0 & e^{-j\omega_1 l} & \ldots & 0 \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & e^{-j\omega_{R-1} l} \end{bmatrix}. \tag{3.25}$$

As an abbreviation, the matrix

$$\mathbf{W} = \mathbf{W}^1 \tag{3.26}$$

is introduced and referred to as the *modulation matrix*. Furthermore, the matrix $\mathbf{M}_n^K$ will be referred to as the *design matrix* at time instant n which is in turn composed of K *observation vectors*. The matrix $\mathbf{M}_n^{K-1}$ will be referred to as the *design matrix* at time instant n which is in turn composed of K *observation vectors*. The optimum solution in the least-squares sense results in the filter design [45]

$$\mathbf{g}_n^K = \left(\mathbf{M}_n^K\right)^\dagger \mathbf{d}_n \tag{3.27}$$

where the Moore Penrose or *generalized inverse* is indicated by [45]

$$\left(\mathbf{M}_n^K\right)^\dagger = \mathbf{P}_n^K \left(\mathbf{M}_n^K\right)^* \tag{3.28}$$

with

$$\mathbf{P}_n^K = \left(\left(\mathbf{M}_n^K\right)^* \mathbf{M}_n^K\right)^{-1}. \tag{3.29}$$

3.4.2 Shared Information in Subsequent Filter Design Problems

Basic Idea

The filter design equations of the previous Section 3.4.1 solve a dedicated filter design problem for each time instant n independent of previous filter design problems performed. This approach disregards the fact that only parts of the design matrix contain new information while the remainder was already part of the previous design problem at time instant $n-1$ in a modified form.

In the following, the principle is described that facilitates a low complexity least-squares filter design. To this end, the dependencies of subsequent filter design problems are illustrated in Fig. 3.4.

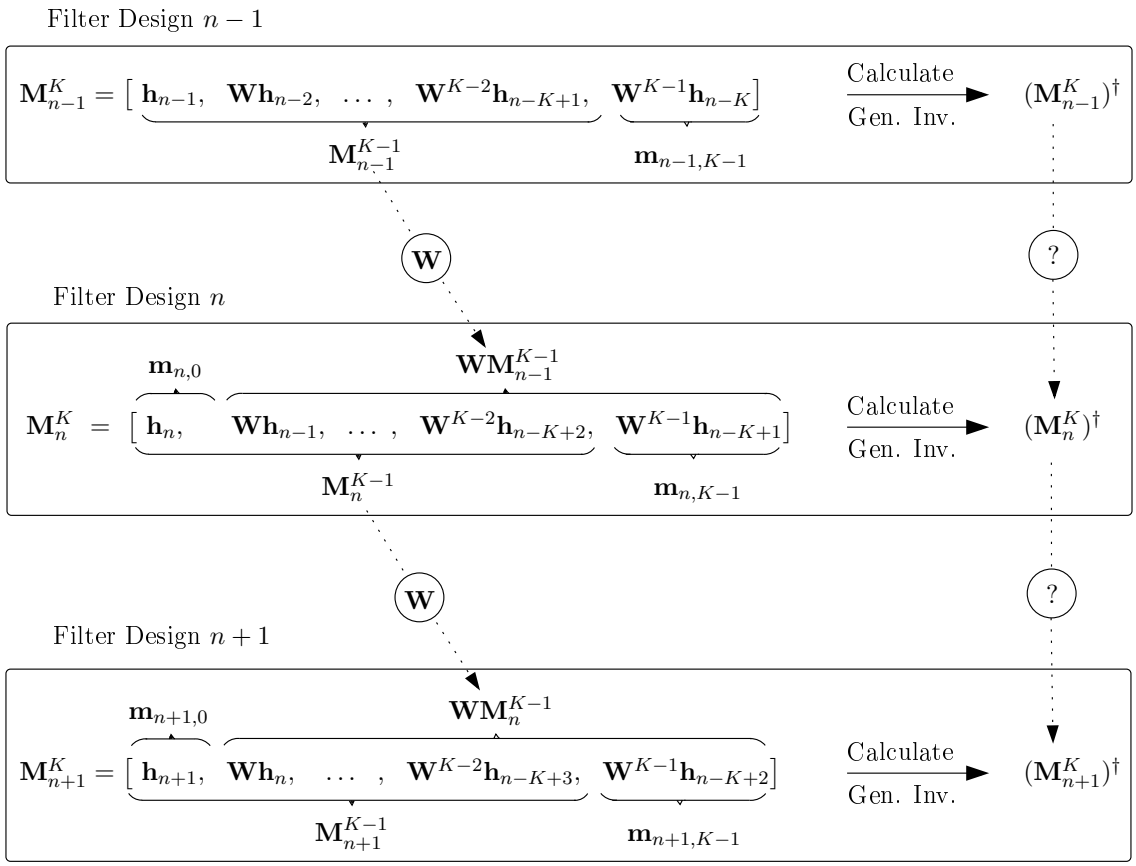


Figure 3.4 Illustration of subsequent filter design problems and the composition of their respective design matrices.

The topmost box represents a filter design at time instant $n-1$ in this illustration. The design matrix associated with this filter design problem is $\mathbf{M}_{n-1}^{K}$ of the form as described

in (3.22). Employing the relation (3.28), the generalized inverse can be calculated from this design matrix which is depicted by the horizontal arrow. To show the relationship of the design matrix $\mathbf{M}_{n-1}^{K}$ with the design matrix of the subsequent filter design problem at time instant n, the matrix $\mathbf{M}_{n-1}^{K}$ is partitioned in the *lower order design matrix* $\mathbf{M}_{n-1}^{K-1}$, aggregating the first $K-1$ column vectors, and the last column vector $\mathbf{m}_{n-1,K-1}$, i.e.,

$$\mathbf{M}_{n-1}^{K} = \left[\mathbf{M}_{n-1}^{K-1},\ \mathbf{m}_{n-1,K-1}\right]. \tag{3.30}$$

The subsequent filter design problem at time instant n with design matrix $\mathbf{M}_{n}^{K}$ is represented by the center box. The first column of $\mathbf{M}_{n}^{K}$ represents the new observation vector $\mathbf{m}_{n,0}$ and the last $K-1$ columns are identical to the matrix product $\mathbf{W}\mathbf{M}_{n-1}^{K-1}$, which is the modulated lower order design matrix of the previous filter design problem. Since the modulated previous design matrix of lower order is part of the current design matrix, it seems a reasonable assumption that the current generalized inverse may also be described by a modified version of the previous generalized inverse.

Update of the Generalized Inverse

To formulate an update relation of the generalized inverse, two tasks need to be performed. An additional task is the actual filter design that is carried out by employing the updated generalized inverse. The required steps for the update of the generalized inverse are illustrated in Fig. 3.5. First, the lower order generalized inverse of the previous design problem $\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger}$ is computed based on the provided generalized inverse $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$, which is indicated by the block "Backward Projection" in Fig. 3.5. Second, in the resulting matrix $\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger}$, the new observation vector $\mathbf{m}_{n,0}$ and the altered structure of the current design needs to be taken into account. This task is indicated by the block "Forward Projection" in the illustration. Consecutively, the filter design can be performed using the updated generalized inverse $\left(\mathbf{M}_{n}^{K}\right)^{\dagger}$ and the desired vector $\mathbf{d}_n$ which results in the coefficient vector $\mathbf{g}_{n}^{K}$. An adaptation of the number of rows R of the design matrix could also be employed to update the generalized inverse in terms of the design domain of the filter design problem by using the recursive least-squares algorithm [42]. However, the following derivations are based on a fixed design domain to promote a clear representation.

3.4.3 Order Recursive Least-Squares Filter Design

The computationally efficient reduction of the order and the update of the generalized inverse according to the current design problem will be investigated in this section. The objective is to obtain the filter design for the time instant n without performing the

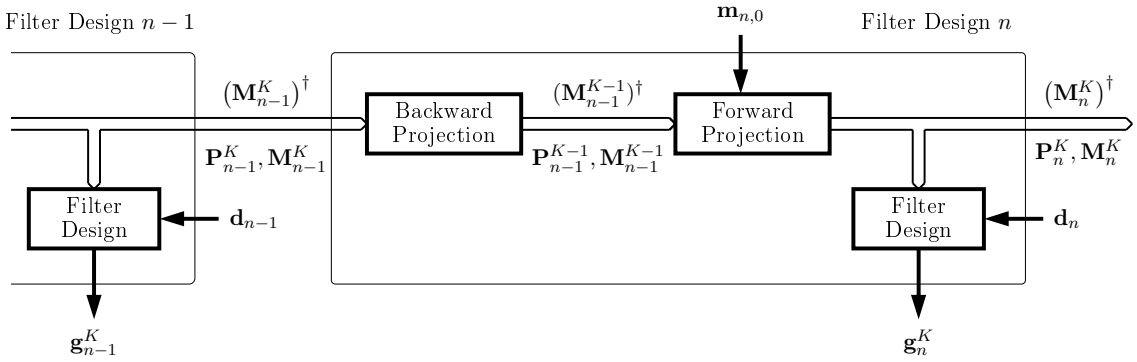


Figure 3.5 Steps for the update of the generalized inverse and the consecutive filter design.

computationally intensive calculation of the generalized inverse $\left(\mathbf{M}_n^K\right)^\dagger$ as presented in (3.28) and (3.29). To this end, the lower order generalized inverse $\left(\mathbf{M}_{n-1}^{K-1}\right)^\dagger$ is computed from the matrix $\left(\mathbf{M}_{n-1}^K\right)^\dagger$, which has been calculated in the previous filter design at time instant $n-1$. Consecutively, it is shown how the new observation vector in the design matrix and the knowledge of $\left(\mathbf{M}_{n-1}^{K-1}\right)^\dagger$ can be used to calculate $\left(\mathbf{M}_n^K\right)^\dagger$ by means of an order recursive update relation. Having determined $\left(\mathbf{M}_n^K\right)^\dagger$ the current filter design problem can be solved by employing the relation in (3.27).

Backward Projection

In order to find a relationship between the a-priori known matrix $\left(\mathbf{M}_{n-1}^K\right)^\dagger$ and the matrix $\left(\mathbf{M}_{n-1}^{K-1}\right)^\dagger$, a backward projection of the filter design problem with K coefficients is performed. The presented framework is in certain aspects similar to the theory presented for joint process estimation [50]. However, in joint estimation theory, the filter coefficients are adapted employing the input and desired signal, whereas the presented algorithm relates two time-varying systems by employing a desired characteristic of the cascaded system. Furthermore, the design matrix changes its structure as a new observation vector becomes available and the old observation vector needs to be discarded which is a consequence of the time-varying nature of the design problem, cf., (3.17) and (3.18). Moreover, the devised algorithm does not only update the quadratic matrix $\mathbf{P}_n^K$ in (3.29) but recursively updates the overall generalized inverse, as presented in (3.28) and (3.29). This is a prerequisite for a computationally efficient filter design algorithm as to obtain the filter design by using $\mathbf{P}_n^K$, an additional matrix multiplication of $\left(\mathbf{M}_n^K\right)^*$ with the updated quadratic

matrix would be required, cf., (3.28). This approach would diminish the computational gain achieved by updating the generalized inverse directly as is proposed in this section.

In order to determine the lower order equivalent of the generalized inverse $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$, this matrix is partitioned in the upper matrix $\mathbf{U}$ and the row vector $\mathbf{v}^{*}$, i.e.,

$$\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger} = \begin{bmatrix} \mathbf{U}_{\langle K-1\times R\rangle} \\ (\mathbf{v}^{*})_{\langle 1\times R\rangle} \end{bmatrix} \tag{3.31}$$

where the subscript operator $\langle . \rangle$ indicates the dimensions of the respective partition. To promote a clear representation, the subscript notation is not going to be used for further calculations involving the matrix partitions.

The lower order generalized inverse can be determined as the $K-1$-by-R matrix (see Appendix (A.1)-(A.12))

$$\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger} = \mathbf{U} + \mathbf{w}_{\mathrm{b}}\mathbf{v}^{*}. \tag{3.32}$$

The backward projection vector $\mathbf{w}_{\mathrm{b}}$ in (3.32) is of length $K-1$ and derived by partitioning the a-priori known matrix

$$\mathbf{P}_{n-1}^{K} = \begin{bmatrix} \mathbf{S}_{\langle K-1\times K-1\rangle}, & \mathbf{t}_{\langle K-1\times 1\rangle} \\ \\ mt^{*}_{\langle 1\times K-1\rangle}, & c_{\langle 1\times 1\rangle} \end{bmatrix} \tag{3.33}$$

where $\mathbf{S}$ represents a real-valued $K-1$-by-$K-1$ matrix, $\mathbf{t}^{*}$ a real-valued row vector of length $K-1$ and c a real scalar value. The backward projection vector can be determined as

$$\mathbf{w}_{\mathrm{b}} = -\frac{\mathbf{t}}{c} \tag{3.34}$$

based on the partitions of $\mathbf{P}_{n-1}^{K}$. The lower order equivalent of $\mathbf{P}_{n-1}^{K}$ can be calculated as (see Appendix (A.1)-(A.8))

$$\mathbf{P}_{n-1}^{K-1} = \mathbf{S} - \mathbf{w}_{\mathrm{b}}\mathbf{t}^{*}. \tag{3.35}$$

Forward Projection

Having established the lower order generalized inverse $\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger}$, we introduce the formulation for the current design matrix

$$\mathbf{M}_{n}^{K} = [\ \underbrace{\mathbf{W}\,\hat{\mathbf{m}}_{n,0}}_{\mathbf{m}_{n,0}},\ \mathbf{W}\,\mathbf{M}_{n-1}^{K-1}\ ] \tag{3.36}$$

with

$$\hat{\mathbf{m}}_{n,0} = \mathbf{W}^{*}\mathbf{m}_{n,0}. \tag{3.37}$$

It is important to note that the first column vector in (3.36) is identical to $\mathbf{m}_{n,0}$ since $\mathbf{W}$ is orthogonal, and thus the matrix product $\mathbf{W}\mathbf{W}^*$ yields the identity matrix.

In order to relate the new observation vector $\hat{\mathbf{m}}_{n,0}$ with the generalized inverse $\left(\mathbf{M}_{n-1}^{K-1}\right)^\dagger$, a forward projection is performed, and to this end, we introduce the forward projection vector [51]

$$\mathbf{w}_\mathrm{f} = \left(\mathbf{M}_{n-1}^{K-1}\right)^\dagger \hat{\mathbf{m}}_{n,0}, \tag{3.38}$$

the forward projection error

$$\mathbf{f} = \hat{\mathbf{m}}_{n,0} - \mathbf{M}_{n-1}^{K-1}\mathbf{w}_\mathrm{f}, \tag{3.39}$$

and the scaling factor

$$\epsilon_\mathrm{f} = \left(\hat{\mathbf{m}}_{n,0}\right)^* \ \mathbf{f} \tag{3.40}$$

which determine the generalized inverse of the current design matrix $\mathbf{M}_n^K$ as

$$\left(\mathbf{M}_n^K\right)^\dagger = \begin{bmatrix} \dfrac{\mathbf{f}^*}{\epsilon_\mathrm{f}} \\ \left(\mathbf{M}_{n-1}^{K-1}\right)^\dagger - \dfrac{\mathbf{w}_\mathrm{f}\,\mathbf{f}^*}{\epsilon_\mathrm{f}} \end{bmatrix} \mathbf{W}^*. \tag{3.41}$$

Using (3.41), the optimum filter design of $\mathbf{g}_n^K$ in the least-squares sense for the time instant n can be calculated by (3.27). In order to allow for an efficient computation of $\mathbf{w}_\mathrm{b}$ during the next filter design process, the matrix $\mathbf{P}_n^K$ is calculated as

$$\mathbf{P}_n^K = \begin{bmatrix} \dfrac{1}{\epsilon_\mathrm{f}}, & -\dfrac{\mathbf{w}_\mathrm{f}^*}{\epsilon_\mathrm{f}} \\ -\dfrac{\mathbf{w}_\mathrm{f}}{\epsilon_\mathrm{f}}, & \mathbf{P}_{n-1}^{K-1} + \dfrac{\mathbf{w}_\mathrm{f}\mathbf{w}_\mathrm{f}^*}{\epsilon_\mathrm{f}} \end{bmatrix}, \tag{3.42}$$

and the backward projection vector $\mathbf{w}_\mathrm{b}$ required for the subsequent filter design problem can be obtained using (3.33) and (3.34). The relations in (3.38)-(3.41) and (3.27) suffice to determine the filter design of $\mathbf{g}_n^K$ for the time instant n. These relations rely on the a-priori known matrices $\left(\mathbf{M}_{n-1}^K\right)^\dagger$ and $\mathbf{P}_{n-1}^K$ that were calculated while performing the previous filter design.

Complexity Comparison

The computation of the generalized inverse, as described by (3.27)-(3.29), provides an algorithm for a filter design in the least-squares sense. The complexity of this algorithm proves to be identical to the complexity of the filter design methodology presented in [9, 20]. In [20], the authors proved the superior efficiency of this least-squares algorithm compared to the method presented in [21]. For the computation of the filter coefficient

in (3.27), Gaussian elimination is used since the linear system of equations can be solved without explicitly calculating the inverse in (3.29) [52]. The resulting algorithm is used as a reference to compare the performance of the proposed order recursive least-squares algorithm for the design of time-varying FIR filter calculation of (3.27)-(3.29).

To this end, we introduce the column vector

$$\mathbf{r}_n = \left(\mathbf{M}_n^K\right)^* \mathbf{d}_n \tag{3.43}$$

and formulate the augmented matrix

$$\left[\, \left(\mathbf{M}_n^K\right)^* \mathbf{M}_n^K,\ \mathbf{r}_n \,\right]. \tag{3.44}$$

Applying an elimination method to (3.44) results in the reduced row echelon form given as

$$\left[\, \mathbf{I},\ \ \mathbf{g}_n^K \,\right] \tag{3.45}$$

where $\mathbf{I}$ indicates the K-by-K identity matrix, and the last column of (3.45) provides the desired solution vector after the elimination process. Employing this formulation to solve the filter design problem, the matrix multiplication in (3.29) is saved [52].

The complexity of the resulting filter design is illustrated in Table 3.1. It can be noted that the computationally most expensive operation for the least-squares filter design algorithm is the inversion of the matrix product $\left(\mathbf{M}_n^K\right)^* \mathbf{M}_n^K$, where the matrix multiplication has a complexity of $\mathcal{O}\left(RK^2\right)$ [53], with $K < R$, and the inversion of the resulting K-by-K matrix has a complexity of $\mathcal{O}\left(K^3\right)$, when applying the Gaussian elimination.

The computational complexity of the reference algorithm and the proposed algorithm are illustrated in Table 3.1 and 3.2 in terms of the $\mathcal{O}$-notation and of the required multiplications and additions.

Comparing the proposed order recursive least-squares filter design algorithm with the matrix inversion based least-squares filter design, a complexity of $\mathcal{O}\left(RK\right)$ for the proposed algorithm can be observed which proves to be one order of magnitude less than the complexity $\mathcal{O}\left(RK^2\right)$ of the reference algorithm.

A comparison of the number of multiplications and additions required by each algorithm for a given filter length is depicted in Fig. 3.6. For very short filter lengths of up to four coefficients, the proposed algorithm requires more multiplications but still less additions than the least-squares algorithm. The proposed algorithm offers reduced complexity both in terms of multiplications and additions for filter designs with more than four coefficients. The ratio of the required multiplication and additions of least-squares and order recursive least-squares based algorithm, respectively, is depicted in Fig. 3.7, where a ratio of one represents identical complexity (dashed line in Fig. 3.7). For any ratio exceeding one, the

Table 3.1 Summary and computational complexity of the least-squares filter design algorithm.

Least-Squares Design	Complexity	# Mult.	# Add.
1) $\mathbf{M}_n^K = [\mathbf{m}_{n,0},\ \mathbf{m}_{n,1},\ \ldots\ ,\mathbf{m}_{n,K-1}]^\dagger$	*n.a.*		
2) $\mathbf{r}_n = \left(\mathbf{M}_n^K\right)^* \mathbf{d}_n$	$\mathcal{O}(RK)$	$2RK$	$2(R-1)K$
3) $\left[\left(\mathbf{M}_n^K\right)^* \mathbf{M}_n^K,\ \mathbf{r}_n\right]$	$\mathcal{O}\left(RK^2\right)$	$2RK^2$	$2(R-1)K^2$
4) $\left[\mathbf{I},\ \ \mathbf{g}_n^K\right]$	$\mathcal{O}\left(K^3\right)$ [a]	$\frac{K^3}{3} + K^2 - \frac{K}{3}$	$\frac{K^3}{3} + \frac{K^2}{2} - \frac{5K}{6}$
Overall Complexity	$\mathcal{O}\left(RK^2\right)$	$\frac{1}{3}K^3 + (2R+1)\,K^2$ $+\left(2R - \frac{1}{3}\right)K$	$\frac{1}{3}K^3 + \left(2R - \frac{3}{2}\right)K^2$ $+\left(2R - \frac{17}{6}\right)K$

[a] Matrix inversion via Gaussian elimination

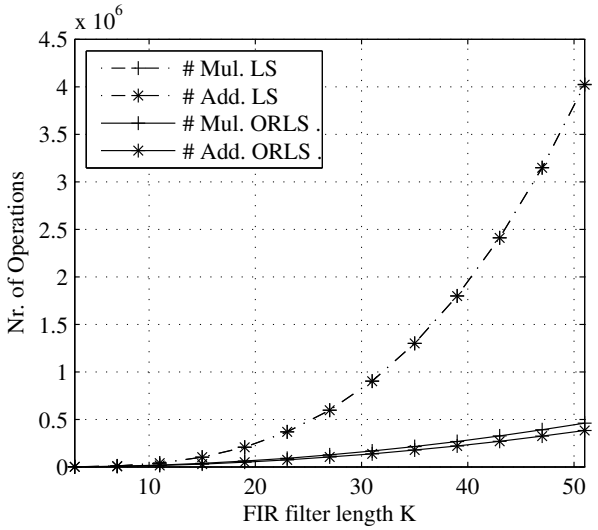


Figure 3.6 Comparison of the computational efficiency of the least-squares and the order recursive least-squares based algorithm. The number of multiplications and additions required for the order recursive least-squares and least-squares based algorithm are shown for different filter lengths K.

order recursive least-squares based algorithm performs more computationally efficient than the least-squares based algorithm. The concrete number of multiplications and additions for selected filter designs are illustrated in Table 3.3.

Table 3.2 Summary and complexity comparison of the order recursive least squares based filter design algorithm. Values that are passed within the algorithm from the previous to the current filter design problem are partitioned in 1). Input values for the algorithm are the modulated observation vector $\hat{\mathbf{m}}_{n,0}$ and the desired vector $\mathbf{d}_n$.

Order Recursive Least-Squares Design	Complexity	# Mult.	# Add.
1) $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger} = \begin{bmatrix} \mathbf{U}^{*} \\ \mathbf{v}^{*} \end{bmatrix}$; $\mathbf{P}_{n-1}^{K} = \begin{bmatrix} \mathbf{S}^{*}, & \mathbf{t} \\ \mathbf{t}^{*}, & c \end{bmatrix}$	$n.a.$[a]		
$\mathbf{M}_{n-1}^{K} = \left[\mathbf{M}_{n-1}^{K-1},\ \mathbf{m}_{n-1,K-1}\right]$	$n.a.$[a]		
Backward Projection			
2) $\mathbf{w}_{\mathrm{b}} = -\dfrac{\mathbf{t}}{c}$	$\mathcal{O}(K)$	$K-1$	
3) $\mathbf{P}_{n-1}^{K-1} = \mathbf{S}^{*} - \mathbf{w}_{\mathrm{b}}\mathbf{t}^{*}$	$\mathcal{O}\left(K^2\right)$	K^2-K	$(K-1)^2$
4) $\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger} = \mathbf{U}^{*} + \mathbf{w}_{\mathrm{b}}\mathbf{v}^{*}$	$\mathcal{O}(RK)$	$2R(K-1)$	$2R(K-1)$
Forward Projection			
5) $\mathbf{w}_{\mathrm{f}} = \left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger}\hat{\mathbf{m}}_{n,0}$	$\mathcal{O}(RK)$	$2R(K-1)$	$(2R-1)$ $(K-1)$
6) $\mathbf{f} = \hat{\mathbf{m}}_{n,0} - \mathbf{M}_{n-1}^{K-1}\mathbf{w}_{\mathrm{f}}$	$\mathcal{O}(RK)$	$2R(K-1)$	$2R(K-2)$ $+R$
7) $\epsilon_{\mathrm{f}} = \hat{\mathbf{m}}_{n,0}^{*}\,\mathbf{f}$	$\mathcal{O}(R)$	$2R$	$2R-1$
8) $\left(\mathbf{M}_{n}^{K}\right)^{\dagger} = \begin{bmatrix} \dfrac{\mathbf{f}^{*}}{\epsilon_{\mathrm{f}}} \\ \left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger} - \dfrac{\mathbf{w}_{\mathrm{f}}\mathbf{f}^{*}}{\epsilon_{\mathrm{f}}} \end{bmatrix}\mathbf{W}^{*}$	$\mathcal{O}(RK)$	$4RK+K$	$2R(K-1)$
9) $\mathbf{P}_{n}^{K} = \begin{bmatrix} \dfrac{1}{\epsilon_{\mathrm{f}}}, & -\dfrac{\mathbf{w}_{\mathrm{f}}^{*}}{\epsilon_{\mathrm{f}}} \\ -\dfrac{\mathbf{w}_{\mathrm{f}}}{\epsilon_{\mathrm{f}}}, & \mathbf{P}_{n-1}^{K-1} + \dfrac{\mathbf{w}_{\mathrm{f}}\mathbf{w}_{\mathrm{f}}^{*}}{\epsilon_{\mathrm{f}}} \end{bmatrix}$	$\mathcal{O}\left(K^2\right)$	K^2	$(K-1)^2$
Filter Design			
10) $\mathbf{g}_{n}^{K} = \left(\mathbf{M}_{n}^{K}\right)^{\dagger}\mathbf{d}_{n}$	$\mathcal{O}(RK)$	$2RK$	$(2R-1)K$
Overall Complexity	$\mathcal{O}(RK)$	$12RK-4R$ $+2K^2+K-1$	$10RK-7R$ $+2K^2-6K+2$

[a] Partitioning of matrices $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$, $\mathbf{P}_{n-1}^{K}$ and $\mathbf{M}_{n-1}^{K}$

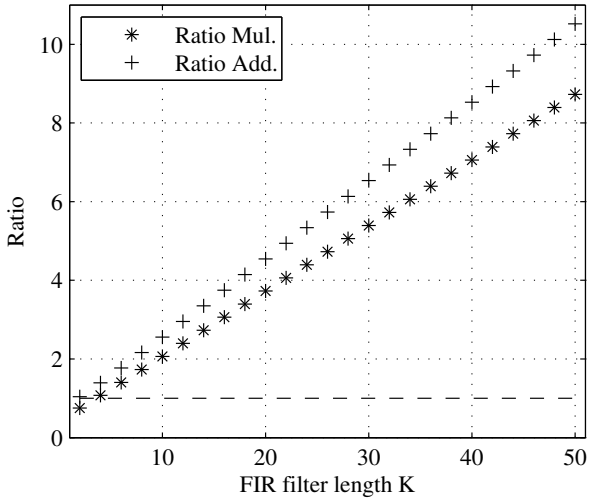


Figure 3.7 Comparison of the computational efficiency of the least-squares and the order recursive least-squares based algorithm. The ratio of the number of multiplications and additions required for the order recursive least-squares and least-squares based algorithm are shown for different filter lengths K.

Table 3.3 Number of required multiplications and additions for selected filter designs with $R = 15(K-1)$.

Filter length K	# Mul. LS	# Add. LS	# Mul. ORLS	# Add. ORLS
3	737	719	980	692
5	3665	3620	3414	2602
31	903681	902193	167552	138088
41	2091041	2088458	296202	244918

Initialization and Adaptation of the Filter Order

In the previous sections considering the backward and forward projection, it was assumed that the matrices $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$ and $\mathbf{P}_{n-1}^{K}$ are known from the previous filter design. At the startup of the algorithm, this assumption does not hold and two different approaches could be employed to resolve this issue. An obvious solution to this problem would be to aggregate observation vectors until a fully populated design matrix with K column vectors is obtained and to calculate $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$, $\mathbf{P}_{n-1}^{K}$ and $\mathbf{g}_{n}^{K}$ subsequently. However, this option

is computationally expensive for the same reason brought forward, when comparing the matrix inversion based filter design and the proposed order recursive least-squares based solution (see Table 3.1 and 3.2).

Alternatively, the order update relationship presented in Section 3.4.3 can be utilized to gradually increase the filter order, without discarding old observations as required to maintain a constant filter order. The generalized inverse is calculated for the time instant $n = 0$, i.e.,

$$\left(\mathbf{M}_0^1\right)^\dagger = \mathbf{P}_0 \ \left(\mathbf{M}_0^1\right)^* \tag{3.46}$$

with

$$\mathbf{P}_0 = \left(\left(\mathbf{M}_0^1\right)^* \ \mathbf{M}_0^1\right)^{-1} \tag{3.47}$$

and

$$\mathbf{M}_0^1 = \mathbf{m}_{0,0}. \tag{3.48}$$

During the time period $1 \leq n < K$, the order of the matrices in (3.46) and (3.47) is increased employing (3.36)-(3.42). The corresponding filter coefficients are obtained using (3.27). The latter alternative is the more favorable one since the matrix inversion can be avoided. Furthermore, the signal reconstruction can be performed immediately by utilizing the coefficients of the filter whose order is gradually increased.

The initialization is completed after performing $K - 1$ filter designs. The initialization case is illustrated in the signal flow graph depicted in Fig. 3.8 by configuration a), where the backward projection block is bypassed and the filter order of the design is increased by the forward projection process for each time instant. This configuration can also be used to increase the filter order during the runtime of the algorithm if a better reconstruction performance is desired. A constant order of the filter design is obtained in configuration b), where the lower order equivalents of $\left(\mathbf{M}_{n-1}^K\right)^\dagger$ and $\mathbf{P}_{n-1}^K$ are calculated during the backward projection process. In order to obtain a filter design of reduced order, the backward projection process is performed twice before the forward projection is performed as illustrated in configuration c). In the latter configuration, the computational complexity of the algorithm rises as the calculations 2)-4) in Table 3.2 are repeated with the intermediate results from the first backward projection process, however, the same number of calculations is saved in configuration a).

Stability Analysis

The stability of the proposed algorithm is investigated numerically both in terms of quantization effects due to finite coefficients lengths and incorrect observations. Although, the

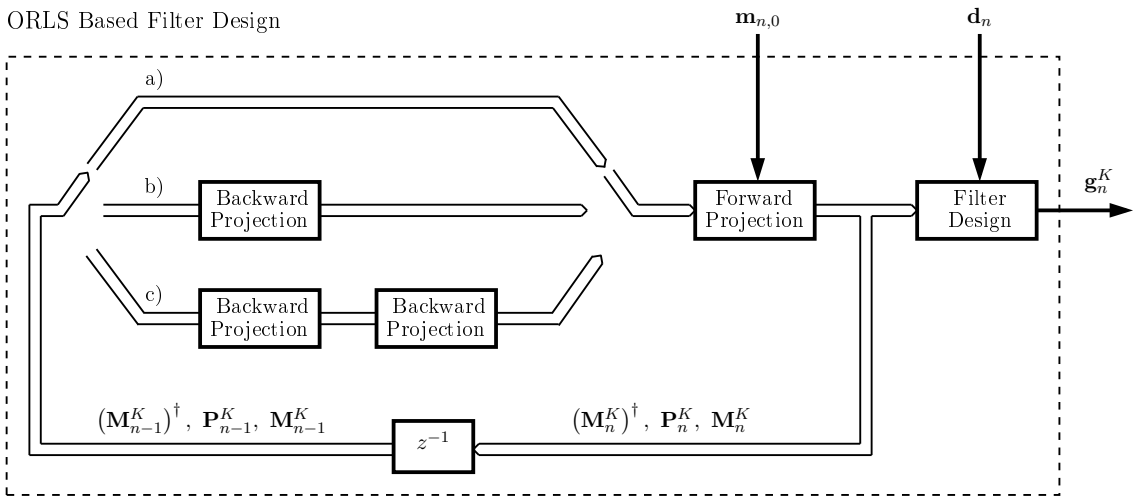


Figure 3.8 Signal flow graph representation of the proposed algorithm for configuration a) initialization/increasing the order of the filter design, configuration b) constant order filter design and configuration c) decreasing the order of the filter design.

proposed algorithm is recursive in nature since results from the previous filter design problem are used, the algorithm exhibits only finite memory. Inspection of (3.36) reveals that any new observation $\hat{\mathbf{m}}_{n,0}$ does only influence the current and the following $K-1$ design matrices. Disregarding finite precision effects, the generalized inverse is directly related to the design matrix via the matrix inversion lemma [42]. Thus, any incorrect observation will only have an impact on K generalized inverses affecting K filter designs before being discarded in the next filter design problem. This behavior of the algorithm corresponds to a memory of $K-1$.

The respective behavior of the algorithm is illustrated by means of a numerical experiment, where the reconstruction performance of two correction systems is compared. The objective of the presented experiments is to demonstrate the ability of the proposed algorithm to recover from incorrect observations. This will be investigated for implementations employing floating point double precision arithmetic in Matlab and fixed point arithmetic. Initially, both correction systems employ floating point double precision arithmetic to design a reconstruction filter of order 4 ($K=5$).

In the first system, the reconstruction filter is designed utilizing the proposed algorithm whose input observations are wrong for parts of the simulation time. In the following, each incorrect observation is synthesized by generating a random impulse response selected according to a standard normal Gaussian distribution ($\mu=0$, $\sigma^2=1$). The second system which serves as a reference determines the reconstruction performance in the least-squares

sense, where the filter design is performed by the reference algorithm presented in Section 3.4.1. The squared error of the reconstructed signals $\check{e}[n]^2$ will serve as a figure of merit to determine the reconstruction performance of the proposed algorithm. The experimental setup is illustrated in Fig. 3.9.

In Fig. 3.10(a), the system behavior for a single incorrect observation affecting the proposed algorithm is depicted. The incorrect observation occurs at $n = 20$ and deteriorates the system's performance for K samples as K filter designs are affected by the incorrect observation while this observation is still comprised in the design matrix, cf., (3.36). A related case is shown in Fig. 3.10(b), where the proposed algorithm is initialized and executed with incorrect observations up to the time instant $n = 20$ and with correct observations for $n > 20$. As in Fig. 3.10(a), the performance returns to the optimum solution in the least-squares sense after all incorrect observation have been replaced by correct observations in the design matrix (for $n \geq 25$). In the presented examples, where both systems were implemented with floating point double precision arithmetic, the proposed algorithm regains the optimum reconstruction performance in the least-squares sense immediately after all incorrect observations are discarded and no longer comprised in the design matrix. The behavior of the proposed algorithm, implemented in fixed point arithmetic, will be investigated in the remainder of this section.

To this end, the effects of a series of incorrect observations is investigated for two fixed point implementations of the proposed algorithm. In these implementations, the filter coefficients and each intermediate variable in the proposed algorithm are calculated using two's complement fixed point fractional numbers. Two different two's complement formats are employed, where the first format describes a fixed point fractional number with 15 integer bits and 16 fractional bits (Q15.16) and the second format describes a 15 integer bits and 32 fractional bits (Q15.32). The achieved performance of each system is again compared with the performance of the reference algorithm employing floating point double precision arithmetic in Matlab. The reconstruction performance of the Q15.16 implementation is illustrated in Fig. 3.11(a), where a recurring series of 500 incorrect observations followed by 1000 correct observations is utilized for the filter design. Each cycle with 500 incorrect observations provides a large number of incorrect observations to ensure that the design matrix is constituted by incorrect observations, and the respective generalized inverse is based on incorrect observations only. As soon as correct observations are available to the filter design algorithm, the squared error returns to approximately -60 dB. The Q15.32 fixed point implementation offers a similar behavior for a minimum reconstruction error of approximately -155 dB. Introducing several cycles with incorrect

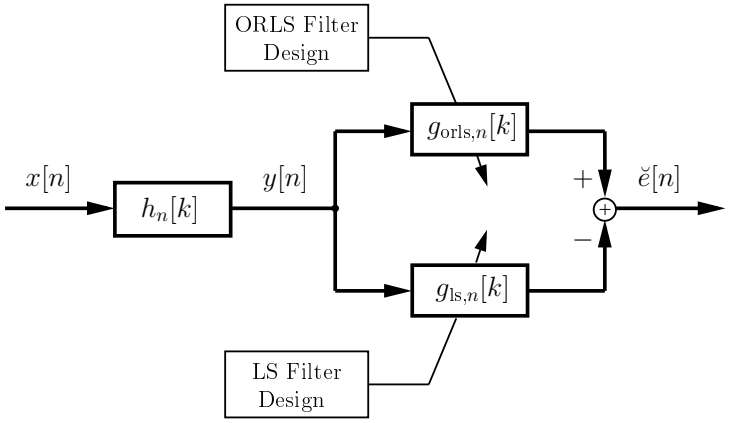
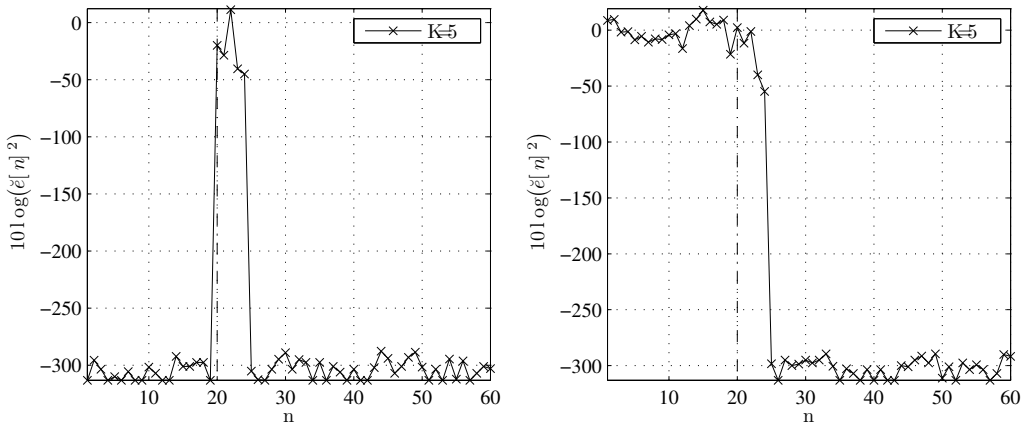


Figure 3.9 Experimental setup for determining the deviation of the perturbed filter design from the optimum filter design in the least-squares sense. The latter filter design is obtained employing the reference algorithm with floating point double precision arithmetic.



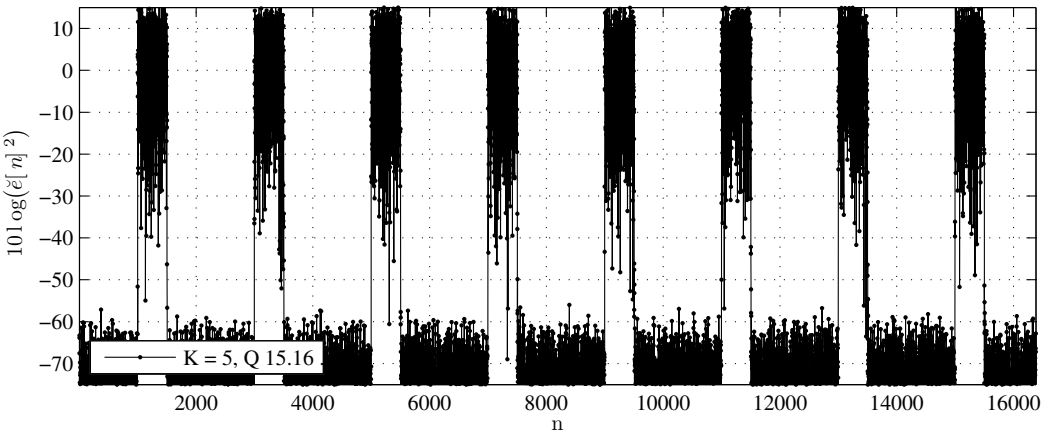
(a) Single incorrect observation for $n = 20$. The occurrence of the incorrect observation is marked by the dashed line

(b) Wrong observation for $n \leq 20$. The occurrence of the last incorrect observation is marked by the dashed line

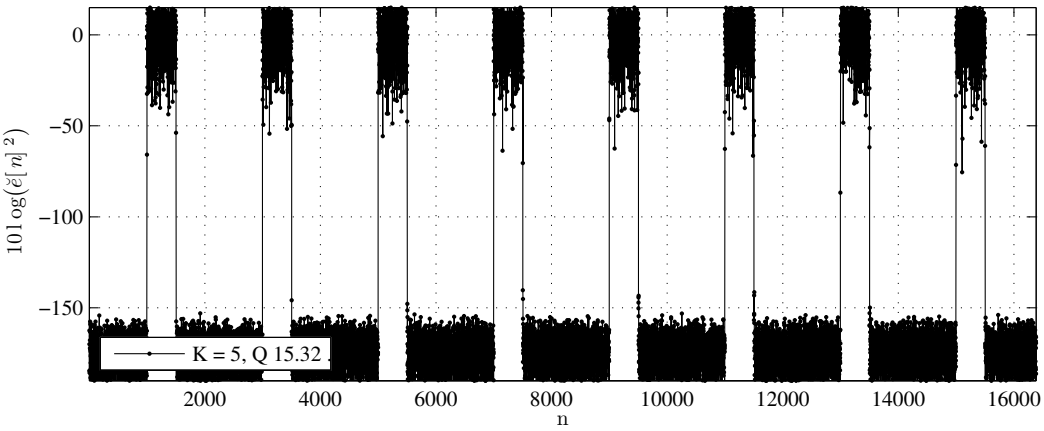
Figure 3.10 Squared difference of the reconstructed signals in dB. The least-squares reconstruction performance is regained for $n \geq 25$ in both cases.

observations does not have a negative effect on the achieved reconstruction performance of the proposed algorithm.

However, the fixed point arithmetic has an impact on the time the algorithm requires to regain the minimum reconstruction error after correct observations are available. The relationship of this delay and the employed arithmetic is illustrated in Fig. 3.12. In this experiment, the proposed filter design algorithm processes incorrect observations up to the time instant $n = 20$ and the incorrect observations are not part of the design matrix for $n \geq 25$, similar to the simulation scenario depicted in Fig. 3.10(b). In Fig. 3.12, three different fixed point implementations and an implementation using floating point double precision are employed by the proposed algorithm and compared with the reference algorithm employing floating point double precision. Each line represents an ensemble average of 500 simulation outcomes of the respective implementation. The fixed point filter designs at time instant $n = 25$ achieve a reconstruction error which is about 20 dB larger than their respective minimum reconstruction errors. This minimum reconstruction error is reached by each implementation after performing about nine consecutive filter designs at time instant $n = 35$.



(a) Two's complement fixed point implementation with 15 integer bits and 16 fractional bits (Q15.16)



(b) Two's complement fixed point implementation with 15 integer bits and 32 fractional bits (Q15.32)

Figure 3.11 Squared difference of the signal reconstructed by the fixed point implementations of the proposed filter design algorithm and the reconstructed signal obtained by the least squared filter design with floating point double precision.

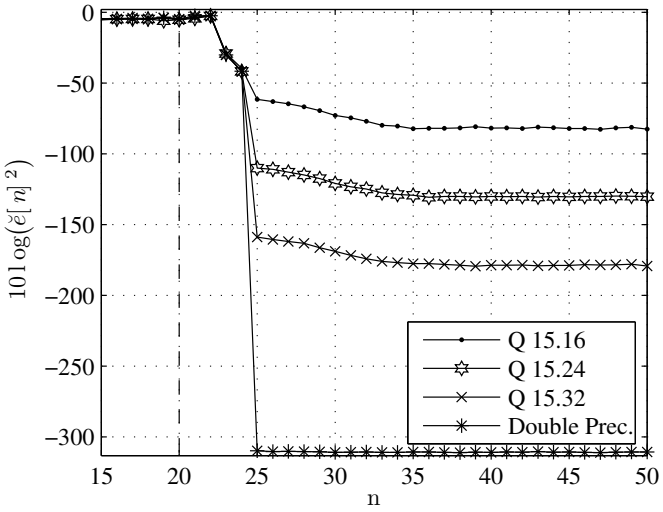


Figure 3.12 Ensemble average of the signal $\check{e}[n]^2$. The signal $\check{e}[n]$ represents the difference of the output of two correction schemes employing the proposed and the reference algorithm, respectively. The correction scheme utilizing the proposed algorithm is implemented either in three different fixed point implementations or in floating point double precision arithmetic. The correction scheme that utilizes the reference algorithm employs floating point double precision arithmetic.

3.5 Application Example

In the following, the presented theory in Section 3.2 and the filter design algorithm as proposed in Section 3.4.3 are employed to precorrect non-uniform sample-and-hold (SH) signals in a DAC. As an example of an SH circuit, we chose the ZOH type due to its practical importance.

3.5.1 Continuous-Time System Model

It is well-known that ZOH signals shape the analog output signal according to the continuous-time Fourier transform (CTFT) of its impulse response [44]. Moreover, spurious tones are introduced if the uniform sampling instants deviate by a time-varying jitter term $\Delta_n T$ [54]. Both effects can be modelled by the time-varying impulse response [23]

$$\tilde{a}(t) = T\left(u(t-\Delta_n T) - u(t-T-\Delta_{n+1}T)\right) \tag{3.49}$$

with T indicating the sampling period. The system employing $\tilde{a}(t)$ to represent the non-uniform ZOH behavior is depicted in Fig. 3.13.

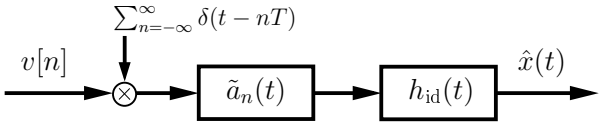


Figure 3.13 DAC model employing a non-uniform ZOH model in continuous-time.

Calculating the CTFT of (3.49) leads to [23]

$$\tilde{A}_n(j\Omega) = \frac{\sin\left(\frac{\Omega T}{2}(1+\Delta_{n+1}-\Delta_n)\right)}{\frac{\Omega T}{2}} e^{-j\frac{\Omega T}{2}(1+\Delta_{n+1}+\Delta_n)}. \tag{3.50}$$

The output signal of the presented model is obtained as

$$\hat{x}(t) = \sum_{k=-\infty}^{\infty} v[k]\tilde{a}_k(t-kT) * h_{\mathrm{id}}(t) \tag{3.51}$$

where $v[n]$ indicates the discrete-time input of the overall model. Furthermore, the impulse response of the ideal low-pass filter $h_{\mathrm{id}}(t)$ with a cut-off frequency $0 < \Omega_D \leq \frac{\pi}{T}$, is given as [44]

$$h_{\mathrm{id}}(t) = \frac{\Omega_D}{\pi}\mathrm{sinc}\left(\frac{\Omega_D}{\pi}t\right). \tag{3.52}$$

3.5.2 Discrete-Time System Model

In order to devise an equivalent formulation of (3.51), we define a representation of a discrete-time filter as [44]

$$\hat{A}_n(e^{j\omega}) = \tilde{A}_n\left(j\frac{\omega}{T}\right) H_{\mathrm{id}}\left(j\frac{\omega}{T}\right) \quad \text{for} - \pi \leq \omega < \pi \tag{3.53}$$

which is constituted by the discrete-time representation of the CTFT of (3.49), as it was presented in [23], and the CTFT of (3.52). With a cut-off frequency $\Omega_D = \frac{\omega_D}{\pi}$, the output of the discrete-time filter results in

$$\hat{x}[n] = \sum_{l=-\infty}^{\infty} v[l]\hat{a}_l[n-l] \tag{3.54}$$

where $\hat{a}_n[k]$ is the inverse DTFT of $\hat{A}_n(e^{j\omega})$. The continuous-time signal $\hat{x}(t)$ is obtained by

$$\hat{x}(t) = \sum_{n=-\infty}^{\infty} \hat{x}[n]h_{\mathrm{id}}(t-nT) \tag{3.55}$$

which represents the same continuous-time signal as obtained by the system in (3.51). Rewriting (3.54) using $k = n - l$ and $l = n - k$, we obtain

$$\hat{x}[n] = \sum_{k=-\infty}^{\infty} \hat{a}_{n-k}[k]v[n-k], \tag{3.56}$$

and comparing it to the time-varying system in (3.2) reveals the identity

$$g_n[k] = \hat{a}_{n-k}[k] \tag{3.57}$$

which agrees with (3.5). Applying this identity to the time-varying system in (3.2), we can represent the ZOH behavior in discrete-time as illustrated in the system model shown in Fig. 3.14.

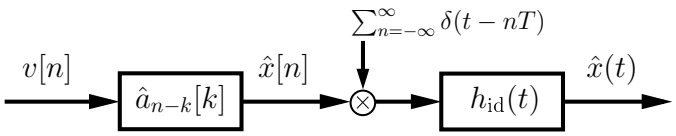


Figure 3.14 DAC model employing a discrete-time ZOH model.

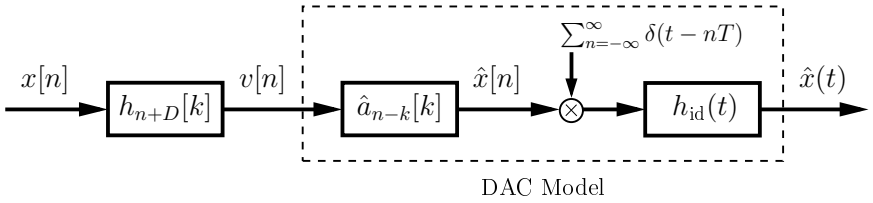


Figure 3.15 System precorrecting the input signal of the DAC that is affected by non-uniform sampling.

3.5.3 Precorrection Scheme and Filter Design

In this section, the overall system is presented, where the time-varying filter $h_n[k]$ precorrects the non-uniform DAC model as shown in Fig. 3.15 where D indicates the delay of the discrete-time ZOH model. The design of the precorrection filter is obtained by defining a design objective in terms of the desired frequency response as

$$\hat{D}_n(e^{j\omega}) = e^{-j\omega D_s} \tag{3.58}$$

where D_s is the accumulated delay induced by the filter delay and the delay of the undesired time-varying system. By specifying $\hat{D}_n(e^{j\omega})$ as a delay D_s, the impact of the undesired in-band attenuation and of the non-uniform sampling are mitigated at the same time. As a consequence, the ideally reconstructed output signal results in $\hat{x}[n] = x[n - D_s]$.

For the precorrection of the DAC model, we obtain for $\hat{G}_n(e^{j\omega}) = \hat{A}_n(e^{j\omega})$ the reformulation of the design equation in (3.11) which results in

$$\hat{F}_n(e^{j\omega}) = \sum_{l=0}^{K-1} \hat{M}_{n,l}(e^{j\omega})\ \hat{h}_n[l] \tag{3.59}$$

with

$$\hat{M}_{n,l}(e^{j\omega}) = \hat{A}_{n+l}(e^{j\omega})e^{j\omega l}. \tag{3.60}$$

The design equations (3.19)-(3.42) of the proposed order recursive filter design algorithm can be adapted accordingly by replacing $D_n(e^{j\omega})$ with $\hat{D}_n(e^{j\omega})$, $H_{n-l}(e^{j\omega})$ with $\hat{A}_{n+l}(e^{j\omega})$ and $\mathbf{W}$ with $\mathbf{W}^*$. Employing this filter design algorithm, the filter coefficients $\hat{h}_n[l]$ are calculated for each time instant. By using (3.57), the coefficients $h_n[l]$ are obtained. Furthermore, the coefficients $h_n[l]$ are employed to continuously update the precorrection filter within the system which is depicted in Fig. 3.15.

To verify the proposed scheme, numerical simulations were performed in Matlab and the achieved correction performance was characterized in terms of the signal-to-noise ratio (SNR) given as

$$\mathrm{SNR} = 10\log_{10}\left(\frac{\sum_{n=0}^{N-1}|x[n-D_s]|^2}{\sum_{n=0}^{N-1}|x[n-D_s]-\hat{x}[n]|^2}\right)\mathrm{dB} \tag{3.61}$$

where N indicates the number of investigated samples. The non-uniform ZOH signals are generated, i.e., Δ_n follows a zero-mean Gaussian distribution with a standard deviation of $\sigma_z = 0.039$ and 2^{12} samples were drawn from this distribution to characterize the non-uniform ZOH behavior. The same number of samples was evaluated, and a coherently sampled multitone input signal was used for the simulations. The impact of the non-uniform ZOH behavior is illustrated in Fig. 3.16, where the spectrum of a non-bandlimited output signal is shown, when the DAC input signal is not precorrected. The depicted spectrum exhibits an in-band attenuation of up to -2 dBc for the highest frequent signal component and a multitude of spurious tones are created in-band and out-of-band.

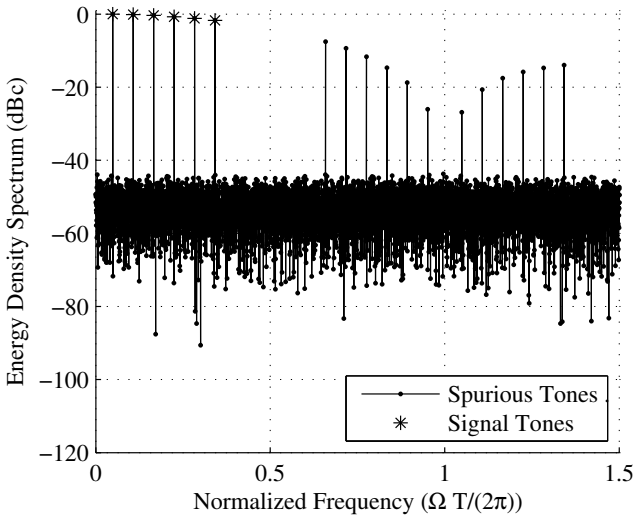


Figure 3.16 Uncorrected output spectrum.

In order to suppress the spurious tones within the bandwidth $|\omega| < 0.75\pi$, a time-varying precorrection with a filter length $K = 13$ and a design domain of $\omega_D = 0.75\pi$ was designed according to the L_2 norm. The precorrected output spectrum of the DAC before the low-pass filter h_{id} is shown in Fig. 3.17 illustrating a considerable attenuation of the in-band spurious tones. Moreover, the in-band attenuation of the signal tones was compensated. The spectrum of the output signal $\hat{x}[n]$ is depicted in Fig. (3.18), where the dashed line indicates the spectral components removed by h_{id}. The signal $\hat{x}[n]$ offers a SNR value of 77.96 dB within the bandwidth $|\Omega| < \frac{0.75\pi}{T}$ which enhances the initial SNR value for the uncorrected case by 59.92 dB.

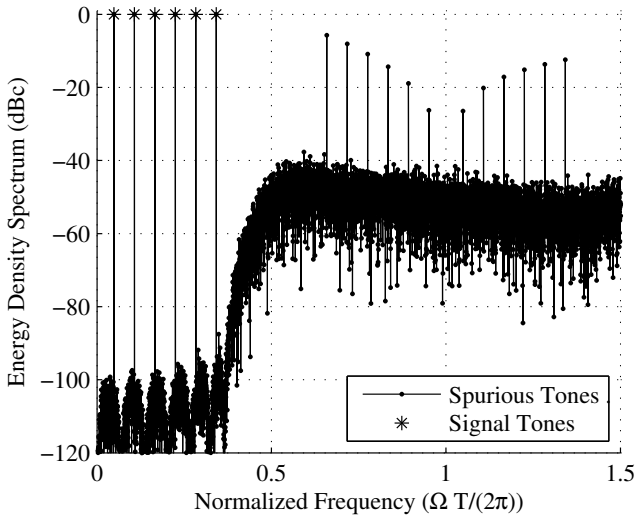


Figure 3.17 Spectrum of precorrected output signal before the low-pass filter h_{id} ($K = 13$).

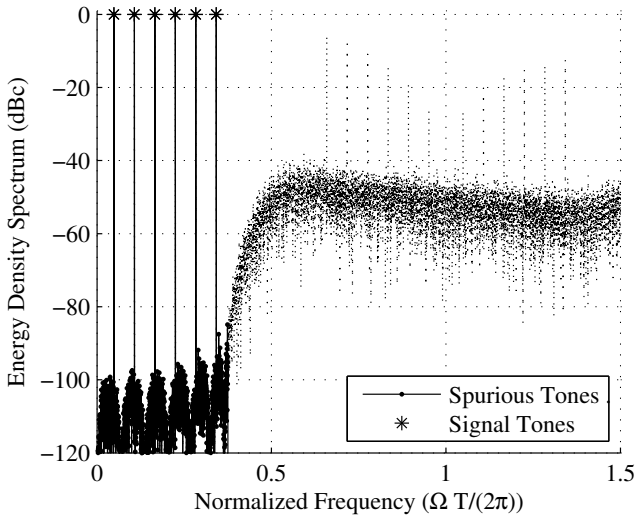


Figure 3.18 Spectrum of precorrected output signal $\hat{x}[n]$ ($K = 13$). The dashed lines indicate the spectral components removed by h_{id}.

3.6 Concluding Remarks

In this chapter, the design of time-varying FIR filters facilitating the precorrection and postcorrection of linear time-varying systems has been presented. By updating the coefficients of a time-varying FIR filter, the impact of a non-periodically time-varying systems

can be corrected. However, in this case, a dedicated filter design for each time instant is required which is computationally intensive. For the design of correction filters enhancing the performance of time-invariant and periodically time-varying systems, an offline design is feasible and the computational burden might be acceptable. To alleviate the computational burden in non-periodically time-varying scenarios, a low complexity algorithm for the design of the correction filter has been presented. The time-varying filter design problem has been posed in terms of a backward and forward projection problem employing order update relations. Moreover, the computational complexity of the proposed algorithm has been compared with a reference filter design algorithm which is based on an explicit matrix inversion and whose computational complexity is identical to the algorithms presented in [9] and [20]. The proposed algorithm proved to be computationally more efficient than the algorithms presented in [9], [20] and [21] both in terms of the $\mathcal{O}$-notation and in terms of the number of operations for filter orders greater than three. This reduced complexity was achieved by formulating the time-varying filter problem, i.e., the results of the previous filter design process could be reused for the current filter design.

Furthermore, the numerical stability of a fixed point implementation of the algorithm and its behavior in the presence of incorrect observations were investigated.

4

Correction of Weakly Time-Varying Linear Systems

4.1 Introduction

This chapter introduces low-complexity structures to correct the effects of these time-varying systems. To significantly reduce the computational complexity, we will narrow the discussion to linear weakly time-varying systems. Similar to the definition of weakly nonlinear systems [55], we can roughly classify linear weakly time-varying systems as systems where we have a dominant time-invariant behavior disturbed by a time-varying behavior reducing the signal quality. This work is presented in the context of digitally enhanced mixed-signal systems, however, the results can be applied to arbitrary linear weakly time-varying systems.

A frequently occurring problem in mixed-signal circuits is non-uniform sampling which can be modeled as a sampled time-varying system. Recently, low-complexity algorithms for signal reconstruction from its non-uniform samples have attracted a lot of research interest [8, 19, 31, 49, 56–58]. As a more general case, the correction of general frequency response mismatches in time-varying systems has been investigated in [9, 59, 60]. Various methods exist to compensate the effect of M-periodically time-varying systems, for example by means of a subsequent M-periodically time-varying reconstruction filter [9], synthesis filters in a filter bank [47] or by utilizing a multichannel bank of filters employing multirate theory [20]. These techniques offer great flexibility to correct for general frequency characteristics, however, the design complexity of these reconstruction filters is challenging as matrix inversions have to be performed or optimization problems need to be solved. The resulting computational load is especially a disadvantage if the characteristics of the time-varying system change during operation, and the correction system has to be redesigned to meet the desired compensation performance. An example of how the

performance of a time-varying system, which is enhanced by an offline designed correction system, may deteriorate as the time-varying system is subject to drift, can be found in [21].

To reduce the computational burden, recently introduced postcorrection systems rely on an approximated solution to correct the time-varying behavior affecting the signal. The utilized structures are constituted by a cascade of correction stages each comprising a time-varying filter to gradually refine the approximated solution. Such a structure was initially proposed in [31], where a cascade of differentiator stages was employed to mitigate the error induced by non-uniform sampling. To enhance the SNR performance for a given number of FIR filters, the number of subfilters comprised in each stage was increased linearly over the number of utilized stages. This approach was extended in [26] to correct for mismatches of general frequency responses. A related approach employing cascades of differentiator banks has been presented in [30], where in each correction stage an equal number of differentiators of higher orders compensate frequency response mismatch errors. Recently, the link between the correction structures in [26, 31] and stationary iterative methods for the solution of systems of linear equations [61] has been introduced in [62] and [28]. Thanks to this significant observation, structures to compensate linear weakly time-varying systems can be seen in a completely different context.

The mathematical framework of time-varying systems may also be used to enhance the performance of a system by precorrecting it, which is required for a different type of application. For example, DACs suffer from timing jitter, which can be modeled by a time-varying system [54]. In [48], a structure to precorrect 2-periodically non-uniform ZOH signals in a DAC was presented. As a result, the in-band spurious frequencies in the DAC spectrum were mitigated considerably. An extended mathematical framework was presented in [23] for the M-periodic case. However, a precorrection structure which can be adapted online was only demonstrated for the 2-periodic case.

4.1.1 Contributions

This chapter presents correction structures for the precorrection and postcorrection of bandlimited signals in order to compensate for the impact of linear weakly time-varying systems. The characteristics of such a system are assumed to be known, which can be achieved by using a method for the blind identification of the time-varying system [42] or simply by means of a-priori knowledge obtained, e.g., via calibration. Although many of the results can be extended to other stationary iterative methods [61], for the sake of consistency we will only utilize the Richardson iteration [61] for our correction structures.

a) Polynomial Transfer Function: In Section 4.3.1, a polynomial transfer function based

on the recurrence relation of the Richardson iteration is derived. Furthermore, the commutativity of this polynomial transfer function with the time-varying system that needs to be corrected is established. Therefore, results regarding reconstruction performance, conditions for convergence and computational complexity obtained for the postcorrection case, as presented in [26–29] can be directly applied to the presented precorrection case.

b) Correction of weakly time-varying systems: In Section 4.3, the recurrence relation is used to devise a causal representation of the postcorrection structure as it has been presented in [28] and [31] to correct for non-uniform sampling and in [26] to postcorrect general frequency response mismatches. By applying the property of commutativity, a correction structure for the precorrection case is established.

c) Precorrection of DACs (online filter design): The precorrection of non-uniform ZOH signals in DACs is employed to illustrate the precorrection of weakly time-varying analog systems. The design of the time-varying filter comprised in each correction stage is performed during operation of the system. This proposed online filter design complements the filter design for the ADC case as presented in [26].

d) Precorrection of DACs (offline filter design): A precorrection of the same system by means of polynomial filters is presented in order to reduce the complexity and avoid the computational burden induced by the online filter design. The presented application of the proposed precorrection structure extends the work in [23] to the non-periodic case.

4.1.2 Outline

In Section 4.2, the principle of time-varying systems, represented by time-varying impulse responses and transition matrices, and the Richardson iteration is reviewed. Furthermore, a definition of weakly time-varying systems is presented. In Section 4.3, a polynomial transfer function is derived characterizing the Richardson iteration, and a causal representation of precorrection and postcorrection structures for weakly time-varying systems is derived. Moreover, the precorrection of non-ideal DACs in a mixed-signal scenario is discussed in Section 4.4.

4.2 Weakly Time-Varying Systems

In this section, the basic input/output relations of time-varying systems and the cascading of time-varying systems is reviewed as has been presented in [7]. Furthermore, an algebraic representation of time-varying systems is presented and the Richardson iteration is reviewed [61, 63].

4.2.1 Input/Output Relations

The output $y[n]$ of a time-varying system is the result of the convolution of the input signal $x[n]$ with the system's time-varying impulse response $h_n[k]$ that is

$$y[n] = \sum_{k=-\infty}^{\infty} h_n[k]x[n-k]. \tag{4.1}$$

Alternatively, a time-varying system may be described as an algebraic system. By defining the infinite input vector $\mathbf{x}$ as

$$\mathbf{x} = [\ldots, x[-1], x[0], x[1], \ldots]^T \tag{4.2}$$

and the infinite output vector $\mathbf{y}$ as

$$\mathbf{y} = [\ldots, y[-1], y[0], y[1], \ldots]^T \tag{4.3}$$

we can rewrite (4.1) as

$$\mathbf{y} = \mathbf{H}\mathbf{x} \tag{4.4}$$

where the time-varying impulse response $h_n[k]$ is represented by the transition matrix

$$\mathbf{H} = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \ddots \\ \ldots & h_{-1}[0] & h_{-1}[-1] & h_{-1}[-2] & \ldots \\ \ldots & h_0[1] & h_0[0] & h_0[-1] & \ldots \\ \ldots & h_1[2] & h_1[1] & h_1[0] & \ldots \\ \ddots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{4.5}$$

4.2.2 Definition

Linear time-varying systems which can be represented by a transition matrix as defined (4.5), while fulfilling the criterion [61]

$$\sum_{k=-\infty}^{\infty} |\delta[k] - h_n[k]| < 1 \tag{4.6}$$

for $-\infty < n < \infty$ and where $\delta[k]$ is the discrete-time impulse [44], will be referred to as linear *weakly time-varying systems* in the remainder of this work. The criterion in (4.6) restricts the impulse response $h_n[k]$ at each time instant n independent of past or future states of the time-varying system. Thus, the maximum deviation of the system is also restricted in terms of its time-varying behavior when progressing from one to the subsequent time instant.

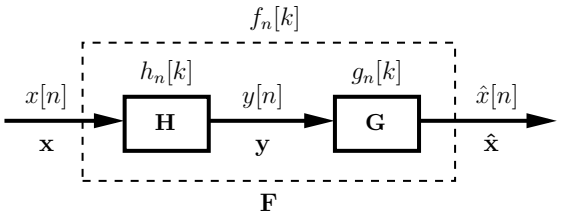


Figure 4.1 Cascade of two time-varying systems.

4.2.3 Correcting Time-Varying Systems

As illustrated in Fig. 4.1, the output $\hat{x}[n]$ of a system characterized by the time-varying impulse response $g_n[k]$ is given by

$$\hat{x}[n] = \sum_{k=-\infty}^{\infty} g_n[k]y[n-k] \tag{4.7}$$

and leads with (4.1) to [7]

$$\hat{x}[n] = \sum_{k=-\infty}^{\infty} f_n[k]x[n-k] \tag{4.8}$$

where

$$f_n[k] = \sum_{l=-\infty}^{\infty} g_n[l]h_{n-l}[k-l] \tag{4.9}$$

is the resulting time-varying impulse response of the cascaded system.

The output of the two cascaded time-varying systems can also be written as

$$\hat{\mathbf{x}} = \mathbf{F}\mathbf{x} \tag{4.10}$$

where the output vector $\hat{\mathbf{x}}$ is

$$\hat{\mathbf{x}} = [\ldots, \hat{x}[-1], \hat{x}[0], \hat{x}[1], \ldots]^T, \tag{4.11}$$

the transition matrix $\mathbf{F}$ is

$$\mathbf{F} = \mathbf{G}\mathbf{H}, \tag{4.12}$$

and the transition matrix $\mathbf{G}$ is of the same form as the transition matrix $\mathbf{H}$ in (4.5).

If the input $\mathbf{x}$ should be reconstructed from the output $\mathbf{y}$ of the first system $\mathbf{H}$ via the second system $\mathbf{G}$, i.e.,

$$\hat{\mathbf{x}} = \mathbf{G}\mathbf{y}, \tag{4.13}$$

a straight forward approach would be to design the second system $\mathbf{G}$ as the inverse of the first system $\mathbf{H}$, i.e., $\mathbf{G} = \mathbf{H}^{-1}$. In the ideal case, $\mathbf{G}$ corrects perfectly for the impact of

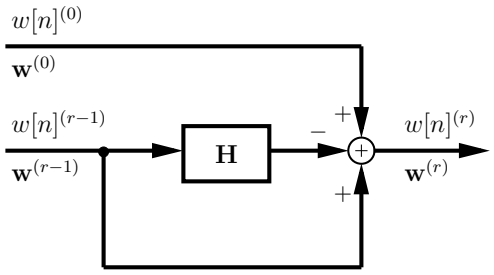


Figure 4.2 Recurrence relation of the Richardson iteration.

the first time-varying system $\mathbf{H}$ and the output $\hat{\mathbf{x}}$ equals the input $\mathbf{x}$. This case will be referred to as the *postcorrection* of the system $\mathbf{H}$ in the remainder of this work. The other possible scenario, where a time-varying system $\mathbf{H}$ is designed to alleviate the impact of a subsequent time-varying system $\mathbf{G}$, which is ideally $\mathbf{H} = \mathbf{G}^{-1}$, will be referred to as *precorrection*. In most situations, a correction system resulting in a delayed version of the input signal, i.e.,

$$\hat{x}[n] = x[n - D_s] \tag{4.14}$$

with D_s being the delay of the overall system, can also be regarded as to yield perfect reconstruction [9, 39]. Consequently, for this case the time-varying impulse response $f_n[k]$ of the cascaded systems results in $f_n[k] = \delta[k - D_s]$.

4.2.4 Richardson Iteration

Direct methods allow for the computation of the inverse of a time-varying filter represented by its transition matrix as shown in (4.5), however, these methods are computationally intensive. Thus, we utilize an iterative method to approximate the inverse characteristics of the weakly time-varying system. Although many iterative methods are possible [26, 28, 31], we only employ the Richardson method to make the further discussion more concise.

The Richardson iteration determines the solution of a linear system of equations, i.e.,

$$\hat{\mathbf{x}} = \mathbf{H}^{-1}\mathbf{y} \tag{4.15}$$

in terms of a linear fixed point iteration [61]. The approximation of the solution vector $\hat{\mathbf{x}}$ after a number of r iterations is indicated by $\mathbf{w}^{(r)}$ and given by the recurrence relation [61]

$$\mathbf{w}^{(r)} = \mathbf{w}^{(r-1)} + \mathbf{w}^{(0)} - \mathbf{H}\mathbf{w}^{(r-1)} \tag{4.16}$$

where the input vector $\mathbf{w}^{(0)} = \mathbf{y}$ for the given system in (4.15). Fig. 4.2 represents this recurrence relation as a block diagram.

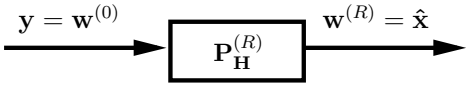


Figure 4.3 Richardson iteration characterized by the polynomial transfer function $\mathbf{P}_{\mathbf{H}}^{(R)}$.

The Richardson method solves a system of linear equations in an element-wise manner by eliminating the n^{th} element of the residual vector $\mathbf{y} - \mathbf{H}\hat{\mathbf{x}}$ [61]. As a consequence, an element in $\mathbf{w}^{(r)}$ can be iterated independently from other elements of the same vector which can be clearly seen from its line-wise formulation given as

$$w[n]^{(r)} = w[n]^{(r-1)} + w[n]^{(0)} - \sum_{k=-\infty}^{\infty} h_n[k] w[n-k]^{(r-1)} \tag{4.17}$$

where $h_n[k]$ is an acausal time-varying impulse response as it was presented in [28].

The component-wise iteration in (4.17) is bound to converge for the given time instant n, if $h_n[k]$ fulfills the criterion in (4.6) for all values of n, and, as a consequence, the system with impulse response $h_n[k]$ is weakly time-varying. Thus, the inverse of the weakly time-varying system can be approximated using the relation in (4.17). The rate of convergence for the Richardson iteration and alternative methods have been compared and examples for mixed-signal and communication applications have been presented in [26, 28] and [62] respectively.

4.3 Precorrection and Postcorrection

After introducing a polynomial transfer function for the Richardson iteration, causal correction structures which can be applied in postcorrection and precorrection scenarios are derived. The precorrection structure is devised by establishing the commutativity of the undesired system and the postcorrection structure. The proposed causal precorrection structure complements the work presented on the postcorrection case [26–29], and the presented postcorrection structure illustrates the causal implementation for the postcorrection case.

4.3.1 Polynomial Relation

In order to characterize the performance of the Richardson iteration, a transfer function is devised. The investigated postcorrection system is depicted in Fig. 4.3 where the output of the Richardson iteration $\mathbf{w}^{(R)}$ for R iterations and input vector $\mathbf{w}^{(0)}$ is given by

$$\mathbf{w}^{(R)} = \mathbf{P}_{\mathbf{H}}^{(R)} \mathbf{w}^{(0)} \tag{4.18}$$

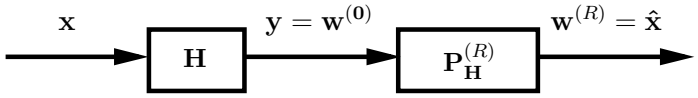


Figure 4.4 Richardson iteration of order R postcorrecting the weakly time-varying system $\mathbf{H}$.

and the matrix $\mathbf{P}_{\mathbf{H}}^{(R)}$ can be obtained in closed form using the relation

$$\mathbf{P}_{\mathbf{H}}^{(R)} = \mathbf{I} + \sum_{r=1}^{R} \binom{R}{r} \left((-\mathbf{H})^r + (-\mathbf{H})^{r-1} \right) \tag{4.19}$$

as shown in Appendix A.3. Therefore, the input $\mathbf{x}$ and the postcorrected output $\hat{\mathbf{x}}$ can be related by means of (4.13), (4.4) and the derived transfer function in (4.19) with $\mathbf{G} = \mathbf{P}_{\mathbf{H}}^{(R)}$, $\mathbf{w}^{(0)} = \mathbf{y}$ and $\hat{\mathbf{x}} = \mathbf{w}^{(R)}$ as

$$\hat{\mathbf{x}} = \mathbf{P}_{\mathbf{H}}^{(R)} \mathbf{H} \mathbf{x}. \tag{4.20}$$

The corresponding postcorrection scenario is depicted in Fig. 4.4. It is important to note that (4.19) represents an univariate matrix polynomial in $\mathbf{H}$. This type of matrix polynomial commutes under matrix multiplication with its matrix variable as shown in [64], i.e., $\mathbf{P}_{\mathbf{H}}^{(R)} \mathbf{H} = \mathbf{H}\, \mathbf{P}_{\mathbf{H}}^{(R)}$. Accordingly, we might rewrite (4.20) by using the commutativity of $\mathbf{P}_{\mathbf{H}}^{(R)}$ as

$$\hat{\mathbf{x}} = \mathbf{G} \mathbf{P}_{\mathbf{G}}^{(R)} \mathbf{x} \tag{4.21}$$

where $\mathbf{H} = \mathbf{P}_{\mathbf{G}}^{(R)}$, $\mathbf{w}^{(0)} = \mathbf{x}$, $\mathbf{v} = \mathbf{w}^{(R)}$, and $\mathbf{H}$ is replaced with $\mathbf{G}$ in order to comply with the notation of the precorrection case that has been used so far. The respective system is shown in Fig. 4.5. A consequence of the commutativity is that for the postcorrection and precorrection case in (4.20) and (4.21), respectively, identical corrected output vectors are obtained for identical transition matrices $\mathbf{G}$ and $\mathbf{H}$ when performing the same number of iterations. Under these conditions, the Richardson iteration achieves the same correction performance for both the postcorrection and precorrection case. Due to this commutativity, results that have been presented for the postcorrection case regarding reconstruction performance, conditions of convergence and computational complexity can be directly applied to the precorrection case [26–29]. Moreover, it is interesting to note that the mentioned permutation is also possible when utilizing a correction system based on the modified Richardson iteration, where the result of the convolution sum in (4.17) is multiplied by an additional time-invariant factor μ [28].

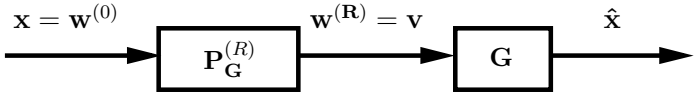


Figure 4.5 Richardson iteration of order R precorrecting the weakly time-varying system $\mathbf{G}$.

4.3.2 Correction Structures

In order to determine a causal implementation of the line-wise Richardson iteration in (4.17), we approximate the characteristics of the acausal time-varying impulse response $h_n[k]$ by the causal impulse response $h_n^{\mathrm{c}}[k]$. To this end, the DTFT of $h_n[k]$ is calculated as

$$H_n(e^{j\omega}) = \sum_{k=-\infty}^{\infty} h_n[k] e^{-j\omega k}. \tag{4.22}$$

The causal representation of the time-varying system is obtained by performing an approximation of $h_n[k]$ by minimizing the approximation error in the frequency domain

$$E_n(e^{j\omega}) = H_n(e^{j\omega}) - H_n^{\mathrm{a}}(e^{j\omega}) \tag{4.23}$$

according to

$$\min ||E_n(e^{j\omega})||_{\mathrm{norm}} \quad \text{for } \omega \in \omega_D \tag{4.24}$$

within a desired design bandwidth ω_D for a given norm. Moreover, the DTFT $H_n^a(e^{j\omega})$ of the resulting time-varying filter is delayed by D samples to obtain the causal system

$$H_n^c(e^{j\omega}) = H_n^a(e^{j\omega}) e^{-j\omega D}. \tag{4.25}$$

For a more general case, where the undesired time-varying impulse response is $\breve{h}_n[k] = h_n[k - k_0]$ and k_0 is some delay, we can use the relation

$$h_n[k] = \breve{h}_n[k + k_0] \tag{4.26}$$

to approximate $\breve{h}_n[k]$ by utilizing (4.22)-(4.25) as well.

Applying the obtained causal filter $h_n^c[k]$ in the recurrence relation as given in (4.17), a single correction stage as depicted in Fig. 4.6 may be derived. Replicating this correction stage R times, an overall correction structure is realized by a cascade of R stages as proposed in [26]. The implementation of the overall postcorrection structure is shown in Fig. 4.7. The adaptation of each filter $H_n^{\mathrm{c}}(e^{j\omega})$ is performed by delaying each filter's time index n by k_0 in order to take the delay of the undesired time-varying system into account. In addition to this delay, the time indices are adapted according to the delay introduced

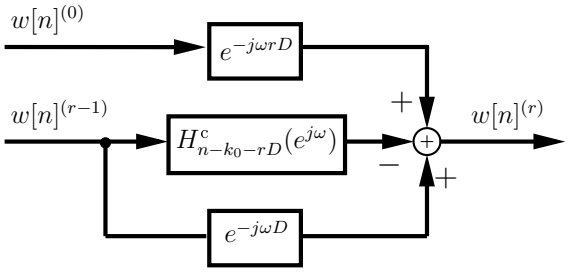


Figure 4.6 Causal representation of the recurrence relation.

by the cascade of filters preceding the respective filter, which accumulates to rD for the r$^{\mathrm{th}}$ correction stage. Various implementations of these time-varying filters would be feasible, e.g., via a look-up table or by means of the Farrow structure [37].

Using the established commutativity of the Richardson iteration, an implementation of the precorrection structure as illustrated in Fig. 4.8 is derived. In Fig. 4.8, the first precorrection filter is updated with $G^{\mathrm{c}}_{n+k_0+RD}(e^{j\omega})$. Therefore, the overall precorrection structure requires knowledge of at least $k_0 + RD$ states of the undesired time-varying system prior to the time-instant n. The reconstruction performance of the postcorrection and precorrection structure can be ensured by the lower bound of the SNR improvement presented in [26] and [29] for a given number of correction stages.

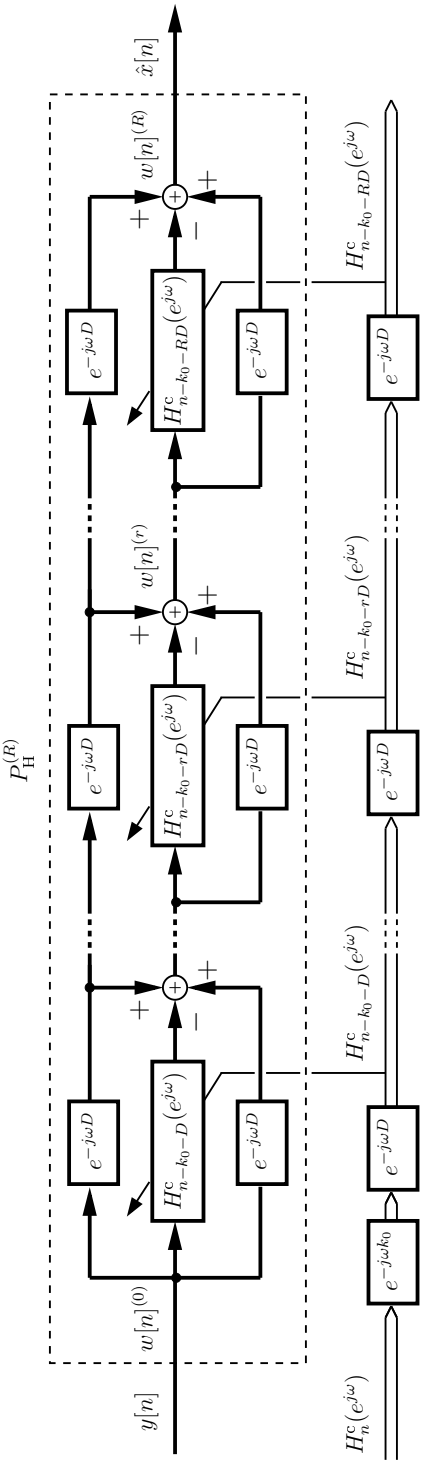


Figure 4.7 Richardson postcorrection structure of order R compensating for the preceding acausal time-varying system $H_n(e^{j\omega})$.

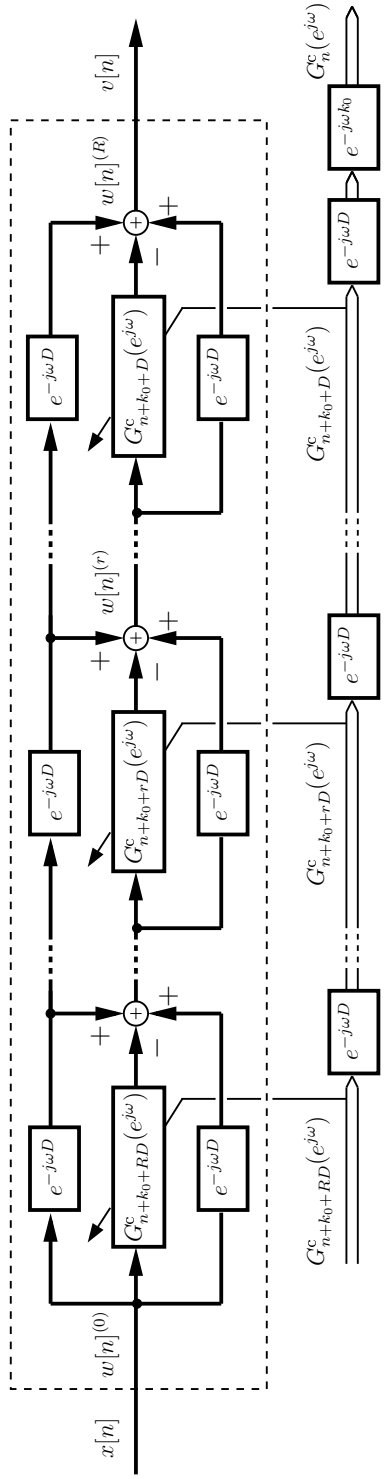


Figure 4.8 Richardson precorrection structure of order R compensating for the subsequent acausal time-varying system $G_n(e^{j\omega})$.

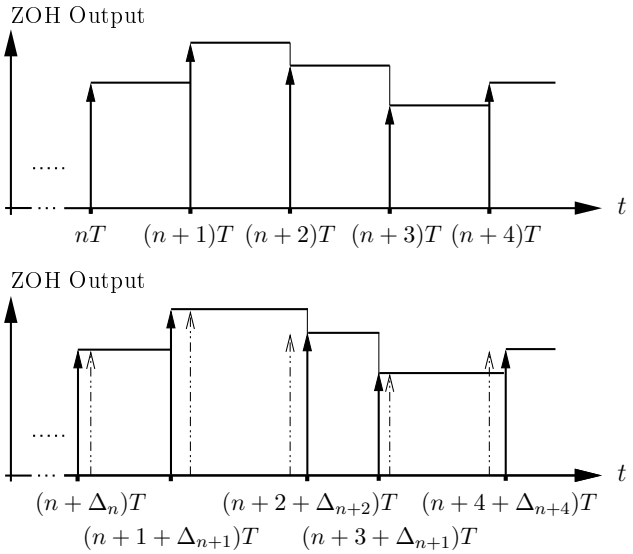


Figure 4.9 (top) Uniform and (bottom) non-uniform ZOH signals.

4.4 Application Example

In the following, we discuss the precorrection of mixed-signal systems by investigating the correction of non-uniform ZOH devices in a DAC. This scenario complements the research on postcorrecting the time-varying behavior of TIADCs [26, 28, 30, 31, 62]. In [23], a framework for describing M-periodically non-uniform SH signals has been introduced and the compensation of 2-periodic ZOH signals has been shown. By using the introduced framework for weakly time-varying systems, we extend the description to non-periodic SH signals and demonstrate the precorrection for non-periodic ZOH signals.

4.4.1 Problem statement

In addition to the well-known shaping of the output spectrum, non-uniform ZOHs induce spurious images in the output spectrum as has been shown in [23, 54]. These undesired spectral components are caused by variation of the ideal sampling instants of period T by a time-varying delay $\Delta_n T$. This delay is in turn a consequence of the clock source that is affected by non-idealities, e.g, jitter and clock skew. Fig. 4.9 (top) depicts uniform ZOH signals representing the ideal case, whereas Fig. 4.9 (bottom) illustrates how the time-varying delay $\Delta_n T$ results in non-uniform ZOH signals. This time-varying delay is caused by jitter in the clock signal inducing a deviation from the ideal sampling instants nT. In order to show the impact non-uniform ZOH signals have on the performance of the overall system, numerical simulations were performed in Matlab. To this end, the time-varying

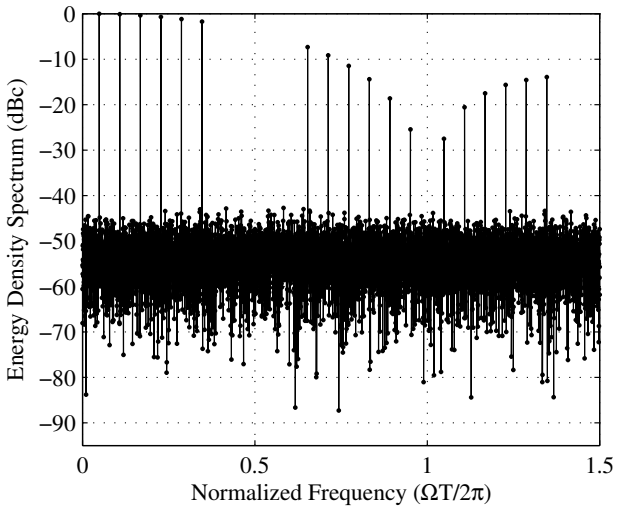


Figure 4.10 Spectrum of the non-uniform ZOH output signals.

delays Δ_n were selected according to a zero mean Gaussian distribution with a standard deviation of $\sigma_\Delta = 0.0391$. A total of 2^{12} samples were drawn from this distribution to characterize the ZOH behavior. As an input signal, we employed a coherently sampled multitone signal constituted by six equally spaced fundamental tones with amplitudes of one and random phases. The fundamental frequencies of the tones were located at $\omega_0 = [0.098,\ 0.217,\ 0.336,\ 0.455,\ 0.574,\ 0.693]\ \pi$. In order to assess the correction performance, the SNR was calculated according to (3.61), where D_s represents the overall delay of the system, cf. (4.14), and N specifies the number of considered samples. The output spectrum of the non-uniform ZOH signals is shown in Fig. 4.10. The random variations of the time shifts result in a multitude of undesired spectral components creating a floor of tonal components located at approximately -45 dBc, considerably degrading the SNR of the analog output signal. In accordance with Fig. 4.9, the output of a non-uniform ZOH can be described by the time-varying impulse responses $\tilde{a}_n(t)$ as [23]

$$\tilde{a}_n(t) = T\left(u(t - \Delta_n T) - u(t - T - \Delta_{n+1} T)\right) \tag{4.27}$$

with $\Delta_n T < \Delta_{n+1} T + T$ and where $u(t)$ denotes the unit step function. The scaling factor T is used to simplify the derivation of the discrete-time model as shown in the next section.

4.4.2 System Modeling and Precorrection structure

As presented in [23], the output of a non-uniform ZOH $\hat{x}(t)$ including an ideal low-pass filter $h_{\mathrm{id}}(t)$ may be described as

$$\hat{x}(t) = \sum_{k=-\infty}^{\infty} v[k]\tilde{a}_k(t - kT) * h_{\mathrm{id}}(t) \tag{4.28}$$

where the CTFT of $h_{\mathrm{id}}(t)$ is defined as

$$H_{\mathrm{id}}(j\Omega) = \begin{cases} T & \text{for } |\Omega| < \frac{\hat{\omega}_D}{T} \leq \frac{\pi}{T} \\ 0 & \text{for } |\Omega| \geq \frac{\hat{\omega}_D}{T}. \end{cases}$$

By specifying a discrete-time filter as [44]

$$A_n(e^{j\omega}) = \tilde{A}_n\left(j\frac{\omega}{T}\right) H_{\mathrm{id}}\left(j\frac{\omega}{T}\right) \quad \text{for} - \pi \leq \omega < \pi, \tag{4.29}$$

we can find an equivalent representation that produces with

$$\hat{x}[n] = \sum_{k=-\infty}^{\infty} v[k]a_k[n - k] \tag{4.30}$$

where $a_n[k]$ is the inverse DTFT of $A_n(e^{j\omega})$, and

$$\hat{x}(t) = \sum_{n=-\infty}^{\infty} \hat{x}[n]h_{\mathrm{id}}(t - nT) \tag{4.31}$$

the same output as the continuous-time system in (4.28). Changing the indices of the discrete-time model of the time-varying ZOH in (4.30) to

$$\hat{x}[n] = \sum_{k=-\infty}^{\infty} a_{n-k}[k]v[n - k] \tag{4.32}$$

and comparing it to the time-varying system in (4.7), which has been used to derive our precorrection structure, reveals that the filter $g_n[k]$ is given by

$$g_n[k] = a_{n-k}[k]. \tag{4.33}$$

A possible implementation of the precorrection structure has been presented in Section 4.3.2. As mentioned in Section 4.3.2, the precorrection structure requires knowledge of RD future jitter values due to the delay induced by the causal correction filter preceeding the ZOH. In contrast to [23], the correction structure developed in this section is not restricted to M-periodically time-varying impulse responses but valid for non-periodically time-varying systems. Although we used a particular ZOH model as defined in (4.27), we can apply the same framework to arbitrary linear weakly time-varying bandlimited analog reconstruction filter.

4.4.3 Convergence

The convergence of the correction process can be ensured by evaluating each impulse response $g_n[k]$ according to the criterion in (4.6). For the given application, a more convenient criterion can be obtained by determining the bounds of the parameters Δ_n, which control the time-varying behavior of $g_n[k]$, with respect to the convergence criterion.

Rewriting the convergence criterion in (4.6) with (4.33), we obtain

$$\sum_{k=-\infty}^{\infty} |\delta[k] - a_{n-k}[k]| < 1 \tag{4.34}$$

where the terms $|a_{n-k}[k]|$ depend on time instants other than the current time instant n. By bounding the time-varying parameter Δ_n according to

$$|\Delta_n| < \epsilon, \tag{4.35}$$

the convergence criterion in (4.34) can be exploited as

$$\sum_{k=-\infty}^{\infty} \max_{|\Delta_n|<\epsilon} |\delta[k] - a_{n-k}[k]| < 1 \tag{4.36}$$

to find ϵ. The maximization in (4.36) gives a conservative convergence bound but can be easily evaluated resulting in a bound of $\epsilon = 0.047$ for the given application, where the utilized impulse responses are highly oversampled instances of (4.29) and an overall number of 2^{12} samples were utilized. Additionally, the reconstruction performance of a multistage scheme ($R = 6$) precorrecting the input signal, as specified in Section 4.4.1, was investigated by means of Monte Carlo simulations, achieving a bound of $\epsilon = 0.5$ which represents the overall viable parameter space according to the constraint of (4.27). Thus, it can be concluded that the application specific convergence criterion in (4.36) offers a signal-independent but conservative convergence bound compared to the simulation-based results.

4.4.4 FIR Filter Design

The precorrection structure in Fig. 4.8 is used to preprocess the input of the non-uniform ZOH model as specified in Section 4.4.1 with respect to the simulation-based convergence results presented in Section 4.4.3. As a consequence, the spectral purity of the output signal within the bandwidth $\hat{\omega}_D$ is enhanced. The utilized filters are designed by approximating the discrete-time frequency response of the ZOH

$$A_n(e^{j\omega}) = e^{-j\omega\frac{1}{2}} \frac{\sin\left(\frac{\omega}{2}(1 + \Delta_{n+1} - \Delta_n)\right)}{\frac{\omega}{2}} e^{-j\frac{\omega}{2}(\Delta_{n+1}+\Delta_n)}. \tag{4.37}$$

which is obtained following the reasoning in Section 4.4.2. The delay term $e^{-j\omega\frac{1}{2}}$ does not introduce any distortions and can be neglected to simplify the precorrection task. The resulting discrete-time frequency response of the ZOH is approximated by $A_n^{\mathrm{a}}(e^{j\omega})$ using the framework outlined in Section 4.3.2 to obtain the frequency response

$$A_n^{\mathrm{c}}(e^{j\omega}) = \sum_{k=0}^{K} a_n^{\mathrm{c}}[k] e^{-j\omega k} \tag{4.38}$$

where $a_n^{\mathrm{c}}[k]$ indicates the coefficients and K the order of the causal FIR filter. Using the relationship in (4.33), the time-varying FIR filter $g_n^{\mathrm{c}}[k]$ is obtained which can be employed in the correction structure. By designing $g_n^{\mathrm{c}}[k]$ and updating the correction filter coefficients as described in Section 4.3.2 on a sample-by-sample basis, the precorrection structure is continuously adapted with respect to the time-varying behavior of the non-uniform ZOH signals.

To verify the proposed structure, we designed filters of order $K = [12,\ 24,\ 36]$, delay $D = 6$ within the band $\omega_D = [0,\ 0.75\pi]$ according to the L_2 norm utilizing the Matlab software CVX for solving the convex optimization problem [18]. Furthermore, the input signal as defined in Section 4.4.1 was used for this simulation.

The output spectrum of the non-uniform ZOH precorrected by three correction stages employing filters of order 12 is shown in Fig. 4.11. The undesired tones within the bandwidth $\hat{\omega}_D = 0.75\pi$ are reduced considerably, and the initial SNR of the output bandlimited to $\hat{\omega}_D$ is enhanced from an initial value of 19.19 dB for the uncorrected case to 62.78 dB via the proposed processing. The achieved SNR performance for an increasing number of precorrection stages and different filter orders are shown in Fig. 4.12. The SNR performance is increased in an approximately linear manner up to a SNR value of about 75.9 dB for four correction stages when filters of order 12 are employed in the correction structure. A maximum SNR value of 80 dB can be achieved by appending additional correction stages. This SNR value can be enhanced when filters of higher order are utilized as exemplified by the SNR performance of reconstruction structures using filters of order 14 and 16, respectively.

Though the obtained improvement of the figures-of-merit are substantial, the presented time-varying filter design suffers from two main drawbacks. The solution of an optimization problem at each instant n is computationally intensive and the set of previously approximated channel responses must be stored and continuously updated in the memory. These two problems will be addressed in the following section.

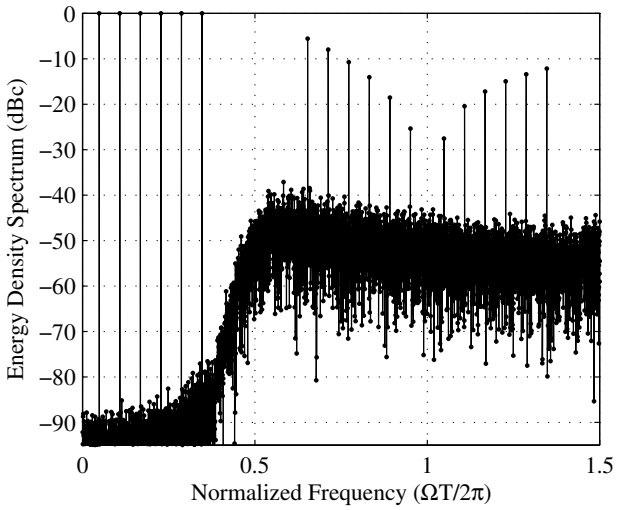


Figure 4.11 Spectrum of the non-uniform ZOH output signals, using the proposed precorrection constituted by three correction stages ($R = 3$).

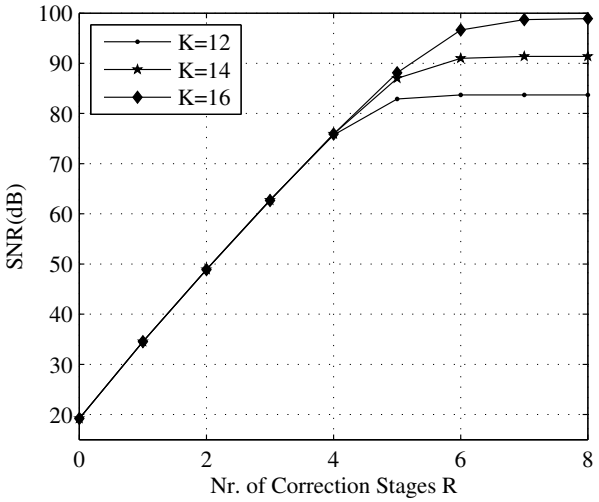


Figure 4.12 SNR performance over the number of correction stages for different filter orders employing Modified Farrow filters.

4.4.5 Farrow Filter Design

To significantly reduce the complexity of an online FIR filter design, a tunable precorrection structure will be devised to adapt to the non-uniform ZOH behavior. This structure can be designed offline and is adapted online by updating a few time-varying multipliers only. To this end, we approximate the ZOH frequency response $A_n(e^{j\omega})$ in terms of two parallel Farrow structures [37] which will be modified to obtain an online tunable time-varying filter that can be employed in the precorrection structure.

The frequency response of a ZOH given in (3.50) can be written as

$$A_n(j\Omega) = e^{-j\Omega T\frac{1}{2}}\frac{1}{j\Omega T}\left(e^{-j\Omega T(-\frac{1}{2}+\Delta_n)} - e^{j\Omega T(-\frac{1}{2}-\Delta_{n+1})}\right). \tag{4.39}$$

As in Section 4.4.4, the delay term $e^{-j\Omega T\frac{1}{2}}$ is neglected to simplify the precorrection task, and by using the relation in (4.29), we can write

$$A_n(e^{j\omega}) = \hat{A}(e^{j\omega}, \Delta_n) + \hat{A}^*(e^{j\omega}, -\Delta_{n+1}) \tag{4.40}$$

where $(.)^*$ denotes the conjugate complex value and $\hat{A}(e^{j\omega}, \lambda)$ is a prototype filter given by

$$\hat{A}(e^{j\omega}, \lambda) = \frac{1}{j\omega}e^{-j\omega(-\frac{1}{2}+\lambda)}. \tag{4.41}$$

Since $\hat{A}(e^{j\omega}, \lambda)$ is an univariate function in λ, it can be approximated in terms of a polynomial in λ by defining the error of the polynomial approximation as

$$E(e^{j\omega}, \lambda) = \hat{A}(e^{j\omega}, \lambda) - \hat{A}^{\mathrm{a}}(e^{j\omega}, \lambda) \tag{4.42}$$

and by solving the optimization problem

$$\min ||E(e^{j\omega}, \lambda)||_2 \tag{4.43}$$

according to the L_2 norm for $\omega \in \omega_D$ and $\lambda \in \lambda_D$. After delaying the approximation result by D samples, the causal Farrow filter is obtained as

$$\hat{A}^{\mathrm{c}}(e^{j\omega}, \lambda) = \sum_{k=0}^{K} \hat{a}^{\mathrm{c}}[k, \lambda]e^{-j\omega k} \tag{4.44}$$

with

$$\hat{a}^{\mathrm{c}}[k, \lambda] = \sum_{l=0}^{L} b_l[k]\ \lambda^l. \tag{4.45}$$

The resulting Farrow filter is constituted by a set of $L+1$ filters of FIR type that define the impulse response $\hat{a}^{\mathrm{c}}[k, \lambda]$. Each subfilter is of order K and $b_l[k]$ indicates the k^{th} coefficient of the l^{th} subfilter.

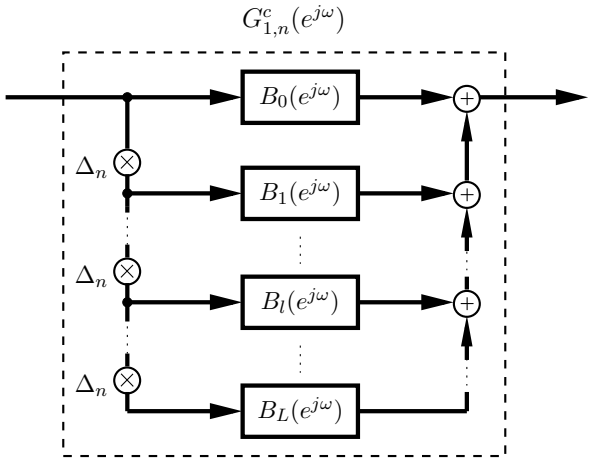


Figure 4.13 Modified Farrow filter.

The inverse discrete-time Fourier transform of (4.40) results with (4.45) in

$$\hat{a}^{\mathrm{c}}_n[k] = \hat{a}^{\mathrm{c}}[k, \Delta_n] + \hat{a}^{\mathrm{c}}[-k, -\Delta_{n+1}]. \tag{4.46}$$

Applying the relationship in (4.33) to the parallel Farrow filter in (4.46), the modified parallel Farrow filter, representing a single correction filter, is obtained as

$$g^{\mathrm{c}}_n[k] = g^{\mathrm{c}}_{1,n}[k] + g^{\mathrm{c}}_{2,n}[k] \tag{4.47}$$

where

$$g^{\mathrm{c}}_{1,n}[k] = \hat{a}^{\mathrm{c}}[k, \Delta_{n-k}] \tag{4.48}$$

$$g^{\mathrm{c}}_{2,n}[k] = \hat{a}^{\mathrm{c}}[-k, -\Delta_{n-k+1}]. \tag{4.49}$$

Rewriting the modified Farrow filter $g^{\mathrm{c}}_{1,n}[k]$ in terms of the FIR coefficients as

$$g^{\mathrm{c}}_{1,n}[k] = \sum_{l=0}^{L} b_l[k]\ \Delta^l_{n-k} \tag{4.50}$$

reveals its time dependency on previous values of Δ_n for all but the first FIR coefficient. Therefore, it presents a cascaded time-varying system as given in (4.9) with $g_n[k] = b_l[k]$ and $h_n[k] = \Delta^l_n \delta[k]$. Thus, by introducing (4.33) the initial order of the Farrow filter is reversed and a modified Farrow filter is obtained as illustrated in Fig. 4.13. A detailed derivation is presented in Section A.4. The same principles can be applied to the modified Farrow filter $g^{\mathrm{a}}_{2,n}[k]$. In order to verify the viability of the proposed correction scheme, the numerical simulations of Section 4.4.4 are repeated. However, the parallel modified

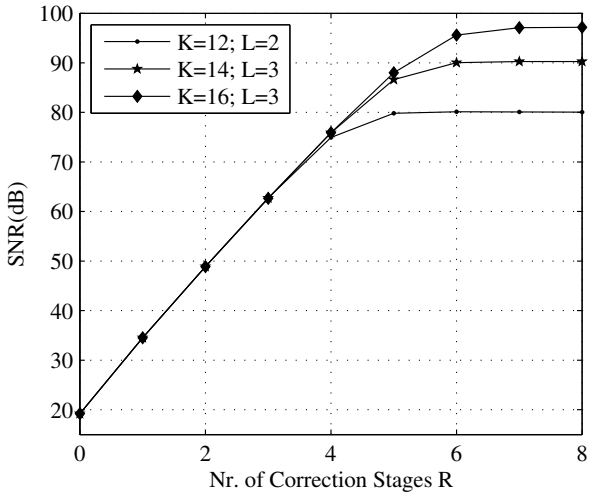


Figure 4.14 SNR performance over the number of correction stages for different filter orders.

Farrow filter is employed in each stage of the correction structure. For these simulations, the same settings, input signal, number of correction stages and realizations of Δ were used as in the previous section. To provide a meaningful comparison with the online filter design, the two modified Farrow filters were approximated according to the L_2 norm with a filter order $K = 12$, which agrees with the previous design. The polynomial order of the prototype Farrow filter was selected as $L = 2$ with a design band of $\omega_D = [0.1\pi,\ 0.75\pi]$ and a parameter vector of $\lambda_D = [-0.08,\ 0.08]$.

The obtained output spectrum of the precorrected DAC is depicted in Fig. 4.11. Again, the distortions in the band $|\omega| < \hat{\omega}_D$ with $\hat{\omega}_D = 0.75\pi$ are reduced considerably. The obtained result and the achieved SNR performance over the number of precorrection stages for different filter order, as shown in Fig. 4.14, compares favorably with the results of the previous section. A similar reconstruction performance compared to case presented in Section 4.4.4 can be observed. The only notable difference is constituted by the increase of the number of Farrow subfilters required to meet an enhanced maximum SNR value for a designed reconstruction system employing filters of higher order [31].

4.5 Concluding Remarks

In this chapter, structures were presented which allow for the precorrection and postcorrection of linear weakly time-varying systems. These structures were derived from the Richardson iteration while the exploration of other stationary iterative methods enabling

the precorrection of time-varying systems was beyond the scope of this chapter. The viability of the precorrection structure was demonstrated by the preprocessing of non-uniform ZOH signals in a DAC. To this end, a correction structure employing modified Farrow filters was presented which can be updated online by tuning a few time-varying multipliers only and considerably enhances the performance of the overall system. Employing more efficient Farrow implementations, e.g., in [65], and Farrow filters with an unequal number of subfilters could be a promising approach to create more computationally efficient structures.

5

Correction of Weakly Time-Varying Volterra Systems

5.1 Introduction

Linear systems are utilized in various fields of engineering. However, for specific applications in the field of communications theory, propagation theory and mixed-signal processing, the capabilities of linear systems might not suffice to adequately represent the behavior of blocks in the signal processing chain [66]. In these applications, an inherent nonlinear behavior of a system block, e.g., of an amplifier or SH device [67], requires nonlinear signal processing [68, 69]. Depending on the type of nonlinear system that is encountered, different mathematical representations could be employed. Nonlinear order statistics, homomorphic, morphological filters and filters based on the Volterra series have been successfully applied in image and seismic signal precessing and the modeling of biological systems [66]. In this chapter, the focus will be on Volterra systems which are able to characterize nonlinear systems with fading memory [70]. Volterra systems and specific types of Volterra systems, i.e., the Hammerstein and Wiener system, have been proposed in order to improve the performance of nonlinear systems in various applications. The enhancement of analog integrated circuits, the digital predistortion of radio frequency (RF) amplifiers and the modeling of SH circuits are selected examples, where an undesired behavior could be mitigated by modeling and correcting the respective nonlinear system [71–73]. In the correction case, it is desired to compensate the impact a Volterra system has on a signal, and the inverse characteristic of the Volterra system is required to process the affected signal [74]. In this scenario, the Volterra system will be referred to as the *undesired Volterra system* to distinguish this system from correction structures that comprise or are constituted by Volterra systems themselves, e.g., in [75]. Not all Volterra systems possess an inverse [74], and for this reason, the theory of the P^{th}-order inverse

was proposed in [74]. Cascading this P^{th}-order inverse with the undesired Volterra system results in an overall Volterra system whose 1^{st}-order up to the P^{th}-order components of the Volterra kernel are canceled. The residual Volterra kernels of the overall system are not extinguished, and as a consequence, the P^{th}-order inverse as presented in [74] does not provide a unique solution since other methods do exist which yield different residual kernels, while still canceling the 1^{st}-order up to the P^{th}-order components of the Volterra kernel. This degree of freedom was exploited in [75] to present recursive methods for synthesizing P^{th}-order inverses which have a lower implementation complexity and are easier to derive. In this recursive method, a truncated instance of the undesired Volterra system was employed, whose order of the Volterra series is gradually increased while the order of the P^{th}-order inverse that is synthesized is gradually increased. Derivations of this methods were reported in [55, 76] which do not alter the order of the Volterra systems employed in the recursive synthesis of the P^{th}-order inverse. In [55], the fixed point framework is employed to investigate the theory on P^{th}-order inverses and their convergence behavior. In this chapter, the representation of time-varying Volterra systems and their correction in terms of a time-varying P^{th}-order inverse will be explored. Furthermore, a correction of time-varying Volterra systems that is based on fixed point iteration methods is presented [77]. The presented work serves as an outlook for further research.

5.2 Volterra Systems

5.2.1 Input/Output Relations

The output $y[n]$ of a nonlinear system can be expressed as the Volterra series expansion for bounded input bounded output (BIBO) stable systems with a time-varying Volterra kernel [78, 79], i.e,

$$\begin{aligned} y[n] &= H_n\left\{x[n]\right\} \\ &= \sum_{p=1}^{\infty} y_p[n] \end{aligned} \tag{5.1}$$

where $H_n\left\{x[n]\right\}$ indicates the time-varying Volterra system $H_n\{.\}$ of infinite order applied to the input signal $x[n]$. The intermediate signals in (5.1) are defined as

$$y_p[n] = H_{p,n}\left\{x[n]\right\}. \tag{5.2}$$

where $H_{p,n}\{.\}$ represents the *p^{th}-order operator* of the Volterra system $H_n\{.\}$. The 0^{th}-order operator in (5.1) is omitted since the correction of the resulting time-varying offset added to the signal is straightforward. The 1^{st}-order operator will also be referred to as

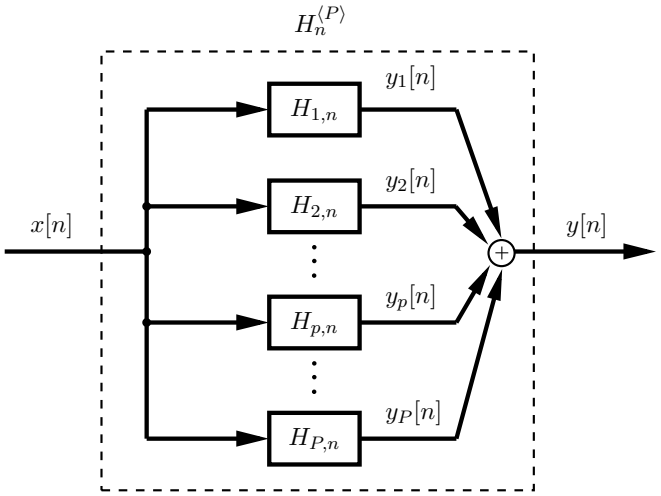


Figure 5.1 Volterra system $H_n^{\langle P \rangle}\{.\}$ that processes the input signal $x[n]$ and results in the output signal $y[n]$. The overall Volterra system generates P intermediate signals $y_p[n]$.

the *linear component* and the p$^{\text{th}}$-order operators for $p > 1$ as the *nonlinear components* of the Volterra system.

The p$^{\text{th}}$-order operators are obtained as the convolution of the input signal $x[n]$ with the respective time-varying multivariate function $h_{p,n}[k_1, k_2 \dots k_p]$ [78, 79], i.e.,

$$H_{p,n}\{x[n]\} = \sum_{k_1=0}^{\infty} \sum_{k_2=0}^{\infty} \cdots \sum_{k_p=0}^{\infty} h_{p,n}[k_1, k_2, \dots, k_p] x[n-k_1] x[n-k_2] \dots x[n-k_p]. \quad (5.3)$$

The P$^{\text{th}}$-order truncation of the Volterra system $H_n\{.\}$ results in a time-varying Volterra system of order P [74], its time-varying formulation is obtained as

$$\begin{aligned} y[n] &= H_n^{\langle P \rangle}\{x[n]\} \\ &= \sum_{p=1}^{P} y_p[n]. \end{aligned} \quad (5.4)$$

The signal flow, when employing the Volterra system $H_n^{\langle P \rangle}\{.\}$ to process the signal $x[n]$, is illustrated in Fig. 5.1, and the relationship between the Volterra system $H_n^{\langle P \rangle}\{.\}$ and its time-varying coefficients $h_{p,n}[k_1, k_2, \dots, k_p]$ of the p$^{\text{th}}$-order Volterra operator $H_{p,n}\{.\}$ in (5.3) is exemplified in the Appendix (see Fig. A.3).

5.2.2 Cascade of Time-Varying Volterra Systems

A representation for a cascade of a Volterra system $H_n\{.\}$ that is postprocessed by a second Volterra system $G_n^{\langle P \rangle}\{.\}$ of order P can be obtained by following the approach in

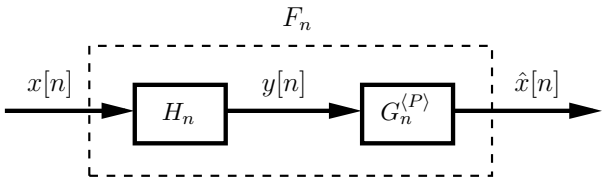


Figure 5.2 Cascade of two Volterra systems, where the first system is of infinite order and the second system is of order P.

[74] and extending it to time-varying Volterra systems. The output of the second Volterra system is given as

$$\hat{x}[n] = G_n^{\langle P \rangle} \{y[n]\} \tag{5.5}$$

and rewritten with (5.1) as

$$\hat{x}[n] = F_n \{x[n]\} \tag{5.6}$$

where

$$F_n \{x[n]\} = G_n^{\langle P \rangle} \{H_n \{x[n]\}\}. \tag{5.7}$$

The overall cascaded system is illustrated in Fig. 5.2.

Using (5.6), (5.7) and (5.4), the Volterra series expansion of the cascade in (5.6) can be rewritten as

$$\hat{x}[n] = \sum_{p=1}^{P} G_{p,n} \{H_n \{x[n]\}\}. \tag{5.8}$$

By utilizing the definition of the intermediate signals in (5.1), we obtain

$$\hat{x}[n] = \sum_{p=1}^{P} G_{p,n} \left\{ \sum_{p=1}^{\infty} y_p[n] \right\} \tag{5.9}$$

which describes the input/output relationship of a time-varying Volterra system cascaded with a subsequent time-varying Volterra system of order P. An equivalent representation of the cascaded system is obtained with

$$\hat{x}[n] = \sum_{p=1}^{\infty} F_{p,n}\left\{x[n]\right\}. \tag{5.10}$$

In order to relate (5.9) and (5.10), a scaled input signal $c\,x[n]$, where c is a time-invariant coefficient, is applied to both representations. The definition of the p$^{\text{th}}$-order operator in (5.3) allows the factoring out of the time-invariant coefficient c, cf., (5.3), and the two representations of the cascaded system in (5.9) and (5.10) can be rewritten as

$$\begin{aligned}\hat{x}[n] &= \sum_{p=1}^{\infty} F_{p,n}\left\{c\;x[n]\right\} \\ &= \sum_{p=1}^{\infty} c^p F_{p,n}\left\{x[n]\right\}\end{aligned} \tag{5.11}$$

and

$$\hat{x}[n] = \sum_{p=1}^{P} G_{p,n}\left\{\sum_{p=1}^{\infty} c^p y_p[n]\right\}. \tag{5.12}$$

Employing the definition of the p$^{\text{th}}$-order operator presented in (5.3) to the relation in (5.12), the result can be reordered and is rewritten as

$$\hat{x}[n] = \sum_{p=1}^{P}\sum_{k_1=1}^{\infty}\sum_{k_2=1}^{\infty}\cdots\sum_{k_p=1}^{\infty} c^{k_1+k_2+\ldots k_p} G_{p,n}\left\{y_{k_1}[n], y_{k_2}[n], \ldots, y_{k_p}[n]\right\} \tag{5.13}$$

with the multivariate p$^{\text{th}}$-order operator

$$\begin{aligned}G_{p,n}\left\{y_{k_1}[n], y_{k_2}[n], \ldots, y_{k_p}[n]\right\} = \sum_{l_1=0}^{\infty}\sum_{l_2=0}^{\infty}\cdots\sum_{l_p=0}^{\infty} g_{p,n}[l_1, l_2, \ldots, l_P] \\ y_{k_1}[n-l_1] y_{k_2}[n-l_2]\ldots y_{k_P}[n-l_p].\end{aligned} \tag{5.14}$$

A comparison of the coefficients c regarding its exponents p in (5.11) and $k_1+k_2+\ldots k_p$ in (5.13), respectively, can be used to relate the operators $F_{p,n}\left\{x[n]\right\}$ to functions of $G_{p,n}\left\{y_{k_1}[n], y_{k_2}[n], \ldots, y_{k_p}[n]\right\}$ for $1 \le p \le P$.

The resulting first three operators of $F_{p,n}\left\{x[n]\right\}$ are listed in (5.15)-(5.17), where the employed multivariate operators are expanded in (5.18) utilizing the approach in [80].

$$F_{1,n}\{x[n]\} = G_{1,n}\{y_1[n]\} \tag{5.15}$$
$$F_{2,n}\{x[n]\} = G_{1,n}\{y_2[n]\} + G_{2,n}\{y_1[n]\} \tag{5.16}$$
$$F_{3,n}\{x[n]\} = G_{1,n}\{y_3[n]\} + G_{3,n}\{y_1[n]\} + 2G_{2,n}\{y_1[n], y_2[n]\} \tag{5.17}$$

with the expansions of the multivariate operators [80]

$$2G_{2,n}\{y_1[n], y_2[n]\} = G_{2,n}\{y_1[n] + y_2[n]\} - G_{2,n}\{y_1[n]\} - G_{2,n}\{y_2[n]\} \tag{5.18}$$

A cascade of two 2$^{\text{nd}}$-order Volterra systems is exemplified in the Appendix A.4.

5.2.3 P$^{\text{th}}$-Order Inverse

In the following, the P$^{\text{th}}$-order inverse for time-varying Volterra systems is derived. To this end, the strong assumption is made that there exists a time-varying linear operator $G_{1,n}\{.\}$ which facilitates the perfect correction of the linear component $H_{1,n}\{.\}$ of the undesired Volterra system, i.e.,

$$x[n] = G_{1,n}\{H_{1,n}\{x[n]\}\}. \tag{5.19}$$

This assumption will be relaxed in Section 5.2.5 where implementation aspects of the correction schemes are discussed. By applying the P$^{\text{th}}$-order inverse, it is desired to obtain a cascaded Volterra system $F_n\{x[n]\}$ with the p$^{\text{th}}$-order operators [74]

$$F_{p,n}\{x[n]\} = \begin{cases} x[n] & \text{for } p = 1 \\ 0 & \text{for } 1 < p \leq P. \end{cases} \tag{5.20}$$

The 1$^{\text{st}}$-order operator of the P$^{\text{th}}$-order inverse is obtained by rewriting (5.15) using (5.2), i.e.,

$$F_{1,n}\{x[n]\} = G_{1,n}\{H_{1,n}\{x[n]\}\}. \tag{5.21}$$

In order to meet the requirement in (5.20) for $p = 1$, $G_{1,n}\{.\}$ is designed to fulfill the assumption in (5.19). Furthermore, the 2$^{\text{nd}}$-order operator of the P$^{\text{th}}$-order inverse is obtained by rewriting (5.16) using (5.2) and (5.20) for $p = 2$, i.e.,

$$0 = G_{1,n}\{H_{2,n}\{x[n]\}\} + G_{2,n}\{H_{1,n}\{x[n]\}\}. \tag{5.22}$$

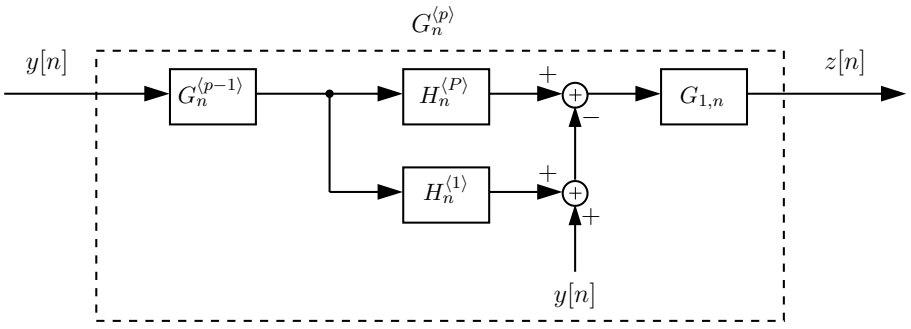


Figure 5.3 Recursive scheme for the composition of the P$^{\text{th}}$-order inverse. The order of the undesired Volterra system employed in the P$^{\text{th}}$-order inverse remains constant.

In order to use the relation in (5.20) for $p = 1$, the output of $H_{1,n}\{x[n]\}$ is postprocessed by $G_{1,n}\{.\}$ yielding

$$G_{2,n}\{x[n]\} = -G_{1,n}\{H_{2,n}\{G_{1,n}\{x[n]\}\}\}. \tag{5.23}$$

The higher order operators of the P$^{\text{th}}$-order inverse can be developed analogously.

In [75], the synthesis of the P$^{\text{th}}$-order inverse was proposed based on a recursive scheme. The derivation of the synthesis equation for time-varying Volterra systems can be derived analogously as in [75] and is given in recursive form as

$$G_n^{\langle p \rangle}\{y[n]\} = G_{1,n}\left\{ y[n] - H_n^{\langle p \rangle}\left\{G_n^{\langle p-1 \rangle}\{y[n]\}\right\} + H_{1,n}\left\{G_n^{\langle p-1 \rangle}\{y[n]\}\right\}\right\}. \tag{5.24}$$

Applying the recursive synthesis relation in (5.24), the order of the instance of the undesired Volterra system $H_n^{\langle p \rangle}\{.\}$ is gradually increased with increasing order of the P$^{\text{th}}$-order inverse, while the linear component of the undesired Volterra system is subtracted. Alternatively, an instance of the undesired Volterra system could be employed which comprises only 2$^{\text{nd}}$-order up to p$^{\text{th}}$-order operators.

A derivations of this recursive synthesis relation was reported in [55, 76] where the order of the Volterra systems employed in the recursive scheme remains constant with order P. The respective recursive synthesis relation is obtained by substituting $H_n^{\langle p \rangle}$ in (5.24) with $H_n^{\langle P \rangle}$. The resulting scheme is depicted in Fig. 5.3.

5.2.4 Fixed Point Iteration Based Correction

The synthesis of the P$^{\text{th}}$-order inverses, as presented in (5.2.3), involves $G_{1,n}\{.\}$ which cancels the linear component of the undesired Volterra system (cf., (5.24) and (5.21) with

(5.20)). In a scenario, where the linear component of the undesired Volterra system is non-periodically time-varying this is an unfavorable property in terms of the required design complexity of $G_{1,n}\{.\}$. To reduce this design complexity, correction schemes based on a fixed point iteration are proposed in the following. To this end, the perfect reconstruction of the signal $y[n]$ that is affected by the undesired Volterra system $H_n\{.\}$ can be represented as

$$\hat{x}[n] = H_n^{-1}\{y[n]\} \tag{5.25}$$

where the existence of $H_n^{-1}\{.\}$ is assumed. Applying a fixed point iteration, $\hat{x}[n]$ can be approximated iteratively using a fixed point iteration [63], to correct time-varying Volterra systems, i.e.,

$$w[n]^{(r)} = w[n]^{(r-1)} + w[n]^{(0)} - T_n^{\langle P\rangle}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\} \tag{5.26}$$

where $w[n]^{(0)} = y[n]$ and $T_n^{\langle P\rangle}\{.\}$ indicates a Volterra system of order P that is employed in the r$^{\text{th}}$ iteration. The principle scheme is illustrated in Fig. 5.4.

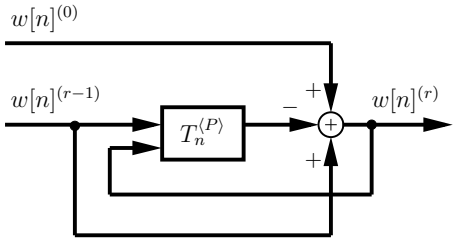


Figure 5.4 Proposed fixed point iteration for the correction of a Volterra system.

The relation in (5.26) can be employed to synthesize a recurrence relation for the successive over-relaxation, Richardson, Jakobi and Gauss-Seidel iteration [77]. However, in the following only correction structures for the nonlinear Richardson iteration and for a modified Gauss-Seidel iteration will be investigated as $G_{1,n}\{.\}$ will not be required in the resulting structures. To obtain the nonlinear Richardson iteration [63], an instance of the undesired Volterra system of order P can be utilized in (5.26) by applying the identity $T_n^{\langle P\rangle}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\} = H_n^{\langle P\rangle}\left\{w[n]^{(r-1)}\right\}$. For a different structure, the already corrected output samples $w[n]^{(r)}$ obtained for the current iteration can be used in the same iteration r in the nonlinear filtering operation. To this end, a recursive system is utilized by rewriting (5.26) using the identity $T_n^{\langle P\rangle}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\} = \tilde{H}_n^{\langle P\rangle}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\}$. The causal system can be defined as

$$\tilde{H}_n^{\langle P\rangle}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\} = \sum_{p=1}^{P} \tilde{H}_{p,n}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\} \tag{5.27}$$

with

$$\begin{aligned}\tilde{H}_{p,n}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\} = &\sum_{k_1=0}^{D}\sum_{k_2=0}^{D}\cdots\sum_{k_p=0}^{D} h_{p,n}[k_1, k_2, \ldots, k_P] \\ &w[n-k_1]^{(r-1)} w[n-k_2]^{(r-1)} \ldots w[n-k_p]^{(r-1)} \\ + &\sum_{l_1=D+1}^{\infty}\sum_{l_2=D+1}^{\infty}\cdots\sum_{l_p=D+1}^{\infty} h_{p,n}[l_1, l_2, \ldots, l_P] \\ &w[n-l_1]^{(r)} w[n-l_2]^{(r)} \ldots w[n-l_p]^{(r)}\end{aligned} \tag{5.28}$$

where D indicates the delay induced by the system.

5.2.5 Correction Structures

In [74, 75], the linear component of the undesired time-invariant Volterra system is assumed to be minimum phase which can be perfectly inverted [44]. An equivalent assumption for a time-varying scenario would be the perfect invertibility of time-varying systems fulfilling the relation presented in (5.19). However, in general the linear component of the undesired Volterra will not be perfectly invertible for time-variant and time-invariant scenarios, and thus an approximation of the inverse characteristics of the linear component is required. To facilitate this approximation, the first order operators $H_{1,n}\{.\}$ and $G_{1,n}\{.\}$ are rewritten in terms of the time-varying impulse responses $h_n[k]$ and $g_n[k]$, respectively, yielding

$$\begin{aligned}H_{1,n}\{x[n]\} &= \sum_{k=-\infty}^{\infty} h_n[k]x[n-k] \\ G_{1,n}\{x[n]\} &= \sum_{k=-\infty}^{\infty} g_n[k]x[n-k]\end{aligned} \tag{5.29}$$

where the cascaded linear component $F_{1,n}\{x[n]\}$, as shown in (5.21), is characterized by $f_n[k]$ as shown in (3.4). The explicit design of the time-varying postcorrection filter $g_n[k]$ is presented in Section 3.3, and to this end, the desired characteristic of the cascaded linear time-varying system is specified in terms of the desired DTFT of the cascade, i.e.,

$$D_n(e^{j\omega}) = e^{-j\omega(D_H+D_G)} \tag{5.30}$$

where D_H is the delay introduced by $H_{1,n}\{.\}$ and D_G specifies the delay introduced by $G_{1,n}\{.\}$. Applying the design framework for time-varying filters, the coefficients of $G_{1,n}\{.\}$ can be obtained by solving the optimization problem in (3.15) according to a given norm using (5.30). The imperfect approximation of the coefficients and a non zero delay induced by $H_{1,n}\{.\}$ and $G_{1,n}\{.\}$ relax the assumption in (5.19).

The implementation of the overall postcorrection structure obtained from the recursive synthesis relation in (5.24) is shown in Fig. 5.5 for the example of a 3^{rd}-order inverse. The adaptation of each block in the correction structure is performed by delaying its time index n with D_H and $(p-1)D_G$, where p indicates the order in the recursive relation in (5.24). This structure will be referred to as correction structure A for the remainder of this work. Correction structure B implements the fixed point iteration given in (5.26) with $T_n^{\langle P\rangle}\left\{w[n]^{(r)}, w[n]^{(r-1)}\right\} = H_n^{\langle P\rangle}\left\{w[n]^{(r-1)}\right\}$ and is illustrated in Fig. 5.6.

Applying the recursive system defined in (5.27) and (5.28) to the fixed point iteration in (5.26), correction structure C is obtained as presented in Fig. 5.7.

Figure 5.5 Correction structure A: The 3rd-order inverse of the undesired Volterra system $H_n\{.\}$ is depicted. Only the signals generated by the 2nd-order and 3rd-order operators contribute within the 3rd-order inverse, as the signal contribution of the linear component is excluded (subtracted).

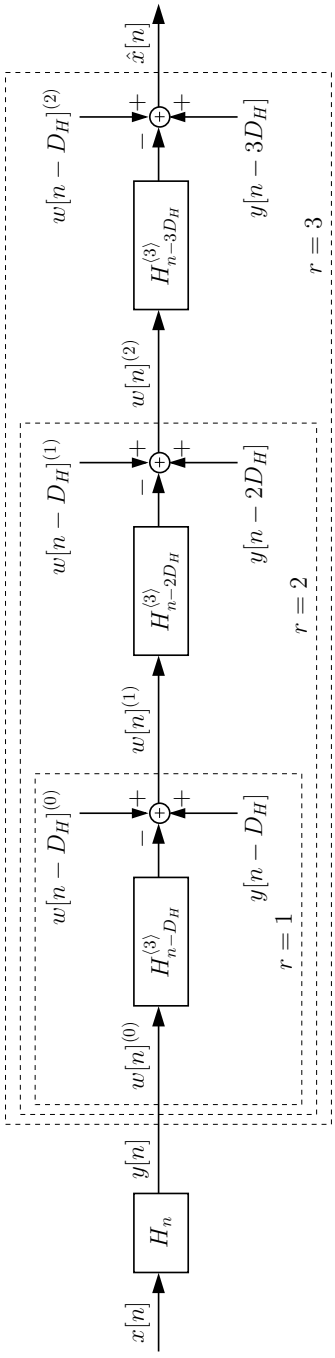


Figure 5.6 Correction structure B: An implementation of a fixed point iteration is depicted which corrects the undesired Volterra system $H_n\{.\}$ in three iterations. The instance of the Volterra system employed in the iteration is identical to the undesired Volterra system of order 3. The iteration index is indicated by r and the output of each iteration is $w[n]^{(r)}$.

Figure 5.7 Correction structure C: An implementation of a fixed point iteration employing recursive Volterra systems is depicted, which corrects the undesired Volterra system $H_n\{\cdot\}$ in three iterations. The output $w[n]^{(r)}$ of each iteration does not only serve as the input for the next iteration but serves also as a feedback to the recursive instance of the Volterra system.

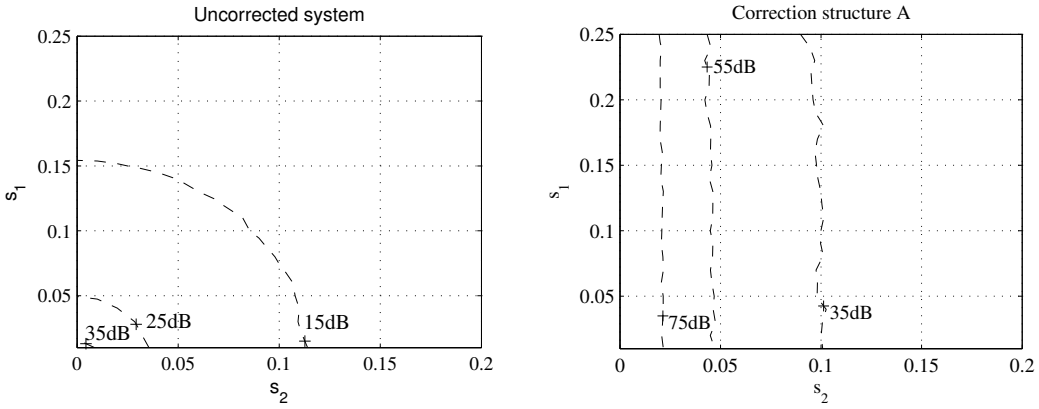
5.3 Correction Example

In the following example, the presented correction structures A, B and C are compared in terms of their performance correcting the linear and nonlinear components of a Volterra system. These correction structures are applied to postcorrect a Volterra system of order 3 with a memory depth of 4, comprising a weakly time-varying linear component. In order to consider the worst-case scenario in terms of the weakly time-varying criterion in (4.6), while still allowing for a variety of possible outcomes, the coefficients of the linear component $H_{1,n}\{.\}$ are obtained by

$$h_{1,n}[k] = \begin{cases} 1 & \text{for } k = D_H \\ c[k] & \text{otherwise} \end{cases} \tag{5.31}$$

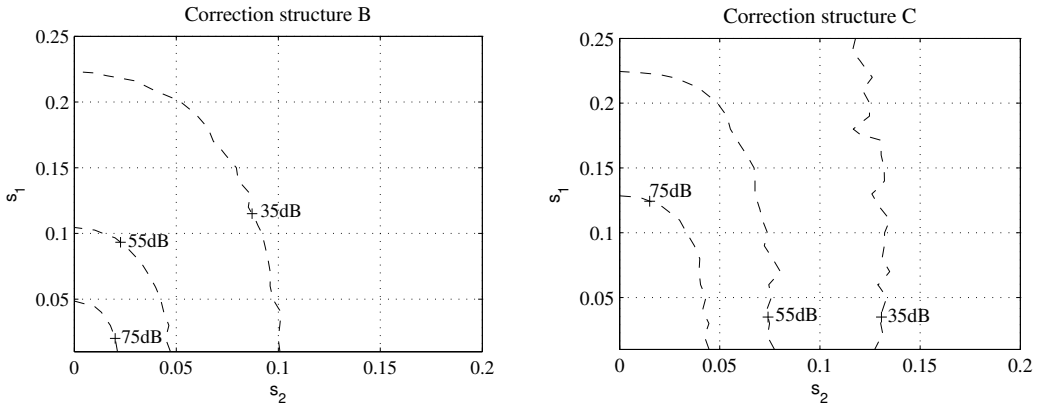
for $0 \leq k \leq 4$ and $D_H = 2$. The time-varying behavior of the coefficients in (5.31) is determined by the zero-mean uniform distribution $c \sim U(-s_1, s_1)$ with $0 \leq s_1 \leq \frac{1}{4}$ and $c[k]$ indicates a single sample drawn from this distribution. The coefficients of the higher order operators follow independent and identically distributed zero-mean uniform distributions, i.e., $h_{p,n} \sim U(-s_2, s_2)$ for $2 \leq p \leq P$. Thus, the parameters s_1 and s_2 control the magnitude of the time-varying behavior of the linear and nonlinear components of the Volterra system, respectively. To investigate a non-periodically time-varying system, a distinct set of coefficients was generated individually for each time instant n to characterize these components.

In order to provide a comparison of the different correction structures with similar implementation complexity, the number of employed Volterra systems in the correction structures is determined by correction structure A. Thus, two Volterra systems of order 3 are comprised in correction structure A, B and C. For correction structure B and C, this results in a two stage correction scheme. In order to minimize the error introduced by the imperfect approximation of $G_{1,n}\{.\}$ in correction structure A, the SNR performance of $G_{1,n}\{.\}$ was evaluated. A filter design of order 80, performed according to the L_2 norm, provided the maximum SNR value possible when using double precision arithmetic in Matlab for the utilized input signal. As an input signal $x[n]$, the bounded multitone signal as specified in Section 4.4.1 was used, however, the signal was coherently sampled to obtain 8192 samples. The contour lines of different SNR values for the uncorrected Volterra system are illustrated in Fig. 5.8(a). The ordinate and abscissa in this figure indicate the parameters s_1 and s_2 which control the magnitude of the time-varying behavior of the linear and nonlinear component, respectively. The contour lines for the SNR values 75, 55 and 35 dB that were achieved by the correction structures A, B and C are depicted in Fig. 5.8(b), Fig. 5.9(a) and Fig. 5.9(b), respectively.



(a) SNR performance of the uncorrection system (b) SNR performance of correction structure A

Figure 5.8 (a) Contour lines indicating the SNR performance of the uncorrected Volterra system and of (b) correction structure A: The ordinate indicates the magnitude of the linear component's time-varying behavior (s_1) and the abscissa indicates the magnitude of the nonlinear component's time-varying behavior (s_2).



(a) SNR performance of correction structure B (b) SNR performance of correction structure C

Figure 5.9 (a) Contour lines indicating the SNR performance of correction structure B and of (b) correction structure C: The ordinate indicates the magnitude of the linear component's time-varying behavior (s_1) and the abscissa indicates the magnitude of the nonlinear component's time-varying behavior (s_2). The 3 contour lines mark the SNR values 75, 55 and 35 dB.

In Fig. 5.8(b), the SNR performance of the correction structure A is independent of the variation of s_1. This independence reflects the ability of this structure to suppress signal contributions induced by the time-varying linear component of the undesired Volterra system. However, this advantage comes at the cost of a high design complexity of $G_{1,n}\{.\}$. In Fig. 5.9(a), the SNR performance of the correction structure B depends on both parameters s_1 and s_2. Furthermore, for a low magnitude of the time-varying behavior of the linear component, the SNR performance of this correction structure is identical to the performance of correction structure A, whereas the performance decreases with an increasing magnitude. This dependency results from the implicit iterative correction of the linear component $H_{1,n}\{.\}$ without resorting to a dedicated correction filter as in correction structure A. In Fig. 5.9(b), the SNR performance of the correction structure C is depicted which shows a similar behavior as the results of correction structure B, however, with an enhanced performance.

5.4 Future Research

In this chapter, the correction of time-varying Volterra systems with weakly time-varying linear components has been investigated as an outlook for future research. The investigated structures have been implemented based on the time-varying equivalent of the P^{th}-order inverse and on fixed point iterative methods. Their reconstruction performance has been compared for different magnitudes of the linear and the nonlinear behavior of a Volterra system.

A point for further research could be the comparison of the presented methods in terms of their convergence for time-varying scenarios. This work could include the proof of the optimality of a correction structure with respect to its implementation complexity and a possible extension of the convergence criterion in (4.6) to nonlinear systems. Furthermore, employing the presented correction structures in blind parameter identification schemes could also prove beneficial. Identification schemes have been presented which utilize an approximated and parameterized inverse of the system to facilitate the gradual adaptation of the identification algorithm, e.g., in [2]. Two beneficial properties of the iteration based correction structure could enhance the identification performance of those algorithms. First, the parameters modeling the undesired system are directly related to the inverse characteristic of the system since the approximated inverse is a function of the undesired system itself. Second, the correction performance in an identification algorithm could be easily improved by increasing the number of employed correction stages. As a consequence, the effect of an inaccurate correction on the identification performance could be mitigated.

6 Conclusion

In this thesis, low complexity methods and structures for the correction of time-varying systems have been presented. In Chapter 2, the mathematical representation of periodically time-varying linear systems as time-varying filters, polynomial filters, modulator banks and filter banks was established. Furthermore, the requirements for the equivalence of these representations were investigated. Thus, a correction scheme can be developed by using the most favorable framework, and the resulting scheme can be implemented choosing the most efficient representation.

In Chapter 3, the problem of correcting non-periodically time-varying linear systems was addressed. Depending on the application, this correction could either result in the compensation of the undesired time-varying linear system or in the synthesis of a desired time-varying behavior of the overall system. To this end, a postcorrection and precorrection method for the design of time-varying FIR correction filters were presented. Applying a different derivation, the derived postcorrection method resulted in the same cost function as was presented in [19]. These correction methods provide a powerful and versatile approach to correct non-periodically time-varying systems since the filter order and the filter design norm can be adapted for each time instant individually in order to meet the application's requirements. However, this filter design process relies on computationally intensive operations, e.g numerical integration or matrix operations [9]. In order to reduce the computational requirements, a low complexity filter design algorithm for the least-squares norm was derived which can be applied to postcorrection and precorrection systems, respectively. This algorithm exploits the time-varying nature of the design problem and reuses results from the previous filter design. As a consequence, the computational complexity of the proposed algorithm was considerably reduced compared to an algorithm performing a dedicated filter design for each time instant. Furthermore, a system model for the precorrection of DAC employing nonuniform ZOH signals was introduced which

complements the postcorrection of ADCs in [22]. The presented application example extends the precorrection of nonuniformly time-varying ZOH signals from the periodic case, as presented in [23], to the non-periodic case.

In Chapter 4, structures for the correction of weakly time-varying systems have been developed. These structures can be applied in weakly time-varying scenarios where the design complexity of the filter design based process, as presented in Chapter 3, can not be afforded. The proposed precorrection structure complements postcorrection structures that have been presented in [26–29]. Furthermore, a condition for weakly time-varying systems has been presented which can be easily evaluated in real-time applications, and the commutativity of the correction structure and the undesired linear system has been established. Therefore, results regarding reconstruction performance, convergence and computational complexity obtained for the postcorrection case, as presented in [26–29] can be directly applied to the proposed precorrection structures. Additionally, the viability of the proposed structures was demonstrated via the precorrection of nonuniform ZOH signals. For this precorrection of nonuniform ZOH signals, a modified Farrow filter structure was developed which can be adapted by tuning a few time-varying multiplies only. Thus, a further reduction of the design complexity could be achieved by applying modified Farrow filters in the precorrection structure.

In Chapter 5, the extension of the principles presented in Chapter 4 to nonlinear systems was explored. To this end, the definition of the P^{th}-order inverse was extended to time-varying Volterra systems to provide a reference correction scheme. Furthermore, fixed point iteration based methods were proposed as a means of correcting Volterra systems with weakly time-varying linear components.

A Appendix

A.1 Order Update of the Inverse $\mathbf{P}_{n-1}^{K}$

The design matrix $\mathbf{M}_{n-1}^{K}$ can be partitioned as the backward prediction problem

$$\mathbf{M}_{n-1}^{K} = \left[\mathbf{M}_{n-1}^{K-1},\ \mathbf{m}_{n-1,K-1}\right]. \tag{A.1}$$

In order to derive $\mathbf{P}_{n-1}^{K} = \left(\left(\mathbf{M}_{n-1}^{K}\right)^{*}\mathbf{M}_{n-1}^{K}\right)^{-1}$, we first consider the matrix product $\left(\mathbf{P}_{n-1}^{K}\right)^{-1} = \left(\mathbf{M}_{n-1}^{K}\right)^{*}\mathbf{M}_{n-1}^{K}$, using (A.1), i.e.,

$$\left(\mathbf{P}_{n-1}^{K}\right)^{-1} = \begin{bmatrix} \left(\mathbf{M}_{n-1}^{K-1}\right)^{*}\mathbf{M}_{n-1}^{K-1}, & \left(\mathbf{M}_{n-1}^{K-1}\right)^{*}\mathbf{m}_{n-1,K-1} \\ \\ \mathbf{m}_{n-1,K-1}^{*}\mathbf{M}_{n-1}^{K-1}, & \mathbf{m}_{n-1,K-1}^{*}\mathbf{m}_{n-1,K-1} \end{bmatrix}. \tag{A.2}$$

The matrix inversion lemma for partitioned matrices can be employed to obtain the inverse of (A.2) as [81]

$$\mathbf{P}_{n-1}^{K} = \begin{bmatrix} \mathbf{P}_{n-1}^{K-1} + \dfrac{\mathbf{w}_{\mathrm{b}}\mathbf{w}_{\mathrm{b}}^{*}}{\epsilon_{\mathrm{b}}}, & -\dfrac{\mathbf{w}_{\mathrm{b}}}{\epsilon_{\mathrm{b}}} \\ -\dfrac{\mathbf{w}_{\mathrm{b}}^{*}}{\epsilon_{\mathrm{b}}}, & \dfrac{1}{\epsilon_{\mathrm{b}}} \end{bmatrix} \tag{A.3}$$

where the notation for joint process estimation in [51] has been used.

The backward projection vector $\mathbf{w}_{\mathrm{b}}$ is given as

$$\mathbf{w}_{\mathrm{b}} = \left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger}\mathbf{m}_{n-1,K-1} \tag{A.4}$$

and the backward scaling factor is

$$\epsilon_{\mathrm{b}} = \mathbf{m}_{n-1,K-1}^{*}\left(\mathbf{m}_{n-1,K-1} - \mathbf{M}_{n-1}^{K-1}\mathbf{w}_{\mathrm{b}}\right). \tag{A.5}$$

There are two possible interpretations of the relation in (A.3). A straight-forward application of (A.3) would be to determine $\mathbf{P}_{n-1}^{K}$ by evaluating its right handside employing the

lower order matrix $\mathbf{P}_{n-1}^{K}$ and the vector $\mathbf{m}_{n-1,K-1}$. However, if it is desired to determine the lower order matrix $\mathbf{P}_{n-1}^{K-1}$ from $\mathbf{P}_{n}^{K}$, the latter matrix needs to be partitioned, i.e.,

$$\mathbf{P}_{n-1}^{K} = \begin{bmatrix} \mathbf{S}, & \mathbf{t} \\ \mathbf{t}^{*}, & c \end{bmatrix} \tag{A.6}$$

where $\mathbf{S}$ represents a real-valued $K-1$-by-$K-1$ matrix, $\mathbf{t}^{*}$ is a real-valued row vector of length $K-1$ and c is a real scalar value. The backward projection vector can be determined as

$$\mathbf{w}_{\mathrm{b}} = -\frac{\mathbf{t}}{c} \tag{A.7}$$

based on the partitions of $\mathbf{P}_{n-1}^{K}$. For this scenario, (A.7) provides an alternative to the relation in (A.4) without requiring the matrix $\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger}$.

The lower order equivalent of $\mathbf{P}_{n-1}^{K}$ can be calculated as

$$\mathbf{P}_{n-1}^{K-1} = \mathbf{S} - \mathbf{w}_{\mathrm{b}}\mathbf{t}^{*} \tag{A.8}$$

The presented derivations can be applied analogously to the respective forward projection problem.

A.2 Order Update of the Generalized Inverse $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$

Multiplying (A.3) with the partitioned version of $\left(\mathbf{M}_{n-1}^{K-1}\right)^{*}$ given in (A.1), an order update relation for the generalized inverse is obtained as

$$\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger} = \begin{bmatrix} \left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger} + \dfrac{\mathbf{w}_{\mathrm{b}}\mathbf{w}_{\mathrm{b}}^{*}\left(\mathbf{M}_{n-1}^{K-1}\right)^{*} - \mathbf{w}_{\mathrm{b}}\mathbf{m}_{n-1,K-1}^{*}}{\epsilon_{\mathrm{b}}} \\ -\dfrac{\mathbf{w}_{\mathrm{b}}^{*}\left(\mathbf{M}_{n-1}^{K-1}\right)^{*} + \mathbf{m}_{n-1,K-1}^{*}}{\epsilon_{\mathrm{b}}} \end{bmatrix} \tag{A.9}$$

for the backward projection scenario. Introducing the backward projection error

$$\mathbf{b} = \mathbf{m}_{n-1,K-1} - \mathbf{M}_{n-1}^{K-1}\mathbf{w}_{\mathrm{b}}, \tag{A.10}$$

(A.9) can be rewritten in a more concise form as

$$\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger} = \begin{bmatrix} \left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger} - \dfrac{\mathbf{w}_{\mathrm{b}}\mathbf{b}^{*}}{\epsilon_{\mathrm{b}}} \\ \dfrac{\mathbf{b}^{*}}{\epsilon_{\mathrm{b}}} \end{bmatrix}. \tag{A.11}$$

For determining $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$ from the lower order generalized inverse $\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger}$, the backward projection vector $\mathbf{w}_{\mathrm{b}}$ and the backward projection error $\mathbf{b}$ are required to calculate the right handside of (A.9). In order to determine the lower order equivalent of the generalized inverse $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$, this matrix is partitioned in the upper matrix $\mathbf{U}$ and the row vector $\mathbf{v}^{*}$, i.e.,

$$\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger} = \begin{bmatrix} \mathbf{U} \\ \mathbf{v}^{*} \end{bmatrix} \tag{A.12}$$

where the matrix $\mathbf{U}$ represents the upper $K-1$ rows and the vector $\mathbf{v}^{*}$ the K^{th} row of $\left(\mathbf{M}_{n-1}^{K}\right)^{\dagger}$.

The lower order generalized inverse can be determined as the $K-1$-by-R matrix

$$\left(\mathbf{M}_{n-1}^{K-1}\right)^{\dagger} = \mathbf{U} + \mathbf{w}_{\mathrm{b}}\mathbf{v}^{*}. \tag{A.13}$$

The presented derivations can be applied analogously to the respective forward projection problem.

A.3 Transfer Function as Matrix Polynomial

The output vector $\mathbf{w}^{(R)}$ after R iterations with respect the input vector $\mathbf{x}$ is given by

$$\mathbf{w}^{(R)} = \left(\mathbf{I} - (\mathbf{I} - \mathbf{H})^{R+1}\right)\mathbf{x}. \tag{A.14}$$

In order to relate $\mathbf{w}^{(R)}$ and the input vector of the iteration $\mathbf{y}$, (A.14) is rewritten as

$$\mathbf{w}^{(R)} = \left(\mathbf{I} - (\mathbf{I} - \mathbf{H})^{R}(\mathbf{I} - \mathbf{H})\right)\mathbf{H}^{-1}\mathbf{y} \tag{A.15}$$

Developing the term $(\mathbf{I} - \mathbf{H})^{R}$ in terms of a binomial series results in

$$\mathbf{w}^{(R)} = \mathbf{P}_{\mathbf{H}}^{(R)}\mathbf{y} \tag{A.16}$$

with

$$\begin{aligned} \mathbf{P}_{\mathbf{H}}^{(R)} &= \mathbf{I} + \sum_{r=1}^{R} \binom{R}{r} (-\mathbf{H})^{r}(\mathbf{I} - \mathbf{H}^{-\mathbf{1}}) \\ &= \mathbf{I} + \sum_{r=1}^{R} \binom{R}{r} \left((-\mathbf{H})^{r} + (-\mathbf{H})^{r-1}\right) \end{aligned} \tag{A.17}$$

A.4 Modified Farrow Filter

The output of the modified Farrow filter $w_1[n]$ may be described relative to the input signal $u[n]$, as depicted in Fig. A.1, i.e.

$$w_1[n] = \sum_{p=-\infty}^{\infty} g_{1,n}^{\mathrm{a}}[p]x[n-p]. \tag{A.18}$$

The impulse response of the modified Farrow filter is given, according to the derivation in Section 4.4.5, as

$$g_{1,n}^{\mathrm{a}}[p] = \sum_{q=-\infty}^{\infty} \sum_{l=0}^{L} b_l[q]\ \hat{\Delta}_{n-q}^{l}[p-q] \tag{A.19}$$

where $\hat{\Delta}_{n-q}[p]$ is defined as

$$\hat{\Delta}_{n-q}[p] = \begin{cases} \Delta_{n-p} & \text{for } p = q \\ 0 & \text{otherwise.} \end{cases} \tag{A.20}$$

Thus, we rewrite (A.18), using (A.19) as

$$w_1[n] = \sum_{p=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} \sum_{l=0}^{L} b_l[q]\ \hat{\Delta}_{n-q}^{l}[p-q]x[n-p] \tag{A.21}$$

Substituting the index $k = p - q$ and $p = q + k$, we obtain

$$w_1[n] = \sum_{k=-\infty}^{\infty} \sum_{q=-\infty}^{\infty} \sum_{l=0}^{L} b_l[q]\ \hat{\Delta}_{n-q}^{l}[k]x[n-q-k] \tag{A.22}$$

Introducing a set of intermediate signal $s_l[n]$, (A.21) can be split into two convolutions, i.e.

$$w_1[n] = \sum_{q=-\infty}^{\infty} \sum_{l=0}^{L} b_l[q]\ s_l[n-q] \tag{A.23}$$

and

$$s_l[n] = \sum_{k=-\infty}^{\infty} \hat{\Delta}_{n}^{l}[k]x[n-k] \tag{A.24}$$

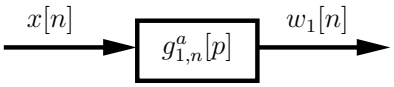


Figure A.1 Signal model for a single modified Farrow filter $g_{1,n}^{\mathrm{a}}[p]$.

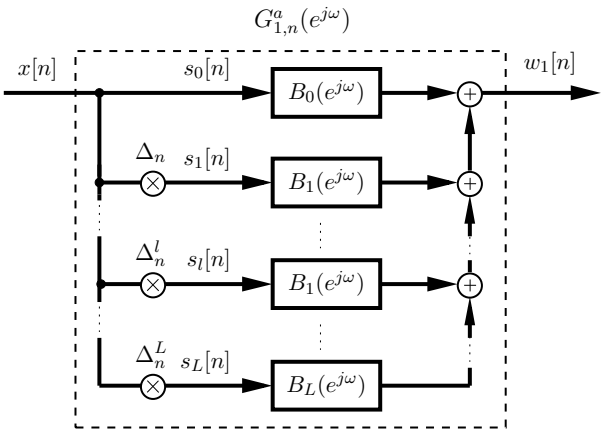


Figure A.2 Modified Farrow filter approximating $g^{\mathrm{a}}_{1,n}[p]$.

Using the relationship in (A.20), i.e. for $p = 0$ the auxiliary impulse response is only non zero for $k = 0$, we can simplify (A.24), i.e.,

$$s_l[n] = \Delta^l_n x[n] \tag{A.25}$$

The resulting simplified structure is shown in Fig. A.2.

A.5 Volterra Systems

A.5.1 Discrete-Time Volterra Filter

An example second order discrete-time Volterra filter with time-varying coefficients and a memory of 2 is illustrated in Fig. A.3.

A.5.2 Cascade of Time-Varying Volterra Filters

In this subsection the time-varying operators $F_{p,n}\{.\}$ that represent a cascade of two first order Volterra systems is exemplified. Applying the relation in (5.2) to the operators of the cascaded Voltera systems as presented in (5.15)-(5.17), we obtain the operators in (A.26).

$$\begin{aligned}
F_1\{x[n]\} &= G_{1,n}\{H_{1,n}\{x[n]\}\} \\
F_2\{x[n]\} &= G_{1,n}\{H_{2,n}\{x[n]\}\} + G_{2,n}\{H_{1,n}\{x[n]\}\} \\
F_3\{x[n]\} &= G_{1,n}\{H_{3,n}\{x[n]\}\} + G_{3,n}\{H_{1,n}\{x[n]\}\} + 2G_{2,n}\{H_{1,n}\{x[n]\}, H_{3,n}\{x[n]\}\}
\end{aligned} \tag{A.26}$$

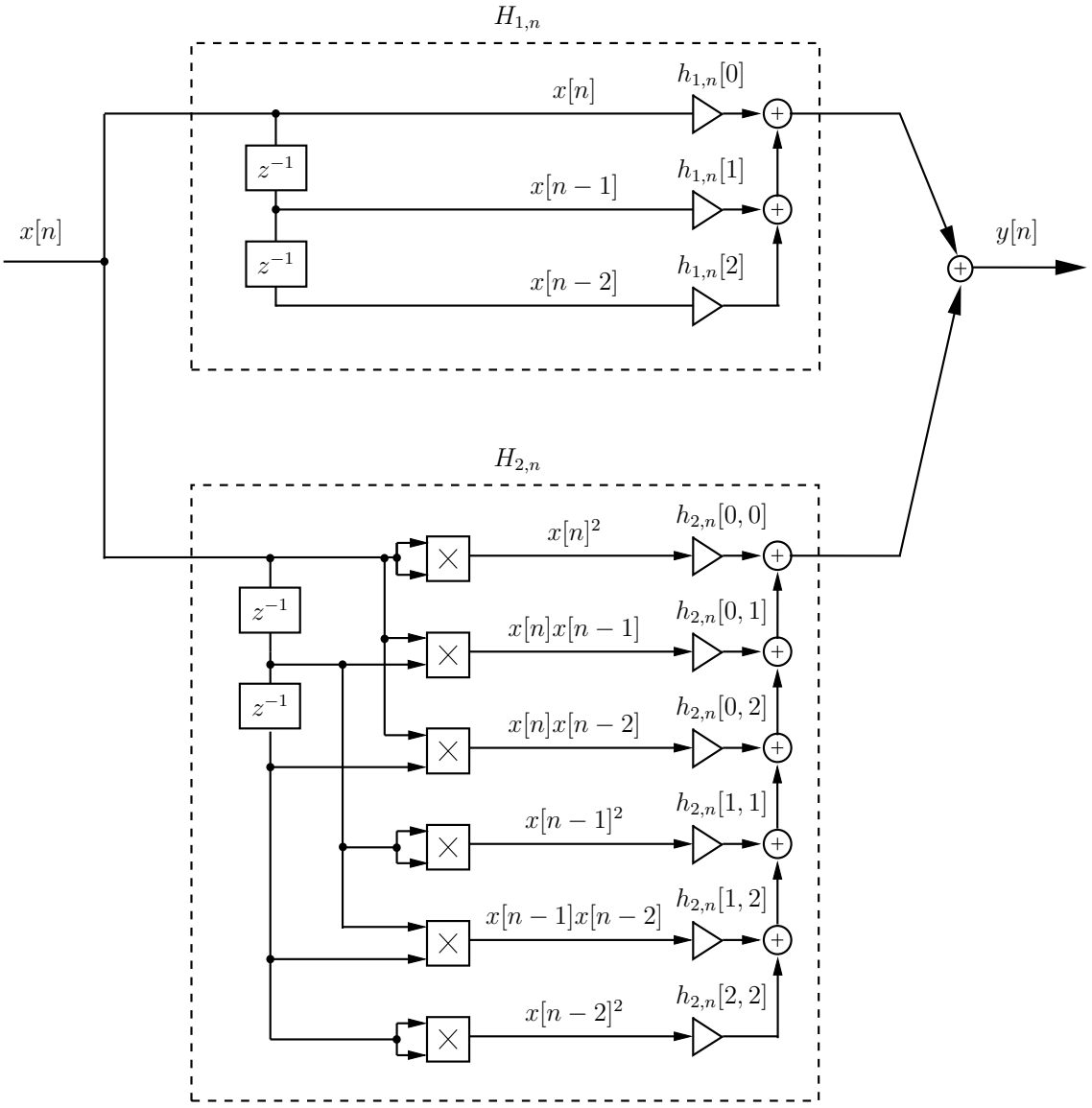


Figure A.3 Discrete Volterra filter of order 2.

.

Since the operator $H_{p,n}\{.\}$ and $G_{p,n}\{.\}$ are zero for $p > 2$ in this example, the relation in (A.26) reduces to

$$\begin{aligned} F_1\{x[n]\} &= G_{1,n}\{H_{1,n}\{x[n]\}\} \\ F_2\{x[n]\} &= G_{1,n}\{H_{2,n}\{x[n]\}\} + G_{2,n}\{H_{1,n}\{x[n]\}\} \\ F_3\{x[n]\} &= 2G_{2,n}\{H_{1,n}\{x[n]\}, H_{2,n}\{x[n]\}\} \end{aligned} \tag{A.27}$$

.

where $F_3\{x[n]\}$ can be expanded, i.e.,

$$F_3\{x[n]\} = G_{2,n}\{y_1[n] + y_2[n]\} - G_{2,n}\{y_1[n]\} - G_{2,n}\{y_2[n]\}. \tag{A.28}$$

The 4^{th}-order operator can be obtained as shown in [74] resulting in a time-varying formulation given as

$$\begin{aligned} F_4\{x[n]\} &= G_{1,n}\{H_{4,n}\{x[n]\}\} + G_{4,n}\{H_{1,n}\{x[n]\}\} + 2G_{2,n}\{H_{1,n}\{x[n]\}, H_{3,n}\{x[n]\}\} \\ &\quad + G_{2,n}\{H_{2,n}\{x[n]\}\} + 3G_{3,n}\{H_{1,n}\{x[n]\}, H_{1,n}\{x[n]\}, H_{2,n}\{x[n]\}\} \\ &= G_{2,n}\{H_{2,n}\{x[n]\}\} \end{aligned} \tag{A.29}$$

The resulting structure is depicted in Fig. A.4.

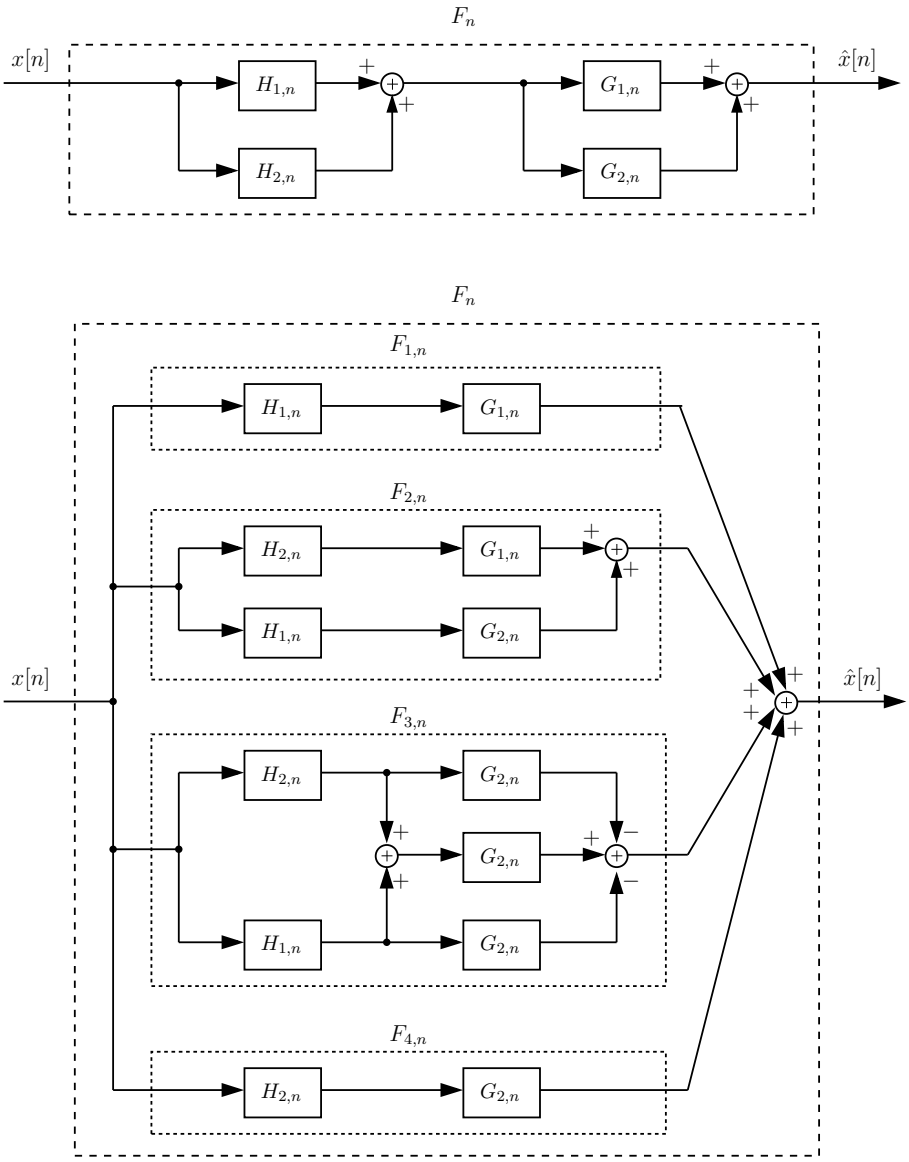


Figure A.4 Cascade of two Volterra systems of order 2.

Bibliography

[1] W. C. Black and D. A. Hodges, "Time-interleaved converter arrays," *IEEE Journal of Solid State Circuits*, vol. 15, no. 6, pp. 1024–1029, December 1980.

[2] V. Divi and G. Wornell, "Blind calibration of timing skew in time-interleaved analog-to-digital converters," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 3, pp. 509–522, June 2009.

[3] T. Kailath, "Sampling models for linear time-variant filters," Ph.D. dissertation, Massachusetts Institute of Technology, M.I.T Research Lab of Electronics, Cambridge, Massachusets, USA: Technical Report, 1959.

[4] G. Matz and F. Hlawatsch, "Time-varying communication channels: Fundamentals recent developments and open problems," in *Proceedings of the 14th European Signal Processing Conference (Invited Paper), EUSIPCO*, Florence, Italy, September 2006.

[5] P. Bello, "Characterization of randomly time-variant linear channels," *IEEE Transactions on Communications Systems*, vol. 11, no. 4, pp. 360–393, December 1963.

[6] G. Giannakis and C. Tepedelenlioglu, "Basis expansion models and diversity techniques for blind identification and equalization of time-varying channels," *Proceedings of the IEEE*, vol. 86, no. 10, pp. 1969–1986, October 1998.

[7] S. Saleem and C. Vogel, "Adaptive compensation of frequency response mismatches in high-resolution time-interleaved ADCs using a low-resolution ADC and a time-varying filter," in *IEEE International Symposium on Circuits and Systems, ISCAS*, Paris, France, May 2010, pp. 561–564.

[8] H. Johansson and P. Löwenborg, "Reconstruction of uniformly sampled bandlimited signals by means of digital fractional delay filters," *IEEE Transactions on Signal Processing*, vol. 50, no. 11, pp. 2757–2767, November 2002.

[9] ——, "A least-squares filter design technique for the compensation of frequency-response mismatch errors in time-interleaved A/D converters," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 55, no. 6, pp. 1154–1158, November 2008.

[10] A. F. Molisch, *Wideband wireless digital communications*. Prentice-Hall, 2001.

[11] B. Murmann, C. Vogel, and H. Koeppl, "Digitally enhanced analog circuits: System aspects," in *IEEE International Symposium on Circuits and Systems, ISCAS*, Seattle, Washington, USA, 2008, pp. 560–563.

[12] C. Vogel and H. Johansson, "Time-interleaved analog-to-digital converters: Status and future directions," in *IEEE International Symposium on Circuits and Systems ISCAS*, Island of Kos, Greece, May 2006, pp. 3386–3389.

[13] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Prentice Hall, 2002.

[14] N. Kurosawa, H. Kobayashi, H. Kogure, T. Komuro, and H. Sakayori, "Sampling clock jitter effects in digital-to-analog converters," *Measurement*, vol. 31, no. 3, pp. 187–199, 2002.

[15] C. Vogel, "Modeling, identification, and compensation of channel mismatch errors in time-interleaved analog-to-digital converters," Ph.D. dissertation, Graz University of Technology, Austria: Doctoral Thesis, 2005.

[16] ——, "The impact of combined channel mismatch effects in time-interleaved ADCs," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 2, pp. 415–427, February 2005.

[17] I. Mitola, J., "Software radios: Survey, critical evaluation and future directions," *IEEE Aerospace and Electronic Systems Magazine*, vol. 8, no. 4, pp. 25–36, April 1993.

[18] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 1.21," http://cvxr.com/cvx, 2011.

[19] H. Johansson and P. Löwenborg, "Reconstruction of nonuniformly sampled bandlimited signals by means of time-varying discrete-time FIR filters," *EURASIP Journal on Applied Signal Processing*, vol. 2006, January 2006, Article ID 64185, 18 pages.

[20] Y. C. Lim, Y.-X. Zou, J. W. Lee, and S.-C. Chan, "Time-interleaved analog-to-digital-converter compensation using multichannel filters," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 10, pp. 2234–2247, October 2009.

[21] M. Seo, M. Rodwell, and U. Madhow, "Comprehensive digital correction of mismatch errors for a 400-Msamples/s 80-dB SFDR time-interleaved analog-to-digital converter," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, no. 3, pp. 1072–1082, March 2005.

[22] H. Johansson and P. Löwenborg, "Reconstruction of nonuniformly sampled bandlimited signals using time-varying discrete-time FIR filters," in *Proceedings of 12th European Signal Processing Conference, EUSIPCO*, Vienna, Austria, September 2004.

[23] C. Vogel and C. Krall, "Digital compensation of in-band image signals caused by M-periodic nonuniform zero-order hold signals," *Ubiquitous Computing and Communication Journal, UBICC*, pp. 1–10, September 2009.

[24] M. Soudan and C. Vogel, "On the correction of linear time-varying systems by means of time-varying FIR filters," in *Proceedings of the 54th Midwest Symposium on Circuits and Systems, MWSCAS*, Seoul, South Korea, August 2011.

[25] ——, "Low complexity least squares filter design for the correction of linear time-varying systems," in *Proceedings of the 20th European Conference on Circuit Theory and Design, ECCTD*, Linköping, Sweden, August 2011, pp. 690–693.

[26] C. Vogel and S. Mendel, "A flexible and scalable structure to compensate frequency response mismatches in time-interleaved ADCs," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 11, pp. 2463–2475, November 2009.

[27] S. Tertinek and C. Vogel, "Reconstruction of two-periodic nonuniformly sampled band-limited signals using a discrete-time differentiator and a time-varying multiplier," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 54, no. 7, pp. 616–620, July 2007.

[28] K. Tsui and S. Chan, "A versatile iterative framework for the reconstruction of bandlimited signals from their nonuniform samples," *Journal of Signal Processing Systems*, pp. 1–10, 2010.

[29] K. M. Tsui and S. C. Chan, "New iterative framework for frequency response mismatch correction in time-interleaved ADCs: Design and performance analysis," *IEEE Transactions on Instrumentation and Measurement*, no. 99, pp. 1–14, 2011.

[30] H. Johansson, "A polynomial-based time-varying filter structure for the compensation of frequency-response mismatch errors in time-interleaved ADCs," *IEEE Journal of Selected Topics in Signal Processing*, vol. 3, no. 3, pp. 384–396, June 2009.

[31] S. Tertinek and C. Vogel, "Reconstruction of nonuniformly sampled bandlimited signals using a differentiator-multiplier cascade," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 8, pp. 2273–2286, September 2008.

[32] H. Johansson, P. Löwenborg, and K. Vengattaramane, "Reconstruction of two-periodic nonuniformly sampled signals using polynomial impulse response time-varying FIR filters," in *IEEE International Symposium on Circuits and Systems, ISCAS*, Island of Kos, Greece, September 2006, pp. 2993–2996.

[33] D. McLernon, "Parameter estimation of autoregressive processes with periodic coefficients," in *IEEE International Symposium on Circuits and Systems, ISCAS*, vol. 2, Portland, Oregon, USA, May 1989, pp. 1350–1353.

[34] S. Ohno, H. Sakai, and H. Yoshida, "Adaptive blind equalization of multichannel FIR systems," in *IEEE International Symposium on Circuits and Systems, ISCAS*, vol. 3, Orlando, Florida, USA, July 1999, pp. 66–69.

[35] N. Kurosawa, H. Kobayashi, K. Maruyama, H. Sugawara, and K. Kobayashi, "Explicit analysis of channel mismatch effects in time-interleaved ADC systems," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, pp. 261–271, March 2001.

[36] Y. Eldar and A. Oppenheim, "Filterbank reconstruction of bandlimited signals from nonuniform and generalized samples," *IEEE Transactions on Signal Processing*, vol. 48, no. 10, pp. 2864–2875, October 2000.

[37] C. Farrow, "A continuously variable digital delay element," in *IEEE International Symposium on Circuits and Systems, ISCAS*, vol. 3, Helsinki, Finland, June 1988, pp. 2641–2645.

[38] K. Tsui, S. Chan, and K. Tse, "Design of complex-valued variable fir digital filters and its application to the realization of arbitrary sampling rate conversion for complex signals," *IEEE Transactions on Circuits and Systems Part 2: Express Briefs*, vol. 52, no. 7, pp. 424–428, June 2005.

[39] P. P. Vaidyanathan, *Multirate Systems and Filter Banks.* Prentice-Hall Signal Processing Series, 1993.

[40] H. H. Dam, A. Cantoni, K. L. Teo, and S. Nordholm, "FIR variable digital filter with signed power-of-two coefficients," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 6, pp. 1348–1357, June 2007.

[41] C. Loeffler and C. Burrus, "Optimal design of periodically time-varying and multirate digital filters," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 32, no. 5, pp. 991–997, October 1984.

[42] S. Haykin, *Adaptive Filter Theory.* Prentice-Hall, 2002.

[43] J. Elbornsson and J.-E. Eklund, "Blind estimation of timing errors in interleaved AD converters," in *Proceedings of the IEEE International Conference on Acoustics,*

Speech, and Signal Processing, ICASSP, vol. 6, Salt Lake City, Utah, USA, 2001, pp. 3913–3916.

[44] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing.* Prentice Hall, 1999.

[45] S. Boyd, *Convex Optimization.* Cambridge University Press, 2004.

[46] A. Antoniou, *Practical Optimization: Algorithms and Engineering.* Springer Science+Business Media, 2007.

[47] R. Prendergast, B. Levy, and P. Hurst, "Reconstruction of band-limited periodic nonuniformly sampled signals through multirate filter banks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 8, pp. 1612–1622, August 2004.

[48] C. Vogel, "Compensation of two-periodic nonuniform holding signal distortions by using a variable FIR filter," in *International Conference on Signals and Electronic Systems, ICSES*, Karkow, Poland, 2008, pp. 323–326.

[49] H. Johansson, P. Löwenborg, and K. Vengattaramane, "Least-squares and minimax design of polynomial impulse response FIR filters for reconstruction of two-periodic nonuniformly sampled signals," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 4, pp. 877–888, April 2007.

[50] V. Madisetti, *The Digital Signal Processing Handbook.* CRC Press, 1997.

[51] A. Sayed, *Fundamentals of Adaptive Filtering.* Wiley & Sons, Inc., 2003.

[52] G. Strang, *Linear Algebra and its Applications.* Harcourth Brace Jovanovich, 1988.

[53] W. Miller, "Computational complexity and numerical stability," in *Proceedings of the sixth annual ACM symposium on Theory of computing*, Seattle, Washington, USA, 1974, pp. 317–322.

[54] L. Angrisani and M. D'Arco, "Modeling timing jitter effects in digital-to-analog converters," *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 2, pp. 330–336, February 2009.

[55] R. Nowak and B. Van Veen, "Volterra filter equalization: a fixed point approach," *IEEE Transactions on Signal Processing*, vol. 45, no. 2, pp. 377–388, February 1997.

[56] E. Iroaga, B. Murmann, and L. Nathawad, "A background correction technique for timing errors in time-interleaved analog-to-digital converters," in *IEEE International Symposium on Circuits and Systems, ISCAS*, vol. 6, Kobe, Japan, May 2005, pp. 5557–5560.

[57] V. Divi and G. Wornell, "Scalable blind calibration of timing skew in high-resolution time-interleaved ADCs," in *IEEE International Symposium on Circuits and Systems, ISCAS*, Island of Kos, Greece, September 2006, pp. 3390–3393.

[58] ——, "Bandlimited signal reconstruction from noisy periodic nonuniform samples in time-interleaved ADCs," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, Las Vegas, Nevada, USA, April 2008, pp. 3721–3724.

[59] M. Seo, M. Rodwell, and U. Madhow, "Generalized blind mismatch correction for two-channel time-interleaved A-to-D converters," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*, vol. 3, Honolulu, Hawaii, USA, April 2007, pp. 1505–1508.

[60] M. Seo and M. Rodwell, "Generalized blind mismatch correction for a two-channel time-interleaved ADC: Analytic approach," in *IEEE International Symposium on Circuits and Systems, ISCAS*, Orleans, Louisiana, USA, May 2007, pp. 109–112.

[61] Y. Saad, *Iterative Methods for Sparse Linear Systems*. Philadelphia, Pennsylvania, USA: Society for Industrial and Applied Mathematics, 2003.

[62] K. Tsui and S. Chan, "Iterative correction of frequency response mismatches in time-interleaved ADCs: A novel framework and case study in OFDM systems," *International Conference on Green Circuits and Systems, ICGCS*, pp. 253–258, June 2010.

[63] T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*. Society for Industrial and Applied Mathematics, 2004.

[64] P. Lancaster and M. Tismenetsky, *The Theory of Matrices, Second Edition: With Applications*. Academic Press, 1985.

[65] C.-C. Tseng and S.-L. Lee, "Efficient design and implementation of variable fractional delay filters using differentiators," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 6, pp. 1311–1322, June 2011.

[66] V. Mathews, "Adaptive polynomial filters," *IEEE Signal Processing Magazine*, vol. 8, no. 3, pp. 10–26, July 1991.

[67] P. Traverso, D. Mirri, G. Pasini, and F. Filicori, "A nonlinear dynamic S/H-ADC device model based on a modified volterra series: identification procedure and commercial CAD tool implementation," *IEEE Transactions on Instrumentation and Measurement*, vol. 52, no. 4, pp. 1129–1135, August 2003.

[68] H. Gandhi and W. Abbott, "A digital signal processing solution for PA linearization and RF impairment correction for multi-standard wireless transceiver systems," in *2010 European Microwave Conference, EuMC*, Paris, France, September 2010, pp. 719–722.

[69] G. Lazzarin, S. Pupolin, and A. Sarti, "Nonlinearity compensation in digital radio systems," *IEEE Transactions on Communications*, vol. 42, no. 234, pp. 988–999, February 1994.

[70] S. Boyd and L. Chua, "Fading memory and the problem of approximating nonlinear operators with volterra series," *IEEE Transactions on Circuits and Systems*, vol. 32, no. 11, pp. 1150–1161, November 1985.

[71] P. Wambacq and W. M. C. Sansen, *Distortion analysis of analog integrated circuits.* Springer Press, 1998.

[72] L. Guan and A. Zhu, "Low-cost FPGA implementation of volterra series-based digital predistorter for RF power amplifiers," *IEEE Transactions on Microwave Theory and Techniques*, vol. 58, no. 4, pp. 866–872, April 2010.

[73] J. Tsimbinos, "Identification and compensation of nonlinear distortion," Ph.D. dissertation, University of South Australia, The Levels, South Australia: Doctoral Thesis, 2002.

[74] M. Schetzen, "Theory of pth-order inverses of nonlinear systems," *IEEE Transactions on Circuits and Systems*, vol. 23, no. 5, pp. 285–291, May 1976.

[75] A. Satri, "Recursive techniques for the synthesis of a pth-order inverse of a volterra system," *European Transactions on Telecommunications*, vol. 3, no. 4, pp. 315–322, July 1992.

[76] V. Kafka, "Rekursive strukturen auf volterra-basis zur aufwandsarmen darstellung und entzerrung von nichtlinearen systemen," Ph.D. dissertation, Universität der Bundeswehr München, Neubiberg, Germany: Doctoral Thesis, 2002.

[77] P. Ortega and W. M. C. Sansen, *Iterative Solution of Nonlinear Equations in Several Variables.* Society for Industrial Mathematics, 1987.

[78] J. Ralston and A. Zoubir, "Identification of time-varying hammerstein systems," in *International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, vol. 3, Detroit, Michigan, USA, May 1995, pp. 1685–1688.

[79] A. Nordsjo and L. Zetterberg, "Identification of certain time-varying nonlinear wiener and hammerstein systems," *IEEE Transactions on Signal Processing*, vol. 49, no. 3, pp. 577–592, March 2001.

[80] M. Schetzen, "Measurement of the kernels of a non-linear system of finite order," *International Journal of Control*, vol. 1, no. 3, pp. 251–263, March 1955.

[81] S. Kay, *Fundamentals of Statistical Signal Processing, Volume 1: Estimation Theory*. Prentice Hall, 1993.