**This document is downloaded from DR-NTU (https://dr.ntu.edu.sg)
Nanyang Technological University, Singapore.**

A parametric approach to adaptive IIR filtering : finite precision implementation and stability aspects

Zhou, Jiong

2006

Zhou, J. (2006). A parametric approach to adaptive IIR filtering : finite precision implementation and stability aspects. Doctoral thesis, Nanyang Technological University, Singapore.; Zhou, J. (2006). A parametric approach to adaptive IIR filtering : finite precision implementation and stability aspects. Doctoral thesis, Nanyang Technological University, Singapore.

https://hdl.handle.net/10356/4066

https://doi.org/10.32657/10356/4066

Nanyang Technological University

Downloaded on 22 Apr 2025 15:46:41 SGT

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

A Parametric Approach to Adaptive IIR Filtering: Finite Precision Implementation and Stability Aspects

Zhou Jiong

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfilment of the requirement for the degree of
Doctor of Philosophy

2006

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Acknowledgments

First of all, I would like to express special thanks to my supervisor, Dr. Li Gang, for his professional and invaluable guidance during the last three years. The completion of this thesis would be impossible without his close supervision and devotion to education.

Three years ago, when starting this Ph.D program, I, as a fresh Bachelor holder, wondered if I could complete it three years later. The past three years is one of the most memorable time I have ever experienced in my life, during which I have not only gained research experience but also made a well-established public relationship with fellow postgraduate students from different countries. Their friendly discussions, strong background and amicable attitudes make our Lab like a "big family" such that I can hardly feel this is an overseas life. In addition, I am grateful to the technicians in Information System Research Lab and Workstation Resource Lab for their dedication to work and kind assistance.

At last, I would like to thank my parents and my brother for their support and encouragement.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Abstract

It is well known that a digital filter can be characterized with different *parametrizations* and realized with different *structures*. The main objective of this thesis is two-fold. The first one is to investigate the effects of parametrizations and structures on the performance of adaptive filters, particularly on the stability aspect, and hence to derive new adaptive algorithms. The second one is to examine how a well-designed adaptive algorithm can be degraded when implemented with finite precision considerations.

The main contributions in this thesis are summarized as follows.

After demonstrating the finite word length (FWL) effects on the adaptive filters using the traditional direct-form-based structures, a simple adaptive state-space filter is derived. This structure is computationally efficient and it is shown that such an adaptive filter yields much better performance than the classical adaptive state-space filters in terms of stability and FWL behavior.

It is shown that when an adaptive filter can be implemented with a normal state-space realization, the corresponding adaptive algorithm is bounded-input bounded-output (BIBO) stable if the instantaneous poles of the filter are all inside the circle $|z| < \gamma$ with γ a constant satisfying $0 < \gamma < 1$. As one of the applications of this new result, a novel adaptive constrained IIR notch filter is derived, where the filter is parametrized using the frequency vector and implemented in a cascade form with

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

each subfilter realized with a normal state-space realization. Normal realizations have a minimal pole sensitivity and are free of overflow oscillations. It is shown that the proposed algorithm needs less computational cost than the lattice-based one.

It is well known that for any (linear) digital filter there exists a class of state-space realizations, called input balanced realizations, which have many nice numerical properties. An input balanced realization-based adaptive filter is derived. This realization can be characterized with two new parameterizations, with which the stability of the (time-invariant) filters can be controlled easily. It is shown that an adaptive filter derived based on any of these two parameterizations is BIBO stable under some mild conditions. Simulations show that the performance of such an adaptive filter is comparable with the famous normalized lattice-based one.

Frequency estimation is required in many applications. Such a problem can be attacked with adaptive constrained IIR filtering. Two new adaptive algorithms are proposed for direct frequency estimation, where the constrained filter is parametrized using its notch frequency. The performance analysis is carried out in terms of steady-state estimation bias and mean square error as well as tracking behavior. Compared with some of the existing adaptive frequency estimation algorithms, the proposed ones have several better properties in terms of stability behavior and computational efficiency.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Contents

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

List of Figures

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Notations

- $x(n)$: a discrete-time signal x
- $\hat{x}(n)$: an approximated or estimated version of $x(n)$, where $x(n)$ can be a signal or a parameter at time n
- z^{-1}: the backward shift operator such that $z^{-1}x(n) = x(n-1)$
- $A(k,:)$, $A(:,j)$: the kth row vector and the jth column vector of matrix A, respectively
- $||v||_2 \triangleq (\sum_{k=1}^{N} |v_k|^2)^{1/2}$: the *Euclidean norm* or l_2-*norm* of vector $v = \{v_k\} \in \mathcal{C}^{N\times 1}$
- $\lambda_k(A)$: the kth eigenvalue of matrix $A = \{a_{ij}\} \in \mathcal{C}^{N\times N}$
- I_K: the identity matrix of dimension K
- A^{-1}, $A^{\mathcal{T}}$: inverse and transpose of matrix A, respectively
- $A^{\mathcal{H}}$: complex conjugate and transpose of matrix A
- $|||A|||_2 \triangleq \max_{||v||_2=1} ||Av||_2$: the *spectral norm* of matrix $A \in \mathcal{C}^{M\times N}$, which is equal to $\max_{\forall k} \sqrt{\lambda_k(AA^{\mathcal{H}})}$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 1

Introduction

Adaptive filtering is a powerful tool that is used in many applications, e.g., in communications, radar, sonar, and biomedical engineering. Although these applications are quite different, the adaptive filters have one common feature. The filter output and a reference signal are used to compute an estimation error, which in turn is used to control a set of adjustable filter coefficients with a *numerical algorithm*.

1.1 Background

Adaptive filtering has been a very important research area for decades. A very good historical review of the developments and classical results can be found from [1], while many recent developments and techniques are available in [2], [3].

The content of linear adaptive filters includes *stochastic gradient* (SG) algorithms and *recursive least-squares* (RLS) algorithms. The earliest work on adaptive filters may be traced back to the late 1950s, during which a number of researchers were working independently on different applications of such filters. In the early work, the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

least-mean-square (LMS) algorithm emerged as a simple, yet effective, algorithm for the operation of adaptive transversal filters. The LMS algorithm is a SG algorithm in which it iterates each tap weight of the filter in the direction of the gradient of the squared magnitude of an error signal with respect to the tap weights. As to the RLS family of adaptive filtering algorithms, the very first paper on the *standard* RLS algorithm appears to be that of Plackett in 1950 (see [2]), although many other investigators had derived or re-derived various versions of the RLS algorithm. In 1974, Godard used Kalman filter theory to derive one variant of the algorithm that is sometimes referred to in the literature as the Godard algorithm. Although prior to that date several investigators had applied Kalman filter theory to solve the adaptive filtering problem, Godard's approach was widely accepted as the most successful application of Kalman filter theory for a span of two decades [2].

The essential difference between the various adaptive filter applications arises in the manner in which the reference signal $r(n)$ is extracted. In this context, we may distinguish four basic classes of adaptive filtering applications as depicted in Figure 1.1. The functions of the four basic classes of adaptive filtering applications are as follows:

- *System Identification Problem*: An adaptive filter is used to identify a system (a plant) transfer function. The reference signal is the plant output.
- *Inverse Modeling*: The adaptive filter is used to model the inverse of the plant transfer function. A delayed version of the plant input is the reference signal. (The delay could also be zero, in certain applications.)
- *Prediction*: The adaptive filter is used to determine the characteristics of a random signal. For example the adaptive filter is used to predict the present value of the random signal using the past values. (Either the predicted value or the prediction error could be useful for subsequent processing.)

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

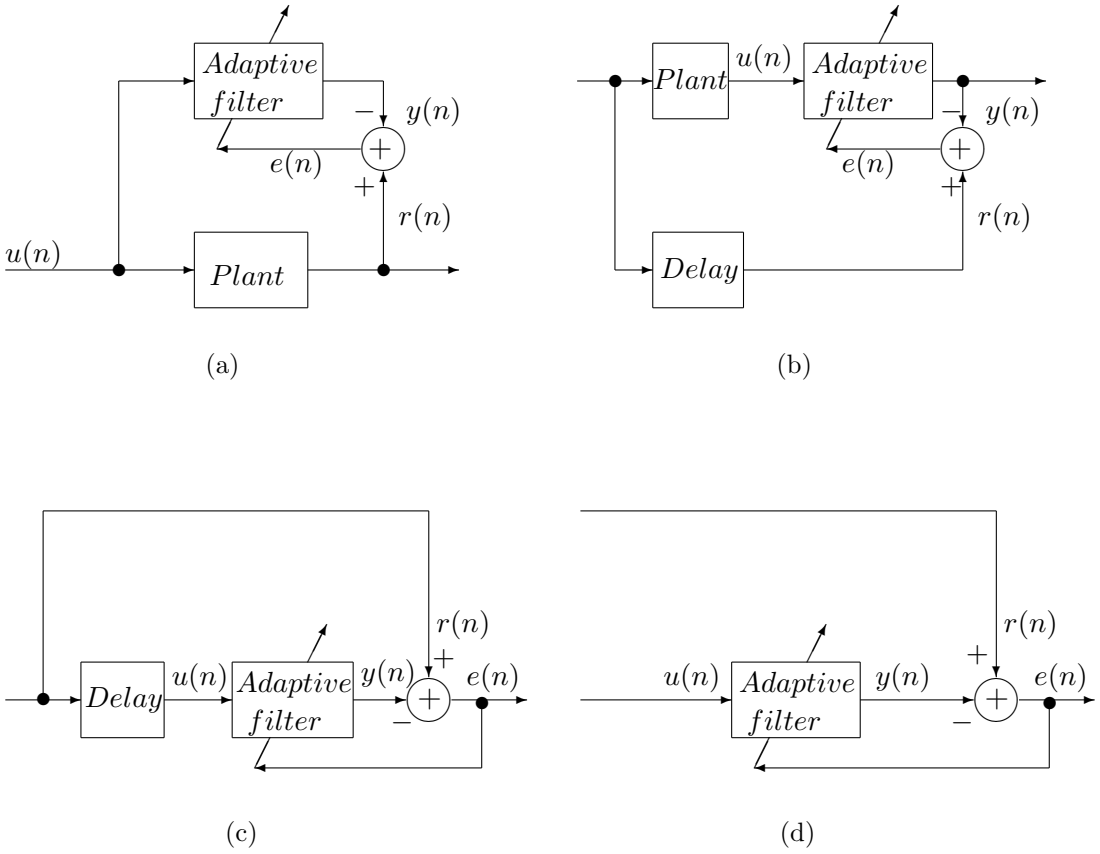


Figure 1.1: Four basic classes of adaptive filtering applications: (a) System identification; (b) Inverse modeling; (c) Prediction; (d) Interference cancellation.

- *Interference Cancellation*: The adaptive filter is used to estimate the unknown interference contained in the reference signal. An auxiliary signal, which is assumed to be correlated with the interference, is used as the input of the filter.

The most widely used adaptive filter is the adaptive *finite impulse response* (FIR) filter, which is usually implemented with the transversal structure. For FIR filters, the output signal $y(n)$ is a linear combination of the filter coefficients, which yields a quadratic mean-square-error function with a unique optimal operation point [1]. Alternative FIR filter realizations are also used in order to obtain improvements as compared to the transversal filter structure in terms of computational complexity

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[42], speed of convergence [43], and finite word length properties [2].

An early attempt to implement an adaptive *infinite impulse response* (IIR) filter was made by White [44] in 1975 and since then a large number of papers have been published in this area. In the last two decades, substantial research effort has been spent to turn adaptive IIR filtering techniques into a reliable alternative to traditional adaptive FIR filters, and adaptive IIR filtering has been considered for a variety of problems in signal processing and communications. Examples of some important applications include linear prediction, adaptive notch filtering, adaptive differential pulse code modulation, channel equalization, echo cancellation, and adaptive array processing [59], [60]. The main advantages of IIR filters are that they are more suitable to model physical systems, due to the pole-zero structure, and also require much less parameters to achieve the same performance level of FIR filters. Unfortunately, these good characteristics come along with some possible drawbacks inherent to adaptive filters with recursive structure such as filter/algorithm stability, convergence to biased and/or local minimum solutions, and slow convergence. Consequently, many new algorithms for adaptive IIR filtering have been proposed in the literature attempting to overcome these problems. Very good summaries on adaptive IIR filters can be found in [7], [40], [59], [60]. In [40], Johnson presented a tutorial on adaptive IIR filtering techniques highlighting the common theoretical basis between adaptive filtering and system identification. This work was the first attempt to unify the concept and the terminology used in the fields of adaptive control and adaptive filtering. Later in 1989, Shynk [60] gave a tutorial on adaptive IIR filtering that deals with different algorithms, error formulations, and realizations. In 1999, Cousseau [7] published a report that briefly reviewed some recent results on adaptive IIR filtering and discussed the research being developed in this matter.

Initially, most of the works on adaptive IIR filters made use of the canonic direct-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

form realization due to its simple implementation and analysis. However, due to some inherent problems of the adaptive IIR filters such as stability monitoring requirement and slow speed of convergence, different realizations were studied attempting to overcome the limitations of the direct-form structure. Among these alternative structures, the cascade, parallel [45], [46], and lattice realizations [3] can be considered by their unique features. The study of alternative realizations is a research direction that has been vastly explored by many authors, specially during the recent years [6], [8], [37], [47].

1.2 Motivations

As mentioned before, the main difficulties that prevent the adaptive IIR filters from being used in real applications are the convergence to local minimum and more seriously, the instability during the adaptation process.

It is well known that a time-invariant digital filter can be characterized in many different *parametrizations* and that these parametrizations are theoretically equivalent since they represent the one and the same system. The important point is, however, that different parametrizations have different properties and for a given application, one parametrization can be better than another. Another very important issue is filter *structure*.[1] A filter can be implemented in different structures that are characterized with the same parametrization. In the time-invariant environment, the design of stable IIR filters can be done with a proper parametrization but it has nothing to do with the filter structure issue. This is no longer true for adaptive IIR filter design, where the filters are time-varying. In such a situation, *the performance of the adaptive filters depends not only on the parametrization but*

[1] A filter structure means a way to compute the filter output with a given input signal.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

also on the structure of the filters.

Traditionally, adaptive IIR filters are designed using the parametrization corresponding to the transfer function coefficients with the filter implemented in one of the *direct-forms*. These filters usually require stability monitoring, which is computationally costly, and have slow convergence speed as well as poor numerical properties (see, [9], [10]). The problem becomes more serious when the filters have poles near the unit circle. The most successful structures in adaptive IIR filter design are those with lattice-based parametrizations. See, e.g., [4] - [10]. One of the nice properties of the lattice-based parametrizations is that the stability can be easily monitored and hence the corresponding implementation is very simple. It was shown in [6] that the normalized lattice structure is also stable in time-varying environment.

The design of digital filters has been a well studied field [71] - [73]. It should be pointed out that in the stage of digital filter design, the transfer function coefficients are calculated to a high degree of precision assuming that the filters would be implemented with infinite precision and no dynamic range constraint. In real-time applications, finite precision implementation is the ultimately inevitable reality that any *a priori* well-designed digital filters have to face. Therefore, the performance of a well-designed digital filter can be totally different from the desired one and hence a computationally efficient structure in infinite precision sense is no more efficient when finite precision/finite word length (FWL) implementation is concerned.

The FWL effects have been a very important issue in digital filter implementation for more than three decades. These effects are usually classified into two categories: parameter perturbation and roundoff noise. Many results have been achieved in terms of finding optimal structures that optimize a certain FWL effect related cost function such as sensitivity measure and roundoff noise gain [74], [75]. The basic idea

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

behind this approach is that for a given digital filter there exist a number of different structures. Theoretically, they are equivalent since they represent the same system transfer function. Different structures, however, have different numerical properties and for a given application (measure or criterion) one structure can be better than another.

It should be pointed out that the FWL effects on time-invariant filters have been well studied, while such effects on adaptive filters have received a little attention. Adaptive filtering, as an on-line processing, is usually implemented with a digital signal processor of finite precision for real-time applications and therefore, it is important to investigate these effects on the performance of a well-designed adaptive algorithm/system.

1.3 Objectives and contributions

The main objective of this thesis is two-fold. The first one is to investigate the effects of parametrizations and structures on the performance of adaptive IIR filters, particularly on the stability aspect, and hence to derive new adaptive algorithms. The second one is to examine how a well-designed adaptive algorithm can be degraded when implemented with finite precision considerations.

A number of new results have been achieved during the three-year's hard work. The main contributions reported in this thesis are summarized as follows.

- *An efficient adaptive state-space digital filter*

  A simple state-space realization is derived. This structure is computationally efficient and can be optimized to reduce FWL effects. More interestingly, this structure is adopted as an adaptive state-space filter. It is shown that such

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

an adaptive filter has much better performance than the classical adaptive state-space filters in terms of stability and FWL behavior.

- *Normal realization-based adaptive filters*

  It is shown that when an adaptive filter can be implemented with a normal state-space realization, the corresponding adaptive algorithm is bounded-input bounded-output (BIBO) stable if the instantaneous poles of the filter are all inside the circle $|z| < \gamma$ with γ a constant satisfying $0 < \gamma < 1$. As one of the applications of this new result, a novel adaptive constrained IIR notch filter is derived, where the filter is parametrized using the frequency vector and implemented in a cascade form with each subfilter realized with a normal state-space realization. Normal realizations have a minimal pole sensitivity and are free of overflow oscillations. Comparison between the proposed normal realization and the lattice notch filter shows that the normal realization has less computational cost than the lattice one.

- *An input balanced realization-based adaptive filter*

  It is well known that for any (linear) digital filter there exists a class of state-space realizations, called input balanced realizations. These realizations have many nice numerical properties, among which is the inherent dynamic scaling that makes the state variables be constrained within a certain range. This property is proved extremely important for digital filter implementation. An input balanced realization-based adaptive filter is derived. This realization can be characterized with two new parameterizations, with which the stability of the (time-invariant) filters can be controlled easily. It is shown that an adaptive filter derived based on any of these two parameterizations is BIBO stable under some mild conditions. Simulations show that the performance of such an adaptive filter is comparable with the famous normalized lattice-based

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

one.

- *Adaptive IIR filtering for direct frequency estimation*

  Frequency estimation is required in many applications. Such a problem can be attacked with adaptive constrained IIR filtering. Two new adaptive algorithms are proposed for direct frequency estimation, where the constrained notch filter is parametrized with its notch frequency. Compared with some of the existing adaptive frequency estimation algorithms, the proposed algorithms have several better properties in terms of stability behavior and computational efficiency. The performance analysis of one algorithm is carried out in terms of steady-state estimation bias, mean square error, and tracking behavior.

1.4 Organization of the thesis

An outline of this thesis is given as follows.

Chapter 2 is devoted to giving some preliminaries that are needed for the development, including the concepts of parametrization and structure, FWL effect analysis, and adaptive systems. One of the objectives in Chapter 3 is to demonstrate the FWL effects on adaptive IIR filters. It is argued that to reduce the FWL effects it is better to implement an adaptive filter with those structures which are well scaled and characterized with a more robust parametrization rather than the direct-form-based ones. A new adaptive state-space filter is also derived in this chapter, which shows better performance than the classical adaptive state-space filters, especially when the initial estimate of the filter is in the neighborhood of the true one. A class of state-space realizations, called normal realizations, is studied in Chapter 4. It is shown that this class of realizations is BIBO stable under some mild conditions. As a special class of IIR adaptive filters, the constrained notch filters are then studied in

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

details. A new adaptive algorithm is derived, in which the constrained notch filter is characterized with the frequency vector and is cascaded into a series of second-order subfilters implemented with a normal realization. One of the remarkable properties of this algorithm, as mentioned before, is that it is BIBO stable. Comparison is made between the normal realization and the lattice notch filter in terms of computational complexity. Some numerical examples and simulations are presented to illustrate the performance of the proposed adaptive algorithm. At the end of this chapter, the application of this algorithm to speech processing is also discussed. Chapter 5 focuses on the general IIR adaptive filters, concentrating on the stability issue. In this chapter, an input balanced realization-based adaptive filter is derived. This realization is parametrized indirectly using a set of parameters, with which the stability of the (time-invariant) filter can be controlled easily with a set of linear inequalities. An alternative pameterization is also proposed. With this parametrization, the stability region of digital filters is actually the entire parameter space. It is shown that the adaptive filter developed based on these parametrizations is BIBO stable under some mild conditions. Several numerical examples and simulations are given to illustrate the performance of the adaptive algorithms. An application of this adaptive filter is also investigated for echo cancellation. The adaptive notch filters discussed in Chapter 4 can be used for frequency estimation. Chapter 6 is devoted to the problem of direct frequency estimation, where two new adaptive algorithms are derived. Performance analysis of one algorithm is carried out in terms of stability, computational efficiency, and steady-state behavior as well as tracking capability. Simulations are given to confirm the theoretical analysis and comparison is made between our proposed algorithms and some of the existing algorithms. Finally, some concluding remarks and possible directions for future work are given in Chapter 7.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 2

Preliminaries

We note that all classes of adaptive systems have a common feature - they can be extracted as a block diagram depicted in Figure 2.1, where $r(n) = y_0(n) + v(n)$ is an available reference signal, often called primary signal in adaptive signal processing, with $y_0(n)$ and $v(n)$ the desired signal and a noise sequence, respectively, $y(n)$ is the output of a model excited with a given input $u(n)$, and the adaptive algorithm is used to drive the model parameters to their optimal value in a certain sense. A typical filtering problem is how to estimate the desired signal $y_0(n)$ with the reference signal $r(n)$ and a given signal $u(n)$. This can be done by minimizing a performance criterion that is based on the prediction error $e(n)$ (sometimes called estimation error).

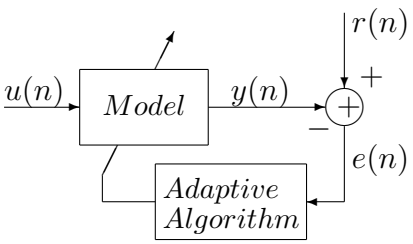


Figure 2.1: Block diagram of a general adaptive system.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

In many applications, the model is a linear time-invariant (LTI) filter described with its transfer function $H(z^{-1})$

$$H(z^{-1}) = \frac{b_0 + b_1 z^{-1} + ... + b_{K-1} z^{-(K-1)} + b_K z^{-K}}{1 + a_1 z^{-1} + ... + a_{K-1} z^{-(K-1)} + a_K z^{-K}} \triangleq \frac{N(z^{-1})}{D(z^{-1})}, \tag{2.1}$$

where $\{a_k, b_k\}$ are constants, called transfer function coefficients/parameters, and z^{-1} is the backward operator, e.g., $z^{-1}u(n) \triangleq u(n-1)$. When $a_k = 0$, $\forall k$, the corresponding $H(z^{-1})$ is called an FIR filter, otherwise it is an IIR filter.

The corresponding input-output (I/O) relationship is

$$y(n) = -\sum_{k=1}^{K} a_k y(n-k) + \sum_{k=0}^{K} b_k u(n-k). \tag{2.2}$$

A system is said BIBO stable if the output of the system is bounded for any bounded input signal. It is well known that a causal LTI system $H(z^{-1})$ is BIBO stable if and only if its poles, which are the roots of the polynomial $z^K D(z^{-1})$, are all inside the unit circle $|z| = 1$.

Denote $e_{op}(n)$ as the prediction error signal with the *output error* scheme, one has[1]

$$e_{op}(n) \triangleq r(n) - y(n) = r(n) - H(z^{-1})u(n), \tag{2.3}$$

This error signal depends on how to choose the transfer function $H(z^{-1})$. The optimal filtering is to find a stable transfer function such that a filtered version of $e_{op}(n)$, denoted as $e(n)$, is minimized:

$$\min_{H(z^{-1}) \in S} \sum_{n} |e(n)|^2, \tag{2.4}$$

[1]Throughout this thesis, we denote $F(z^{-1})\nu(n)$ as the output of the system $F(z^{-1})$ excited with $\nu(n)$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

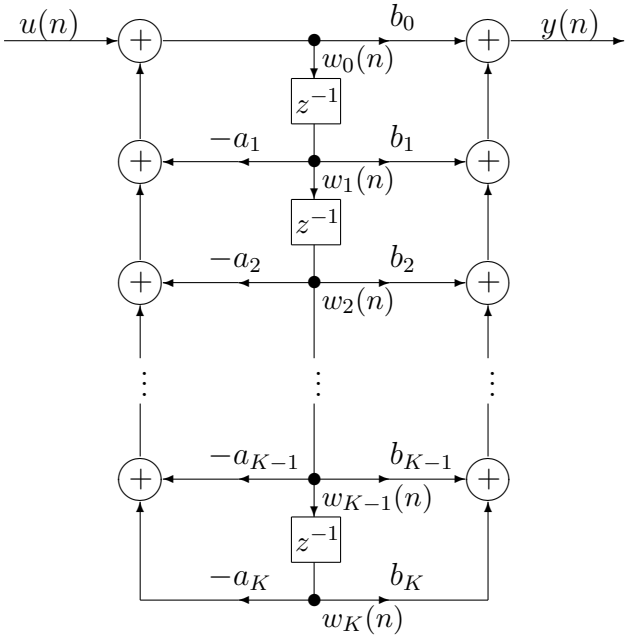


Figure 2.2: Direct-form II structure.

where S is the set of all possible stable transfer functions of order K.

2.1 Filter structures and parametrizations

A filter *structure* means a way to compute the output $y(n)$ with a given input $u(n)$. So, the input-output relationship (2.2) is a structure of $H(z^{-1})$, usually referred to *direct-form I* (DFI). It is well known that for a given $H(z^{-1})$, there exist a number of different structures. Figure 2.2 shows an alternative way to compute $y(n)$, called *direct-form II* (DFII) structure.

Denote

$$\theta_d \triangleq \begin{bmatrix} a_1 & a_2 & \cdots & a_K & b_0 & b_1 & \cdots & b_K \end{bmatrix}^{\mathcal{T}}, \tag{2.5}$$

where $\mathcal{T}$ denotes the transpose operator. One can see that the filter $H(z^{-1})$ is uniquely characterized with the parameter vector θ_d. This is a minimal[2] *parametriza-*

[2]For the class of Kth-order IIR filters, a parametrization θ is said *minimal* if the dimension of θ is $2K+1$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

tion of the filter. It is well known that $H(z^{-1})$ can be characterized with many different parametrizations. In fact, applying any one-to-one mapping to θ_d one can obtain a new parametrization.

Let θ be the parameter vector corresponding to a parametrization. We use $H(\theta, z^{-1}) = \frac{N(\theta,z^{-1})}{D(\theta,z^{-1})}$ to denote the fact that the transfer function is parameterized using this parametrization.

One observes that both DFI and DFII structures are directly parametrized using θ_d. A parametrization is called *a direct parametrization* of a given structure if the elements in the parameter vector are directly from the structure parameters. Clearly, θ_d is a direct parametrization of the two DF structures.

The choice of parametrizations and structures of an adaptive filter can influence the computational complexity (amount of arithmetic operations per iteration) of the processing and also the necessary number of iterations to achieve a desired performance level. Moreover, it is strongly related to the stability and convergence properties of the adaptive filters.

2.1.1 State-space realizations of digital filters

It is well known that the *state-space* description is a powerful tool in system theory. Although the state-space description of a system still involves a relationship between the input and output signals, it also involves an additional set of variables, called *state variables*. The state variables provide information about all the internal signals in the system. As a result, the state-space description provides a more detailed description of the system than the input-output description. The state-space techniques can be applied to nonlinear systems, time-variant systems, and multiple input-multiple output systems. In this thesis, the emphasis is on discrete-time linear

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

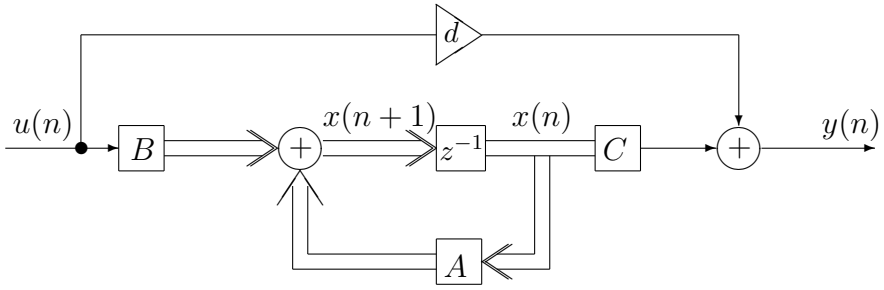


Figure 2.3: Block diagram of the state-space description of an LTI system.

systems .

Consider an LTI filter $H(z^{-1})$ given by (2.1). The state-space description is given by

$$\begin{cases} x(n+1) &= Ax(n) + Bu(n) \\ y(n) &= Cx(n) + du(n), \end{cases} \tag{2.6}$$

where $x(n) \in \mathcal{R}^{K\times 1}$ is the state vector with elements $x_k(n)$, and $A = \{a_{lk}\} \in \mathcal{R}^{K\times K}$, $B = \{\tilde{b}_k\} \in \mathcal{R}^{K\times 1}$, $C = \{c_k\} \in \mathcal{R}^{1\times K}$, $d = b_0 \in \mathcal{R}$ are constant matrices. $R \stackrel{\triangle}{=} (A, B, C, d)$ is called a (state-space) realization of $H(z^{-1})$ and satisfies

$$H(z^{-1}) = d + C(zI - A)^{-1}B. \tag{2.7}$$

In the sequel, a state-space realization R refers to the structure (2.6) depicted in Figure 2.3, where the double lines represent vector quantities and the blocks represent the vector or matrix coefficients.

It is interesting to note that for one and the same filter there exists an infinite set of realizations, denoted as

$$S_H \stackrel{\triangle}{=} \{R:\ (A, B, C, d)\ satisfies\ (2.7)\}.$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

One of the canonical realizations is the controllable realization, denoted as $R_c \in S_H$:

$$A_c = \begin{bmatrix} -a_1 & -a_2 & \dots & -a_{K-1} & -a_K \\ 1 & 0 & \dots & 0 & 0 \\ & & \vdots & & \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix}, \; B_c = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \; C_c = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_K \end{bmatrix}^{\mathcal{T}}, \quad (2.8)$$

where $c_k \stackrel{\triangle}{=} b_k - b_0 a_k$, $\forall k$. S_H can be characterized with

$$A = T^{-1} A_c T, \quad B = T^{-1} B_c, \quad C = C_c T, \quad (2.9)$$

where $T \in \mathcal{R}^{K \times K}$ is any nonsingular real valued matrix, performing similarity transformation. Clearly, the controllable realization/structure R_c is parametrized directly with

$$\theta_c \stackrel{\triangle}{=} \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_K & b_0 & C_c \end{bmatrix}^{\mathcal{T}}, \quad (2.10)$$

while θ_d is an *indirect* parametrization of this structure.

2.1.2 Normalized lattice structure

Any filter structure can be implemented/represented with the following revised state-space realization:

$$\begin{cases} \begin{bmatrix} x(n+1) \\ w(n) \end{bmatrix} = Q \begin{bmatrix} x(n) \\ u(n) \end{bmatrix} \\ y(n) = \bar{h}^{\mathcal{T}} \begin{bmatrix} x(n+1) \\ w(n) \end{bmatrix}, \end{cases} \quad (2.11)$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $Q \in \mathcal{R}^{(K+1)\times(K+1)}$ and $\bar{h} \in \mathcal{R}^{K\times 1}$ are constant, while $w(n)$ is an intermediate (auxiliary) signal.

For the DFII structure depicted in Figure 2.2, which is *directly* parameterized with θ_d, one has

$$Q = \begin{bmatrix} -\bar{a}^{\mathcal{T}} & 1 \\ I_K & 0_K \end{bmatrix}, \ \bar{h}^{\mathcal{T}} = \begin{bmatrix} b_0 & b_1 & \cdots & b_K \end{bmatrix}, \ \bar{a}^{\mathcal{T}} = \begin{bmatrix} a_1 \cdots a_k \end{bmatrix}, \tag{2.12}$$

where I_K denotes the identity matrix of order K, while 0_K is the zero column vector of dimension K.

Lattice-based structures have found many applications in speech signal processing and digital filter design/implementation. Among these structures, the normalized lattice structure depicted in Figure 2.4 has many nice properties. See, e.g., [6], [49]. This structure is directly parameterized by

$$\theta_l \triangleq \begin{bmatrix} \phi_1 & \cdots & \phi_K & \bar{h}^{\mathcal{T}} \end{bmatrix}^{\mathcal{T}}. \tag{2.13}$$

In [6], it is shown that the normalized lattice filter is equivalent to the revised state-space realization (2.11) with Q an *orthogonal* matrix, having the following factorization form

$$Q = Q_1 Q_2 \cdots Q_k \cdots Q_{K-1} Q_K \tag{2.14}$$

with Q_k the identity matrix except

$$Q_k(k:k+1, k:k+1) = \begin{bmatrix} -\sin\phi_k & \cos\phi_k \\ \cos\phi_k & \sin\phi_k \end{bmatrix}, \ \forall k, \tag{2.15}$$

where $M(i_1 : i_2, j_1 : j_2)$ denotes the $(i_2 - i_1 + 1) \times (j_2 - j_1 + 1)$ submatrix of M, resulting from deletion of all the (i, j)th elements of M with (i, j) not belonging to

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

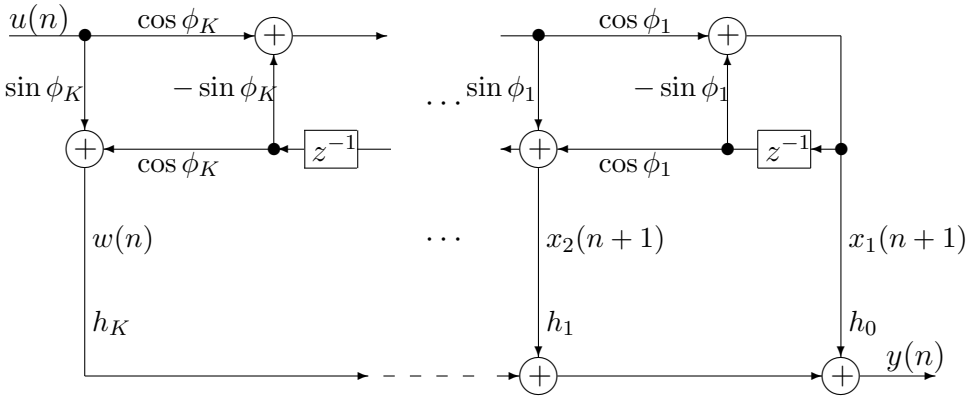


Figure 2.4: The tapped-state normalized lattice structure.

$\{i_1 \le k \le i_2,\ j_1 \le l \le j_2\}$.

It is easy to see that the equivalent (standard) state-space realization to (2.11) is

$$\begin{cases} A &= Q(1:K, 1:K) \\ B &= Q(1:K, K+1) \\ C &= \bar{h}^{\mathcal{T}}(1:K)A + \bar{h}(K+1)Q(K+1, 1:K) \\ d &= \bar{h}^{\mathcal{T}}(1:K)B + \bar{h}(K+1)Q(K+1, K+1). \end{cases} \tag{2.16}$$

A rational transfer function can be characterized with many different parametrizations. One parametrization is totally equivalent to another if the mapping between them is one-to-one. The important point is that different parametrizations have different properties. For example, θ_l is better than θ_d in terms of stability checking. In fact, $H(\theta_l, z^{-1})$ is stable if and only if

$$|\phi_k| < \frac{\pi}{2},\ \forall k, \tag{2.17}$$

and hence the stability can be checked very easily in the θ_l-parametrization, which is not the case for the θ_d-parametrization.

For a given LTI filter, there exist many different structures which are theoretically

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

equivalent, but different structures have different numerical properties. Some of the properties have been exploited in filter implementation with finite word length considerations.

2.2 Finite precision implementation of LTI filters

Let $H(z^{-1})$ be a well-designed single-input-single-output (SISO) LTI digital filter with $H(z^{-1})$ given by (2.1). This filter, as mentioned before, can be implemented in many different structures.

When the filter is implemented in a given structure with a digital device of finite word length, the actual performance of this filter may degrade greatly. The degrees of such degradation are traditionally measured by the transfer function sensitivity and roundoff noise gain.

2.2.1 Sensitivity measure

Denote $\{\tau_k\}$ as the parameter set of a filter structure and

$$S_{FWL} \stackrel{\triangle}{=} \{\tau_k^*\} \in \{\tau_k\}$$

as the set of the parameters τ_k^* which can be represented exactly with a given finite word length implementation (say, of 16 bits). For an actual implementation of this structure, these parameters, which have been calculated to very high precision at the design stage and hence usually do not belong to S_{FWL}, have to be truncated into an FWL format. Therefore, the actually implemented transfer function, denoted by

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$\tilde{H}(z^{-1})$, is different from $H(z^{-1})$. Using the first order approximation,

$$\Delta H(z^{-1}) \triangleq \tilde{H}(z^{-1}) - H(z^{-1}) = \sum_{\tau_k \notin S_{FWL}, \forall k} \frac{\partial H(z^{-1})}{\partial \tau_k} \Delta\tau_k,$$

where $\Delta\tau_k$ is the FWL error of the parameter τ_k. In the traditional FWL analysis, the parameter errors are modelled as zero mean uniformly distributed independent random variables with a variance σ_0^2 [88], [89]. Keeping this in mind, the variance of the degradation $\Delta H(z^{-1})$ is

$$\epsilon^2(\omega) \triangleq E[|\Delta H(e^{-j\omega})|^2] = \sum_{\tau_k \notin S_{FWL}, \forall k} |\frac{\partial H(e^{-j\omega})}{\partial \tau_k}|^2 \sigma_0^2,$$

where $E[.]$ denotes statistical average operation.

Since $\epsilon^2(\omega)$ is a function in frequency, the overall behavior of this function can be measured by the following

$$\frac{1}{2\pi} \int_0^{2\pi} \epsilon^2(\omega) d\omega = M_{L_2} \sigma_0^2$$

with M_{L_2} being called sensitivity measure:

$$M_{L_2} \triangleq \sum_{\tau_k \notin S_{FWL}, \forall k} ||\frac{\partial H(z^{-1})}{\partial \tau_k}||_2^2, \tag{2.18}$$

where $||.||_2$ is the L_2-norm defined below:

$$\begin{aligned} \| S(z) \|_2 &\triangleq \left[\frac{1}{2\pi} \int_0^{2\pi} \sum_{i=1}^{l} \sum_{k=1}^{m} |s_{ik}(e^{j\omega})|^2 d\omega \right]^{1/2} \\ &= \left[tr(\frac{1}{j2\pi} \oint_{|z|=1} S(z) S^{\mathcal{H}}(z) z^{-1} dz) \right]^{1/2} \end{aligned} \tag{2.19}$$

with $S(z) = \{s_{ik}(z)\} \in \mathcal{C}^{l \times m}$ any complex matrix-valued function of the complex variable z, and $\mathcal{H}$, $tr(\cdot)$ denoting the conjugate-transpose and trace operators, re-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

spectively.

Let $S(z) = D + L(zI - \Phi)^{-1}K$. It can be shown that

$$\|S(z)\|_2^2 = tr[DD^{\mathcal{T}} + LW_cL^{\mathcal{T}}] = tr[D^{\mathcal{T}}D + K^{\mathcal{T}}W_oK], \tag{2.20}$$

where W_c and W_o are the controllability and observability Gramians of the realization (Φ, K, L, D), respectively, which are the solutions of the Lyapunov equations

$$W_c = \Phi W_c \Phi^{\mathcal{T}} + KK^{\mathcal{T}}, \quad W_o = \Phi^{\mathcal{T}} W_o \Phi + L^{\mathcal{T}}L. \tag{2.21}$$

When the filter is implemented with a realization (A, B, C, d), it can be shown [91] that M_{L_2} can be evaluated with

$$\begin{aligned} M_{L_2} &= \sum_{l=1}^{K}\sum_{k=1}^{K} \phi(a_{lk}) \begin{bmatrix} C & \mathbf{0} \end{bmatrix} R_{lk} \begin{bmatrix} C & \mathbf{0} \end{bmatrix}^{\mathcal{T}} \\ &\quad + \sum_{k=1}^{K} [\phi(\tilde{b}_k) W_o(k,k) + \phi(c_k) W_c(k,k)] + \phi(d), \end{aligned} \tag{2.22}$$

where $\phi(x) = 0,\ \forall x \in S_{FWL}$, otherwise $\phi(x) = 1$, and R_{lk} is the solution of the following Lyapunov equation

$$R_{lk} - \begin{bmatrix} A & e_l e_k^{\mathcal{T}} \\ \mathbf{0} & A \end{bmatrix} R_{lk} \begin{bmatrix} A & e_l e_k^{\mathcal{T}} \\ \mathbf{0} & A \end{bmatrix}^{\mathcal{T}} = \begin{bmatrix} \mathbf{0} \\ B \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ B \end{bmatrix}^{\mathcal{T}}, \ \forall (l,k) \tag{2.23}$$

with e_m denoting the mth elementary (column) vector, whose elements are all zeros except the mth which is one.

Though for a given realization, one can compute this sensitivity measure, how to minimize M_{L_2} within the realization space S_H is very difficult and, as far as we know, this is still an open problem. When, however, S_{FWL} of the realizations is nil

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

(empty), one has

$$M_{L_2} = \| \frac{\partial H}{\partial A} \|_2^2 + \| \frac{\partial H}{\partial B} \|_2^2 + \| \frac{\partial H}{\partial C^{\mathcal{T}}} \|_2^2 + \| \frac{\partial H}{\partial d} \|_2^2,$$

where $\frac{\partial H(z^{-1})}{\partial A}$, $\frac{\partial H(z^{-1})}{\partial B}$, $\frac{\partial H(z^{-1})}{\partial C^{\mathcal{T}}}$ and $\frac{\partial H(z^{-1})}{\partial d}$ are the sensitivity matrices and can be computed with [75], [78]

$$\begin{cases} \frac{\partial H(z^{-1})}{\partial A} = G(z^{-1})F^{\mathcal{T}}(z^{-1}), & \frac{\partial H(z^{-1})}{\partial B} = G(z^{-1}) \\ \frac{\partial H(z^{-1})}{\partial C^{\mathcal{T}}} = F(z^{-1}), & \frac{\partial H(z^{-1})}{\partial d} = 1 \end{cases} \tag{2.24}$$

with

$$\begin{cases} F(z^{-1}) = (zI - A)^{-1}B \\ G(z^{-1}) = (zI - A^{\mathcal{T}})^{-1}C^{\mathcal{T}}. \end{cases} \tag{2.25}$$

The corresponding optimal sensitivity problem has been solved [75], [92].

2.2.2 Roundoff noise gain

A rounding operation has to be applied if less-than-double precision fixed-point arithmetic is utilized. Assuming rounding occurs *after multiplication* (RAM), a variable, say x, computed with a multiplication, has to be replaced by its quantized version, denoted as $q[x]$, in the ideal computation model. The difference $q[x] - x$ is the corresponding roundoff noise, which is usually modeled as a white noise sequence and statistically independent of those produced by other sources.

When the filter is implemented with (2.6), it can be shown [74] - [76] that the variance of the output error due to the roundoff of the states is given by

$$\sigma_{\Delta y}^2 = [tr(W_o Q) + m_{K+1}]\sigma_0^2,$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where σ_0^2 is a constant, determined by the word length used for representing the states,

$$Q = diag(m_1, \cdots, m_k, \cdots, m_K)$$

with $m_k = \sum_{l=1}^{K} \psi(a_{kl}) + \psi(\tilde{b}_k)$, where $\psi(x) = 0$ if $x = 0, \pm 1$, otherwise $\psi(x) = 1$, which is actually the number of nontrivial parameters in the kth row vector of $\begin{bmatrix} A & B \end{bmatrix}$ for $k = 1, \cdots, K$, and $m_{K+1} = \sum_{l=1}^{K} \psi(c_l) + \psi(d)$ the number of nontrivial parameters in the row vector $\begin{bmatrix} C & d \end{bmatrix}$. The roundoff noise gain is defined as

$$G \triangleq \frac{\sigma_{\Delta y}^2}{\sigma_0^2} = tr(W_o Q) + m_{K+1}. \tag{2.26}$$

It is well known [83], [93] that in an implementation structure, all the signals should be sustained within a certain dynamic range in order to avoid overflow. Under the assumption that the input $u(n)$ and the output $y(n)$ of the filter are properly pre-scaled, the only signals which may overflow are the states $\{x_k(n)\}$, which, therefore, have to be scaled. There exist different scaling schemes for preventing variables from overflow. The popularly used ones are the l_2- and l_∞-scalings. In what follows, we will concentrate on the l_2-scaling scheme.

The l_2-scaling means that each state variable should have a unit variance when the input is a white noise with a unit variance. This can be achieved if [74], [76]

$$W_c(i, i) = 1, \forall i. \tag{2.27}$$

(2.27) defines a subset, denoted as $S_H^{l_2}$, of the realization set S_H. The optimal roundoff noise realization problem is to find those realizations in $S_H^{l_2}$ that minimize G given by (2.26):

$$\min_{R \in S_H^{l_2}} G. \tag{2.28}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Just as the optimal sensitivity realization problem, (2.28) seems very difficult and is still an open problem. In [81], a numerical algorithm was proposed to solve a sub-optimal problem. We note that if the realization is fully parametrized (with nontrivial parameters[3]), that is $m_k = K + 1$, $\forall k$, one has

$$G = [tr(W_o) + 1](K + 1).$$

The optimal realization problem (2.28) was solved in [74] and [76] independently. The corresponding realizations are denoted by R_G.

It has been found that R_G yields a very good performance in terms of reducing FWL effects. It, however, has $(K + 1)^2$ parameters to implement, which makes the implementation very complicated and slows down the processing speed. Noting the fact that the trivial parameters in a sparse structure produce no FWL errors at all, researchers have put a lot of efforts in finding more efficient structures that can also beat the fully parametrized R_G in terms of roundoff noise gain [81] - [87].

2.3 Adaptive filters

For a given parametrization θ, the optimal filtering problem, defined by (2.4), is equivalent to finding a parameter vector, say θ_{opt},[4] such that the error signal is minimized subject to $H(\theta_{opt}, z^{-1})$ is stable. Such a problem can be solved with an adaptive way of the following form (see, [6], [7])

$$\theta(n + 1) = \theta(n) + e(n)P_\theta(n)\psi_\theta(n), \tag{2.29}$$

[3]By *trivial parameters* we mean those that are 0 and ± 1. Other parameters are, therefore, referred to *nontrivial parameters*.

[4]In many applications, the desired signal $y_0(n)$ can be modelled as the output of a transfer function in S_H, excited with $u(n)$, i.e., $y_0(n) = H(\theta^0, z^{-1})u(n)$, which will be assumed in the sequel. In that case, ideally one should have $\theta_{opt} = \theta^0$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $e(n)$ is a filtered version of $e_{op}(n)$ defined in (2.3), $P_\theta(n)$ is a matrix updated recursively, and $\psi_\theta(n)$, usually referred to regressor or gradient vector, is the output of a system $H_g(z^{-1})$, called gradient filter, excited with $u(n)$.

Both $H(z^{-1})$ and $H_g(z^{-1})$ can be implemented with many different structures. Denote $y(n) = \mathcal{S}[H, \theta, u(n)]$ as the output of the filter $H(z^{-1})$ parametrized with θ and excited by an input $u(n)$ using a structure $\mathcal{S}$. Similarly, one can denote $\psi_\theta(n) = \mathcal{G}[H_g, \theta, u(n)]$ with $\mathcal{G}$ indicating a structure of $H_g(z^{-1})$ parametrized with θ. The whole adaptive filter is then described with

$$\begin{cases} y(n) &= \mathcal{S}[H, \theta(n), u(n)] \\ \psi_\theta(n) &= \mathcal{G}[H_g, \theta(n), u(n)] \\ \theta(n+1) &= \theta(n) + e(n)P_\theta(n)\psi_\theta(n) \end{cases} \tag{2.30}$$

For convenience, in the sequel (2.30) is referred as an adaptive algorithm or an adaptive filter, which is actually a time-varying system and is said stable if the three sub-systems are BIBO stable.

It is very useful to notice that an adaptive filter/algorithm (2.30) mainly consists of the following elements.

- *Parametrization and structure*

  This is concerned with how the filters $H(z^{-1})$ and $H_g(z^{-1})$are characterized and the signals $y(n)$ and $\psi_\theta(n)$ are computed.

- *Adaptive scheme*

  This is how to define the filtered error signal $e(n)$ and hence $\psi_\theta(n) \triangleq -\frac{\partial e(n)}{\partial \theta}|_{\theta=\theta(n)}$ [7]. For example, $e(n) = e_{op}(n)$ is for the *output error* scheme, $e(n) = D(\theta(n), z^{-1})e_{op}(n)$ is for the *equation error* scheme, and $e(n) = \frac{D(\theta(n),z^{-1})}{D(\theta(n-1),z^{-1})}e_{op}(n)$ corresponds to the famous *Steiglitz-McBride* scheme.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

- *Numerical algorithm*

  It defines $P_\theta(n)$. The least mean square (LMS) and recursive *Gauss-Newton* prediction error (RPE) algorithms are the two most popularly used ones. For LMS, $P_\theta(n) = \mu I$ with μ a positive constant and I the identity matrix, and for RPE

$$\begin{cases} P_\theta(n) &= \varrho^{-1}(n)[P_\theta(n-1) - \frac{P_\theta(n-1)\psi_\theta(n)\psi_\theta^{\mathcal{T}}(n)P_\theta(n-1)}{\varrho(n)+\psi_\theta^{\mathcal{T}}(n)P_\theta(n-1)\psi_\theta(n)}] \\ \varrho(n+1) &= \varrho_\infty - [\varrho_\infty - \varrho(n)]\varrho_0 \end{cases} \tag{2.31}$$

  with ϱ_∞, ϱ_0 two constants governing the forgetting factor $\varrho(n)$.

These three aspects give us a structural and simple way to interpret, analyze, and study an adaptive algorithm. In fact, most of the adaptive algorithms can be visualized in this form, or in a slight variation of this generalization. Any new combination of the three aspects leads to a novel adaptive algorithm.

Many existing adaptive schemes can be found from the survey paper [7]. In this thesis, the *output error* scheme and the *Steiglitz-McBride* scheme are considered. As to the numerical algorithms, RPE and LMS will be used in simulations.

For a given adaptive filter, the performance analysis usually involves convergence, stability and implementation complexity. The third term is mainly concerned with the number of multiplications and additions required for computing each sample of the adaptive filter output.

2.3.1 Convergence

One of the fundamental issues concerned with the system (2.30) is the *convergence*. This involves the behavior of $\theta(n)$ when n goes to infinity and is traditionally investigated using the ordinary differential equation (ODE) approach. This approach, in

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

fact, is quite general, and draws from the field of stochastic approximation theory. It should be pointed out that this approach is not limited to direct-form filters, nor to gradient descent algorithms. Denote

$$\begin{cases} f(\tilde{\theta}) & \triangleq & E[e(n)\psi_{\tilde{\theta}}(n)] \\ R(\tilde{\theta}) & \triangleq & E[(1-\varrho(n))P_{\tilde{\theta}}^{-1}(n)] \\ G(\tilde{\theta}) & \triangleq & E[\psi_{\tilde{\theta}}(n)\psi_{\tilde{\theta}}^{\mathcal{T}}(n)], \end{cases} \tag{2.32}$$

where $\tilde{\theta} = \tilde{\theta}(t)$ is a function of a continuous variable t and $\theta(n)$ can be considered as the sampled version of $\tilde{\theta}(t)$. $E[\cdot]$ denotes the statistical average operator, $\psi_{\tilde{\theta}}(n) = \psi_{\theta}(n)_{|_{\theta=\tilde{\theta}}}$ and $P_{\tilde{\theta}}(n) = P_{\theta}(n)_{|_{\theta=\tilde{\theta}}}$ which is evaluated using (2.31). It was shown (see, e.g., [21]) that (2.29) - (2.31) converges to the solutions of the following ordinary differential equations

$$\begin{cases} \frac{d\tilde{\theta}}{dt} & = & R^{-1}(\tilde{\theta})f(\tilde{\theta}) & \triangleq & V_{\tilde{\theta}} \\ \frac{dR(\tilde{\theta})}{dt} & = & G(\tilde{\theta}) - R(\tilde{\theta}) & \triangleq & U_{\tilde{\theta}}(\tilde{\theta}), \end{cases} \tag{2.33}$$

where all terms are evaluated at constant $\tilde{\theta}(t)$. The convergence of a given adaptive algorithm depends on the three factors mentioned above.

Our interest at this point is to use the connections between adaptation algorithms and differential equations as a tool towards studying convergence. The ODE method dose not prove convergence of the algorithms to the global minimum of the cost function, except when there are no local minima. Furthermore, it does not provide any information concerning the *rate* of convergence, only that the algorithm will converge asymptotically.

The basic connections linking parameter adaptation algorithms with ODEs were pioneered by Ljung in two seminal papers [33], [34], where the adaptation gain $P_{\theta}(n)$ is time-varying and tending asymptotically to zero. Strong convergence results, which

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

allow one to prove parameter convergence with probability one, can be established for specific algorithmic structures [34]. Algorithms for which the adaptation step-size tends asymptotically to zero are sometimes called *vanishing gain* algorithms. Although such algorithms are suitable for signal environments that are truly stationary, they are theoretically incapable of tracking nonstationary signals, for the simple reason that the algorithm asymptotically turns off. This is one reason why *constant gain* algorithms, where the adaptation step-size $P_\theta(n)$ is held at a constant value, are often preferred (like LMS algorithm) in that case. Convergence in constant gain algorithms (when applicable) tends to be exponentially fast [35], whereas the convergence speed of vanishing gain algorithms (when applicable) is typically at a rate $1/L$, where L is the iteration number [3].

For the LMS algorithm, as indicated in [3], we have

$$\frac{d\tilde{\theta}}{dt} = f(\tilde{\theta}). \tag{2.34}$$

Consider the general update form (2.29) and assume that $u(n)$ and $y(n)$ are jointly stationary stochastic processes, and the filters which generate $e(n)$ and $\psi_\theta(n)$ are exponentially stable for all θ in some well defined domain. We have following theorem [3],

Theorem 1 *For sufficiently small $P_\theta(n) = \mu I$, a value θ^* in the parameter space is a convergent point in mean of the adaptation algorithm (2.29) if and only if this same θ^* is a stable stationary point of the differential equation (2.34).*

The stationary points of the differential equation (2.34) are those values of $\tilde{\theta}$ which make the driving term $f(\tilde{\theta})$ vanishes. This gives $\frac{d\tilde{\theta}}{dt} = 0$ so that the motion of the trajectory θ ceases; hence the term "stationary point". Let $\{\theta^*\}$ denote the set of

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

all such points, i.e.,

$$\{\theta^*\} = \{\tilde{\theta} : f(\tilde{\theta}) = 0\}. \tag{2.35}$$

Assuming that the set of stationary points can be deduced, two standard methods are used to test for convergence, and are summarized in [3], the *direct approach of Lyapunov* and the *indirect method of Lyapunov*. For the *direct approach of Lyapunov*, one needs to find a Lyapunov function for the differential equation, which is a positive valued function that is decreasing along all trajectories. The indirect method is known as local linearization. The idea is to linearize the differential equation about a stationary point, and then check whether the locally linearized system is stable. This approach is often the only resort for dealing with nonlinear differential equations, and the final conclusions rely on the stability theory of linear systems which amounts to an eigenvalue check of a feedback matrix [36].

2.3.2 Stability

Stability is one of the essential requirements in system design. As said before, an LTI filter can be implemented/described in a state-space realization (2.6). For this situation, the filter is BIBO stable if and only if all the eigenvalues of A matrix are inside the unit circle $|z| = 1$. An adaptive system described by (2.30) is actually a time-varying system and is said stable if the three sub-systems are BIBO stable.

Let

$$\begin{cases} x(n+1) &= A(n)x(n) + B(n)u(n) \\ y(n) &= C(n)x(n) + d(n)u(n) \end{cases} \tag{2.36}$$

be a state-space realization of a time-varying system with $u(n)$, $y(n)$ the input and output, respectively, or the equivalent state-space realization of a structure (say DFII) with which the system is implemented.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Denote $||v||_2$ as the l_2-norm (i.e., Euclidean norm) of a vector v. The homogeneous system

$$x(n+1) = A(n)x(n)$$

of (2.36) is said exponentially stable if there exist two non-negative constants η and $\xi < 1$ such that

$$||x(m)||_2 \leq \eta \xi^{m-n} ||x(n)||_2, \ \forall m \geq n$$

for any $||x(n)||_2 < +\infty$. The following theorem plays a very important role in stability analysis of time-varying systems [3], [102]:

Theorem 2 *The system described by (2.36) is BIBO stable if*

- *the elements in the realization are bounded (for all n) and*

- *the homogeneous system $x(n+1) = A(n)x(n)$ is exponentially stable.*

In an adaptive algorithm, the first condition in the above theorem can be easily ensured, while the second one is usually difficult to be guaranteed. The second condition in Theorem 2 is strongly related to the type of realizations/structures. It is well known that the adaptive IIR filters parameterized using θ_d with a direct-form structure yields a very bad performance due to the instability problem. In fact, all the direct-form structures correspond to an equivalent state-space representation in which the A matrix is A_c or $A_c^{\mathcal{T}}$. $A_c(n)$ usually does not satisfy the condition of exponential stability even if $A_c(n)$ has its eigenvalues all inside the unit circle for all n. This is the reason why the corresponding algorithm is easy to become unstable. Consider the following time-varying controllable realization R_c, where the A-matrix

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$A_c(n)$ is given by

$$A_c(2m-1) = \begin{bmatrix} 1.4400 & -0.9801 \\ 1 & 0 \end{bmatrix}, \; A_c(2m) = \begin{bmatrix} -0.8021 & -0.9801 \\ 1 & 0 \end{bmatrix}, \; \forall m,$$

where $A_c(2m-1)$ and $A_c(2m)$ correspond to A_c in (2.8), having poles at $z = 0.99e^{\pm j0.7565}$ and $z = 0.99e^{\pm j1.9879}$, respectively. It is easy to see that the corresponding homogeneous system yields $x(2m) = A_s^m x(0)$, where $x(0)$ is an initial state vector and $A_s \triangleq A_c(2m-1)A_c(2m)$, which has an eigenvalue of -2.7682. Therefore, $||x(2m)||_2$ may go to infinity and hence the system is unstable though the instantaneous eigenvalues of $A_c(n)$ are all on the circle $|z| = 0.99$, inside the unit circle.

For the normalized lattice structure parameterized using θ_l, the BIBO stability can be ensured under some modest conditions. See, e.g., [6]. One of the main contributions in this thesis is to provide a number of new structures which can guarantee the exponential stability. Based on these results, some new adaptive algorithms will be developed.

2.4 Summary

In this chapter, we have given some preliminaries that are needed for the development in the sequel, including the concepts of parametrization and structure, the optimal FWL structure design for LTI filters, and the adaptive algorithms. A number of new results will be presented in the chapters that follow.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 3

FWL Effects on Adaptive Recursive Digital Filters

An adaptive filter of form (2.30) consists of three sub-systems. Traditionally, the adaptive filter is parametrized using θ_d, in which the $\mathcal{S}$ and $\mathcal{G}$ of $H(z^{-1})$ and $H_g(z^{-1})$, respectively, are of direct-forms. These structures parametrized using θ_d, though very simple, usually have very poor numerical properties ([93], [94]). When implemented with FWL devices, a well-designed adaptive filter may not work properly. The FWL effect is more serious for IIR models than for FIR models. Generally speaking, the severity of FWL effects is strongly related to the distribution of the poles. When they are close to the unit circle and clustered together, the problem becomes extremely serious.

As mentioned in the previous chapters, for a linear system there exist a number of different structures. Theoretically, they are equivalent since they represent one and the same system. The most important point, however, is that different structures have different numerical properties and for a given application (measure criterion) one structure can be better than another, which leads to *optimal structure design*

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

problem.

The state-space representation provides a class of realizations for filter implementation. A lot of results have been achieved in exploiting those state-space realizations which can greatly reduce the FWL effects in LTI digital filters [75], [93], while the problem how to apply these realizations for designing adaptive filters has been studied by a few researchers. The first paper to deal with adaptive IIR state-space filters is perhaps by John *et al.* [52], where several realizations were examined by simulations, which showed that a properly chosen realization can yield a much better convergence speed than the traditional DFII structure parametrized using θ_d, but no systematic way on how to find a proper parametrization was given.

Compared with time-invariant filters, the FWL effects in adaptive filters have been studied by fewer researchers. Fan [97] examined the convergence behavior of the third equation in (2.30) for several adaptive algorithms towards a small perturbation on $\theta(n)$. One can see that the parameter updating equation is excited with the outputs $y(n)$ and $\psi_\theta(n)$ of the filters $H(z^{-1})$ and $H_g(z^{-1})$. Therefore, the FWL behavior of the adaptive filter (2.30) is strongly determined by $\mathcal{S}$, $\mathcal{G}$ and the parametrization θ used. As far as we know, there are no concrete work and even proper simulations reported yet.

The objective of this chapter is twofold. The first one is to demonstrate the FWL effects on the performance of the classical DFII-based adaptive algorithms with concrete examples. The second one is to develop a new adaptive state-space filter, which is comparable with the traditional canonical realizations in terms of implementation complexity. It is shown that this proposed adaptive state-space filter yields much better performance than the traditional DF-based ones when implemented with FWL considerations.

Throughout this chapter, the output error adaptive scheme is assumed. Therefore,

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$\psi_\theta(n) = \frac{\partial y(n)}{\partial \theta} = H_g(z^{-1})u(n)$.

3.1 Direct-form II adaptive filters

Traditionally, an adaptive IIR filter is parametrized using θ_d and the filter $H(z^{-1})$ and the gradient filter $H_g(z^{-1})$ are implemented in one of the direct-forms. With the consideration of computational efficiency, the DFII structure is usually used [108].

For the direct-form parametrization θ_d defined by (2.5), the gradient filter $H_g(z^{-1}) = \frac{\partial H(z^{-1})}{\partial \theta}$ is specified with

$$\begin{cases} \frac{\partial H(z^{-1})}{\partial a_k} &= -\frac{z^{-k}}{D(z^{-1})}H(z^{-1}), \quad k = 1, 2, \cdots, K \\ \frac{\partial H(z^{-1})}{\partial b_k} &= \frac{z^{-k}}{D(z^{-1})}, \qquad\qquad\quad k = 0, 1, \cdots, K, \end{cases} \tag{3.1}$$

where $H(z^{-1}) = \frac{N(z^{-1})}{D(z^{-1})}$ is given by (2.1). The filter output $y(n)$ and the gradient signals $\psi_{b_k}(n) = w_k(n)$, $\forall k$ are obtained from the DFII structure by Figure 2.2, while the gradient signals $\psi_{a_k}(n) = -\frac{z^{-k}}{D(z^{-1})}y(n)$, $\forall k$ can be computed with the same DFII structure specified by Figure 3.1.

In what follows, we will discuss the FWL effects on the DFII-based adaptive filters.

3.1.1 Pole sensitivities and stability

As mentioned in Chapter 2, the equivalent state-space realization of the DFII structure is the controllable realization R_c. The transition matrix A_c is of companion form and such a form has very poor stability robustness, especially when the dimension becomes large [94]. When the adaptive filter is implemented using the DFII structure, the corresponding transition matrix $A_c(n)$ usually does not satisfy the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

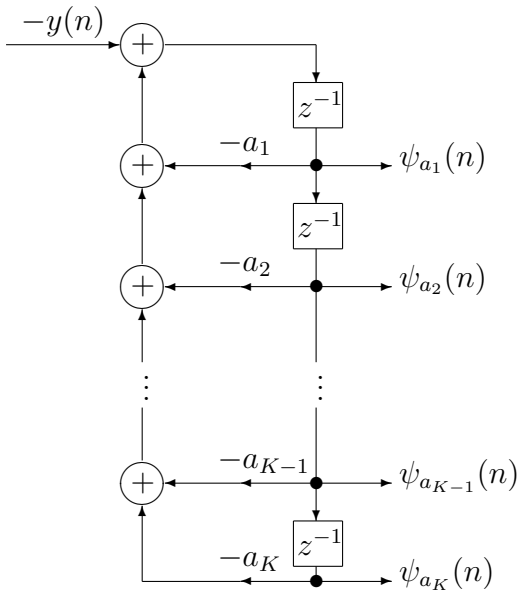


Figure 3.1: Direct-form II structure implementation of $\psi_{a_k}(n),\ \forall k$.

condition of exponential stability and hence the adaptive filter has a poor stability behavior.

The poor stability behavior of the DFII-based adaptive filters parametrized using θ_d can be seen from the angle of pole sensitivity. Let $\{\lambda_k\}$ be the set of the system poles. Then the denominator of the transfer function in z is

$$\bar{D} \triangleq z^K + \sum_{k=1}^{K} a_k z^{K-k} = \prod_{m=1}^{K} (z - \lambda_m). \tag{3.2}$$

Assume λ_k is not a repeated pole. According to the implicit function theorem, one has the following pole sensitivity (see [75])

$$\frac{\partial \lambda_k}{\partial a_l} = -\frac{\partial \bar{D}/\partial a_l}{\partial \bar{D}/\partial \lambda_k}\Big|_{z=\lambda_k} = -\frac{\lambda_k^{K-l}}{\prod_{m\neq k}(\lambda_k - \lambda_m)},\ \forall k, l. \tag{3.3}$$

One can see that the pole sensitivities of the system are very high if its poles are clustered together and close to the unit circle $|z| = 1$. This happens when the system is of narrow bandwidth. For example, for low-pass narrow bandwidth filters, their poles are located around $z = 1$. As observed by many researchers, it is very

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

difficult to make an adaptive system converge even the initial system is very close to its true one since a small increment $\Delta\theta_d(n) = \theta_d(n) - \theta_d(n-1)$ may drive the instantaneous poles outside the unit circle in such kind of situations. More seriously, when implemented with an FWL device, the parameter vector $\theta_d(n)$ of the adaptive filter has to be truncated/rounded at each iteration in order to sustain the word length requirement. If the transfer function of the true filter is very sensitive to such a parametrization, the adaptive filter can never converge. Therefore, we argue that *it would be better to use a parametrization to which the systems are insensitive.*

3.1.2 Effect of dynamic range

Another FWL effect is called roundoff noise due to the signal quantization. The severity of such an effect depends strongly on the filter structure used. This issue is also related to the so-called dynamical range. For an adaptive filter described in (2.30), one has $H(z^{-1})$ and $H_g(z^{-1})$ to implement. Any digital device has a limited dynamic range, denoted as DR, for the input, output, and (internal) state signals $s(n)$ (say $|s(n)| \leq DR$, $\forall n$ with $DR = 5$ volts). The signals within this range are then quantized with B_{dr} bits.

It is well known that the dynamic range of the state signals is structure dependent. In order to sustain the dynamic range of the device, the filter structures used for implementation have to be well-scaled. One of the popularly used scaling schemes is the l_2-scaling, which can be achieved if (2.27) is satisfied [75], [93].

It should be pointed out that the dynamic constraint (2.27) is not held inherently by most of the filter structures. For the DFII structure or equivalently, the controllable realization R_c, the dynamic range of the states varies with the filters. For the adaptive filter (2.30) when implemented using these structures, the dynamic range of

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

the states may vary dramatically during the adaptation and consequently, the states will be truncated when they are beyond the dynamic range specified by the FWL digital device. This may cause a serious problem to the adaptive filter. An DFII structure-based adaptive filter that works well without dynamic range constraint can not work properly at all once the dynamic range is taken into account. This issue, as far as we know, has not been considered in the literature.

As a conclusion, it is recommended that the adaptive filters be implemented in those well-scaled realizations.

3.2 Adaptive state-space filters

Consider the LTI filter $H(z^{-1})$, described with a realization (2.6):

$$\begin{cases} x(n+1) &= Ax(n) + Bu(n) \\ y(n) &= Cx(n) + du(n). \end{cases}$$

Depending on the structures of the matrices A, B, and C, the number of structure parameters varies. For the controllable realization R_c, it contains $2K+1$ parameters and can be parametrized directly using θ_c defined by (2.10). Generally speaking, for a given realization there are $(K+1)^2$ elements, $2K+1$ of which are independent. Assume that θ is the parameter vector, containing $2K+1$ independent elements, of a realization. Let's consider how to compute the gradient signals $\psi_\theta(n)$.

Applying z-transform to (2.6) leads to

$$X(z) = (zI - A)^{-1}BU(z), \quad Y(z) = CX(z) + dU(z),$$

where $X(z)$, $Y(z)$ and $U(z)$ are the z-transform of the state vector $x(n)$, the output

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$y(n)$ and input $u(n)$ of the filter. Noting $Y(z) = H(z)U(z)$ with $H(z) = d + C(zI - A)^{-1}B$, one has

$$\begin{cases} \frac{\partial Y(z)}{\partial A} = \frac{\partial H(z)}{\partial A}U(z), & \frac{\partial Y(z)}{\partial B} = \frac{\partial H(z)}{\partial B}U(z) \\ \frac{\partial Y(z)}{\partial C} = \frac{\partial H(z)}{\partial C}U(z), & \frac{\partial Y(z)}{\partial d} = \frac{\partial H(z)}{\partial d}U(z). \end{cases} \tag{3.4}$$

It then follows from (2.25) and $X(z) = F(z^{-1})U(z)$

$$\begin{cases} \frac{\partial Y(z)}{\partial A} = G(z)X^{\mathcal{T}}(z), & \frac{\partial Y(z)}{\partial B} = G(z)U(z) \\ \frac{\partial Y(z)}{\partial C^{\mathcal{T}}} = X(z), & \frac{\partial Y(z)}{\partial d} = U(z), \end{cases} \tag{3.5}$$

where $G(z^{-1}) = (zI - A^{\mathcal{T}})^{-1}C^{\mathcal{T}}$ as defined in (2.25).

Denote $\psi_V(n) \stackrel{\triangle}{=} \frac{\partial y(n)}{\partial V}$ for $V = B,\ C^{\mathcal{T}},\ d,\ A(:,k),\ \forall k$. Clearly,

$$\begin{cases} \psi_{A(:,k)}(n+1) = A^{\mathcal{T}}\psi_{A(:,k)}(n) + C^{\mathcal{T}}x_k(n),\ \forall k \\ \psi_B(n+1) = A^{\mathcal{T}}\psi_B(n) + C^{\mathcal{T}}u(n) \\ \psi_{C^{\mathcal{T}}}(n) = x(n) \\ \psi_d(n) = u(n). \end{cases} \tag{3.6}$$

Generally speaking, $\psi_\theta(n)$ can be computed with some equations of (3.6) for any given parametrization θ. One observes that (3.6) means that p state-space systems of dimension K have to be implemented if θ involves p elements that are positioned in different columns of the A-matrix. For example, the controllable realization R_c is minimally parametrized using $\theta_c = \begin{bmatrix} A_c(1,:) & C_c & d \end{bmatrix}^{\mathcal{T}}$ (see (2.8) and (2.10)). In this realization, the K parameters $\{-a_k\}$ are located in the first row of A_c. To compute the corresponding gradient signals, one needs the first equation (though only the first element of $\psi_{A(:,k)}(n)$ is needed) for $k = 1, 2, \cdots, K$ and the 3rd and 4th equations of (3.6).

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Note that the transposed R_c, denoted as $R_{ct} \triangleq (A_{ct}, B_{ct}, C_{ct}, d) \in S_H$:

$$A_{ct} = A_c^{\mathcal{T}},\ B_{ct} = C_c^{\mathcal{T}},\ C_{ct} = B_c^{\mathcal{T}}$$

is also directly parametrized using θ_c. In this situation, one has $\theta_c = \begin{bmatrix} A_{ct}^{\mathcal{T}}(:,1) & B_{ct}^{\mathcal{T}} & d \end{bmatrix}^{\mathcal{T}}$ and hence the corresponding gradient signals can be computed with

$$\begin{cases} \psi_{A_{ct}(:,1)}(n+1) &= A_c\psi_{A_{ct}(:,1)}(n) + B_c x_1(n) \\ \psi_{B_{ct}}(n+1) &= A_c\psi_{B_{ct}}(n) + B_c u(n) \\ \psi_d(n) &= u(n), \end{cases} \tag{3.7}$$

where $x_1(n)$ is the first state variable in the state-space realization R_{ct} with which the filter $H(z^{-1})$ is implemented. Clearly, the adaptive state-space filter implemented in this way is much more efficient in terms of reducing computational load.

This was observed in [52], where it is suggested to use the so-called *single-column* adaptive filter which is implemented in a realization (A, B, C, d) and parametrized using

$$\theta = \begin{bmatrix} A^{\mathcal{T}}(:,k) & C & d \end{bmatrix}$$

for some index k. In that case, the gradient signals can be simply computed with the 1st, 3rd and 4th equations of (3.6). The problem, however, is that it requires checking the observability of the pair (e_k, A) at each adaptation iteration, where e_k is the kth elementary vector of order K (whose elements are all zero except the kth one which is 1). This is not convenient and needs a burden of computation. In addition, the performance of the structures suggested in [52] depends strongly on the initial realization used. For detailed discussions, we refer to [52].

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

3.3 A new state-space realization with improved pole sensitivities

Simulations have shown that the DFII adaptive filter parametrized using θ_d and the adaptive filteres realized by the state-space realizations R_c, R_{ct}, both parametrized using θ_c, usually have very poor performance in terms of structure stability and parametrization sensitivity, especially when the true filter has poles close to the unit circle. In this section, we will propose a new adaptive state-space filter which is simple and can yield better performance.

3.3.1 Reparametrization

Let $p \triangleq K/2$ if K is even, $p \triangleq (K+1)/2$ if K is odd and

$$\rho_k(z) \triangleq \frac{z-\gamma_k}{\Delta_k}, \ k = 1, 2, \cdots, K-1, K, \tag{3.8}$$

where $\{\gamma_k\}$ and $\{\Delta_k\}$ are two sets of parameters to be discussed later. One then has a set of polynomials $\{q_k(z), \ k = 0, 1, \cdots, K\}$ obtained with the following recursive equations

$$\begin{cases} q_K(z) = 1, \quad q_{K-1}(z) = \rho_K(z) \\ q_{K-2m}(z) = \bar{\rho}_{p-m+1}(z) q_{K-2m+2}(z) \\ q_{K-2m-1}(z) = \rho_{K-2m}(z) q_{K-2m}(z) \end{cases} \tag{3.9}$$

for all $m = 1, 2, \cdots, p-1$, where

$$\begin{cases} \bar{\rho}_{p-m+1}(z) \triangleq \rho_{K-2m+1}(z)\rho_{K-2m+2}(z) + \eta_{p-m+1} \\ \eta_{p-m+1} \triangleq \frac{\bar{\eta}_{p-m+1}}{\Delta_{K-2m+1}\Delta_{K-2m+2}} \end{cases} \tag{3.10}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

with $\{\bar{\eta}_m\}$ a set of free parameters. It should be pointed out that when K is odd, $\rho_0(z) \triangleq 1$, $\bar{\eta}_1 = 0$ are assumed, which leads to $\bar{\rho}_1(z) = \rho_1(z)$ and hence $q_0(z) = \bar{\rho}_1(z)q_1(z)$.

It is interesting to note that

$$q_k(z) = [\prod_{l=k+1}^{K} \Delta_l^{-1}]\bar{q}_k(z), \ \forall k \tag{3.11}$$

where all $\bar{q}_k(z)$ are obtained using (3.9) with $\Delta_k = 1, \forall k$. Clearly, these polynomials are uniquely determined by the two sets of free parameters $\{\gamma_k\}$ and $\{\bar{\eta}_m\}$.

Denoting

$$V_a \triangleq \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_k \\ \vdots \\ a_K \end{bmatrix}, \quad V_b \triangleq \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \\ \vdots \\ b_K \end{bmatrix}, \quad \bar{z} \triangleq \begin{bmatrix} z^K \\ z^{K-1} \\ \vdots \\ z^{K-k} \\ \vdots \\ 1 \end{bmatrix}, \tag{3.12}$$

one has

$$V_q(z) \triangleq \begin{bmatrix} q_0(z) & q_1(z) & \cdots & q_{K-k}(z) & \cdots & q_K(z) \end{bmatrix}^{\mathcal{T}} = T_q\bar{z}, \tag{3.13}$$

where $T_q \in \mathcal{R}^{(K+1)\times(K+1)}$ is an upper triangular matrix with its kth row determined by the coefficients of $q_k(z)$ for $k = 1, 2, \cdots, K$ and $T_q(K+1, K+1) = 1$. It follows from (3.8) and (3.9) that

$$H(z^{-1}) = \frac{V_b^{\mathcal{T}}\bar{z}}{V_a^{\mathcal{T}}\bar{z}} = \frac{V_\beta^{\mathcal{T}}V_q(z)}{V_\alpha^{\mathcal{T}}V_q(z)} = \frac{\sum_{k=0}^{K}\beta_k q_k(z)}{q_0(z) + \sum_{k=1}^{K}\alpha_k q_k(z)}, \tag{3.14}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where

$$\begin{cases} V_\alpha \triangleq \begin{bmatrix} 1 & \alpha_1 & \cdots & \alpha_k & \cdots & \alpha_K \end{bmatrix}^{\mathcal{T}} = \kappa^{-1} T_q^{-\mathcal{T}} V_a \\ V_\beta \triangleq \begin{bmatrix} \beta_0 & \beta_1 & \cdots & \beta_k & \cdots & \beta_K \end{bmatrix}^{\mathcal{T}} = \kappa^{-1} T_q^{-\mathcal{T}} V_b \end{cases} \tag{3.15}$$

with $\kappa \triangleq \prod_{k=1}^{K} \Delta_k$ leading to $V_\alpha(1) = 1$.

Define

$$\begin{cases} B_{p-m}(z) & \triangleq & \beta_{K-2m-1}\rho_{K-2m}(z) + \beta_{K-2m} \\ A_{p-m}(z) & \triangleq & \alpha_{K-2m-1}\rho_{K-2m}(z) + \alpha_{K-2m} \end{cases} \tag{3.16}$$

for all possible m except $m = p-1$ when K is odd, for which, $A_1(z) = \alpha_1,\ B_1(z) = \beta_1$. The transfer function can finally be rewritten as

$$H(z^{-1}) = \frac{\beta_0 q_0(z) + \sum_{m=0}^{p-1} B_{p-m}(z) q_{K-2m}(z)}{q_0(z) + \sum_{m=0}^{p-1} A_{p-m}(z) q_{K-2m}(z)}. \tag{3.17}$$

Now, let us consider how to implement the filter based on the polynomial-parametrized transfer function (3.17). First of all, it follows from (3.17) that

$$\begin{aligned} y(n) &= \beta_0 u(n) + \sum_{m=0}^{p-1} [B_{p-m}(z) \frac{q_{K-2m}(z)}{q_0(z)} u(n) - A_{p-m}(z) \frac{q_{K-2m}(z)}{q_0(z)} y(n)] \\ &\triangleq \beta_0 u(n) + w_1(n). \end{aligned} \tag{3.18}$$

Noting $q_{K-2m}(z) = \bar{\rho}_{p-m+1}(z) q_{K-2m+2}(z)$ and $q_0(z) = \prod_{m=1}^{p} \bar{\rho}_m(z)$ (for any K), one can show that

$$\begin{aligned} w_1(n) &= \bar{\rho}_1^{-1}(z)[B_1(z)u(n) - A_1(z)y(n) + \sum_{m=0}^{p-2} [B_{p-m}(z) \frac{q_{K-2m}(z)}{\prod_{m=2}^{p} \bar{\rho}_m(z)} u(n) \\ &\quad - A_{p-m}(z) \frac{q_{K-2m}(z)}{\prod_{m=2}^{p} \bar{\rho}_m(z)} y(n)] \triangleq \bar{\rho}_1^{-1}(z)[B_1(z)u(n) - A_1(z)y(n) + w_2(n)]. \end{aligned}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

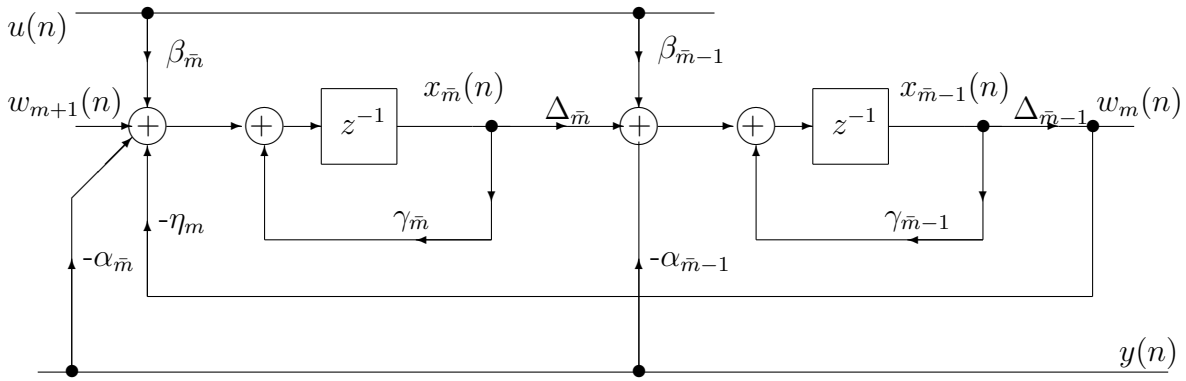


Figure 3.2: Implementation of (3.19), where $\bar{m} \triangleq K - 2p + 2m$ and $w_{K+1}(n) = 0$.

Using the same procedure, one can show that with $w_{p+1}(n) = 0$

$$w_m(n) \quad = \quad \bar{\rho}_m^{-1}(z)[B_m(z)u(n) - A_m(z)y(n) + w_{m+1}(n)], \ \forall m, \qquad (3.19)$$

which can be computed with the structure depicted in Figure 3.2.

Choose $\{x_k(n)\}$ indicated in Figure 3.2 as the state variables and denote $x(n)$ as the state vector. It can be shown that the corresponding state-space realization, denoted as $R_\rho \triangleq (A_\rho, B_\rho, C_\rho, \beta_0)$ is given by (for K even)

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$$\begin{cases} A_\rho = \begin{bmatrix} a_{11} & \Delta_2 & 0 & \cdots & 0 & 0 & 0 \\ a_{21} & \gamma_2 & \Delta_3 & \cdots & 0 & 0 & 0 \\ a_{31} & 0 & \gamma_3 & \cdots & 0 & 0 & 0 \\ a_{41} & 0 & -\eta_2\Delta_3 & \cdots & 0 & 0 & 0 \\ & & & \vdots & & & \\ a_{(K-1)1} & 0 & 0 & \cdots & 0 & \gamma_{(K-1)} & \Delta_K \\ a_{K1} & 0 & 0 & \cdots & 0 & -\eta_p\Delta_{(K-1)} & \gamma_K \end{bmatrix} \\ B_\rho = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \cdots \\ \beta_{K-1} \\ \beta_K \end{bmatrix} - \beta_0 \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdots \\ \alpha_{K-1} \\ \alpha_K \end{bmatrix}, \quad C_\rho = \begin{bmatrix} \Delta_1 \\ 0 \\ \cdots \\ 0 \\ 0 \end{bmatrix}^{\mathcal{T}}, \end{cases} \tag{3.20}$$

where

$$a_{11} = -\alpha_1\Delta_1 + \gamma_1,\ a_{21} = -\alpha_2\Delta_1 - \eta_1,\ a_{k1} = -\alpha_k\Delta_1,\ \forall k. \tag{3.21}$$

The transfer function from $u(n)$ to $y(n)$ is then

$$H(z^{-1}) = \beta_0 + C_\rho(zI - A_\rho)^{-1}B_\rho. \tag{3.22}$$

With (3.11), it can be shown that

$$A_\rho = T_{sc}\bar{A}_\rho T_{sc}^{-1},\ B_\rho = T_{sc}\bar{B}_\rho,\ C_\rho = \bar{C}_\rho T_{sc}^{-1}, \tag{3.23}$$

where $(\bar{A}_\rho, \bar{B}_\rho, \bar{C}_\rho, \beta_0)$ is the equivalent realization corresponding to $\Delta_k = 1,\ \forall k$ and

$$T_{sc} = diag(d_1, d_2, \cdots, d_K),\ d_k = \prod_{l=1}^{k} \Delta_l^{-1},\ \forall k. \tag{3.24}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Such T_{sc}^{-1} is actually a diagonal scaling similarity transformation.

As mentioned before, a structure has to be properly scaled in order to avoid signal overflow. The l_2-scaling (2.27) can be achieved if $W_c^\rho(k,k) = 1, \forall k$, where W_c^ρ is the controllability Gramian of $(A_\rho, B_\rho, C_\rho, \beta_0)$, which can be computed using MATALB command $dgram.m$.

Denote $\bar{W}_c^\rho$ as the controllability Gramian of $(\bar{A}_\rho, \bar{B}_\rho, \bar{C}_\rho, \beta_0)$. It follows from (3.23) - (3.24) that $W_c^\rho = T_{sc}\bar{W}_c^\rho T_{sc}^{\mathcal{T}}$. Therefore, the l_2-scaling can be achieved if $d_k^2\bar{W}_c^\rho(k,k) = 1, \forall k$, which leads to

$$\Delta_1 = \sqrt{\bar{W}_c^\rho(1,1)},\ \Delta_k = \sqrt{\frac{\bar{W}_c^\rho(k,k)}{\bar{W}_c^\rho(k-1,k-1)}},\ k = 2, \cdots, K.$$

In the sequel, the realization R_ρ is assumed to have been l_2-scaled. Therefore, one has

$$\begin{cases} \kappa &= \sqrt{\bar{W}_c^\rho(K,K)} \\ q_0(z) &= \sqrt{\frac{1}{\bar{W}_c^\rho(K,K)}}\bar{q}_0(z),\ q_K(z) = 1 \\ q_k(z) &= \sqrt{\frac{\bar{W}_c^\rho(k,k)}{\bar{W}_c^\rho(K,K)}}\bar{q}_k(z),\ k = 1, \cdots, K-1 \end{cases} \tag{3.25}$$

3.3.2 Pole sensitivity analysis

Clearly, any transfer function of order K can be implemented in the realization specified by (3.20), which is uniquely determined by the minimal parametrization

$$\theta_I = \begin{bmatrix} A_\rho^{\mathcal{T}}(:,1) & B_\rho^{\mathcal{T}} & \beta_0 \end{bmatrix}^{\mathcal{T}} \tag{3.26}$$

for any given sets $\{\gamma_k\}$, $\{\Delta_k\}$ and $\{\bar{\eta}_k\}$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Therefore, the corresponding gradient vector $\psi_{\theta_I}(n) = \frac{\partial y(n)}{\partial \theta_I}$ can be computed with

$$\begin{cases} \psi_{A_\rho(:,1)}(n+1) &= A_\rho^{\mathcal{T}} \psi_{A_\rho(:,1)}(n) + C_\rho^{\mathcal{T}} x_1(n) \\ \psi_{B_\rho}(n+1) &= A_\rho^{\mathcal{T}} \psi_{B_\rho}(n) + C_\rho^{\mathcal{T}} u(n) \\ \psi_d(n) &= u(n), \end{cases} \tag{3.27}$$

where $x_1(n)$ is the first state variable in the R_ρ realization with which the filter $H(z^{-1})$ is implemented.

The proposed adaptive filter is realized in R_ρ parametrized using θ_I. The output error scheme is assumed and the filter output is computed by (2.6) with (A, B, C, d) replaced by R_ρ. The gradient signal is given by (3.27) and the numerical algorithm used is (2.31).

As mentioned before, one of the main reasons for the structures parametrized using θ_d and θ_c having a poor capability against the FWL effects is that these parametrizations have $\{a_k\}$ as parameters and the poles of the system are very sensitive to these z-operator based transfer function coefficients, especially when the poles are clustered together and close to the unit circle $|z| = 1$.

Consider (3.14). The denominator of the transfer function in z is

$$\bar{D} \triangleq \kappa q_0(z) + \sum_{k=1}^{K} \alpha_k \kappa q_k(z) = \prod_{m=1}^{K} (z - \lambda_m), \tag{3.28}$$

where $\kappa = \prod_{k=1}^{K} \Delta_k$. Once again, assume λ_k is not a repeated pole, according to the implicit function theorem one has the following pole sensitivity

$$\frac{\partial \lambda_k}{\partial \alpha_l} = -\frac{\partial \bar{D}/\partial \alpha_l}{\partial \bar{D}/\partial \lambda_k}\Big|_{z=\lambda_k} = -\frac{\kappa q_l(\lambda_k)}{\prod_{m \neq k}(\lambda_k - \lambda_m)}, \ \forall k, \ l. \tag{3.29}$$

It then follows from (3.21) that the pole sensitivities for the parametrization θ_I are

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

then

$$\frac{\partial \lambda_k}{\partial a_{l1}} = \frac{\partial \lambda_k}{\partial \alpha_l} \frac{\partial \alpha_l}{\partial a_{l1}} = \frac{\Delta_1^{-1} \kappa q_l(\lambda_k)}{\prod_{m \neq k}(\lambda_k - \lambda_m)}, \ \forall k, \ l. \tag{3.30}$$

One observes that when $q_l(z)$ has roots equal or close to the pole λ_k, $\frac{\partial \lambda_k}{\partial \alpha_l}$ and hence $\frac{\partial \lambda_k}{\partial \theta_I(l)}$, $l = 1, 2, \cdots, K$, can be much reduced, which coincides with the arguments on how to choose these polynomial operators for reducing roundoff noise in Reference [9] of Author's Publications. This observation gives an guideline for us to choose these polynomial operators $q_k(z)$.

Based on the initially estimated system, say $\hat{H}(z^{-1})$, one can determine a realization R_ρ using the procedure described in Reference [9] of Author's Publications. Since this structure has better numerical properties than its DFII counterpart within the neighborhood of this system, it is expected that the corresponding adaptive algorithm has better FWL performance, which has been supported by simulations to be presented in Section 3.4.

3.4 Numerical examples and simulations

In this section, we will present two numerical examples to demonstrate the FWL performance of the adaptive filters using the structures DFII, R_{ct} and the proposed R_ρ, parametrized using θ_d, θ_c and θ_I, respectively. The three adaptive algorithms/filters are denoted as Alg_{DF}, $\text{Alg}_{R_{ct}}$, and Alg_I for convenience.

Example 3.1: The true system is generated with $[V_b, V_a] = butter(3, 0.1)$, whose magnitude response is given in Figure 3.3 with solid line. The corresponding poles are $\lambda_{1,2} = 0.8238 \pm j0.2318$, $\lambda_3 = 0.7265$. With an input signal $u(n)$, $n = 0, 1, 2, \cdots, 4999$, which is a white noise signal with unit variance, the output of the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

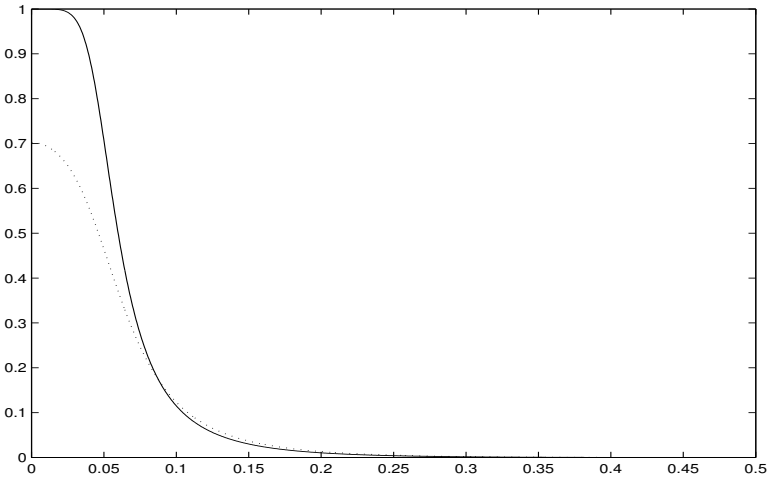


Figure 3.3: Magnitude responses for Example 3.1, where the solid line is the true system, while the dotted line, for the initial estimate, with the x-axis denoting the normalized digital frequency.

true system is generated.

Based on the following initial estimate of the true system, whose magnitude response is given in Figure 3.3 with dotted line,

$$\hat{H}(z^{-1}) = 10^{-2} \times \frac{0.2898 + 0.9564z^{-1} + 1.0520z^{-2} + 0.3857z^{-3}}{1 - 2.1367z^{-1} + 1.5628z^{-2} - 0.3879z^{-3}},$$

a R_ρ realization is obtained, which is presented below

$$A_\rho(0) = \begin{bmatrix} 0.6367 & 0.4052 & 0 \\ -0.1117 & 0.7500 & 0.3019 \\ -0.0343 & 0 & 0.7500 \end{bmatrix}, \; B_\rho(0) = \begin{bmatrix} 0.0716 \\ 0.3323 \\ 0.6813 \end{bmatrix}, \; C_\rho^{\mathcal{T}}(0) = \begin{bmatrix} 0.2201 \\ 0 \\ 0 \end{bmatrix}$$

with $\beta_0(0) = 0.2898 \times 10^{-2}$. This will be the initial realization of the proposed algorithm.

We have computed the poles sensitivities for θ_d and θ_I, both evaluated at the point corresponding to the true system. Denote $M_{ps}^{\theta_d}$ with $M_{ps}^{\theta_d}(i,j) = |\frac{\partial \lambda_i}{\partial a_j}|$ as the pole

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

sensitivity matrix corresponding to θ_d. Computations show

$$M_{ps}^{\theta_d} = \begin{bmatrix} 6.2843 & 7.3434 & 8.5811 \\ 6.2843 & 7.3434 & 8.5811 \\ 8.3541 & 11.4984 & 15.8262 \end{bmatrix}.$$

With $M_{ps}^{\theta_I}$ defined in the same way, one has

$$M_{ps}^{\theta_I} = \begin{bmatrix} 0.5078 & 0.8458 & 1.0497 \\ 0.5078 & 0.8458 & 1.0497 \\ 0.0087 & 0.1504 & 1.9359 \end{bmatrix}.$$

As observed, the pole sensitivities can be much reduced with the parametrization θ_I. It is expected that the proposed adaptive state-space filter should have a better FWL performance than the classical DFII structure-based one. This has been supported by the following simulations, where the structure parameters for each algorithm are implemented with B_c bits.

Starting from this initial system, we run Alg_{DF} and $\text{Alg}_{R_{ct}}$ with the same settings and $P_\theta(0) = 0.01 I_{2K+1}$. In Figure 3.4, both algorithms converge to the true system under infinite precision environment. In Figure 3.5, when the algorithms are implemented with $B_c = 16$ bits, both of them are out of work. This simulation supports our analysis in the previous sections, where we mentioned that both $\text{Alg}_{R_{ct}}$ and Alg_{DF} have a poor capability against the FWL effects.

We then run Alg_{DF} and Alg_I with the same settings, particularly, $P_\theta(0) = 0.5 I_{2K+1}$. As shown in Figure 3.6, both algorithms converge to true system when the parameters are implemented with infinite precision.

When implemented with $B_c = 16$ bits, the proposed algorithm still performs well, while the DFII-based algorithm is out of work. See Figure 3.7. To make the DFII-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

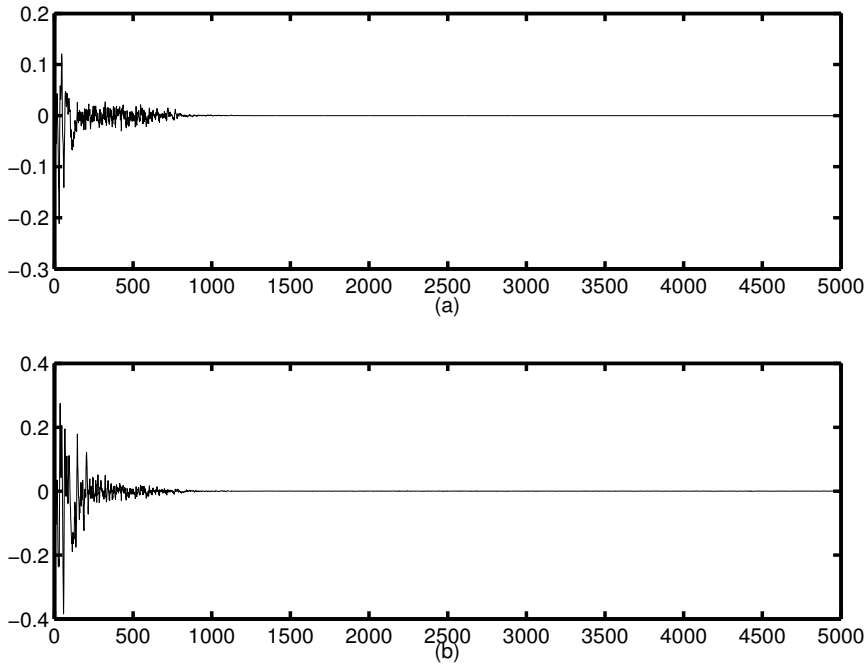


Figure 3.4: Prediction errors for Example 3.1: (a) – the Alg_{DF} implemented with $B_c = +\infty$ bits, (b) – the $\text{Alg}_{R_{ct}}$ implemented with $B_c = +\infty$ bits, where the x-axis denotes the iteration number.

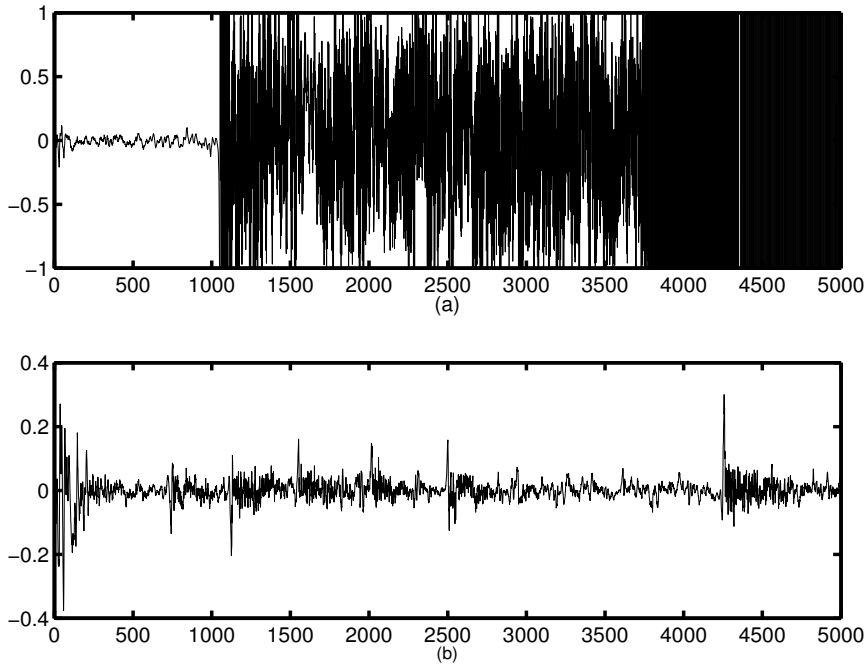


Figure 3.5: Prediction errors for Example 3.1: (a) – the Alg_{DF} implemented with $B_c = 16$ bits, (b) – the $\text{Alg}_{R_{ct}}$ implemented with $B_c = 16$ bits, where the x-axis denotes the iteration number.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

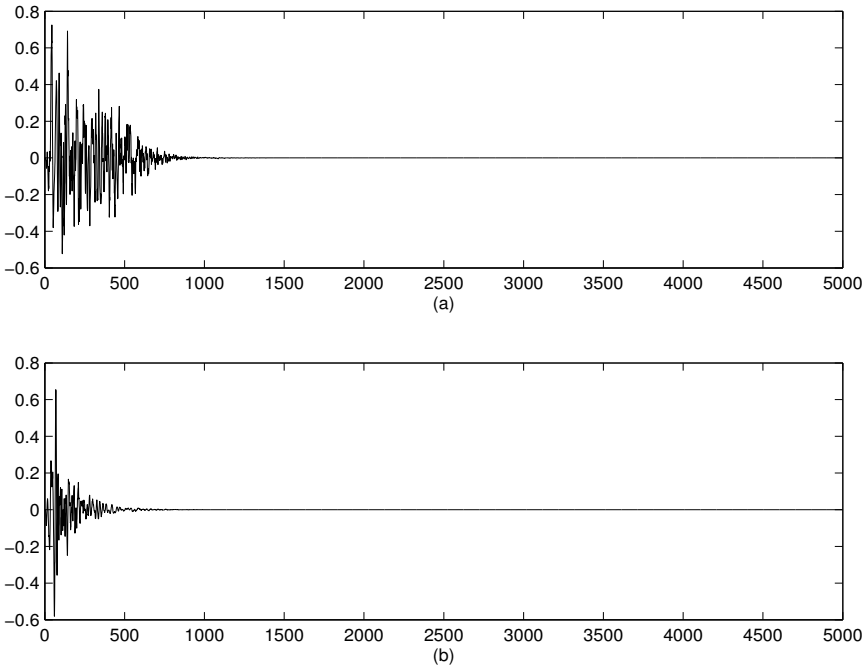


Figure 3.6: Prediction errors for Example 3.1: (a) – the Alg_{DF} implemented with $B_c = +\infty$ bits, (b) – the proposed Alg_I implemented with $B_c = +\infty$ bits, where the x-axis denotes the iteration number.

based Alg_{DF} back to normal, B_c is at least 20 bits.

Example 3.2: In this example, the true system is a sixth order bandpass filter, generated with $[V_b, V_a] = butter(3, [0.15 \quad 0.25])$, whose magnitude response is given in Figure 3.8 with solid line. The corresponding poles are $\lambda_{1,2} = 0.6622 \pm j0.6260$, $\lambda_{3,4} = 0.8316 \pm j0.4362$ and $\lambda_{5,6} = 0.7071 \pm j0.4760$.

Similarly, with an input signal $u(n)$, $n = 0, 1, 2, \cdots, 4999$, which is a white noise signal with unit variance, the output of the true system is generated.

With an initial estimate $\theta_d(0)$, which yields a magnitude response depicted in Fig-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

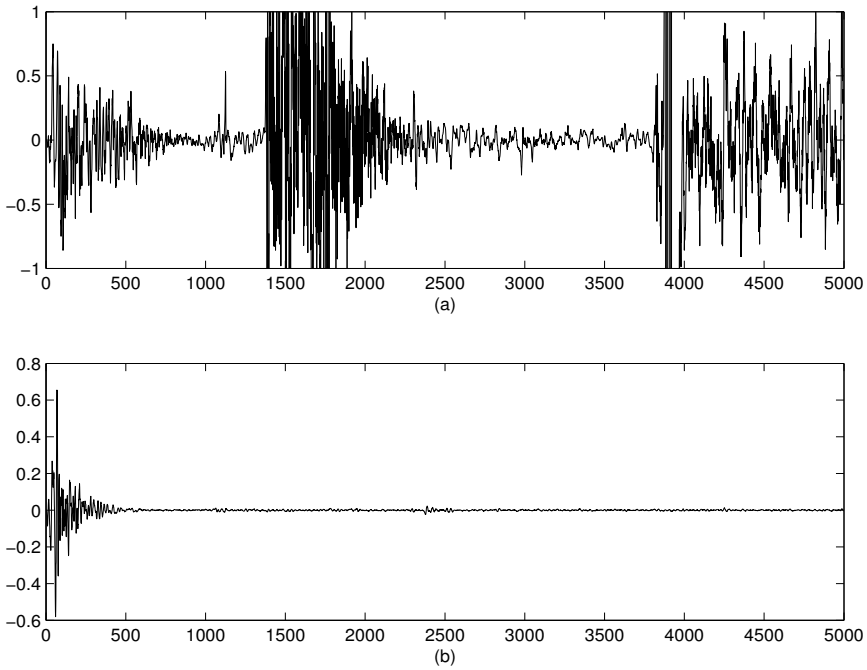


Figure 3.7: Prediction errors for Example 3.1: (a) – the Alg_{DF} implemented with $B_c = 16$ bits, (b) – the proposed Alg_I implemented with $B_c = 16$ bits, where the x-axis denotes the iteration number.

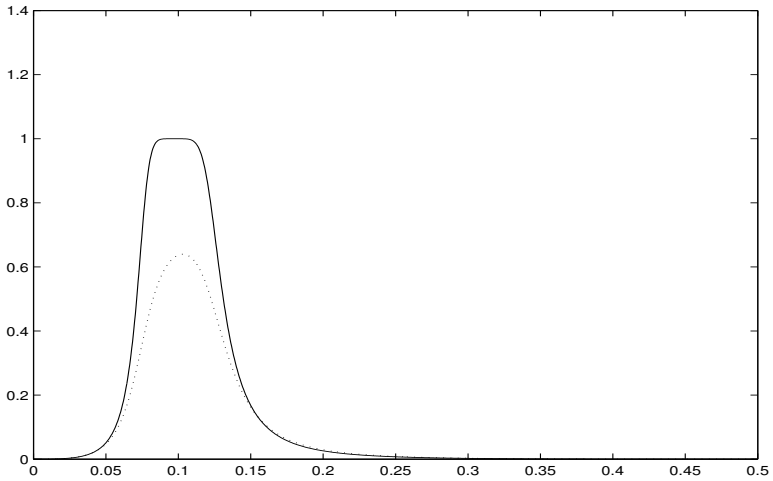


Figure 3.8: Magnitude responses for Example 3.2, where the solid line is the true system, while the dotted line, for the initial estimate, with the x-axis denoting the normalized digital frequency.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ure 3.8 with dotted line, a R_ρ is obtained

$$A_\rho(0) = \begin{bmatrix} 0.6818 & 0.6252 & 0 & 0 & 0 & 0 \\ -0.5855 & 0.7500 & 0.2611 & 0 & 0 & 0 \\ 0.1235 & 0 & 0.5000 & 0.6589 & 0 & 0 \\ -0.0472 & 0 & -0.3794 & 0.7500 & 0.2772 & 0 \\ -0.0519 & 0 & 0 & 0 & 0.7500 & 0.2488 \\ -0.1841 & 0 & 0 & 0 & -0.5023 & 0.7500 \end{bmatrix}$$

$$B_\rho(0) = \begin{bmatrix} 0.0609 & 0.0832 & -0.2973 & -0.5560 & 0.5391 & 0.5740 \end{bmatrix}^\mathcal{T}$$

$$C_\rho(0) = \begin{bmatrix} 0.1990 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

with $\beta_0(0) = 0.2898^{-2}$.

We have computed the poles sensitivities for Alg_{DF} and Alg_I, both evaluated at the point corresponding to true system:

$$M_{ps}^{\theta_d} = \begin{bmatrix} 10.6149 & 11.6483 & 12.7823 & 14.0266 & 15.3921 & 16.8906 \\ 10.6149 & 11.6483 & 12.7823 & 14.0266 & 15.3921 & 16.8906 \\ 25.4126 & 27.0612 & 28.8168 & 30.6863 & 32.6770 & 34.7969 \\ 25.4126 & 27.0612 & 28.8168 & 30.6863 & 32.6770 & 34.7969 \\ 22.7318 & 26.6688 & 31.2876 & 36.7064 & 43.0637 & 50.5220 \\ 22.7318 & 726.6688 & 31.2876 & 36.7064 & 43.0637 & 50.5220 \end{bmatrix}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

and

$$M_{ps}^{\theta_I} = \begin{bmatrix} 0.4899 & 0.4845 & 0.4908 & 0.5115 & 0.3184 & 0.1253 \\ 0.4899 & 0.4845 & 0.4908 & 0.5115 & 0.3184 & 0.1253 \\ 0.2849 & 0.4014 & 0.2324 & 0.3452 & 0.4604 & 0.2582 \\ 0.2849 & 0.4014 & 0.2324 & 0.3452 & 0.4604 & 0.2582 \\ 0.2068 & 0.2706 & 0.4247 & 0.5855 & 0.7199 & 0.3748 \\ 0.2068 & 0.2706 & 0.4247 & 0.5855 & 0.7199 & 0.3748 \end{bmatrix}.$$

Compared with the system in Example 3.1, this system has higher pole sensitivities due to the fact that there are more poles clustered together. As expected, a higher precision implementation is needed in order to keep the adaptive algorithms working properly. As observed again, the pole sensitivities can be much reduced with the parametrization θ_I. It is expected that the proposed Alg_I should have a better FWL performance than the classical DFII structure-based Alg_{DF}.

As shown in Figure 3.9, both algorithms converge to true system when the parameters are implemented with infinite precision.

When implemented with $B_c = 21$ bits, the proposed algorithm still performs well, while for the DFII-based one, an implementation of at least $B_c = 26$ bits is needed.

Now, let's examine the dynamic range constraint issue with Example 3.1. For this example, the measurements, i.e., the input and output signals of the true system are bounded by 3.6 and 1.2, respectively. All the state variables in the DFII-based Alg_{DF} and the proposed Alg_I are bounded by 67 and 6, respectively.

Now, we implement both algorithms with $DR = 5$ and $B_{dr} = 16$ bits. The corresponding prediction errors are presented in Figure 3.10, from which one can see that Alg_{DF} can not converge at all due to the dynamic constraint, while this constraint

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

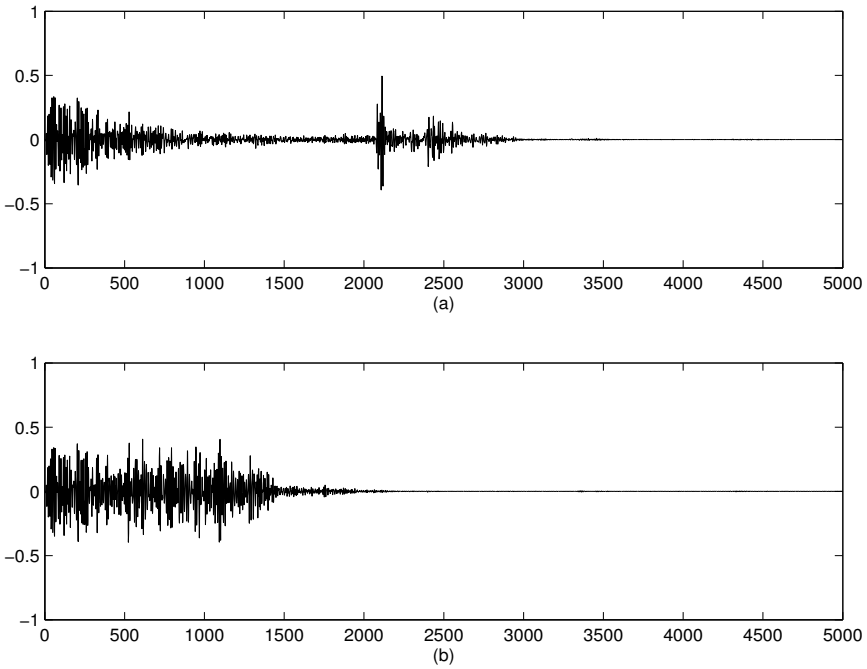


Figure 3.9: Prediction errors for Example 3.2: (a) – the Alg_{DF} implemented with $B_c = +\infty$ bits, (b) – the proposed Alg_I implemented with $B_c = +\infty$ bits, where the x-axis denotes the iteration number.

has no effect on the proposed Alg_I at all.

Figure 3.11 shows the prediction errors for $DR = 20$. As seen, an overflow oscillation occurs at (around) the 4350th iteration for the DFII-based algorithm.

It should be pointed out that in a real-time implementation problem, the dynamic range constraint DR and B_{dr} are usually determined by the digital device. The measurements are usually pre-scaled such that the resulting signals are within this specified range. The dynamic range of the state variable, however, is structure and system dependent. For the DFII structure, this range varies with the systems. The structures satisfying l_2-scaling (2.27) make the states have the same dynamic range in a statistical sense.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

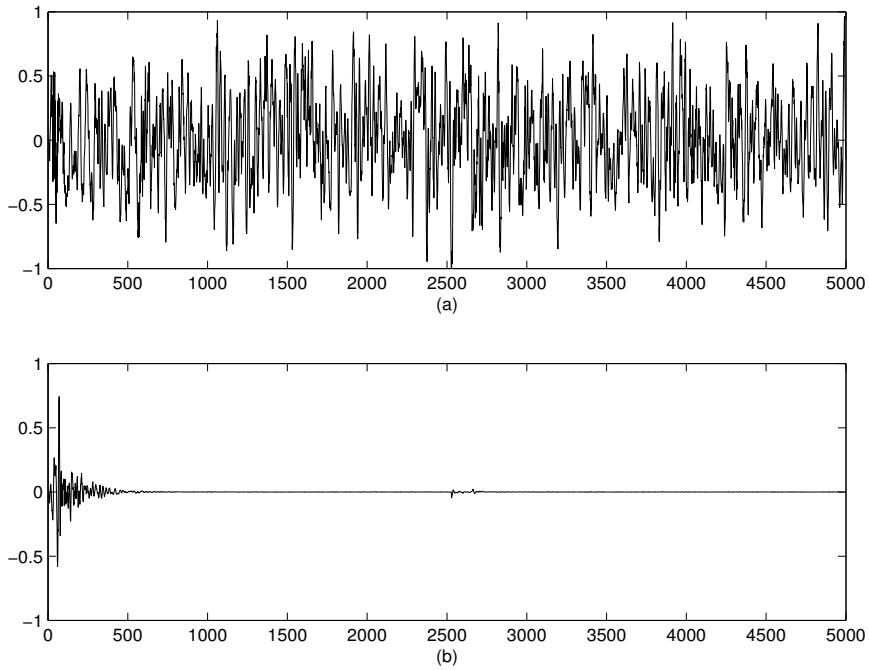


Figure 3.10: Prediction errors for Example 3.1: (a) – the Alg_{DF}; (b) – the proposed Alg_I, with both implemented using $DR = 5$ and $B_{dr} = 16$ bits, where the x-axis denotes the iteration number.

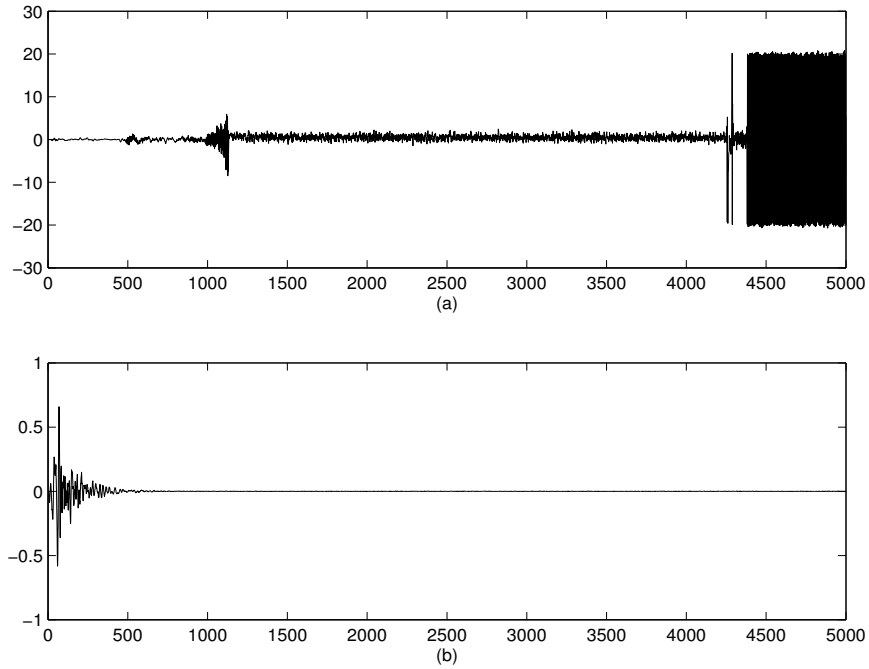


Figure 3.11: Prediction errors for Example 3.1: (a) – the Alg_{DF}; (b) – the proposed Alg_I, with both implemented using $DR = 20$ and $B_{dr} = 16$ bits, where the x-axis denotes the iteration number.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

3.5 Conclusions

The FWL effects have been well studied in time-invariant filter implementation, while they have received a little attention in adaptive system design in the literature. In this chapter, we have discussed some of the FWL issues in adaptive filter design and implementation in terms of pole sensitivity, stability, and dynamic range constraint. It is argued that to improve the stability behavior and FWL performance of the traditional DFII-based adaptive filters, alternative structures with more robust parametrizations should be used. A new adaptive state-space filter has been proposed. It has been shown that the pole sensitivity and hence the stability behavior of the adaptive filter can be much improved with this structure. Simulations have been done to demonstrate the FWL behavior of the classical DFII-based adaptive filters and the proposed one.

It should be pointed out that the nice performance of the proposed adaptive state-space filter depends on a relatively good estimate of the true system and that the BIBO stabilty and l_2-scaling can not be guarantted during the adaptation. In the following chapters, alternative structures will be proposed to overcome these problems.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 4

Normal State-Space Realizations for Adaptive IIR Filters

Stability and convergence, considered as the two most important issues in adaptive IIR filter design, are strongly related to the filter structure. The convergence behavior of several adaptive IIR filter structures was analyzed by Fan [9], [97]. Due to the time-varying nature, the relation of the stability of an adaptive IIR filter with its structures is a very hard problem to attack and has been studied by a few researchers. The stability behavior of an adaptive filter is strongly related to the two structures $\mathcal{S}$ and $\mathcal{G}$. As mentioned in Chapter 3, these two filters are traditionally implemented in direct-form-based structures. These structures, though very simple, usually have very poor numerical and stability properties ([75], [93], [94]) and require stability monitoring which is based on checking the magnitude of instantaneous eigenvalues of the adaptive filters.

It has been pointed out in Section 2.3.2, however, that the BIBO stability may not hold even when an adaptive filter has all its instantaneous eigenvalues absolutely less than one. To avoid the risk of the filter instability during the convergence pro-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

cess, alternative structures of adaptive IIR filters are also used, such as lattice-based structures, cascade structure [10] and parallel structure [8], [45]. So far as we know, the lattice-based structures are perhaps the most advocated ones for adaptive IIR filters regardless their computational complexity. Dasgupta *et al.* [95] studied the time-varying unnormalized lattice filters and provided conditions for which stability can be ensured and, more interestingly, Regalia [6] proved that under a set of inequalities the adaptive IIR filters using the normalized lattice (NL) structure are absolutely stable. In fact, the lattice-based structures are orthogonal realizations of adaptive filters. In [8], alternative orthogonal realizations for adaptive IIR filters were analyzed in terms of convergence and computational efficiency.

The main objective of this chapter is to present a number of new results, which are based on the normal realizations. The basic feature of these results is that the proposed adaptive IIR filters are BIBO stable. An outline of this chapter is given as follows. In Section 4.1, it is shown that an adaptive filter is BIBO stable under some simple conditions if it can be implemented in a normal state-space realization. The application of this important result to adaptive filter design relies on how to parametrize the normal realizations. A normal state-space realization is parametrized for a class of 2nd-order IIR filters. In Section 4.2, a new adaptive notch filter is derived based on this normal state-space realization. In this proposed adaptive algorithm, the Steiglitz-McBride adaptive scheme is used. The stability of this algorithm is proved in Section 4.3. The convergence is also analyzed in this section, where it is shown that the algorithm converges to the desired notch filter whose notch frequencies are those underlying the input signal. In Section 4.4, the computational efficiency and implementation issues are discussed. It is shown that the proposed algorithm is BIBO stable even when implemented with finite precision. Compared with the normalized lattice structure, our proposed normal realization is simpler and more interestingly, has minimal pole/zero sensitivities,

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

which is very important for FWL implementation of notch filters whose poles and zeros are close to and on the unit circle $|z| = 1$ and hence very sensitive to the FWL errors. An application of pitch detection in speech signal using our proposed algorithm is discussed in Section 4.5. Simulation results are provided to illustrate the performance of the proposed algorithm in Section 4.6, in which it is seen that the proposed algorithm outperforms the one in [19] in terms of stability, convergence and estimation accuracy.

4.1 Normal state-space realizations

It has been pointed out in Chapter 2 that the first criterion in Theorem 2 is intuitive and it can be easily ensured in a real-time implementation of an adaptive system. The second one depends strongly on the filter structures used. As far as we know, there are a few results on how to find the structures which can guarantee the exponential stability [6], [95].

Our first new result in this chapter is presented in the following theorem, which is based on normal matrices.[1]

Theorem 3 *The homogeneous system $x(n+1) = A(n)x(n)$ is exponentially stable if $A(n)$ is normal and the instantaneous eigenvalues of $A(n)$ are all inside the circle $|z| = \gamma$ for any n, where $\gamma < 1$ is a constant.*

Proof: Denote $\lambda_l(M)$ and $|||M|||_2$ as the lth eigenvalue and the *spectral* norm (or 2-norm) of a real matrix M, respectively. It is well known [26] that $|||M|||_2 =$

[1] A matrix M is said normal if $MM^T = M^T M$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$\max_{\forall l} \sqrt{\lambda_l(MM^T)}$. It then follows from

$$x(n+m) = A(n+m-1)\cdots A(n+1)A(n)x(n) = \prod_{k=n}^{n+m-1} A(k)x(n)$$

for any integers m, n that

$$||x(n+m)||_2 \leq ||| \prod_{k=n}^{n+m-1} A(k)|||_2 ||x(n)||_2 \leq \prod_{k=n}^{n+m-1} |||A(k)|||_2 ||x(n)||_2.$$

For a normal matrix $A(k)$, $|||A(k)|||_2 = \max_{\forall l} |\lambda_l(A(k))|$ [104]. Since $|\lambda_l(A(k))| \leq \gamma$ for any possible l and k, one has $||x(n+m)||_2 \leq \gamma^m ||x(n)||_2$, equivalently,

$$||x(m)||_2 \leq \gamma^{m-n} ||x(n)||_2, \ \forall m \geq n,$$

which, according to the definition, implies that the system is exponentially stable. ∎

Theorem 3 means that an adaptive IIR filter implemented in a *normal* realization[2] or a structure, whose equivalent realization is normal, can ensure the exponential stability if their instantaneous poles can be confined within a specified region inside the unit circle.

Generally speaking, it is very difficult to parametrize a normal realization if the filters are of high order. Consider the following 2nd-order filter

$$F(z^{-1}) = q_0 + \frac{q_1 z^{-1} + q_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}, \tag{4.1}$$

whose input and output are $\nu(n)$ and $w(n)$, respectively. It is well known that this

[2] A realization is called *normal* if its transition matrix is normal.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

filter can be implemented in its state-space equations:

$$\begin{cases} x(n+1) &= Ax(n) + B\nu(n) \\ w(n) &= Cx(n) + q_0\nu(n), \end{cases} \tag{4.2}$$

where $(A,\ B,\ C,\ q_0)$ is called a realization of $F(z^{-1})$ with $A \in \mathcal{R}^{2\times 2}, B \in \mathcal{R}^{2\times 1}, C \in \mathcal{R}^{1\times 2}$, satisfying

$$F(z^{-1}) = q_0 + C(zI - A)^{-1}B. \tag{4.3}$$

All the realizations satisfying (4.3) form a set, which can be characterized with the following:

$$A = T^{-1}A^cT, \quad B = T^{-1}B^c, \quad C = C^cT, \tag{4.4}$$

where $T \in \mathcal{R}^{2\times 2}$ is any nonsingular matrix, performing similarity transformation, and

$$\begin{cases} A^c &\triangleq \begin{bmatrix} -a_1 & -a_2 \\ 1 & 0 \end{bmatrix}, \ B^c \triangleq \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ C^c &\triangleq \begin{bmatrix} q_1 & q_2 \end{bmatrix}. \end{cases} \tag{4.5}$$

$R^c \triangleq (A^c,\ B^c,\ C^c,\ q_0)$ is called controllable state-space realization of $F(z^{-1})$. When $F(z^{-1})$ has complex poles $\rho e^{\pm j\omega}$ with $\rho \neq 0,\ 0 < \omega < \pi,\ a_1 = -2\rho\cos\omega,\ a_2 = \rho^2$. Applying the following similarity transformation to the controllable realization R^c:

$$T = \begin{bmatrix} -\frac{\cos\omega}{\sin\omega} & 1 \\ -\frac{1}{\rho\sin\omega} & 0 \end{bmatrix},$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

one has the following normal realization of $F(z^{-1})$:

$$\begin{cases} A &= \rho \begin{bmatrix} \cos\omega & -\sin\omega \\ \sin\omega & \cos\omega \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ C &= \begin{bmatrix} q_1 & q_2 \end{bmatrix} T. \end{cases} \tag{4.6}$$

Therefore, such a 2nd-order transfer function $F(z^{-1})$ can be implemented in the normal realization (4.6) parametrized using

$$\theta_{nr} \triangleq \begin{bmatrix} \rho & \omega & C & q_0 \end{bmatrix}^{\mathcal{T}}. \tag{4.7}$$

A 2nd-order adaptive filter implemented in such a realization is exponentially stable if the parameter ρ is forced to $0 < \rho \leq \gamma < 1$ with γ a pre-determined constant. It should be pointed out that this normal realization is valid for only the class of 2nd-order systems whose poles are complex.

For higher order filters, one can use the cascade form, where each 2nd-order stage, say $F_k(z^{-1})$, is modeled with (4.6). The corresponding adaptive algorithm can then be derived easily and the corresponding stability can be ensured smoothly. This model, though limited to transfer functions of complex poles, presents a wide range of important linear systems used in many applications, for example, constrained IIR notch filters to be discussed in the next section. On the bottom line, any system having real poles can be approximated with a proper system whose poles are all complex.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

4.2 A new adaptive constrained notch filter

Notch filters find a wide range of applications in the area of signal processing. The main function of such filters is to remove/block the sinusoidal components of a signal $u(n)$:

$$u(n) = \sum_{k=1}^{N} M_k cos(\omega_k^0 n - \phi_k) + v(n), \tag{4.8}$$

where $v(n)$ is a broadband signal or a white noise, statistically independent of the components, and M_k, ω_k^0, ϕ_k are the magnitude, (angular) frequency and phase of the kth component, respectively, with $0 < \omega_k^0 < \pi$, $\forall k$ and $\omega_k^0 \neq \omega_j^0$ if $k \neq j$, $\forall j$, k. Usually, ϕ_k, $\forall k$ are assumed as statistically independent variables distributed uniformly within $[-\pi, \pi]$.

A popularly used notch filter is given by the following transfer function

$$H(z^{-1}) = \frac{D(z^{-1})}{D(\rho z^{-1})}, \tag{4.9}$$

where ρ is a constant, satisfying $0 < \rho < 1$, and $D(z^{-1}) = 1 + z^{-2N} + \sum_{k=1}^{N-1} a_k(z^{-k} + z^{-2N+k}) + a_N z^{-N}$. This filter is constrained to have a zero transmission gain at $z = e^{\pm j\omega_k}$, $k = 1, 2, \ldots, N$, where $\{\omega_k\}$ are the notch frequencies. When the filter parameters are properly set such that $\omega_k = \omega_k^0$, $\forall k$, the output $e(n)$ of the filter $H(z^{-1})$ excited with $u(n)$ is almost equal to $v(n)$. When $\{\omega_k^0\}$ are not available, one has to adjust the filter parameters such that the notch frequencies of the filter are equal to the frequencies underlying the signal, which is actually equivalent to an adaptive frequency estimation problem [15] - [19], [65].

Denote

$$\theta_a \triangleq \begin{bmatrix} a_1 & \ldots & a_k & \ldots & a_N \end{bmatrix}^{\mathcal{T}}$$

with $\{a_k\}$ the N independent coefficients of $D(z^{-1})$, as defined below (4.9). Nehorai

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[17] derived a minimal parameter adaptive notch filter parametrized using θ_a, where the notch frequencies can be computed by finding the roots of $D(z^{-1})$. Though the stability can not be strictly guaranteed, such an adaptive filter, as pointed out in [17], [22], does show a much better stability behavior than the unconstrained notch filters when the poles of the filter are set close to the origin at the start of the data processing and a time-varying $\rho(n)$ is used, which increases from a value smaller than one (e.g., $\rho(0) = 0.80$) to that close to one. One notes, however, that for a real-time implementation with a digital device of finite word length (FWL) (say, 16 bits), instability may occur due to the FWL errors on the parameters $\{a_k\}$. Therefore, a stability monitoring scheme is required in this case. In [18], a *direct* frequency estimation algorithm was proposed using an adaptive notch filter which is *partially* parametrized using the notch frequencies. This algorithm can estimate the frequencies directly. It, however, suffers from converting frequencies to transfer function coefficients $\{a_k\}$ and, *more seriously*, most of its properties such as ideal notch performance (due to the zeros are automatically ensured on $|z| = 1$) and fast convergence speed may be lost when this algorithm is implemented with an FWL device, which is always the case for real-time applications. Based on alternative models to (4.9), the lattice structures were used in design of adaptive notch filters [48], [105]. Of particular interest is the normalized lattice realization proposed in [105], which is notch frequency parametrized and, according to [6], BIBO stable.

Motivated by [18], a fully frequency parametrized adaptive algorithm was derived in [19]. Though this adaptive algorithm, compared with the ones in [17], [18], is of a better stability behavior and more robust, the stability still can not be ensured during the adaptation. The main objective of the following sections is to derive a new adaptive notch filter, which can be considered as an improved version of the one in [19] in terms of having guaranteed stability and a better convergence behavior.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

The key feature of this proposed adaptive notch filter is that the whole adaptive system (2.30) is implemented based on a *normal* state-space realization.

4.2.1 A normal realization-based constrained notch filter

The notch filter (4.9) can be rewritten as [18], [19]

$$H(z^{-1}) = \prod_{k=1}^{N} \frac{1-2\cos\omega_k z^{-1}+z^{-2}}{1-2\rho\cos\omega_k z^{-1}+\rho^2 z^{-2}} \triangleq \prod_{k=1}^{N} \frac{D_k(z^{-1})}{D_k(\rho z^{-1})} \triangleq \frac{D(z^{-1})}{D(\rho z^{-1})}, \quad (4.10)$$

which is parametrized directly using the notch frequency vector

$$\theta_\omega \triangleq \begin{bmatrix} \omega_1 & \dots & \omega_k & \dots & \omega_N \end{bmatrix}^{\mathcal{T}}. \quad (4.11)$$

It is easy to see that $H(z^{-1})$ has $2N$ zeros at $z = e^{\pm j\omega_k}$, $\forall k$, which means a zero transmission gain at the notch frequencies $\{\omega_k\}$, and $2N$ poles, on the circle of radius ρ. The notch filtering performance depends on the choice of the constant ρ, which is usually very close to one. In fact, the bandwidth of

$$H_k(z^{-1}) \triangleq \frac{1-2\cos\omega_k z^{-1}+z^{-2}}{1-2\rho\cos\omega_k z^{-1}+\rho^2 z^{-2}}$$

is approximately given by $BW \approx \pi(1-\rho)$. See [17]. In our simulations to be presented later, $\rho = 1-2^{-8} \approx 0.9961$ is used. As pointed out in Section 4.1, $H_k(z^{-1})$ can be realized by using the normal realization given by (4.6) with ω replaced by ω_k.

Noting the fact that $0 < \omega_k^0 < \pi$, $\forall k$, the following constraints are assumed in the sequel:

$$\epsilon \le \omega_k \le \pi - \epsilon, \ \forall k, \quad (4.12)$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where ϵ is a small positive constant. With such constraints, the transformation T is well defined. Though all the realizations are totally equivalent in infinite precision, they do have different numerical properties and stability behavior [75], [93]. The newly proposed adaptive notch filter is based on this normal realization.

Denote $e_k(n)$ as the output of $H_k(z^{-1})$ defined before, for $k = 1, 2, \ldots, N$, and $e_0(n) = u(n)$ and $e(n) = e_N(n)$. Applying (4.6) to $H_k(z^{-1}) = 1 + \frac{-2(1-\rho)\cos\omega_k z^{-1} + (1-\rho^2)z^{-2}}{1 - 2\rho\cos\omega_k z^{-1} + \rho^2 z^{-2}}$, one can obtain the corresponding normal realization $(A_k, B_k, C_k^*, 1)$, where A_k is equal to A but with ω replaced by ω_k, $B_k = B$ and

$$\begin{aligned} C_k^* &= \begin{bmatrix} -2(1-\rho)\cos\omega_k & 1-\rho^2 \end{bmatrix} \begin{bmatrix} -\frac{\cos\omega_k}{\sin\omega_k} & 1 \\ -\frac{1}{\rho\sin\omega_k} & 0 \end{bmatrix} \\ &= \begin{bmatrix} \frac{(1-\rho)[2\rho\cos^2\omega_k - (1+\rho)]}{\rho\sin\omega_k} & -2(1-\rho)\cos\omega_k \end{bmatrix}. \end{aligned}$$

Since, as assumed, $\rho \approx 1$, it is easy to understand that one can compute $e(n)$ using the following normal realizations $R_k^{nor} \triangleq (A_k, B_k, C_k, 1)$:

$$\begin{cases} x_k(n+1) &= \rho \begin{bmatrix} \cos\omega_k & -\sin\omega_k \\ \sin\omega_k & \cos\omega_k \end{bmatrix} x_k(n) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} e_{k-1}(n) \\ e_k(n) &= 2(\rho-1) \begin{bmatrix} \sin\omega_k & \cos\omega_k \end{bmatrix} x_k(n) + e_{k-1}(n) \end{cases} \tag{4.13}$$

Our proposed adaptive notch filter is of the form (2.30), where the output $e(n) = e_N(n)$ of the filter is computed in the way described by (4.13), where θ_ω is replaced by $\theta_\omega(n)$. For convenience, we use $e(n) = \mathcal{S}_{nor}[H, \theta_\omega(n), u(n)]$ to indicate the corresponding adaptive filter structure.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

4.2.2 Steiglitz-McBride adaptive scheme and computing the gradient signals

The Steiglitz-McBride adaptive scheme [6], [56] has some interesting convergence properties. As shown in [53], under some mild conditions any stochastic gradient-based algorithm has a single possible stationary point corresponding to the true system if the order of the model parametrized using the transfer function coefficients is at least as large as that of the true system. Though an error bound for the Steiglitz-McBride algorithm was obtained for the undermodelled situations [106], the uncertain convergence behavior of the algorithm is still a concern in these situations. The order of the true notch filter to be identified here, however, is related to the number of the sinusoids in the input signal $u(n)$ and this number or its upper bound is usually known in many applications such as communication systems and speech/audio signal processing. In this chapter, we assume that the number of the sinusoids N is known and hence the order of the model for the notch filter is $2N$.

The complexity of an adaptive filter described by (2.30) depends strongly on how the gradient vector $\psi_\theta(n)$ is computed. Based on the controllable realization of $H_k(z^{-1})$, an adaptive notch filter was derived in [19] using the output error scheme. Noting the fact that implementing the corresponding gradient vector $\psi_{\theta_\omega}(n)$ requires a lot of computation, the notch filter was approximated with $\prod_{k=1}^{N} D_k(\beta z^{-1})/D_k(\rho z^{-1})$, where $D_k(z^{-1})$ is defined in (4.10) and β is a constant, satisfying $0 < \rho < \beta < 1$. By doing so, the computational burden can be greatly reduced but at the price of sacrificing model accuracy. In what follows, we will develop an adaptive notch filter based on the Steiglitz-McBride adaptive scheme without modifying the model of the notch filter. As to be seen, the resulting adaptive filter is computationally as efficient as the one in [19].

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

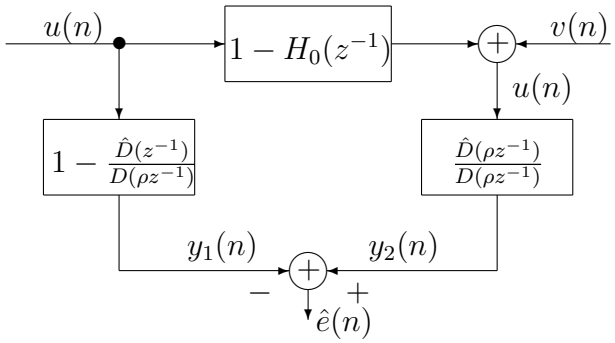


Figure 4.1: The Steiglitz-McBride identification scheme of the adaptive notch filter.

Denote $H_0(z^{-1})$ as the notch filter corresponding to $\omega_k = \omega_k^0$, $\forall k$, where $\{\omega_k^0\}$ are the frequencies of the sinusoids underlying the input signal $u(n)$. As mentioned before, the notch filter $H(z^{-1})$ yields a frequency response almost the same as that of the ideal notch filter when $\rho \approx 1$. It is easy to understand that $v(n) \approx H_0(z^{-1})u(n)$, that is $u(n) - v(n) \approx [1 - H_0(z^{-1})]u(n)$. Therefore, the true system to be identified in the Steiglitz-McBride adaptive scheme is $[1 - H_0(z^{-1})]$ and hence this scheme can be depicted in Figure 4.1, where the measurement is equal to the input signal $u(n)$ and the notation $\hat{F}(z^{-1})$ is used to indicate the fact that $F(z^{-1})$ is parametrized using $\hat{\theta}_\omega = \begin{bmatrix} \hat{\omega}_1 & \cdots & \hat{\omega}_k & \cdots & \hat{\omega}_N \end{bmatrix}^{\mathcal{T}}$ for $F(z^{-1}) = D(z^{-1}),\ D(\rho z^{-1})$, both defined in (4.10).

Note that $\hat{e}(n) = y_2(n) - y_1(n)$ is the output of adaptive filter $e(n)$ when both $\hat{\theta}_\omega$ and θ_ω are replaced with $\theta_\omega(n)$. In this scheme [6], [56], the corresponding gradient vector $\psi_{\theta_\omega}(n)$ is the output of the gradient filter $H_g(z^{-1}) = -\frac{d[H_2(z^{-1}) - H_1(z^{-1})]}{d\hat{\theta}_\omega}|_{\hat{\theta}_\omega = \theta_\omega}$, where $H_1(z^{-1}) \triangleq 1 - \frac{\hat{D}(z^{-1})}{D(\rho z^{-1})}$ and $H_2(z^{-1}) \triangleq \frac{\hat{D}(\rho z^{-1})}{D(\rho z^{-1})}$.

Noting $D(\rho z^{-1}) = \prod_{l=1}^{N} D_l(\rho z^{-1})$ with $D_l(\rho z^{-1})$ defined in (4.10), one has

$$\frac{dH_2(z^{-1})}{d\hat{\omega}_k}|_{\hat{\theta}_\omega = \theta_\omega} = \frac{d\hat{D}(\rho z^{-1})/d\hat{\omega}_k|_{\hat{\theta}_\omega = \theta_\omega}}{D(\rho z^{-1})} = \frac{2\rho \sin \omega_k z^{-1} \prod_{l \neq k}^{N} D_l(\rho z^{-1})}{D(\rho z^{-1})} = \frac{2\rho \sin \omega_k z^{-1}}{D_k(\rho z^{-1})}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

and

$$\frac{dH_1(z^{-1})}{d\hat{\omega}_k}|_{\hat{\theta}_\omega=\theta_\omega} = -\frac{d\hat{D}(z^{-1})/d\hat{\omega}_k|_{\hat{\theta}_\omega=\theta_\omega}}{D(\rho z^{-1})} = -\frac{2\sin\omega_k z^{-1}\prod_{l\neq k}^{N} D_l(z^{-1})}{D(\rho z^{-1})}.$$

Denote $g_k(n)$ as the kth element of $\psi_{\theta_\omega}(n)$. One then has

$$g_k(n) = -\frac{d[H_2(z^{-1}) - H_1(z^{-1})]}{d\hat{\omega}_k}|_{\hat{\theta}_\omega=\theta_\omega} u(n) \triangleq g_{2k}(n) + g_{1k}(n),$$

where

$$g_{1k}(n) = \frac{d[H_1(z^{-1})]}{d\hat{\omega}_k}|_{\hat{\theta}_\omega=\theta_\omega} u(n) = -\frac{2\sin\omega_k z^{-1}\prod_{l\neq k}^{N} D_l(z^{-1})}{D(\rho z^{-1})} u(n).$$

To reduce the implementation complexity of $g_{1k}(n)$, $\forall k$, let us consider the following revised $g_{1k}(n)$:

$$g^*_{1k}(n) \triangleq \frac{D_k(z^{-1})}{D_k(\rho z^{-1})} g_{1k}(n) = -\frac{2\sin\omega_k z^{-1}}{D_k(\rho z^{-1})} e(n)$$

with $e(n) = H(z^{-1})u(n)$ the output of the adaptive notch filter, computed using (4.13). The revised gradient signal is then

$$g^*_k(n) \triangleq g_{2k}(n) + g^*_{1k}(n) = -\frac{2\sin\omega_k z^{-1}}{D_k(\rho z^{-1})}[\rho u(n) + e(n)].$$

Applying (4.6) to the transfer function $-\frac{2\sin\omega_k z^{-1}}{D_k(\rho z^{-1})}$, one can then compute $g^*_k(n)$ with the following normal realization

$$\begin{cases} s_k(n+1) &= A_k s_k(n) + B_k[\rho u(n) + e(n)] \\ g^*_k(n) &= 2\begin{bmatrix} \cos\omega_k & -\sin\omega_k \end{bmatrix} s_k(n) \\ k &= 1,\ 2, \cdots,\ N, \end{cases} \tag{4.14}$$

where A_k, B_k are the same as used in (4.13).

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Denote $\mathcal{G}_{nor}$ as the structure of the transfer function $H_g^*(z^{-1})$ specified with (4.14), our proposed adaptive notch filter can be presented below

$$\begin{cases} e(n) &= \mathcal{S}_{nor}[H, \theta_\omega(n), u(n)] \\ \psi_{\theta_\omega}^*(n) &= \mathcal{G}_{nor}[H_g^*, \theta_\omega(n), \rho u(n) + e(n)] \\ \theta(n+1) &= \theta(n) + e(n) P_{\theta_\omega}(n) \psi_{\theta_\omega}^*(n) \end{cases} \tag{4.15}$$

In the following sections, we will analyze the performance of the proposed adaptive algorithm (4.15) in terms of stability, convergence and computational complexity, and compare it with the one in [19].

4.3 Performance analysis

In this section, performance analysis is carried out to the proposed alogirhm in terms of stability and convergence.

4.3.1 Stability analysis

The proposed adaptive algorithm (4.15) actually consists of three time-varying systems. It is said stable if the three systems are all BIBO stable. Since $P_{\theta_\omega}(n)$ is a positive-definite matrix, associated with the so-called information matrix, $\theta(n)$ is bounded if so are $e(n)$ and $\psi_{\theta_\omega}^*(n)$. In what follows, we will show that the two time-varying systems $\mathcal{S}_{nor}$ and $\mathcal{G}_{nor}$ in (4.15) are BIBO stable.

One of the main results in this chapter is given by Theorem 3, with which one can show that our proposed adaptive system (4.15) is BIBO stable. One notes that the elements/parameters in the two systems $\mathcal{S}_{nor}$ and $\mathcal{G}_{nor}$, as specified by (4.13) and (4.14), are all bounded for any $\theta_\omega(n)$. Let $u(n)$ be any bounded input

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

signal. Since $e_1(n)$ is computed with the normal realization $[A_1(n), B_1(n), C_1(n), 1]$, where $|\lambda_k[A_1(n)]| = \rho < 1$, $\forall k$ for any n, $e_1(n)$, according to Theorems 2 and 3, is bounded. With the same argument, one can show that $e(n) = e_N(n)$, the output of the adaptive filter is bounded. Therefore, the proposed adaptive notch filter is BIBO stable. In the same way, we can show the revised gradient filter $H_g^*(z^{-1})$ yielding $\psi_{\theta_\omega}^*(n)$ is BIBO stable. As a conclusion, our proposed adaptive algorithm (4.15) is strictly stable.

Adaptive IIR filters are traditionally implemented in the direct-form-based structures. The transition matrix $A(n)$ of the equivalent state-space realization to such a time-varying structure is of companion form (see, e.g., A^c in (4.5)). Such a form has very poor stability robustness, especially when the dimension becomes large [94], and usually does not satisfy the condition of exponential stability. That is why the traditional adaptive notch filters [15] - [18] demonstrate a poor stability behavior.

In [19], the adaptive notch filter was realized in the same cascade-form but with each stage implemented in the controllable realization. This algorithm showed a better stability behavior than the ones in [17], [18] since the corresponding transition matrix of the equivalent state-space realization for the entire notch filter is no longer of a companion form, the stability of this adaptive algorithm, however, can not be ensured.

4.3.2 Convergence study

The convergence of adaptive algorithms of type (2.29) - (2.31) is usually investigated with the associated ordinary differential equations given by (2.33) [9], [21], the following is the first term of (2.33):

$$\frac{d\tilde{\theta}}{dt} = R^{-1}(\tilde{\theta}) f(\tilde{\theta}) \triangleq V_{\tilde{\theta}}, \tag{4.16}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $\tilde{\theta} = \tilde{\theta}(t)$ is a function of a continuous variable t. If (2.29) converges to θ^*, there should exist a solution $\tilde{\theta}(t)$ to (4.16), such that $\lim_{t\to+\infty} \tilde{\theta}(t) = \theta^*$.

It is well known [6], [9], [21] that the class of the gradient-based adaptive algorithms (2.29) converges to one of the stationary points θ^s defined by $f(\tilde{\theta}) = 0$, that is

$$E[e(n)\psi_{\tilde{\theta}}(n)]_{|_{\tilde{\theta}=\theta^s}} = 0. \tag{4.17}$$

Denote $S_\theta(\theta^s) = \left[\begin{array}{cccc} \frac{dV_{\tilde{\theta}}}{d\tilde{\theta}_1} & \frac{dV_{\tilde{\theta}}}{d\tilde{\theta}_2} & \cdots & \frac{dV_{\tilde{\theta}}}{d\tilde{\theta}_N} \end{array} \right]_{|_{\tilde{\theta}=\theta^s}} \triangleq \frac{dV_{\tilde{\theta}}}{d\tilde{\theta}}_{|_{\tilde{\theta}=\theta^s}}$ and $P_\theta^s \triangleq R^{-1}(\tilde{\theta})_{|_{\tilde{\theta}=\theta^s}}$. With (4.17), one has

$$S_\theta(\theta^s) = P_\theta^s \frac{dE[e(n)\psi_{\tilde{\theta}}(n)]}{d\tilde{\theta}}|_{\tilde{\theta}=\theta^s}. \tag{4.18}$$

Linearizing (4.16) locally at $\tilde{\theta} = \theta^s$ yields

$$\frac{d\tilde{\theta}}{dt} \approx S_\theta(\theta^s)(\tilde{\theta} - \theta^s). \tag{4.19}$$

Clearly, a stationary point θ^s is a convergent point of (2.29) if and only if $S_\theta(\theta^s)$ is stable, that is the real part of each eigenvalue of $S_\theta(\theta^s)$ is negative. In the sequel, we will drop the superscript tilde to simplify the notations.

In what follows, we will analyze the convergence of the proposed algorithm (4.15). First of all, we show that this algorithm has exactly the same convergence behavior as the one by the same (4.15) but with $\psi^*_{\theta_\omega}(n)$ replaced by the exact $\psi_{\theta_\omega}(n)$. One notes that $g^*_{1k}(n) = \frac{D_k(z^{-1})}{D_k(\rho z^{-1})} g_{1k}(n)$. Without loss of generality, assume that the closest notch frequency to ω_l^0 is $\omega_l, \forall l$. Since $g_{1k}(n)$ can be decomposed as $g_{1k}(n) = P_k cos(\omega_k^0 n + \psi_k - \phi_k) + \zeta_k(n)$, where ψ_k is the phase response of $\frac{dH_1(z^{-1})}{d\hat{\omega}_k}|_{\hat{\theta}_\omega=\theta_\omega}$ evaluated at $\omega = \omega_k^0$ and $\zeta_k(n)$ contains no frequency component of ω_k^0. Noting the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

fact that $\frac{D_k(z^{-1})}{D_k(\rho z^{-1})}$ can be considered as the ideal notch filter, one has

$$g_{1k}^*(n) = \begin{cases} g_{1k}(n) - P_k cos(\omega_k^0 n + \psi_k - \phi_k), & if\ \omega_k = \omega_k^0 \\ g_{1k}(n), & if\ \omega_k \neq \omega_k^0 \end{cases}$$

When $\omega_k = \omega_k^0$, the notch filter output $e(n)$ has no sinusoidal component corresponding to ω_k^0. With the assumption in mind that $v(n)$ and $\phi_k,\ \forall k$ are all statistically independent, it then follows that $E[e(n)P_k cos(\omega_k^0 n + \psi_k - \phi_k)] = 0$. Therefore, one has $E[e(n)g_{1k}^*(n)] = E[e(n)g_{1k}(n)],\ \forall k$, and hence

$$E[e(n)\psi_{\theta_\omega}^*(n)] = E[e(n)\psi_{\theta_\omega}(n)]. \tag{4.20}$$

As shown above, the convergence behavior of the adaptive algorithms of type (2.29) is determined by $E[e(n)\psi(n)]$, (4.20) implies that the our proposed algorithm (4.15) has the same convergence properties as these of the same algorithm but with the revised gradient vector $\psi_{\theta_\omega}^*(n)$ replaced by the exact $\psi_{\theta_\omega}(n)$. Therefore, the convergence of the former can be analyzed via the latter, which is to be carried out below.

For the direct-form-based parametrization θ_a, it has been shown [6], [56] that $\theta_a(n)$ under the Steiglitz-McBride adaptive algorithm converges to the parameter vector θ_a^s, which is the unique stationary point and corresponds to the true frequencies underlying the signal $u(n)$.

Note $\psi_{\theta_\omega}(n) = J\psi_{\theta_a}(n)$, where J is the Jacobian matrix with its kth row given by $J(k,:) = \frac{d\theta_a^{\mathcal{T}}}{d\omega_k}$ and is nonsingular if and only if $\omega_i \neq \omega_j,\ \forall i \neq j$ [18], which is assumed in the sequel. Clearly, $E[e(n)\psi_{\theta_\omega}(n)] = 0$ leads to $E[e(n)\psi_{\theta_a}(n)] = 0$, which implies that any stationary θ_ω^s in the parametrization θ_ω corresponds to the unique stationary point θ_a^s in θ_a. The question to be asked is if such a θ_ω^s is a convergent

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

point, that is if $S_{\theta_\omega}(\theta_\omega^s)$ is stable. The answer is positive. In fact, it follows from (4.18) that

$$S_{\theta_a}(\theta_a^s) = P_{\theta_a}^s \frac{dE[e(n)\psi_{\theta_a}(n)]}{d\theta_a}|_{\theta_a=\theta_a^s}, \quad \frac{dV_{\theta_\omega}}{d\omega_k}_{|_{\theta\omega=\theta_\omega^s}} = P_{\theta_\omega}^s \frac{dE[e(n)\psi_{\theta_\omega}(n)]}{d\omega_k}|_{\theta_\omega=\theta_\omega^s}.$$

With $\psi_{\theta_\omega}(n) = J\psi_{\theta_a}(n)$ and $E[e(n)\psi_{\theta_a}]_{|\theta a=\theta_a^s} = 0$, it turns out that

$$\frac{dV_{\theta_\omega}}{d\omega_k}_{|_{\theta\omega=\theta_\omega^s}} = P_{\theta_\omega}^s J_s \frac{dE[e(n)\psi_{\theta_a}(n)]}{d\omega_k}|_{\theta_\omega=\theta_\omega^s}$$

with $J_s \stackrel{\triangle}{=} J$, evaluated at $\theta_\omega = \theta_\omega^s$. Noting that $P_{\theta_\omega}^s = J_s^{-\mathcal{T}} P_{\theta_a}^s J_s^{-1}$ (see (2.31)) and $\frac{dE[e(n)\psi_{\theta a}(n)]}{d\omega_k}_{|_{\theta\omega=\theta_\omega^s}} = \frac{dE[e(n)\psi_{\theta a}(n)]}{d\theta_a}_{|_{\theta a=\theta_a^s}} J_s^{\mathcal{T}}(k,:)$, one has

$$\frac{dV_{\theta_\omega}}{d\omega_k}_{|_{\theta\omega=\theta_\omega^s}} = J_s^{-\mathcal{T}} P_{\theta_a}^s \frac{dE[e(n)\psi_{\theta_a}(n)]}{d\theta_a}_{|_{\theta a=\theta_a^s}} J_s^{\mathcal{T}}(k,:)$$

and hence

$$S_{\theta_\omega}(\theta_\omega^s) = J_s^{-\mathcal{T}} S_{\theta_a}(\theta_a^s) J_s^{\mathcal{T}}. \tag{4.21}$$

Since $S_{\theta_a}(\theta_a^s)$ is stable, $S_{\theta_\omega}(\theta_\omega^s)$ is stable. To the second term of (2.33), using the same approach, it is easy to see that: $U_{\theta_\omega}(\theta_\omega^s) = J_s U_{\theta_a}(\theta_a^s) J_s^{\mathcal{T}}$, which means

$$U_{\theta_\omega}(\theta_\omega^s) = 0 \Leftrightarrow U_{\theta_a}(\theta_a^s) = 0.$$

We conclude that our proposed adaptive filter (4.15) converges to the true frequencies when the RPE algorithm (2.31) is used for updating the parameter vectors.

It is interesting to note that $S_{\theta_\omega}(\theta_\omega^s)$ and $S_{\theta_a}(\theta_a^s)$ have exactly the same eigenvalues. This implies that the two parametrizations θ_ω and θ_a have the same convergence speed when the RPE is used. Applying the same procedure, one can show that for

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

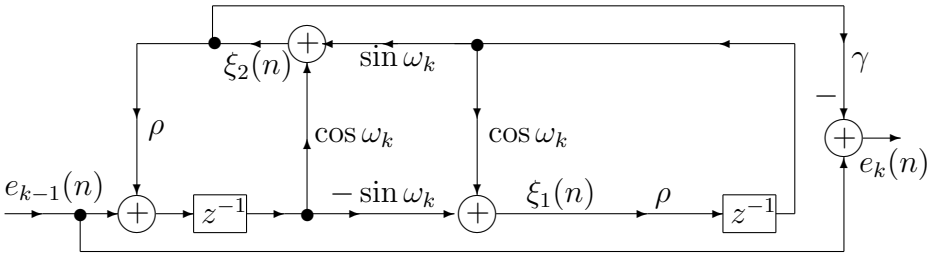


Figure 4.2: Implementation structure of the normal state-space realization (4.13).

the LMS algorithm, one has

$$S_{\theta_\omega}(\theta_\omega^s) = J_s S_{\theta_a}(\theta_a^s) J_s^{\mathcal{T}}.$$

In this case, $S_{\theta_\omega}(\theta_\omega^s)$ is stable due to $S_{\theta_a}(\theta_a^s)$ stable, but has different eigenvalues from those of the latter. Therefore, these two parametrizations have different convergence speeds.

4.4 Implementation issues

First of all, one notes that the normal state-space realization (4.13) can be implemented with the structure depicted by Figure 4.2, where $\gamma = -2(1-\rho)$.

4.4.1 Implementation complexity and stability

Now, let's compare the implementation complexity of the proposed algorithm and the one in [19].[3] One observes that the two algorithms have to compute the factors $\cos\omega_k$ and $\sin\omega_k$. These can be done efficiently using the CORDIC technique (see [3], [6] and the references therein). In what follows, we consider that the computation

[3]Here, we assume that the parameters ρ and β in [19] are time-invariant. Otherwise, more computations are required for this algorithm.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

for the product of a coefficient such as $\cos\omega_k$ and $\sin\omega_k$ with any signal requires one multiplication. The implementation complexity is normally measured by the amount of multiplications and additions needed for computing the signals and updating the states at each iteration.

- *Structure complexity*

  For the proposed algorithm, as shown in Figure 4.2, $7N$ multiplications and $4N$ additions are needed, while for the one, that is Equation (14) in [19], the corresponding values are $6N$ and $4N$, respectively.

- *Computing gradient signals*

  The gradient signals for the proposed algorithm are given by (4.14), which can be implemented with the same structure as Figure 4.2, where $e_{k-1}(n)$ and $e_k(n) = \gamma\xi_2(n) + e_{k-1}(n)$ are replaced with $e(n) + \rho u(n)$ and $g_k^*(n) = 2\xi_1(n)$, respectively. Clearly, $8N$ multiplications and $4N$ additions are needed, while for the counterpart of (4.14), that is Equations (17) - (18) of [19], the corresponding values are $10N$ and $5N$, respectively.

Clearly, the proposed algorithm has almost the same implementation complexity as the one in [19].

FWL implementation is the ultimately inevitable reality of all adaptive filters. A well-designed stable adaptive filter with infinite precision may become unstable since the coefficients of the filter have to be truncated into an FWL format. When the coefficients ρ, $\xi_c \triangleq \cos\omega_k$ and $\xi_s \triangleq \sin\omega_k$ in Figure 4.2 are truncated with B_c bits, their FWL version is $\hat{\rho} \triangleq \rho - \epsilon_\rho$, $\hat{\xi}_c \triangleq \xi_c - \epsilon_c$ and $\hat{\xi}_s \triangleq \xi_s - \epsilon_s$, respectively, where $0 \leq \epsilon_\rho,\ \epsilon_c,\ \epsilon_s \leq 2^{-B_c}$.[4] With $\rho_k \triangleq \hat{\rho}\sqrt{\hat{\xi}_c^2 + \hat{\xi}_s^2}$ and $\hat{\omega}_k \triangleq atan\frac{\hat{\xi}_s}{\hat{\xi}_c}$, one can see that the A-matrix of the corresponding state-space realization to the FWL implemented

[4]Here, $\omega_k \leq \frac{\pi}{2}$ is assumed. When $\frac{\pi}{2} < \omega_k < \pi$, $\hat{\xi}_c = \xi_c + \epsilon_c$ and the same argument can be obtained.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

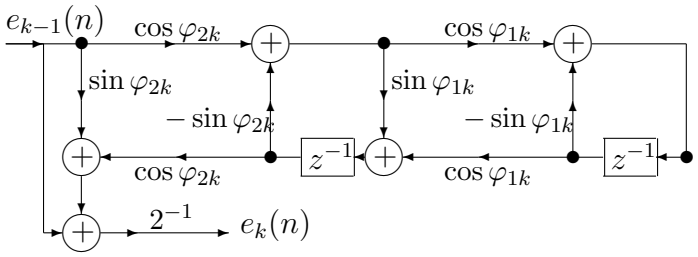


Figure 4.3: Normalized lattice structure of the 2nd-order notch filter (4.22).

structure Figure 4.2 is

$$\hat{A}_k = \rho_k \begin{bmatrix} \cos\hat{\omega}_k & -\sin\hat{\omega}_k \\ \sin\hat{\omega}_k & \cos\hat{\omega}_k \end{bmatrix},$$

which is normal and satisfies $|\lambda_l(\hat{A}_k)| = \rho_k \leq \rho < 1$, $\forall l,\ k$. It is then easy to understand that our proposed adaptive algorithm is BIBO stable even when the filter parameters are implemented with FWL. This is of great importance for real-time applications.

4.4.2 The NL structure-based notch filters

The lattice-based structures have been popularly used in digital filter design and implementation. In [105], an alternative constrained notch filter model

$$H(z^{-1}) = \prod_{k=1}^{N} \frac{1+\sin\varphi_{2k}}{2} \frac{1+2\sin\varphi_{1k}z^{-1}+z^{-2}}{1+\sin\varphi_{1k}(1+\sin\varphi_{2k})z^{-1}+\sin\varphi_{2k}z^{-2}} \tag{4.22}$$

was used and implemented with the normalized lattice structure depicted in Figure 4.3.

The relationship of φ_{1k}, φ_{2k} with the notch frequency ω_k and 3dB attenuation band-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

width BW of this transfer function is

$$\omega_k = \varphi_{1k} + \frac{\pi}{2}, \ |\varphi_{1k}| < \frac{\pi}{2}, \ \sin\varphi_{2k} = \frac{1 - tan\frac{BW}{2}}{1 + tan\frac{BW}{2}} \ \forall k. \tag{4.23}$$

It is interesting to note that both models (4.10) and (4.22) are almost identical when BW is very small. Like the proposed normal state-space structure depicted in Figure 4.2, the normalized lattice structure Figure 4.3, as shown in [6], is also BIBO stable. It is well known that multiplications are much more costly than additions and shifting operations and that a multiplication of a signal with a power-of-two coefficient 2^{-L}, where L is an integer, can be much more efficiently implemented with a shifting operation [107]. One can see that with the notch filter implemented using the structure by Figure 4.3, it requires $8N$ multiplications, $5N$ additions and N shifting operations to compute each output sample $e_k(n)$. Look at Figure 4.2. Since ρ is very close one, we can take $\rho = 1-2^{-8}$, approximately equal to 0.9961, and hence the multiplications of $\rho\xi_l(n)$, $l = 1, 2$ can be implemented with $\xi_l(n)-2^{-8}\xi_l(n)$, which needs one addition and one shifting operation. Similarly, the multiplier $\gamma = -2(1-\rho) = -2^{-7}$ can be implemented with a shifting operation. With these in mind, one can see that our proposed structure requires only $4N$ multiplications plus $6N$ additions and $3N$ shifting operations.

Look at the normal realization R_k^{nor} in (4.13). The zeros of the corresponding transfer function is the eigenvalues of $Z_k \stackrel{\triangle}{=} A_k - B_kC_k$ [75]. Since ρ is very close to one, $C_k \approx 0$ and hence $Z_k \approx A_k$. It is shown [75] that a matrix M achieves a minimal eigenvalue sensitivity if and only if M is normal. This means that our proposed structure has a minimal pole sensitivity as well as an almost minimal zero sensitivity. Consequently, this structure has the capability to reduce the FWL effects. Consider a 2nd-order constrained notch filter by (4.10) with a notch frequency equal to $\pi/4$ and $\rho = 1 - 2^{-8}$. The corresponding 3dB bandwidth is $BW = 2^{-8}\pi$. With

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

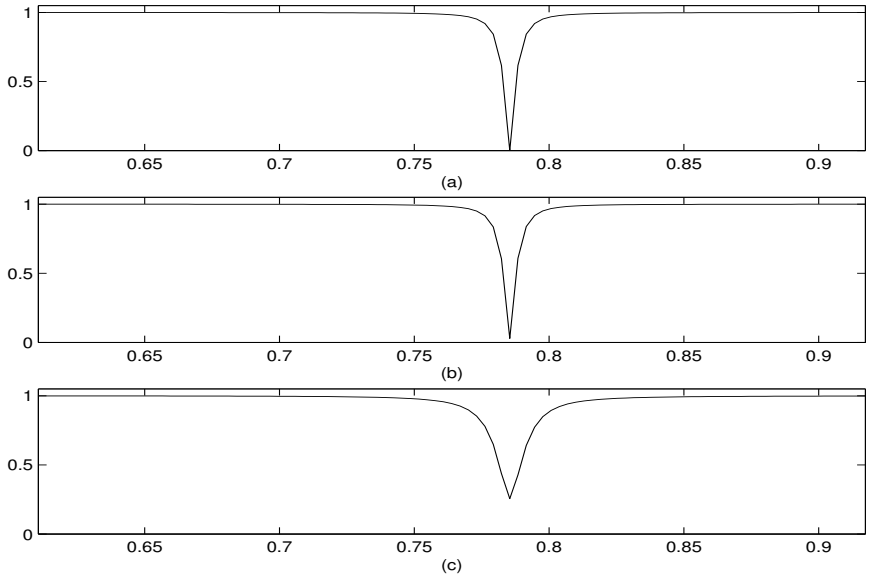


Figure 4.4: Magnitude responses of 2nd-order notch filters: (a) the ideal response of (4.10) with $\rho = 1 - 2^{-8} \approx 0.9961$, $\omega_1 = \pi/4$; (b) the response obtained for Figure 4.2 implemented with 8 bits; (c) the response obtained for Figure 4.3 implemented with 8 bits, where the x-axis denotes the angular frequency.

(4.23), the corresponding notch filter by (4.22) can be obtained. These two transfer functions have almost the same magnitude response which is shown in Figure 4.4 (a). Now, we implement the two structures depicted in Figures 4.2 and 4.3 with all their coefficients truncated into a 8-bit format and then compute the corresponding magnitude responses which are presented in Figures 4.4 (b) and 4.4 (c), respectively. Clearly, the proposed structure has a better performance against the FWL errors which can not be avoided in real-time applications.

4.5 Application in speech processing

It is well known that speech signal is usually processed frame by frame with an interval of $10 - 30$ ms. Voiced speech signals can be modeled with (4.8) and for most of the cases the speech signal shows highly periodicity for such a short time interval. The period, called *pitch*, is a very important parameter in speech encoder

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

design and there exist several traditionally used algorithms to detect the pitch such as the average magnitude difference function and zero crossing measure [29], [30]. The problem with such algorithms is that the obtained pitch is always an integer, therefore, there exists an estimation error constantly for most of the cases. Such an error may greatly affect the quality of the synthesized speech signal.

We can model a voiced speech signal using the following:

$$u(n) = \sum_{k=1}^{N} M_k cos(k\omega_0 n - \phi_k) + v(n), \tag{4.24}$$

where ω_0 is the fundamental (angular) frequency. In this case, we have one frequency, ω_0, to determine instead of N. Our proposed algorithm can be adapted for this situation easily. In fact, with the constraint

$$\theta_\omega(n) = \begin{bmatrix} 1 & 2 & \cdots & k & \cdots & N \end{bmatrix}^{\mathcal{T}} \omega_0(n),$$

one has

$$-\frac{de(n)}{d\omega_0} = -\sum_{k=1}^{N} \frac{de(n)}{d\omega_k}\frac{d\omega_k}{d\omega_0} = \begin{bmatrix} 1 & 2 & \cdots & k & \cdots & N \end{bmatrix} \psi_\omega(n) \tag{4.25}$$

where ψ_ω, as defined before, is the derivative of the prediction error with respect to the frequency vector θ_ω. Therefore, the corresponding adaptive algorithm to the fundamental frequency estimation is the same as (2.29) but with the vector $\theta(n)$ replaced by the (scalar) estimated fundamental frequency $\hat{\omega}_0(n)$ and hence $P_{\omega_0}(n)$ is a *scalar* instead of a *matrix* of order $N \times N$.

With the obtained fundamental frequency $\hat{\omega}_0$, the corresponding synthesized speech

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

signal $\hat{u}(n)$ can be obtained with

$$\hat{u}(n) = \sum_{k=1}^{N} M_k cos(k\hat{\omega}_0 n - \phi_k), \tag{4.26}$$

where $\{M_k, \phi_k\}$ can be determined by minimizing

$$\sigma^2 \triangleq \frac{1}{L} \sum_{n=1}^{L} [u(n) - \hat{u}(n)]^2 \tag{4.27}$$

with respect to these variables, where L is the number of samples in one frame. In fact, denoting $M_k^c \triangleq M_k cos\phi_k$ and $M_k^s \triangleq M_k sin\phi_k$ for all k, one can see that σ^2 is a quadratic function in $\{M_k^c,\ M_k^s\}$ with $\hat{\omega}_0$ and $u(n)$ given. Therefore, σ^2 can be minimized with respect to $\{M_k^c,\ M_k^s\}$ easily. With the obtained optimal $\{M_k^c,\ M_k^s\}$, one can then convert them into $\{M_k,\ \phi_k\}$ steadily.

4.6 Numerical examples and simulations

In this section, we present two numerical examples to examine the performance of our proposed algorithm and compare it with some existing ones. Moreover, an application of our proposed algorithm in pitch detection of speech signals is given as the third example. For convenience, we denote the proposed algorithm, the normalized lattice-based one, and the ones in [17], [19] as Alg_{new}, Alg_{nl}, Alg_{Neh} and Alg_{Li}. For the simulations, the same RPE given by (2.31) is utilized, where $P_\theta(0) = 0.01 I_N$, $\varrho_\infty = 1$, $\varrho_0 = 0.99$, $\varrho(0) = 0.85$.

Example 4.1: This example is for demonstrating the performance of Alg_{new} and

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Alg_{nl}. The signal $u(n)$ is generated with

$$u(n) = 2\cos(\omega_1^0 n - \phi_1) + \sigma_0 v_0(n),\ \ n = 0, 1, 2, \cdots, 2000,$$

where ω_1^0 is set to one with ϕ_1 randomly generated, $v_0(n)$ is a white noise with zero-mean and unit variance, and σ_0 is chosen such that the signal-to-noise ratio (SNR) is 10dB for this frequency component.

With the same Steiglitz-McBride scheme shown in Figure 4.1, we run Alg_{new} and Alg_{nl}, where the exact gradient signals are implemented with the proposed normal state-space realization and a normalized lattice structure (to ensure the BIBO stability), respectively. In a total of 20 trials, both algorithms all converge to the true frequency $\omega_1^0 = 1$ with random initial values for $\omega_1(0)$. Figures 4.5 (a) and (b) show the evolution of the notch frequency $\omega_1(n)$ with Alg_{new} and Alg_{nl} for a typical trial, respectively. Figure 4.5 (c) shows the corresponding mean square error (MSE) computed with the 20 trials, where the dotted line is for Alg_{nl} and the solid line, for Alg_{new}. Both Alg_{new} and Alg_{nl} demonstrate an excellent stability behavior and yield almost the same performance in terms of convergence speed and frequency estimation accuracy.

Example 4.2: In this example, the signal $u(n)$ is generated in the same way as in Example 4.1 with three frequency components:

$$u(n) = 2\cos(\omega_1^0 n - \phi_1) + 2\cos(\omega_2^0 n - \phi_2) + 2\cos(\omega_3^0 n - \phi_3) + \sigma_0 v_0(n),$$

where $\omega_1^0 = 0.5,\ \omega_2^0 = 1,\ \omega_3^0 = 2$. Once again, σ_0 is chosen such that the signal-to-noise ratio (SNR) is 10dB for each frequency component.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

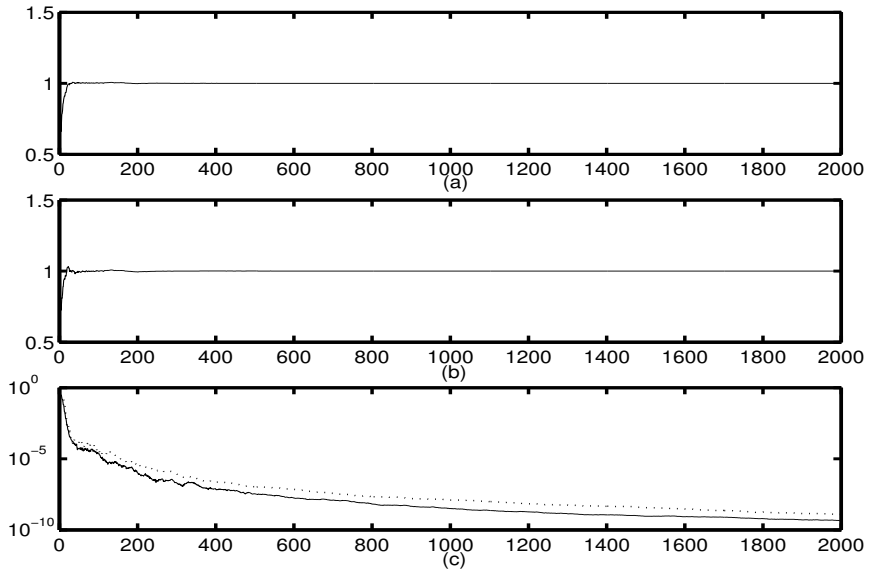


Figure 4.5: (a) and (b): Evolution of $\omega_1(n)$ in Example 4.1 for a typical trial using Alg_{new} and Alg_{nl}, respectively; (c) MSE learning curves for $\omega_1(n)$ with the solid-line denoting the one by Alg_{new} and the dotted-line, by Alg_{nl}, where the x-axis denotes the iteration number.

We generate 50 different sets of $u(n)$, each of them contains 2000 samples. For each run, in order to have a fair comparison we first generate $\theta_\omega(0)$ randomly for Alg_{new} and Alg_{Li}, then convert it to the equivalent $\theta_a(0)$ for Alg_{Neh}.

In a total of 50 trials, the number of runs, for which Alg_{new}, Alg_{Neh} and Alg_{Li} converge after 2000 iterations, is 50, 28 and 30, respectively. Figure 4.6 shows the evolution of the estimated frequencies by the proposed Alg_{new} for a typical trial. After 2000 iterations,

$$\theta_\omega(2000) = \begin{bmatrix} 0.5000 & 1.0002 & 2.0001 \end{bmatrix}^{\mathcal{T}}.$$

The evolution of MSE for each frequency is computed with the 30 trials, for which both Alg_{new} and Alg_{Li} converge to the true frequencies. Figure 4.7 shows the evolution of the MSE for $\omega_2(n)$ ($\omega_2^0 = 1$), where the solid line is for Alg_{new} and the dotted line, for Alg_{Li}. As seen, Alg_{new} yields a higher accuracy of frequency estimation.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

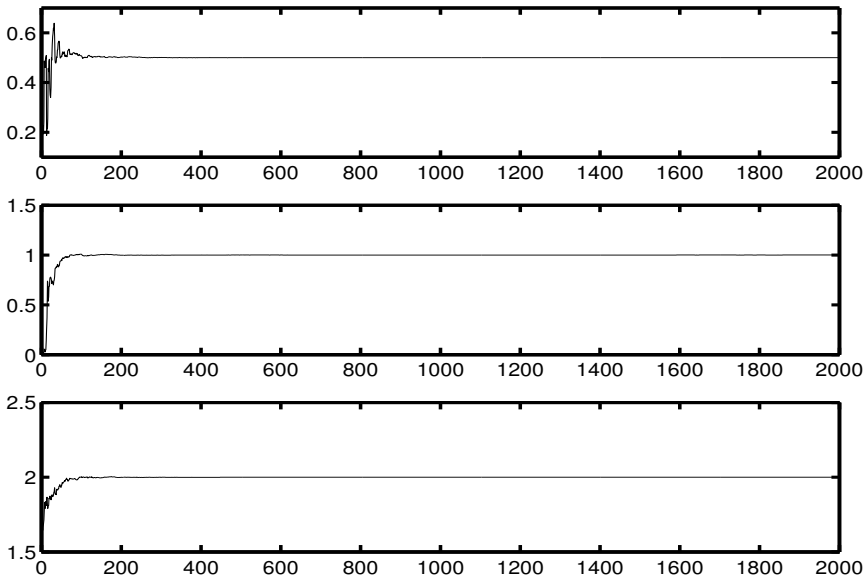


Figure 4.6: Evolution of $\omega_k(n),\ k = 1, 2, 3$ in Example 4.2 for a typical trial using Alg_{new}, where the x-axis denotes the iteration number.

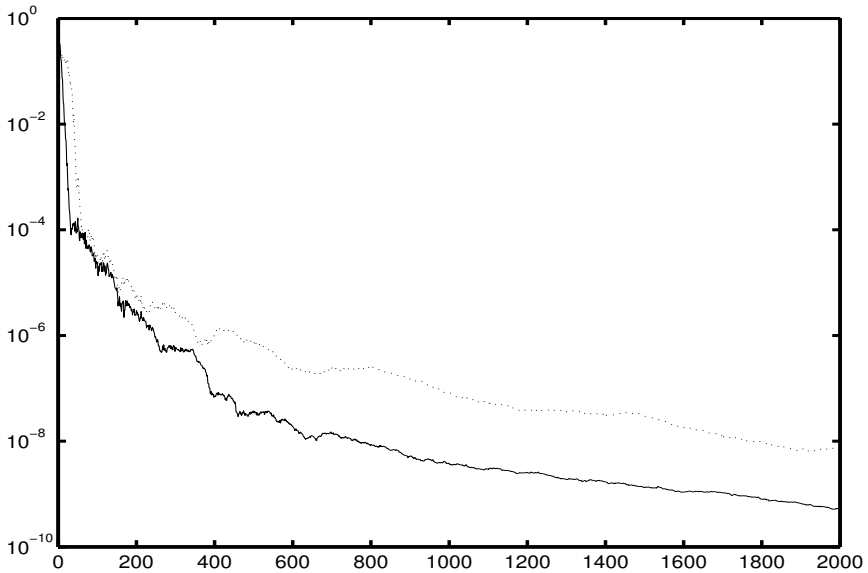


Figure 4.7: MSE learning curves for $\omega_2(n)$ in Example 4.2 with the solid-line denoting the one by Alg_{new} and the dotted-line, by Alg_{Li}, where x-axis denotes the iteration number.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

During the 22 trials in which Alg_{Neh} fails to converge, instability has been observed (since random initial values for frequencies are used), while Alg_{Li} remains stable even when it does not converge after 2000 iterations. When the true frequencies ω_k^0 are small and close to each other, both Alg_{Neh} and Alg_{Li} have difficulty in converging to their true parameters, while Alg_{new} yields a much better convergence performance. This seems due to, besides the well known convergence properties of the Steigliz-McBride adaptive scheme, the poor stability of these two algorithms. In fact, it has been observed that the magnitudes of signals in Alg_{Neh} and Alg_{Li} have a very large dynamic range. This is a crucial disadvantage for both algorithms to be used for real-time implementations, in which signals should be well confined within a certain dynamic range.

Example 4.3: In this example, the signal is a frame of voiced speech signal with $L = 500$. This speech signal is sampled with $20k$Hz. We estimate the fundamental frequency of the speech signal, or called pitch, by using the classical average magnitude difference function (AMDF) [30] and our proposed algorithm with $N = 16$. With AMDF, the fundamental frequency is 0.0706. The corresponding error variance, as defined in (4.27) where $\hat{u}(n)$ is computed with (4.26) with $N = 16$, is 339.6915. We run the proposed adaptive algorithm with $\hat{\omega}_0(0) = 0.05$ and $P_{\omega_0}(0) = 20$, and obtain the fundamental frequency $\hat{\omega}_0(500) = 0.0708$. Then the corresponding synthesized speech signal is computed with (4.26) for $N = 16$. The error variance is 144.5735, much smaller than the one obtained with AMDF. In Figure 4.8, the solid line is the original speech signal while the pointed line is the synthesized one computed with the pitch estimated with our proposed algorithm.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

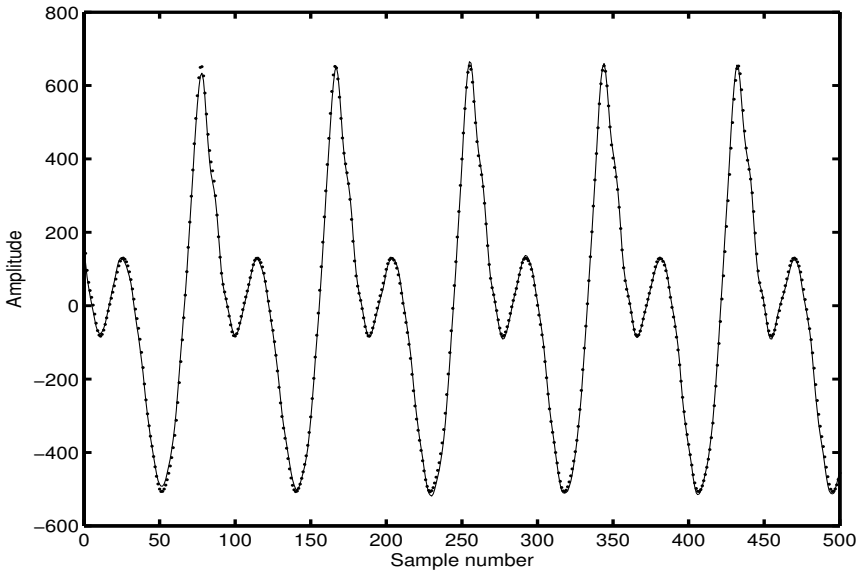


Figure 4.8: The original speech signal and the synthesized one.

4.7 Conclusions

Our contribution in this chapter is three-fold. The first is to have derived a new condition with which the BIBO stability of the adaptive IIR filters can be guaranteed. The second one is to have derived a new adaptive algorithm for the constrained notch filters based on a state-space realization approach. This algorithm is as efficient as the one in [19]. It is shown that this adaptive algorithm is absolutely stable no matter it is implemented with infinite or finite precision, which is very important in real-time application. The proof of convergence of the algorithm, as the third one, has been provided. In addition, the proposed structure has been compared with the normalized lattice realization of the constrained notch filters in terms of structure complexity and pole/zeros sensitivities. Simulations have been carried out, which show that the proposed adaptive algorithm has a much better stability and convergence behavior than those in [17] and [19], and an application of the proposed adaptive algorithm in speech signal processing has also been given.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 5

An Input Balanced Realization-Based Adaptive IIR Filter

It should be pointed out that there have been many results available in designing adaptive filters [7], [60], where it is assumed that the filter parameters and the signals are computed with infinite precision. In real-time applications, finite precision implementation is the ultimately inevitable reality that any *a priori* well-designed adaptive system has to confront. The essential features of such an implementation, as pointed out in Chapter 3, are that computations involved are carried out in finite precision, and that all the signals, particularly those intermediate variables or states, should be constrained to a certain dynamical range. Therefore, the performance of a well-designed adaptive filter can be totally different from the desired one when finite precision implementation is concerned.

As mentioned in Chapter 3, the state-space representation provides a class of realizations for filter implementation. The adaptive state-space filter using the proposed

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

realization R_ρ with parametrization θ_I, is shown to yield better performance in terms of reducing parametrization sensitivity and sustaining the dynamic range for the signals than the classical DFII-based ones that are parametrized using θ_d and θ_c. This, however, relies on the initial estimate of the true system. In fact, this proposed realization is system dependent. In addition, the stability of this structure can not be ensured in a strict sense.

Among all the realizations, the internally and input balanced realizations demonstrate many interesting properties such as very low parameter sensitivity, very small roundoff noise gain and free of overflow oscillation [75], [93]. All these suggest that it is desired to use the balanced realizations for adaptive filter design [50], [51], [98] - [100]. The main difficulty in using the balanced realizations for adaptive filter design is how to parametrize these realizations. In [100], an interesting adaptive filtering algorithm was developed using the internally balanced realizations characterized with a canonical parametrization developed in [101]. The main drawbacks of this parametrization are that it is not minimal and can not represent those filters whose Hankel singular values are not distinct. In addition, whether the adaptive filter is BIBO stable is still an open problem.

The objective of this chapter is to develop an adaptive IIR state-space filter using an input balanced realization, which was used for digital filter implementation in [51]. In this proposed state-space filter, the input balanced realization is characterized with a minimal parametrization. It is shown that this new parametrization does not bring extra stationary points and the corresponding filter has exactly the same convergent points as the one based on the traditional direct-form (DF) structures, and more importantly, that the proposed adaptive filter is BIBO stable, which does not hold for most of the existing adaptive IIR filters. Another outstanding property of this proposed structure is that the instantaneous filter is always l_2-scaled. This

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

makes the signals involved be well confined within a certain dynamic range.

An outline of this chapter is given as follows. Section 5.1 provides some preliminaries on the balanced state-space realizations. Our main contributions in this chapter are found in Sections 5.2 and 5.3. An input balanced realization-based adaptive IIR filter is derived in Section 5.2 using the output error (OE) scheme. Section 5.3 is devoted to analyzing the performance of this filter structure in terms of parametrization sensitivity, stability and convergence. An alternative parametrization is proposed in Section 5.4. One of the exciting properties of this alternative parametrization is that its stability region is exactly the entire parameter space, which means that this parametrization can model any stable transfer function with no constraints required. In Section 5.5, an application of this input balanced realization-based IIR filter in acoustic echo cancellation is presented. Numerical examples and simulations are given in Section 5.6, which support our theoretical analysis and more interestingly, the echo canceller using our proposed input balanced realization shows very nice performance.

5.1 Balanced realizations

Consider the state-space representation (2.6) of a linear filter:

$$\begin{cases} x(n+1) &= Ax(n) + Bu(n) \\ y(n) &= Cx(n) + du(n) \end{cases}$$

The controllability and observability Gramians, denoted as W_c and W_o, respectively,

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

of a realization (A, B, C, d) are the solutions of the following Lyapunov equations

$$\begin{cases} W_c &= AW_cA^{\mathcal{T}} + BB^{\mathcal{T}} \\ W_o &= A^{\mathcal{T}}W_oA + C^{\mathcal{T}}C. \end{cases} \tag{5.1}$$

A realization is called *internally balanced* if its controllability and observability Gramians W_c and W_o satisfy

$$W_c = W_o = \begin{bmatrix} \sigma_1 & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2 & 0 & \cdots & 0 \\ & & & \vdots & \\ 0 & 0 & 0 & \cdots & \sigma_K \end{bmatrix}, \tag{5.2}$$

where $\sigma_k > 0$, $\forall k$ are called *Hankel singular values* of the filter $H(z^{-1})$. Such a realization is of minimal transfer function sensitivity [78].

Among the infinite set of realizations of a linear filter, there exists another class of interesting realizations, called *input balanced* (IB) realizations [75]. A realization, denoted as R_{IB}, is said IB if its controllability Gramian W_c given by (5.1) is the identity matrix:

$$I_K = AA^{\mathcal{T}} + BB^{\mathcal{T}}. \tag{5.3}$$

One notices that an IB realization means $W_c = I_K$ and hence is inherently l_2-scaled. Therefore, an adaptive filter implemented in the IB realizations can solve the dynamic range problem easily. In addition, the IB realizations have very good performance gainst the FWL effects [38], [75].

Intuitively, one would suggest to use the optimal FWL realizations for these filters in order to reduce the FWL effects. The problem, however, is that these realizations are usually fully parametrized using nontrivial parameters and hence computationally

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

inefficient [52]. In addition, it seems impossible to find a direct parametrization to characterize those realizations.

Obser [101] showed that under the assumption that the Hankel singular values are all *distinct* an internally balanced realization can be parametrized using the following parametrization

$$\theta_{Bal} = \begin{bmatrix} \sigma_1 & \cdots & \sigma_k & \cdots & \sigma_K & \tau_1 & \cdots & \tau_k & \cdots & \tau_K & s_1 & \cdots & s_k & \cdots & s_K & d \end{bmatrix}^{\mathcal{T}}, \tag{5.4}$$

where $s_k = \pm 1$, $\forall k$. Based on such a parametrization, Sankaran and Beex [100] developed an internally balanced realization-based adaptive IIR filter. Such a parametrization, on the one hand, can ensure that the instantaneous realizations of $H(z^{-1})$ and $H_g(z^{-1})$ are internally balanced and hence the corresponding adaptive filter has very good FWL behavior, on the other hand, however, it is not a minimal parametrization and hence may introduce extra local minima or saddle points [3]. Another drawback of their algorithm is its high complexity due to the fact that the internally balanced realization are fully parametrized.

5.2 An IB-based state-space adaptive filter

Like the internally balanced realizations, the IB realizations are usually dense, fully parametrized using nontrivial elements. The key to make use of the IB realizations in adaptive filter design is how to parametrize such realizations and to reduce the implementation complexity.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

5.2.1 A minimal parametrization of IB realizations

Johns *et al* [38] showed that any stable analog transfer function $F(s)$ can be implemented in the following state-space realization (Φ, L, J, D), called orthonormal ladder filter:

$$\Phi = \begin{bmatrix} 0 & \alpha_1 & 0 & 0 & \cdots & 0 & 0 \\ -\alpha_1 & 0 & \alpha_2 & 0 & \cdots & 0 & 0 \\ 0 & -\alpha_2 & 0 & \alpha_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & \alpha_{K-1} \\ 0 & 0 & 0 & 0 & \cdots & -\alpha_{K-1} & -\frac{1}{2}\alpha_K^2 \end{bmatrix}, \quad L = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \alpha_K \end{bmatrix} \tag{5.5}$$

with $J \in \mathcal{R}^{1\times K}$ having no special form, where $\alpha_k > 0, \ \forall k$.

With the Bilinear transfer transformation, the discrete-time counterpart of $F(s)$ can be obtained $H(z^{-1}) = F(s)|_{s=\frac{z-1}{z+1}}$, which is also stable and has a realization $(A_{ib}, B_{ib}, C_{ib}, d)$ given by

$$\begin{cases} A_{ib} &= (I_K + \Phi)(I_K - \Phi)^{-1}, \quad B_{ib} = \frac{\sqrt{2}}{2}(I_K + A_{ib})L \\ C_{ib} &= \frac{\sqrt{2}}{2}J(I_K + A_{ib}), \qquad\qquad d = D + C_{ib}(I_K + A_{ib})^{-1}B_{ib}. \end{cases} \tag{5.6}$$

It is easy to verify that $\Phi + \Phi^{\mathcal{T}} = -LL^{\mathcal{T}}$, which implies that the continuous-time realization (Φ, L, J, D) has a controllability Gramian equal to the identity matrix and hence so does its discrete-time counterpart $(A_{ib}, B_{ib}, C_{ib}, d)$ [101]. Therefore, the realization (5.6) is input balanced.

Denote

$$\theta_{ib} \triangleq \begin{bmatrix} \alpha_1 & \cdots & \alpha_K & C_{ib} & d \end{bmatrix}^{\mathcal{T}}. \tag{5.7}$$

One can see that this IB realization and hence $H(z^{-1})$ can be characterized with

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

the minimal parametrization θ_{ib}. Such a parametrization has the property that the set of all stable transfer functions $H(z^{-1})$ (of order K) is one-to-one mapped into the parameter space defined as

$$S_{\theta_{ib}} \triangleq \{\theta_{ib} : \alpha_k > 0, \forall k\}. \tag{5.8}$$

Based on this new parametrization θ_{ib}, a new adaptive filter is developed, where $H(z^{-1})$ and $H_g(z^{-1})$ are implemented in the IB realization (5.6).

5.2.2 An OE scheme-based adaptive algorithm

When the OE adaptive scheme is used, $\psi_{\theta_{ib}}(n) = \frac{\partial y(n)}{\partial \theta_{ib}}|_{\theta_{ib}=\theta_{ib}(n)}$. First of all, with $y(n) = C_{ib}x(n) + du(n)$ we have directly

$$\frac{\partial y(n)}{\partial C_{ib}^{\mathcal{T}}} = x(n), \quad \frac{\partial y(n)}{\partial d} = u(n),$$

while $\frac{\partial y(n)}{\partial \alpha_k} = C_{in}\frac{\partial x(n)}{\partial \alpha_k}$, with $x_{\alpha_k}(n) \triangleq \frac{\partial x(n)}{\partial \alpha_k}$ computed from

$$x_{\alpha_k}(n+1) = A_{ib}x_{\alpha_k}(n) + [\frac{\partial A_{ib}}{\partial \alpha_k}x(n) + \frac{\partial B_{ib}}{\partial \alpha_k}u(n)], \ \forall k.$$

Therefore, our proposed adaptive algorithm can be entirely described with the following equations

$$\begin{cases} y(n) &= C_{ib}(n)x(n) + d(n)u(n) \\ \frac{\partial y(n)}{\partial \alpha_k} &= C_{in}(n)x_{\alpha_k}(n), \qquad \forall k \\ \frac{\partial y(n)}{\partial C_{ib}^{\mathcal{T}}} &= x(n) \\ \frac{\partial y(n)}{\partial d} &= u(n) \end{cases} \tag{5.9}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

and

$$\begin{cases} x(n+1) &= A_{ib}(n)x(n) + B_{ib}(n)u(n) \\ x_{\alpha_k}(n+1) &= A_{ib}(n)x_{\alpha_k}(n) + [\frac{\partial A_{ib}}{\partial \alpha_k}x(n) + \frac{\partial B_{ib}}{\partial \alpha_k}u(n)],\ \forall k, \\ \theta_{\theta_{ib}}(n+1) &= \theta_{\theta_{ib}}(n) + [r(n) - y(n)]P_{\theta_{ib}}(n)\psi_{\theta_{ib}}(n), \end{cases} \tag{5.10}$$

where

$$\begin{cases} \frac{\partial A_{ib}}{\partial \alpha_k} &= (I_K + A_{ib})\frac{\partial \Phi}{\partial \alpha_k}(I_K - \Phi)^{-1} \\ \frac{\partial B_{ib}}{\partial \alpha_k} &= \frac{\sqrt{2}}{2}[\frac{\partial A_{ib}}{\partial \alpha_k}L + (I_K + A_{ib})\frac{\partial L}{\partial \alpha_k}], \end{cases} \tag{5.11}$$

which are obtained from (5.6). Simple computations show that

$$\begin{cases} \frac{\partial \Phi}{\partial \alpha_k} &= \begin{cases} e_k e_{k+1}^{\mathcal{T}} - e_{k+1}e_k^{\mathcal{T}} & k = 1, 2, \cdots, K-1 \\ -\alpha_K e_K e_K^{\mathcal{T}} & k = K \end{cases} \\ \frac{\partial L}{\partial \alpha_k} &= \begin{cases} 0 & k = 1, 2, \cdots, K-1 \\ e_K & k = K \end{cases} \end{cases} \tag{5.12}$$

where e_k denotes the kth elementary (column) vector, whose elements are all zero except the kth one which is 1.

Let $R_{nl} \triangleq (A_{nl}, B_{nl}, C_{nl}, d)$ be the equivalent state-space realization of the normalized lattice structure. It is interesting to note from (2.14) - (2.16) that

$$A_{nl}A_{nl}^{\mathcal{T}} + B_{nl}B_{nl}^{\mathcal{T}} = I_K,$$

which means that R_{nl} is actually an input balanced state-space realization.

The normalized lattice structure has been considered as one of the most important structures for adaptive IIR filter implementation. It is shown in [6], [96] that the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

homogeneous system $x_{nl}(n+1) = A_{nl}(n)x_{nl}(n)$ is exponentially stable if

$$|\phi_k(n)| \leq \frac{\pi}{2} - \epsilon, \ \forall k, \ n, \tag{5.13}$$

where ϵ is a small positive constant. Therefore, an adaptive system, parametrized using θ_l defined in (2.13), is BIBO stable if $H(z^{-1})$ and $H_g(z^{-1})$ are BIBO stable. The main problem involved with the normalized lattice parametrization is that the implementation of $H_g(z^{-1})$ is usually very complicated.

5.3 Performance analysis

In this section, we analyze the performance of proposed adaptive filter in terms of parametrization sensitivity, stability, convergence and implementation.

5.3.1 Parametrization sensitivity

As argued before, it is desired in adaptive filter design to use such a parametrization that the transfer function is insensitive to the parameter perturbation due to the FWL effects.

Let $\theta = \{\theta_k\} \in \mathcal{R}^{(2K+1)\times 1}$ be the parameter vector characterizing the filter model $H(z^{-1})$. The transfer function sensitivity measure is defined as

$$M_\theta \triangleq ||\frac{\partial H(z^{-1})}{\partial \theta}||_2^2 = \sum_{\forall k} ||\frac{\partial H(z^{-1})}{\partial \theta_k}||_2^2, \tag{5.14}$$

where the norm $||.||_2$ is defined in (2.19).

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Let $S(z^{-1}) = \Xi + \Upsilon(zI - \Psi)^{-1}\Gamma$. It follows from (2.20) that

$$\|S(z^{-1})\|_2^2 = tr[\Xi\Xi^{\mathcal{T}} + \Upsilon W_c \Upsilon^{\mathcal{T}}] = tr[\Xi^{\mathcal{T}}\Xi + \Gamma^{\mathcal{T}} W_o \Gamma], \tag{5.15}$$

where W_c, W_o are the controllability and observability Gramians of the realization $(\Psi, \Gamma, \Upsilon, \Xi)$, respectively.

When $S(z^{-1}) = S_2(z^{-1})S_1(z^{-1})$, where $S_m(z^{-1}) = D_m + C_m(zI_m - A_m)^{-1}B_m$ with I_m the identity matrix of a proper dimension for $m = 1, 2$. It is easy to verify that

$$\begin{aligned} S(z^{-1}) &= D_2D_1 + \begin{bmatrix} D_2C_1 & C_2 \end{bmatrix} [z \begin{bmatrix} I_1 & \mathbf{0} \\ \mathbf{0} & I_2 \end{bmatrix} - \begin{bmatrix} A_1 & \mathbf{0} \\ B_2C_1 & A_2 \end{bmatrix}]^{-1} \begin{bmatrix} B_1 \\ B_2D_1 \end{bmatrix} \\ &\triangleq \Xi + \Upsilon(zI - \Psi)^{-1}\Gamma. \end{aligned} \tag{5.16}$$

The corresponding sensitivity measure can then be computed with (5.15).

A. The new parametrization θ_{ib}

For the filter $H(z^{-1}) = C_{ib}(zI_K - A_{ib})^{-1}B_{ib} + d$, the sensitivity $\partial H(z^{-1})/\partial\theta_{ib}$ of this transfer function with respect to the parametrization θ_{ib} can be computed with

$$\begin{cases} \frac{\partial H(z^{-1})}{\partial \alpha_k} &= C_{ib}(zI_K - A_{ib})^{-1}[\frac{\partial A_{ib}}{\partial \alpha_k}(zI_K - A_{ib})^{-1}B_{ib} + \frac{\partial B_{ib}}{\partial \alpha_k}], \ \forall k \\ \frac{\partial H(z^{-1})}{\partial C_{ib}^{\mathcal{T}}} &= (zI_K - A_{ib})^{-1}B_{ib} \\ \frac{\partial H(z^{-1})}{\partial d} &= 1. \end{cases} \tag{5.17}$$

It follows from (5.15) that

$$||\frac{\partial H(z^{-1})}{\partial C_{ib}^{\mathcal{T}}}||_2^2 = tr(W_c^{ib}) = K, \quad ||\frac{\partial H(z^{-1})}{\partial d}||_2^2 = 1,$$

where $W_c^{ib} = I_K$ is the controllability Gramian of the input balanced realization.

For $\frac{\partial H(z^{-1})}{\partial \alpha_k}$, one has $A_2 = A_{ib}$, $B_2 = I_K$, $C_2 = C_{ib}$, $D_2 = 0$ and $A_1 = A_{ib}$, $B_1 = B_{ib}$,

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$C_1 = \frac{\partial A_{ib}}{\partial \alpha_k}$, $D_1 = \frac{\partial B_{ib}}{\partial \alpha_k}$. Therefore, $||\frac{\partial H(z^{-1})}{\partial \alpha_k}||_2^2$ can be evaluated easily with (5.15) - (5.16) and hence the total sensitivity measure is

$$M_{\theta_{ib}} = (K+1) + \sum_{k=1}^{K} ||\frac{\partial H(z^{-1})}{\partial \alpha_k}||_2^2. \tag{5.18}$$

It can be shown that $||\frac{\partial H(z^{-1})}{\partial \alpha_k}||_2^2 = ||\frac{\partial F(s)}{\partial \alpha_k}||_2^2$, where the latter is the L_2-norm-based sensitivity measure of analog filter $F(s)$. As shown in [38], analog filter, when implemented in the orthonormal ladder structure/realization, has a very small sensitivity measure. Noting that the parameters α_k for all k except $k = K$ are the structure parameters of the orthonormal ladder filter, one can expect a small $M_{\theta_{ib}}$.

B. Direct-form parametrization θ_d

For the direct-form parametrization θ_d, it follows from (3.1) that

$$\frac{\partial H(z^{-1})}{\partial b_k} = \frac{z^{-k}}{D(z^{-1})} = e_k^{\mathcal{T}}(zI_K - A_c)^{-1}B_c,$$

where A_c, B_c corresponds to the controllable realization R_c defined in (2.8) and e_k is the kth elementary vector. According to (5.15),

$$||\frac{\partial H(z^{-1})}{\partial b_k}||_2^2 = e_k^{\mathcal{T}} W_{cc} e_k, \quad k = 1, 2, \cdots, K,$$

where W_{cc} is the controllability Gramian corresponding to A_c, B_c.

Denote $\bar{a} \triangleq \begin{bmatrix} a_1 & \cdots & a_k & \cdots & a_K \end{bmatrix}^{\mathcal{T}}$. Since $\frac{\partial H(z^{-1})}{\partial b_0} = \frac{1}{D(z^{-1})} = 1 - \bar{a}^{\mathcal{T}}(zI_K - A_c)^{-1}B_c$, one has

$$||\frac{\partial H(z^{-1})}{\partial b_0}||_2^2 = 1 + \bar{a}^{\mathcal{T}} W_{cc} \bar{a}.$$

Noting $\frac{\partial H(z^{-1})}{\partial a_k} = -\frac{z^{-k}}{D(z^{-1})} H(z^{-1}) = -e_k^{\mathcal{T}}(zI - A_c)^{-1}B_c[b_0 + C_c(zI_K - A_c)^{-1}B_c]$, one can compute $||\frac{\partial H(z^{-1})}{\partial a_k}||_2^2$ using (5.15) and (5.16).

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Therefore,

$$M_{\theta_d} = 1 + \bar{a}^{\mathcal{T}} W_{cc} \bar{a} + tr(W_{cc}) + \sum_{k=1}^{K} ||\frac{\partial H(z^{-1})}{\partial a_k}||_2^2 \tag{5.19}$$

can be computed.

It is well known that $tr(W_{cc})$ is very large when $H(z^{-1})$ has poles near the unit circle. Therefore, an adaptive filter using the direct-form based parametrization has a big difficulty in converging to the true filter that has poles around the unit circle.

5.3.2 Guaranteed BIBO stability and convergence

Our proposed adaptive algorithm (5.9) - (5.10) consists of three feedback systems. It is said stable if all the three system is BIBO stable. Looking at the third equation of (5.10), $\theta_{ib}(n)$ is bounded if so are $y(n)$, $\psi_{\theta_{ib}}(n)$ and $P_{\theta_{ib}}(n)$. Since $P_{\theta_{ib}}(n)$ is a positive-definite matrix, associated with the so-called information matrix matrix, $\theta_{ib}(n)$ is bounded if so are $y(n)$ and $\psi_{\theta_{ib}}(n)$. At the bottom line, it will be limited by the dynamic range of the parameters and hence the parameters involved in (5.9) - (5.10) are all bounded. Therefore, the whole adaptive algorithm will be stable if the corresponding homogeneous systems to $y(n)$ and $\psi_{\theta_{ib}}(n)$ are exponentially stable, which is actually true.

For a given transfer function $H(z^{-1})$, one can implement it with the proposed IB realization $R_{ib} \triangleq (A_{ib}, B_{ib}, C_{ib}, d)$ as well as the NL structure. R_{ib} can then be obtained from R_{nl} with a proper orthogonal similarity transformation U. Noting this fact, we have the following theorem.

Theorem 4 *Let $A_{ib}(n)$ be the time varying transition matrix of R_{ib}. The homogeneous system $x(n+1) = A_{ib}(n)x(n)$ is exponentially stable if*

$$0 < \delta \leq \alpha_k(n) < \infty, \ \forall k, n, \tag{5.20}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where δ is a small constant. Further more, the whole adaptive algorithm is stable.

Proof: Note that for any given n, there exists an orthogonal matrix $U(n)$ such that $A_{ib}(n) = U^{\mathcal{T}}(n)A_{nl}(n)U(n)$, where $A_{nl}(n)$ is the transition matrix of the corresponding normalized lattice structure. Since with (5.20), there exist a constant ϵ such that (5.13) holds. Therefore, $x_{nl}(n+1) = A_{nl}(n)x_{nl}(n)$ is exponentially stable, that is $\lim_{n\to\infty} e^{\gamma n}||x_{nl}(n)||^2 = 0$ for some constant $\gamma \geq 0$. Since $x(n) = U(n)^{\mathcal{T}}x_{nl}(n)$, $\lim_{n\to\infty} e^{\gamma n}||x(n)||^2 = 0$ for the same γ. Therefore, the adaptive filter and the gradient systems are all exponentially stable and hence, with the arguments above, the whole adaptive algorithm is stable. ∎

Like in the NL structure, the linear conditions (5.20) are very easy to implement. The parameter δ is determined by the designer or the relative precision of the implementation environment [6]. The spectral norm $|||A|||$ of the transition matrix is usually considered as the signal propagation gain. For companion forms, this gain can be very large, especially when the poles of the filters are close to each other near the unit circle, which leads to the internal signals (states) having a very big dynamic range. While for the IB realizations, the corresponding states are not bigger than one. This is the reason why such systems have a very good signal dynamic range.

It is well known the gradient based numerical algorithm converges to one of the solutions:

$$E[e(n)\psi_{\theta(n)}] = 0. \tag{5.21}$$

The convergence behavior of the OE in θ_d has been well studied in the literature [7], [60]. Since mapping between the proposed parametrization θ_{ib} and the traditional direct-form based θ_d is one-to-one, no extra stationary points can be introduced by using θ_{ib}. In fact, $\psi_{\theta_{ib}}(n) = J(\theta_{ib})\psi_{\theta_d}(n)$, where $J(\theta_{ib})$ is the Jacobian matrix

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

between the two parametrizations and nonsingular for all θ_{ib} under the constraints (5.20). Clearly,

$$E[e(n)\psi_{\theta_{ib}(n)}] = 0 \quad \Leftrightarrow \quad E[e(n)\psi_{\theta_d(n)}] = 0$$

It can be shown that θ_{ib}^* is a convergent point if and only if the corresponding θ_d^* is a convergent point.

5.3.3 Implementation issue

In the proposed adaptive algorithm (5.9) - (5.10), one can see that both are involved in computation of type $A_{ib}v$, where $v \in \mathcal{R}^{K\times 1}$. Due to the fact that A_{ib} obtained with (5.6) is fully parameterized due to the matrix inversion $(I-\Phi)^{-1}$ involved in (5.6) (Here, we drop the subscript K of I for convenience), a direct computation of $A_{ib}v$ requires K^2 multiplications and $K(K-1)$ additions. In what follows, we show that $(I-\Phi)^{-1}$ and hence A_{ib} can be decomposed into a series of simple (sparse) matrices. Using this decomposition, computation for $A_{ib}v$ can be much simplified.

Denote by $U(i,j,x)$ the identity matrix except that its (i,j)th element is x, $\forall(i,j)$, and let $T_1 = U(2,2,\gamma_1)U(2,1,-\alpha_1)$ with $\gamma_1 \triangleq \frac{1}{1+\alpha_1^2}$. Now we note that

$$I-\Phi = T_1^{-1}T_1(I-\Phi) = T_1^{-1}\begin{bmatrix} 1 & -\alpha_1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \beta_2 & 0 & \cdots & 0 & 0 \\ 0 & \alpha_2 & 1 & -\alpha_3 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -\alpha_{K-1} \\ 0 & 0 & 0 & 0 & \cdots & \alpha_{K-1} & 1+\frac{1}{2}\alpha_K^2 \end{bmatrix},$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $\beta_2 \stackrel{\triangle}{=} -\frac{\alpha_2}{1+\alpha_1^2}$. Repeating this procedure, one can show that

$$I - \Phi = T_1^{-1} T_2^{-1} \cdots T_k^{-1} \cdots T_{K-1}^{-1} \Delta, \tag{5.22}$$

where Δ is the identity matrix except $\Delta(k, k+1) = \beta_k$, $\forall k$ with β_k given by the following recursive equations:

$$\begin{cases} \beta_{k+1} = -\frac{\alpha_{k+1}}{s_k}, & s_k = 1 - \alpha_k \beta_k, \quad k = 1, \cdots, K-2 \\ \beta_1 \stackrel{\triangle}{=} -\alpha_1, & s_{K-1} = 1 + \frac{1}{2}\alpha_K^2 + \alpha_{K-1}\beta_{K-1}, \end{cases} \tag{5.23}$$

and

$$T_k = U(k+1, k+1, \gamma_k) U(k+1, k, -\alpha_k), \quad k = 1, 2, \cdots, K-1, \tag{5.24}$$

where $\gamma_k = s_k^{-1}$ with s_k defined in (5.23).

Noting that $\Delta^{-1} = U(1, 2, -\beta_1) \cdots U(k, k+1, -\beta_k) \cdots U(K-1, K, -\beta_{K-1})$, one has

$$\begin{aligned} (I - \Phi)^{-1} &= U(1, 2, -\beta_1) \cdots U(k, k+1, -\beta_k) \cdots U(K-1, K, -\beta_{K-1}) \\ &\quad U(K, K, \gamma_{K-1}) U(K, K-1, -\alpha_{K-1}) \cdots U(2, 2, \gamma_1) U(2, 1, -\alpha_1) \\ &\stackrel{\triangle}{=} \prod_{k=1}^{N} A_{N-k+1}, \end{aligned} \tag{5.25}$$

where $N = 3(K-1)$.

It follows from $A_{ib} = (I + \Phi)(I - \Phi)^{-1} = 2(I - \Phi)^{-1} - I$ that

$$A_{ib} = A_{N+1} A_{N-1} \cdots A_{(2)} A_{(1)} - I \stackrel{\triangle}{=} \prod_{k=1}^{N+1} A_k - I, \tag{5.26}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $A_{N+1} = 2I$. Therefore, $A_{ib}v$ can be computed with

$$\begin{cases} v_k &= A_k v_{k-1}, \quad v_0 = v, \quad k = 1, 2, \cdots, N \\ A_{ib}v &= v_N - v + v_N. \end{cases} \tag{5.27}$$

Noting the sparse structure of each A_k, one can see that computing $A_{ib}v$ with (5.27) requires *only* $3(K-1)$ multiplications and $4K-2$ additions, rather than K^2 and $K(K-1)$, respectively. This is a significant reduction of computational complexity. Similarly, it turns out from (5.11) that $\frac{\partial A_{ib}}{\partial \alpha_k}$ and $\frac{\partial B_{ib}}{\partial \alpha_k}$ can be computed efficiently using the decompositions (5.25) and (5.26).

5.4 An alternative parametrization of the IB realization

We note that the stability region $S_{\theta_{ib}}$ in (5.8) is defined by a set of inequalities. This means that a stability checking system has to be implemented during the adaptation though it is much simpler than the one needed for the algorithms based on θ_d or θ_c.

Define

$$\alpha_k = e^{\beta_k}, \ \forall k \tag{5.28}$$

and hence a new parametrization can be obtained

$$\theta_\beta \triangleq \begin{bmatrix} \beta_1 & \cdots & \beta_K & C_{ib} & d \end{bmatrix}^{\mathcal{T}}. \tag{5.29}$$

Clearly, the stable region $S_{\theta_{ib}}$ in parametrization θ_{ib} is one-to-one mapped into the entire parameter space of this new parametrization θ_β. Therefore, any transfer function parametrized using (finite) θ_β is stable without constraint at all!

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

The corresponding regressor $\psi_{\theta_\beta}(n) = \frac{\partial y(n)}{\partial \theta_\beta}|_{\theta_\beta=\theta_\beta(n)}$ can be obtained steadily. In fact,

$$\begin{cases} \frac{\partial y(n)}{\partial \theta_\beta(k)} &= \frac{\partial y(n)}{\partial \theta_{ib}(k)} e^{\beta_k}, \quad k = 1, \ \cdots, \ K \\ \frac{\partial y(n)}{\partial \theta_\beta(m)} &= \frac{\partial y(n)}{\partial \theta_{ib}(m)}, \qquad m = K+1, K+2, \cdots, 2K+1 \end{cases} \tag{5.30}$$

with $\frac{\partial y(n)}{\partial \theta_{ib}}$ given by (5.9) - (5.12). The corresponding OE scheme-based algorithm can then be obtained easily and no stability checking system is need.

In what follows, the IB realization-based algorithm in discussion uses the parametrization θ_{ib}.

5.5 IIR filtering for acoustic echo cancellation

The acoustic coupling between loudspeaker and microphone is a severe problem for modern audio terminals such as teleconference rooms, hands-free and radio-mobile telephones. See Figure 5.1. Acoustic echo cancellation is based on the identification of the echo path by an adaptive filter. The estimated echo signal is then subtracted from the real echo to give the residual echo. The echo path is characterized by a very long and time-varying impulse response. The actual structure of the echo path is of course unknown. Traditionally, it is modelled by an FIR filter with the advantages of guaranteed stability during adaptation and unimodul MSE surface. However, the FIR filter is computationally rather expensive, and the adaptive filter may require up to thousands of coefficients to match the impulse response. It is well established [139] that the echo path can be modelled as a rational function with poles and zeros. Consequently, there has been a growth of interest in adaptive IIR filters for acoustic echo cancellation, with the hope of a significant reduction of complexity.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

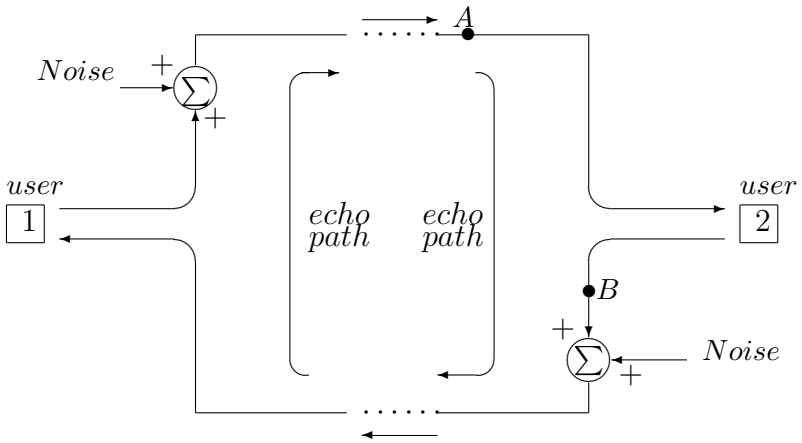


Figure 5.1: A 4-wire long-distance telephone system.

An adaptive echo canceller is an adaptive filter placed in parallel to the echo path. Its task is to approximate the transfer function $H_0(z^{-1})$ of the echo path, whose input and output points are A and B, respectively, as indicated in Figure 5.1. The diagram of the echo canceller (assume it is installed at End 2) is depicted in Figure 5.2, where $H(z^{-1})$ is the adaptive filter which is the estimate of $H_0(z^{-1})$.

By subtracting the estimated returning echo $\hat{y}(n)$ form the real returning echo $r(n)$, the new returning echo $e(n)$ is suppressed greatly. A figure of merit for the effectiveness of an echo canceller is the echo return loss enhancement (ERLE) which is defined as follows [136] - [138]

$$ERLE = 10log\frac{E[r^2(n)]}{E[e^2(n)]}. \tag{5.31}$$

Usually, an ERLE of 20dB or more is expected for effective echo cancellation. The signal $v(n)$ (see Figure 5.2) is generally the sum of the voice signal from End 2 plus a white noise. In the following discussion, it is assumed that End 2 is quiet. In case of double-talking (when user 2 tries to interrupt), it is assumed that there is a mechanism (speech detector) to freeze the adaptation process [139]. Note that although there are some interesting works on solving the double-talking problem to

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

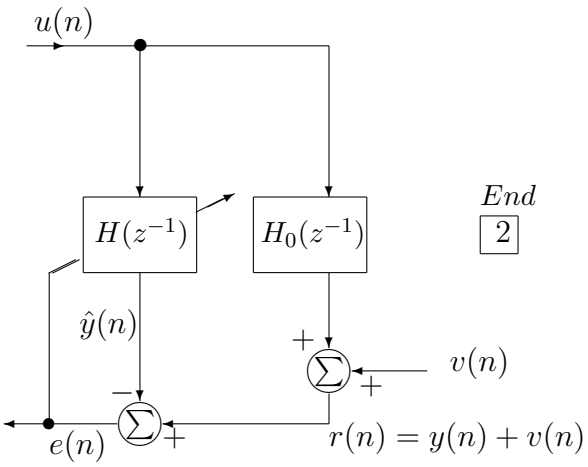


Figure 5.2: Echo cancellation using an echo canceller.

avoid parameter freezing [140], we do not attempt to cope with this problem here since our objective is to investigate the potential of the adaptive IIR canceller using our input balanced realization. If the transfer function of the echo patch is considered as an unknown dynamic plant, then the system described in Figure 5.2 is exactly the same as that in Figure 1.1 (a). As discussed earlier, since the transfer function of the echo path can be more accurately modelled as a rational function [139], it is more natural to use an adaptive IIR filter to approximate it than an adaptive FIR filter.

From above discussion, it is clear that in order to achieve satisfactory cancellation we need to minimize $E[e^2(n)]$. In other words, we want to use the adaptive filter (canceller) to simulate the transfer function of the echo path as closely as possible.

In [134], Fan used his adaptive IIR filtering algorithm [56] to implement an adaptive IIR echo canceller. This algorithm is attractive for its simplicity and the proved convergence. Simulations showed that this IIR canceller performed better than the FIR canceller after convergence. In general, this algorithm converges very slowly, and more seriously, the stability can't be guaranteed during the adaptation.

Here, we want to improve the IIR echo canceller from two aspects:

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

- Using Newton-type algorithm (2.31) to speed up convergence.

- Using IB realization (5.6) which has a guaranteed stability to implement adaptive IIR echo canceller.

The IB realization derived in Section 5.2 can be directly applied to the echo cancellation as depicted in Figure 5.2. Rational function of 2nd-order are used to model the echo path transfer function which is well justified by extensive studies on real echo paths [139]. The disturbance $v(n)$ is assumed to be a white noise sequence. Simulation results which are obtained for sufficient order case will be given in the next section.

5.6 Numerical examples and simulations

In this section, we will present some numerical examples to demonstrate the performance of the IB-based adaptive filter and compare it with some of the algorithms. Moreover, an application of echo cancellation using the IB-based adaptive IIR filter is given.

Example 5.1: In the example, the true system is generated with $[V_b, V_a] = butter(3, 0.1)$. The corresponding poles are $\lambda_{1,2} = 0.8238 \pm j0.2318,\ \lambda_3 = 0.7265$. With an input signal $u(n),\ n = 0, 1, 2, \cdots, 4999$, which is a white noise signal with unit variance, the output $y(n)$ of the true system is generated. We note $|u(n)| < 3.6,\ |y(n)| < 1,\ \forall n$.

The true θ_d is

$$\theta_d^0 = \begin{bmatrix} -2.3741 & 1.9294 & -0.5321 & 0.002898 & 0.008695 & 0.008695 & 0.002898 \end{bmatrix}^{\mathcal{T}}.$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

The corresponding equivalent θ_{ib}^0 is given below

$$\theta_{ib}^0 = \begin{bmatrix} 0.1120 & 0.1940 & 0.7960 & 0.3210 & 0.0350 & 0.5149 \times 10^{-2} & 0.2898 \times 10^{-2} \end{bmatrix}^{\mathcal{T}}.$$

Starting from the following initial estimate of θ_d^0

$$\theta_d(0) = \begin{bmatrix} -1.1870 & 0.4823 & -0.0665 & 0.02898 & 0.09564 & 0.1052 & 0.03857 \end{bmatrix}^{\mathcal{T}}$$

and its equivalent counterpart

$$\theta_{ib}(0) = \begin{bmatrix} 0.2555 & 0.7029 & 1.6007 & 46.1441 & 8.2277 & 2.5614 & 2.8982 \end{bmatrix}^{\mathcal{T}},$$

we run the DFII-based algorithm, denoted as Alg_{DFII} and the proposed IB-based one, denoted as Alg_{IB} using the same numerical algorithm (2.31) with $P_\theta(0) = I_7$. Figure 5.3 shows the evolution of prediction error signal for each algorithm. The corresponding instantaneous transfer function sensitivity measures are given in Figure 5.4.

One observes that Alg_{DFII} has a much higher transfer function sensitivity measure than Alg_{IB}. So, the latter is expected to have a better FWL performance. Now, we implement the two algorithms with $DR = 5$ and $B_{dr} = 16$ bits. Figure 5.5 depicts the evolution of prediction error signal for each algorithm.

It is observed that Alg_{DFII} can not work properly, while Alg_{IB} yields almost the same performance as the one when implemented with infinite precision. It is interesting to note that for this example, the state variables in the DFII structure are all within $[-184.1669, 99.4021]$, while for the IB-based one, the corresponding range is $[-3.9817,\ 3.8198]$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

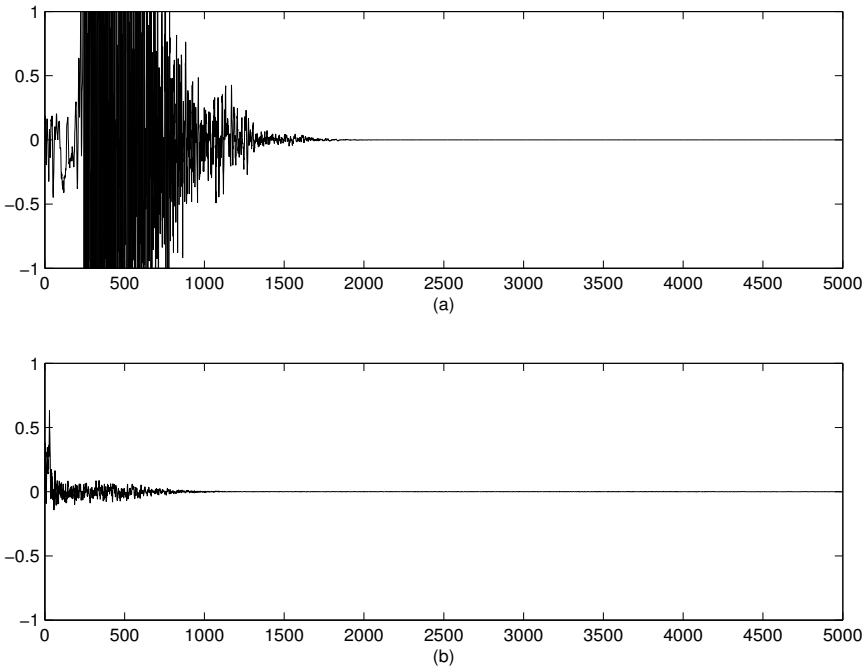


Figure 5.3: Prediction errors for Example 5.1: (a) – Alg_{DFII}, (b) – Alg_{IB}, where the x-axis denotes the iteration number.

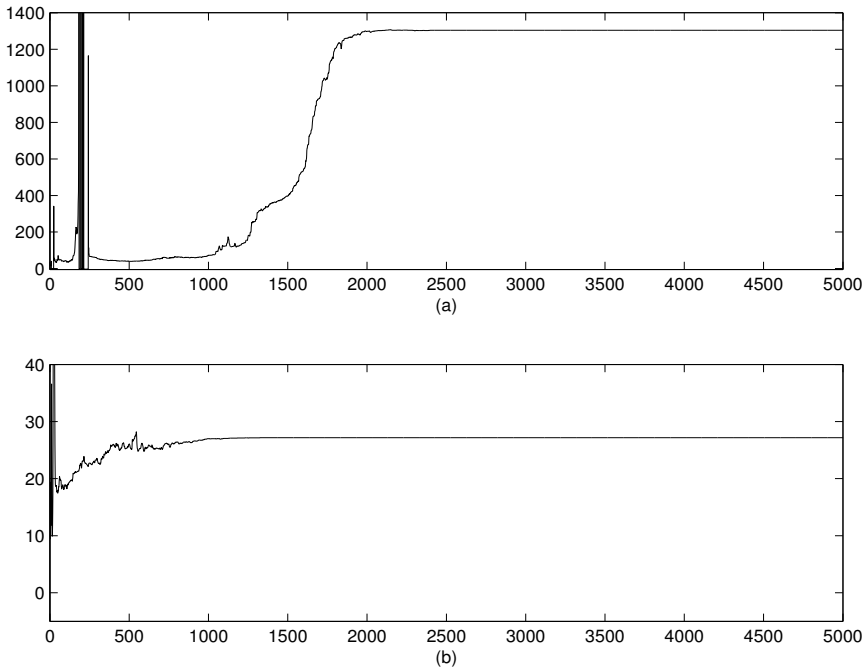


Figure 5.4: Instantaneous transfer function sensitivity measures for Example 5.1: (a) – Alg_{DFII}, (b) – Alg_{IB}, where the x-axis denotes the iteration number.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

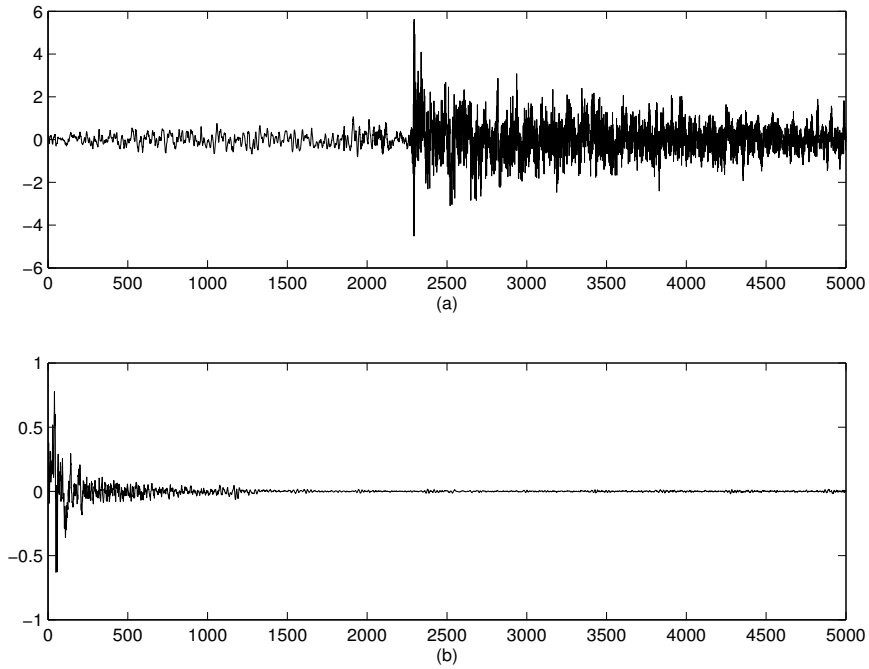


Figure 5.5: Prediction errors for Example 5.1: (a) – Alg_{DFII}, (b) – Alg_{IB}, where $DR = 5, B_{dr} = B_c = 16$ bits and the x-axis denotes the iteration number.

Example 5.2: An echo path transfer function usually exhibits an all-pass characteristic [139]. Hence the echo path in this case is modelled as a 2nd-order all-pass filter with two complex conjugate poles located at $0.95e^{\pm j\pi/6}$ and two zeros at $1.0526e^{\pm j\pi/6}$. The gain is selected as 0.44 so that the loss for all frequencies is about 6dB. This model is the same as the one in [134]. The model transfer function is

$$H(z^{-1}) = \frac{0.4400 - 0.8022z^{-1} + 0.4875z^{-2}}{1 - 1.6454z^{-1} + 0.9025z^{-2}}. \tag{5.32}$$

The canceller in [134] is a second-order adaptive IIR filter with parameters

$$\theta_d = [a_1 \ a_2 \ b_0 \ b_1 \ b_2]^{\mathcal{T}}. \tag{5.33}$$

In practice, the generalization of white noise to speech signals when one proceeds from theory to practice is widely accepted. The input signal here is a zeros mean, unit variance stationary white Gaussian noise sequence. Using the input balanced

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

realization (5.6) in Section 5.2.1 and the corresponding adaptive algorithm, our echo canceller is an adaptive IIR filter paramtrized with

$$\theta_{ib} = [\alpha_1 \ \alpha_2 \ c_1 \ c_2 \ d]^{\mathcal{T}} \tag{5.34}$$

First, the disturbance $v(n) = 0$, and θ_{ib} starts from an initial values which are generated randomly,

$$\theta_{ib}(0) = [-0.58683 \ 0.76144 \ -0.25129 \ -1.149 \ -1.1294]^{\mathcal{T}}$$

and the numerical algorithm (2.31) is used with $P_{\theta_{ib}}(0) = 0.01 I_5$. Figure 5.6 gives the learning curve of ERLE of the echo canceller based on the IB realization. The results presented are obtained by averaging the values of 10 independent runs. Figure 5.7 shows the learning curve under the same initial conditions except

$$\theta_{ib}(0) = [1 \ 2 \ 0 \ 0 \ 0]^{\mathcal{T}}, \tag{5.35}$$

which is equivalent to $\theta_d(0) = [0 \ 0 \ 0 \ 0 \ 0]^{\mathcal{T}}$.

From Figures 5.6 and 5.7, it is easy to see that our proposed echo canceller based on the input balanced realization performs much better than the one in [134] and the one in [135] in terms of convergence speed and ERLE. For example, (see Figure 7 in [134]), the proposed echo canceller in [134] can achieve only 110dB of ERLE while needs 5×10^4 iterations with the normalized LMS algorithm under the same input and initial conditions as in Figure 5.7.

Actually, $v(n)$ is generally nonzero, and can be modeled as a white noise sequence. An experiment similar to that shown in Figure 5.7 is performed with $v(n)$ taken to be a zero mean white Gaussian noise sequence with variance 0.0001, which corresponds

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

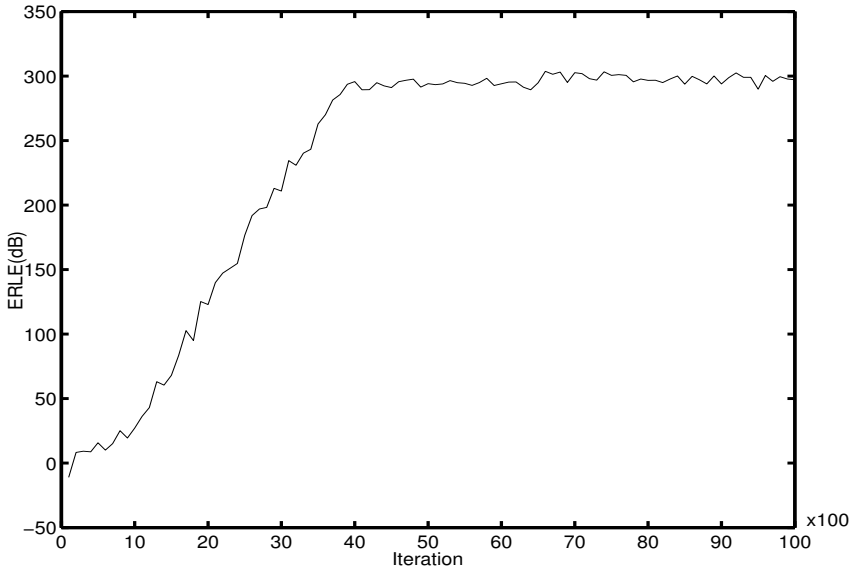


Figure 5.6: ERLE for white noise input, where the initial condition is generated randomly and $v(n) = 0$.

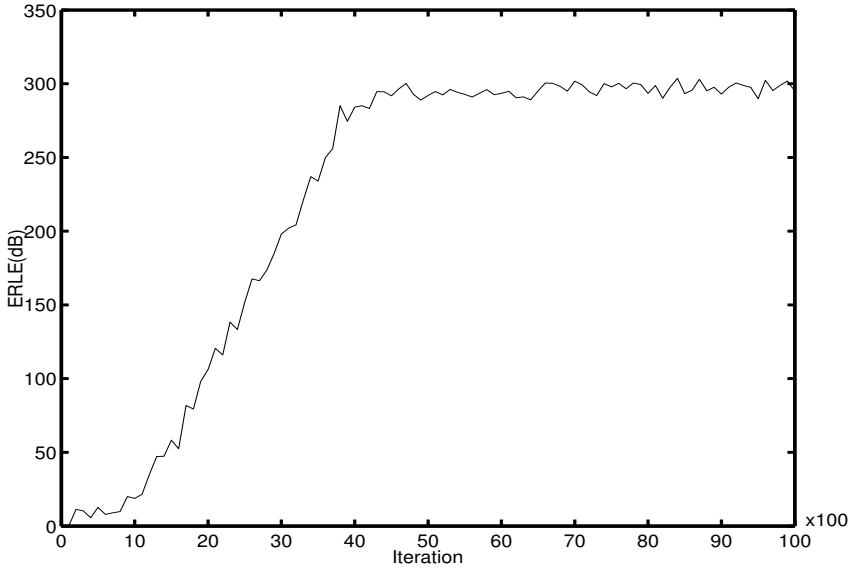


Figure 5.7: ERLE for white noise input, where the initial condition $\theta_{ib}(0)$ is set equal to $\theta_d(0) = [0\ 0\ 0\ 0\ 0]^{\mathcal{T}}$ and $v(n) = 0$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

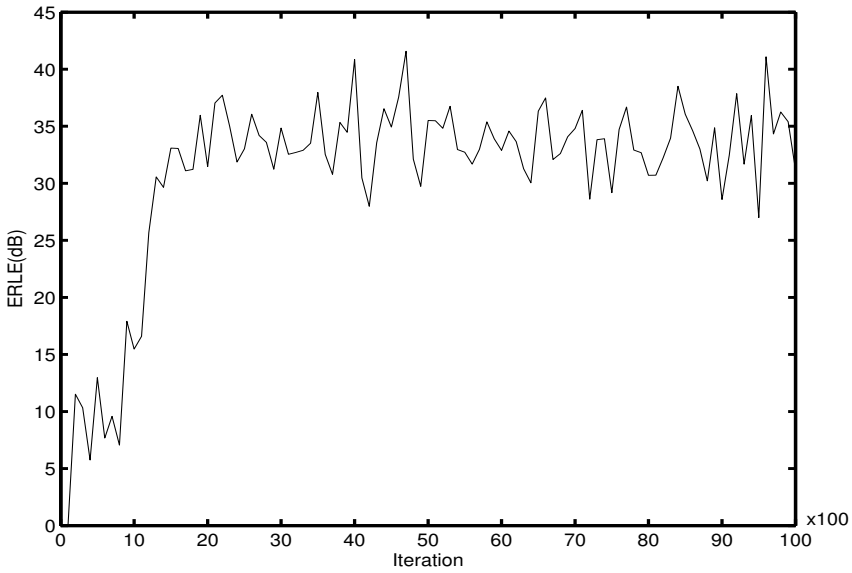


Figure 5.8: ERLE for white noise input with nonzero disturbance, where the $SNR = 34$dB.

to $SNR = 34$dB. The initial value of θ_{ib} is the same as (5.35). Figure 5.8 shows the results of this experiment, where the ERLE is computed by (5.31). The ERLE curve in Figure 5.8 is the result of averaging 10 independent runs. It can be seen that our proposed echo canceller can achieve a higher ERLE than the ones in [134] and [135] within a smaller number of iterations.

To give a better understanding of the proposed input balanced realization-based echo canceller, another experiment which uses real speech signal as the input is provided. The input signals are 10 different speech sequences with the length of $10,000$. Theses sequences are extracted from a Japanese speech which is plotted in Figure 5.9. The sampling frequency is $16k$Hz and the resolution 16 bits at each sample. The results of this experiment are shown in Figure 5.10. Note that for the definition of ERLE as in (5.31), when the signal part in $r(n)$ becomes small, as a speech signal often does, both $r(n)$ and $e(n)$ are dominated by $v(n)$ and the ERLE becomes zero. Thus, the ERLE curve constantly oscillates between zero and its functional value. Simulation indicates that the situation can be so bad that one cannot obtain any information from the ERLE curve thus defined. A modified

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

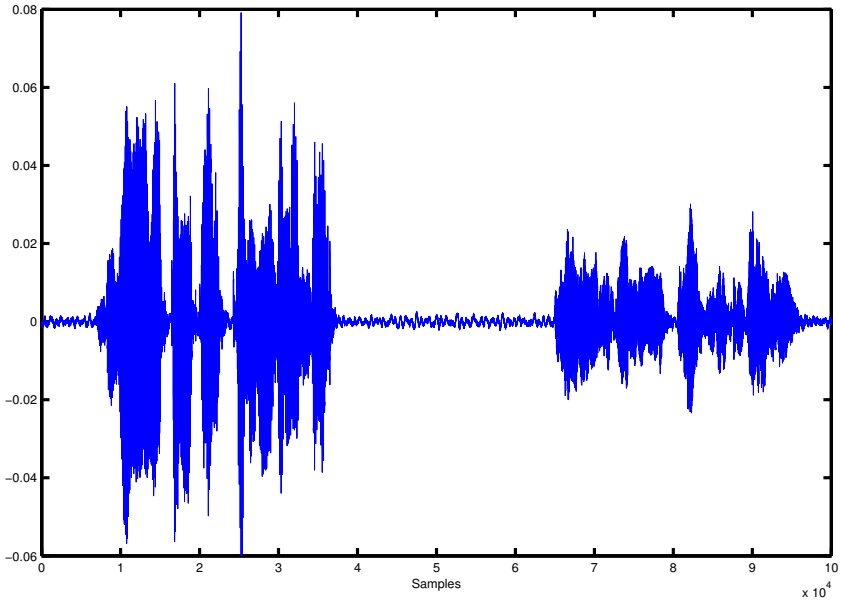


Figure 5.9: Recorded speech signal.

ERLE definition can avoid this problem,

$$ERLE = 10log\frac{E\{[r(n) - v(n)]^2\}}{E\{[e(n) - v(n)]^2\}}. \tag{5.36}$$

For $v(n) = 0$, (5.36) coincides with (5.31). The ERLE curve in Figure 5.10 is the result of using definition (5.36) by averaging 10 independent runs. The added noise is a zero mean white Gaussian noise sequence with variance 10^{-8} and the initial condition is given by (5.35). The same RPE algorithm is used. It can be seen from this figure that the proposed canceller still performs well.

5.7 Conclusions

The input balanced realizations are attractive filter structures. In this chapter, we have derived an input balanced realization for digital IIR filer implementation. This IB-based algorithm inherently leads to adaptive filters that are insensitive to

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

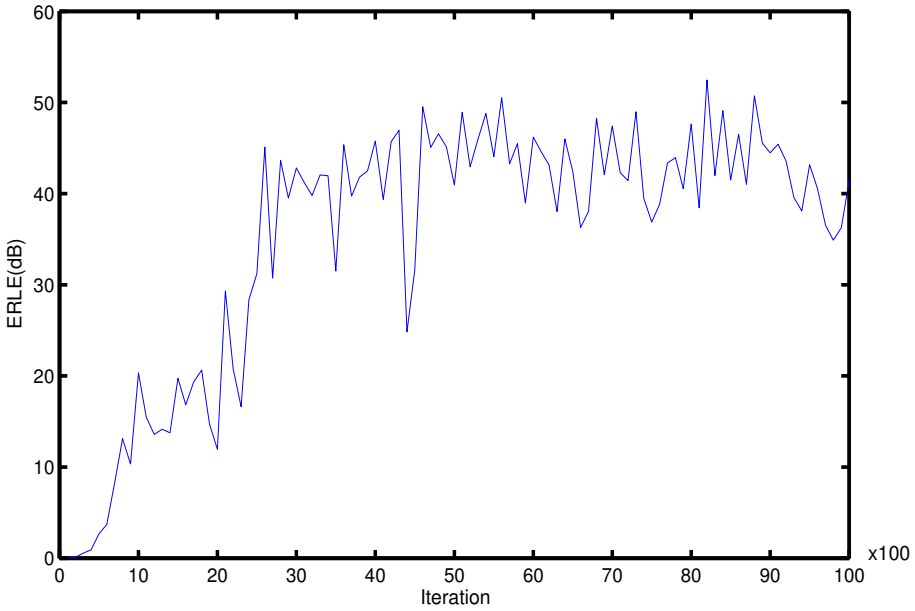


Figure 5.10: ERLE for speech input with nonzero disturbance.

parameter perturbation due to the FWL effects. More importantly, this IB-based adaptive algorithm is proved to be exponentially stable and hence BIBO stable under a set of constrains (5.20). This character is of great interest in adaptive IIR filtering. An application of adaptive IIR echo canceller using our proposed IB-based algorithm is also given in this chapter. Simulations showed that the echo canceller using our proposed algorithm performs much better than some existing ones due to the use of RPE algorithm and the balanced realization. Other simulations and examples have been given to evaluate the performance of the IB-based adaptive filter in the FWL environment.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 6

Direct Frequency Estimation Algorithms

The problem of estimating frequency from a desired signal of sinusoid buried in an additive noise can be found from many practical situations, such as radar, sonar, biomedical engineering, control systems, communication systems, and so on. The adaptive notch filtering technique, discussed in Chapter 4, is one of the popularly used approaches to solve such a problem on-line. Sometimes, the frequency varies with time, it is also useful to apply adaptive notch filters that adapt their notch frequency as a function of the observed time series. The subject of notch filtering has been widely studied [17]. So far, several IIR notch filter structures have been proposed in the literature [17], [48], [105], [121] and [126]. Three typical ones are the IIR lattice notch filter [48], [121], the bilinear 2nd-order IIR notch filter [126], and the 2nd-order IIR notch filter with constrained poles and zeros [17]. In this chapter, the third one will be the subject of interest.

The performance analysis for adaptive IIR filters is very difficult due to the non-linearity in the IIR model. Generally, the insight we have into the characteristics

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

of adaptive IIR notch filters is mainly from simulation experiments. However, it only covers a small part of the statistical properties of adaptive IIR notch filters and is not conclusive. Since the end of 1980s, there have been several trials to make clear analytically the performance of some of the adaptive algorithms for IIR notch filter. The steady-state and tracking properties of the RPE algorithm for the notch filter with constrained poles and zeros in [17] were analytically assessed by Stoica *et al.* [22], [122], [124]. Cho and Lee analyzed the lattice-based algorithm using the ordinary differential equation and the mean square error (MSE) techniques [121], [123]. Nishimura *et al.* proposed a technique to analyze the steady-state performance of the plain gradient (PG) algorithm for a notch filter that is similar to the lattice notch filter in structure [125]. Although a possible way to analyze the gradient-type algorithms was reported in [125], the analysis is based on gradient linearization and is not accurate enough due to the coarse approximations involved. Recently, steady-state analysis of the PG algorithm for the constrained IIR notch filter was conducted by Xiao [66]. The estimation bias and MSE are derived in a closed-form based on more accurate approximations of the error and gradient signals of the notch filter, where the nonlinear terms of the estimation error are also included to improve the accuracy of the analysis. Moreover, the tracking properties of this PG algorithm were analyzed by the same author in [67], where the input signal of the notch filter is a linear chirp signal in noise. Closed-form expressions for its asymptotic tracking error and MSE have been derived and clarified.

It should be pointed out that the 2nd-order IIR notch filter used in the PG-based algorithm in [66] and [67] is parametrized using the polynomial coefficient of its transfer function. The coefficient is a function of the notch frequency for which the notch filter has or nearly has a zero gain. The frequency is then computed from the estimated transfer function coefficient. These techniques, referred as *indirect* frequency estimation methods, require stability monitoring at each iteration.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

The main contribution of this chapter is three-fold. The first one is to derive a PG-based direct frequency estimation algorithm. The constrained 2nd-order IIR notch filter used in this algorithm is purely parametrized using the notch frequency, which makes the frequency estimation algorithm very efficient. Based on the Taylor series expansions, the steady-state analysis and the tracking behavior of the proposed PG-based direct frequency estimation algorithm are analyzed in detail as the second one. The third one is to develop an alternative algorithm for direct frequency estimation, which is based on the linear prediction property of sinusoid singals. Simulations are given to confirm our theoretical analysis and comparison is made between our proposed algorithms and the traditional indirect frequency estimation algorithm.

6.1 A PG-based direct frequency estimation algorithm

Let $u(n)$ be a measurable signal, which can be given by the same model as (4.8) with $N = 1$:

$$u(n) \quad = \quad Mcos(\omega_0 n - \phi) + v(n) \stackrel{\triangle}{=} s(n) + v(n) \tag{6.1}$$

where the sinusoid signal $s(n)$ and the additive zero-mean white Gaussian noise $v(n)$ are assumed to be statistically independent. The variance of $v(n)$ is σ_v^2, and the definition of $M,\ \omega_0,\ \phi$ are the same as (4.8).

To estimate the frequency ω_0 with the *only* available signal $u(n)$, the following 2nd-order constrained IIR notch filter can be used:

$$H(z^{-1}) = \frac{1 + az^{-1} + z^{-2}}{1 + \rho az^{-1} + \rho^2 z^{-2}}, \tag{6.2}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $0 < \rho < 1$ is a constant close to one, controlling the notch bandwidth. Traditionally, (6.2) is implemented in the following direct-form I structure [66]:

$$\hat{v}(n) = -\rho a\hat{v}(n-1) - \rho^2\hat{v}(n-2) + u(n) + au(n-1) + u(n-2), \quad (6.3)$$

where $u(n)$ and $\hat{v}(n)$ are the input and output of $H(z^{-1})$.

It is easy to see that when $a = -2\cos\omega_0 \triangleq a_0$, $\hat{v}(n)$, the output of the filter $H(z^{-1})$, is almost equal to $v(n)$. Therefore, if the estimate of a_0 is available, the frequency ω_0 can then be estimated indirectly using the above relation. The parameter a_0 can be estimated with one of the adaptive algorithms. It is well known that the PG algorithm is popularly used because of its simplicity and its well balanced complexity and performance [65] - [67]. In [66], the following PG algorithm is used to estimate the parameter a_0

$$a(n+1) = a(n) - \mu\hat{v}(n)s_a(n), \quad (6.4)$$

where $s_a(n)$ is the gradient of $\hat{v}(n)$ with respect to a at $a = a(n)$. Actually, $s_a(n)$ is the output of the gradient transfer function, denoted as $H_{ga}(z^{-1})$, when excited with $u(n)$. In [66], $H_{ga}(z^{-1})$ is approximated with:

$$H_{ga}(z^{-1}) \approx z^{-1} - \rho z^{-1}H(z^{-1}) = (1-\rho)z^{-1}\frac{1-\rho z^{-2}}{1+a\rho z^{-1}+\rho^2 z^{-2}} \quad (6.5)$$

and hence $s_a(n)$ can be computed by

$$s_a(n) = u(n-1) - \rho\hat{v}(n-1). \quad (6.6)$$

The closed-form expressions for estimation bias and MSE of the adaptive algorithm defined with (6.3) - (6.6), where a is replaced by $a(n)$, can be found in [66]. It should

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

be pointed out that this algorithm is an indirect frequency estimation algorithm. In many applications, the real frequency at each iteration, denoted as $\hat{\omega}_0(n)$, is needed and hence it has to be calculated from $a(n) = -2\cos\hat{\omega}_0(n)$ at each iteration if (6.4) is used. On the one hand, this conversion is computationally costly. On the other hand, it is possible to have such a situation from algorithm (6.4) that $|a(n)| > 2$, for which there is no frequency $\hat{\omega}_0(n)$ can be found from $a(n) = -2\cos\hat{\omega}_0(n)$ and, *more seriously*, the whole adaptive system may become unstable.

We propose the following direct frequency estimation algorithm. The basic idea is to replace a in (6.2) with $-2\cos\hat{\omega}_0$. In this case, the transfer function $H(z^{-1})$ becomes

$$H(z^{-1}) = \frac{1 - 2\cos\hat{\omega}_0 z^{-1} + z^{-2}}{1 - 2\rho\cos\hat{\omega}_0 z^{-1} + \rho^2 z^{-2}} \triangleq H(z^{-1}, \hat{\omega}_0), \tag{6.7}$$

which has been discussed in Section 4.2.1, see (4.10). With this transfer function $H(z^{-1}, \hat{\omega}_0)$, it is easy to understand that the true frequency ω_0 can be estimated directly with the following new PG algorithm:

$$\hat{\omega}_0(n+1) = \hat{\omega}_0(n) - \mu\hat{v}(n)s_{\hat{\omega}_0}(n), \tag{6.8}$$

where $s_{\hat{\omega}_0}(n)$ is the gradient of $\hat{v}(n)$ with respect to $\hat{\omega}_0$ at $\hat{\omega}_0 = \hat{\omega}_0(n)$. It is the output of the corresponding gradient transfer function, denoted as $H_{g\hat{\omega}_0}(z^{-1})$, when excited by $u(n)$. Using the same approximation idea as the one used in [6] and [66], one can see that the output of (6.7) can be written as:

$$\hat{v}(n) = u(n) - 2\cos\hat{\omega}_0 u(n-1) + u(n-2) + 2\rho\cos\hat{\omega}_0\hat{v}(n-1) - \rho^2\hat{v}(n-2).$$

Then, the gradient of $\hat{v}(n)$ with respect to $\hat{\omega}_0$ at $\hat{\omega}_0 = \hat{\omega}_0(n)$ can be obtained,

$$s_{\hat{\omega}_0}(n) = \frac{\partial\hat{v}(n)}{\partial\hat{\omega}_0(n)} = 2\sin\hat{\omega}_0(n)u(n-1) - 2\rho\sin\hat{\omega}(n)\hat{v}(n-1)$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

So, the relationship between $s_{\hat{\omega}_0}(n)$ and $u(n)$, which is the gradient transfer function, can be given by given by

$$H_{g\hat{\omega}_0}(z^{-1}) \approx 2\sin\hat{\omega}_0[z^{-1} - \rho z^{-1}H(z^{-1},\hat{\omega}_0)] \triangleq H_{g\hat{\omega}_0}(z^{-1},\hat{\omega}_0), \tag{6.9}$$

and hence the gradient signal $s_{\hat{\omega}_0}(n)$, by

$$s_{\hat{\omega}_0}(n) = 2\sin\hat{\omega}_0[u(n-1) - \rho\hat{v}(n-1)]. \tag{6.10}$$

6.2 Steady-state performance analysis

In this section, we will analyze the performance of our proposed PG-based direct frequency estimation algorithm which consists of (6.3) where a is replaced by $-2\cos\hat{\omega}_0(n)$, (6.8) and (6.10) with $\hat{\omega}_0$ replaced by $\hat{\omega}_0(n)$.

6.2.1 Steady-state error and gradient signals

To study the steady-state estimation bias and MSE of the proposed algorithm, let's look at the behavior of the frequency response $H(e^{-j\omega},\hat{\omega}_0)$ of the notch filter given by (6.7) around $\omega = \hat{\omega}_0$. We note that $H(e^{-j\omega},\hat{\omega}_0)$ is a function of $\hat{\omega}_0$ (for a given frequency ω) and $H(e^{-j\omega},\omega) = 0$. Assume that at the steady-state, the parameter $\hat{\omega}_0$ is close to the given frequency ω, which in fact is the frequency of the input sinusoid signal. Using the Taylor series expansion, the transfer function $H(e^{-j\omega},\hat{\omega}_0)$ and the gradient transfer function $H_{g\hat{\omega}_0}(e^{-j\omega},\hat{\omega}_0)$ in the vicinity of $\hat{\omega}_0 = \omega$ can be approximated as follows:

$$H(e^{-j\omega},\hat{\omega}_0) \approx \frac{2\sin\omega\delta(\omega)}{-2\cos\omega(\rho-1)+(\rho^2-1)e^{-j\omega}} + \frac{\cos\omega\delta^2(\omega)}{-2\cos\omega(\rho-1)+(\rho^2-1)e^{-j\omega}}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$$
\begin{aligned}
& -\frac{\rho 4\sin^2\omega\delta^2(\omega)}{[-2\cos\omega(\rho-1)+(\rho^2-1)e^{-j\omega}]^2} \\
= \; & [B_1(\omega)\delta(\omega)+B_2(\omega)\delta^2(\omega)]e^{-j\vartheta(\omega)}+B_3(\omega)\delta^2(\omega)e^{-j2\vartheta(\omega)} \qquad (6.11)
\end{aligned}
$$

and

$$
\begin{aligned}
H_{g\hat{\omega}_0}(e^{-j\omega},\hat{\omega}_0) = \; & 2\sin\omega e^{-j\omega}+2\cos\omega e^{-j\omega}\delta(\omega)-2\rho\delta^2(\omega)e^{-j\omega}\cos\omega B_1(\omega)e^{-j\vartheta(\omega)} \\
& -\sin\omega e^{-j\omega}[2\rho B_1(\omega)e^{-j\vartheta(\omega)}\delta(\omega)+\delta^2(\omega)] \\
& -2\rho\delta^2(\omega)e^{-j\omega}\sin\omega[B_2(\omega)e^{-j\vartheta(\omega)}+B_3(\omega)e^{-j2\vartheta(\omega)}], \qquad (6.12)
\end{aligned}
$$

where

$$
\begin{cases}
B_1(\omega) \triangleq \frac{2\sin\omega}{(1-\rho)\sqrt{(1+\rho)^2-4\rho\cos^2\omega}} \\
B_2(\omega) \triangleq \frac{\cos\omega}{2\sin\omega}B_1(\omega) \\
B_3(\omega) \triangleq -\rho B_1^2(\omega) \\
\vartheta(\omega) \triangleq \begin{cases} tan^{-1}\frac{(1+\rho)\sin\omega}{(1-\rho)\cos\omega}, & \omega \le \frac{\pi}{2} \\ \pi + tan^{-1}\frac{(1+\rho)\sin\omega}{(1-\rho)\cos\omega}, & \omega > \frac{\pi}{2} \end{cases} \\
\delta(\omega) \triangleq \hat{\omega}_0-\omega
\end{cases} \qquad (6.13)
$$

and the expansion terms of $\delta^m(\omega)(m \ge 3)$ are omitted.

Using the input signal $u(n)$ given by (6.1) with the frequency ω_0, the output of the notch filter $\hat{v}(n)$ at its steady-state can then be expressed as:

$$
\begin{aligned}
\hat{v}(n) = \; & M[B_1(\omega_0)\delta(\omega_0)+B_2(\omega_0)\delta^2(\omega_0)]\cos[\varphi(n)-\vartheta(\omega_0)] \\
& +MB_3(\omega_0)\delta^2(\omega_0)\cos[\varphi(n)-2\vartheta(\omega_0)]+v_1(n), \qquad (6.14)
\end{aligned}
$$

where $\varphi(n) \triangleq \omega_0 n-\phi$ and $v_1(n)$ is the zero-mean noise signal at the output of the notch filter excited with $v(n)$.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Similarly, the gradient signal can be approximated as

$$\begin{aligned} s_{\hat{\omega}_0}(n) &= -2\rho M \sin\omega_0 B_1(\omega_0)\cos[\varphi^*(n)-\vartheta(\omega_0)]\delta(\omega_0) + 2Acos\omega_0\cos[\varphi^*(n)]\delta(\omega_0) \\ &\quad +2M\sin\omega_0\cos[\varphi^*(n)] - \rho\cos\omega_0 M B_1(\omega_0)\cos[\varphi^*(n)-\vartheta(\omega_0)]\delta^2(\omega_0) \\ &\quad -2\rho\sin\omega_0 M B_3(\omega_0)\cos[\varphi^*(n)-2\vartheta(\omega_0)]\delta^2(\omega_0) - M\sin\omega_0\cos[\varphi^*(n)]\delta^2(\omega_0) \\ &\quad -2\rho M B_1(\omega_0)\cos\omega_0\cos[\varphi^*(n)-\vartheta(\omega_0)]\delta^2(\omega_0) + v_2(n), \end{aligned} \tag{6.15}$$

where $\varphi^*(n) \triangleq \varphi(n) - \omega_0$ and $v_2(n)$, is the output of the gradient filter $H_{g\hat{\omega}_0}(z^{-1})$ excited with $v(n)$.

6.2.2 Steady-state estimation bias and MSE

Now, look at the equation (6.8), from which we have

$$\delta_\omega(n+1) = \delta_\omega(n) - \mu\hat{v}(n)s_{\hat{\omega}_0}(n), \tag{6.16}$$

where $\delta_\omega(n)$ is actually equal to $\delta(\omega_0)$ with $\hat{\omega}_0$ replaced by the (slowly) time varying frequency parameter $\hat{\omega}_0(n)$ updated using (6.8) and $\hat{v}(n)$, $s_{\hat{\omega}_0}(n)$ are given by (6.14) and (6.15) but with $\delta(\omega_0)$ replaced by $\delta_\omega(n)$, respectively. Taking ensemble average of (6.16) leads to

$$E[\delta_\omega(n+1)] = E[\delta_\omega(n)] - \mu E[\hat{v}(n)s_{\hat{\omega}_0}(n)].$$

It follows from (6.14) and (6.15)

$$E[\hat{v}(n)s_{\hat{\omega}_0}(n)] = \xi_{11}E[\delta_\omega(n)] - \xi_{12}E[\delta_\omega^2(n)] + R_{1,2}, \tag{6.17}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where

$$\begin{cases} \xi_{11} &= M^2 B_1(\omega_0)\sin\omega_0\cos[\omega_0 - \vartheta(\omega_0)] \\ \xi_{12} &= -0.5\xi_{11}cot\omega_0 - M^2 B_1(\omega_0)\cos\omega_0\cos[\omega_0 - \vartheta(\omega_0)] \\ & \quad +\rho M^2 B_1^2(\omega_0)\sin\omega_0\{\cos[\omega_0 - 2\vartheta(\omega_0)] + \cos\omega_0\} \\ R_{1,2} &= E[v_1(n)v_2(n)] \\ &= 2\sigma_v^2[-\frac{2\rho(1-\rho)\cos\omega_0}{1+\rho} - \frac{2}{1+\rho}\frac{(1-\rho)^3\cos\omega_0}{\rho^4 - 2\rho^2\cos 2\omega_0 + 1}]\sin\omega_0, \end{cases} \tag{6.18}$$

with $E[\cdot]$ denoting the ensemble-averaging operator. All the simulated values of bias and MSE given in Section 6.5 are obtained by ensemble averaging a number of independent runs.

It should be pointed out that in the development of (6.17), the following two simplification conditions are used:

- the terms $\delta_\omega^m(n)(m \geq 3)$ are ignored and $\delta_\omega(n)$ and the noise signals $v_1(n)$, $v_2(n)$ are assumed to be statistically uncorrelated,
- the sine and cosine terms, $\sin[\varphi(n) \pm something]$ and $\cos[\varphi(n) \pm something]$, are treated as the zero-mean pseudo-random signals whose variance are 0.5.

Therefore, one has the following equation steadily,

$$E[\delta_\omega(n+1)] = (1 - \mu\xi_{11})E[\delta_\omega(n)] + \mu\xi_{12}E[\delta_\omega^2(n)] - \mu R_{1,2} \tag{6.19}$$

At the steady state (i.e., n tends to infinity), $E[\delta_\omega(n+1)]|_{n\to\infty} = E[\delta_\omega(n)]|_{n\to\infty} = E[\delta_\omega(\infty)]$. Using (6.19), one reaches a closed-form expression for the steady-state estimation error as follows:

$$E[\delta_\omega(\infty)] = \frac{\xi_{12}E[\delta_\omega^2(\infty)] - R_{1,2}}{\xi_{11}}, \tag{6.20}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $E[\delta_\omega^2(\infty)]$ is the steady-state estimation MSE.

To derive the expression of steady-state MSE, it follows from (6.16) that

$$E[\delta_\omega^2(n+1)] = E[\delta_\omega^2(n)] + \mu^2 E[\hat{v}^2(n)s_{\hat{\omega}_0}^2(n)] - 2\mu E[\delta_\omega(n)\hat{v}(n)s_{\hat{\omega}_0}(n)]. \quad (6.21)$$

Inserting (6.14) and (6.15) into (6.21), one can show that

$$\begin{aligned} E[\delta_\omega^2(n+1)] &= (1-2\mu\xi_{11}+\mu^2\xi_{22})E[\delta_\omega^2(n)] \\ &\quad -(2\mu R_{1,2}+\mu^2\xi_{21})E[\delta_\omega(n)]+\mu^2\xi_{23}, \end{aligned} \quad (6.22)$$

where

$$\left\{\begin{aligned} \xi_{21} &= 4\rho\sigma_1^2M^2B_1(\omega_0)\sin^2\omega_0\cos\vartheta(\omega_0) - 4\sigma_1^2M^2\sin\omega_0\cos\omega_0 \\ &\quad -4M^2B_1(\omega_0)\sin\omega_0\cos(\omega_0-\vartheta(\omega_0))R_{1,2} \\ \xi_{22} &= 2M^2\sigma_1^2(\cos^2\omega_0-\sin^2\omega_0+\rho^2B_1(\omega_0)^2\sin^2\omega_0) \\ &\quad +M^4B_1(\omega_0)^2\sin^2\omega_0[1+0.5\cos(2\omega_0-2\vartheta(\omega_0))] \\ &\quad -4\rho M^2\sigma_1^2[B_1(\omega_0)\sin\omega_0\cos\omega_0\cos\vartheta(\omega_0) \\ &\quad +\sin^2\omega_0(B_2(\omega_0)\cos\vartheta(\omega_0)+B_3(\omega_0)\cos2\vartheta(\omega_0))] \\ &\quad +4M^2\sin\omega_0R_{1,2}[B_2(\omega_0)\cos(\omega_0-\vartheta(\omega_0)) \\ &\quad +B_3(\omega_0)\cos(\omega_0-2\vartheta(\omega_0))]+0.5M^2B_1(\omega_0)^2\sigma_2^2 \\ &\quad +2M^2B_1(\omega_0)R_{1,2}[2\cos\omega_0\cos(\omega_0-\vartheta(\omega_0))-\rho B_1(\omega_0)\sin2\omega_0] \\ \xi_{23} &= 2M^2\sigma_1^2\sin^2\omega_0+\sigma_1^2\sigma_2^2+2R_{1,2}^2 \end{aligned}\right. \quad (6.23)$$

with

$$\left\{\begin{aligned} \sigma_1^2 &\triangleq E[\sigma_1^2(n)] = \tfrac{\sigma_v^2}{\rho^2} - \tfrac{1-\rho}{1+\rho}\tfrac{(1+\rho^2)(1+\rho)^2-8\rho^2\cos^2\omega_0}{\rho^2(\rho^4-2\rho^2\cos2\omega_0+1)}\sigma_v^2 \\ \sigma_2^2 &\triangleq E[\sigma_2^2(n)] = 4\sigma_v^2[\tfrac{(1-\rho)^3}{1+\rho}\tfrac{1+\rho^2}{\rho^4-2\rho^2\cos2\omega_0+1} + \tfrac{2\rho(1-\rho)}{1+\rho}]\sin^2\omega_0. \end{aligned}\right. \quad (6.24)$$

It should be pointed out that in the derivation of (6.22), in addition to the two

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

simplification rules mentioned previously, the following relationships are used (see [66]):

$$\begin{cases} E[v_1^2(n)v_2(n)] &= 0, \quad E[v_1(n)v_2^2(n)] = 0 \\ E[v_1^2(n)v_2^2(n)] &= \sigma_{v1}^2\sigma_{v2}^2 + 2R_{1,2}^2. \end{cases} \tag{6.25}$$

At the steady state, we have $E[\delta_\omega^2(n+1)]|_{n\to\infty} = E[\delta_\omega^2(n)]|_{n\to\infty} = E[\delta_\omega^2(\infty)]$. It then follows from (6.20) and (6.22) that the expression of the steady-state MSE is given by

$$E[\delta_\omega^2(\infty)] = \frac{\frac{2R_{1,2}^2}{\xi_{11}} + \mu(\frac{\xi_{21}R_{1,2}}{\xi_{11}} + \xi_{23})}{2\xi_{11} + \frac{2\xi_{12}R_{1,2}}{\xi_{11}} - \mu(\xi_{22} - \frac{\xi_{12}\xi_{21}}{\xi_{11}})}. \tag{6.26}$$

It can be seen from (6.26) that an additional term due to the existence of the estimation bias is involved in the estimation MSE, implying that the MSE will not vanish no matter how small the step size μ will be.

6.3 Tracking properties

In the last section, the input signal is the sinusoid signal buried in white noise, where the frequency of the sinusoid is time-invariant. Now, let the input signal be a linear chirp signal in additive white noise, which is given by:

$$\begin{cases} u(n) &= M\cos\varphi(n) + v(n) \\ \varphi(n) &= \omega_0 n + \frac{1}{2}\alpha n^2 - \phi, \end{cases} \tag{6.27}$$

where ω_0 is an initial frequency at a discrete time instant $n = 0$, and M, ϕ, $v(n)$ are the same as those in (6.1). α is a very small constant called chirp rate, implying that $u(n)$ is a slowly time-varying sinusoid in noise. $v(n)$ is assumed to be statistically independent with the noiseless linear chirp signal. The instantaneous frequency of

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

the linear chirp signal is

$$\omega(n) = \frac{d\varphi(n)}{dn} = \omega_0 + \alpha n \tag{6.28}$$

which changes linearly in time. To track the the instantaneous frequency $\omega(n)$, one can use the constrained notch filter (6.7) and the proposed PG-based algorithm (6.8), where $\hat{\omega}_0(n)$ and $s_{\hat{\omega}_0}(n)$ are replaced by $\hat{\omega}(n)$ and $s_{\hat{\omega}}(n)$, respectively, then becomes

$$\hat{\omega}(n+1) = \hat{\omega}(n) - \mu\hat{v}(n)s_{\hat{\omega}}(n). \tag{6.29}$$

$\hat{\omega}(n)$ is the estimate of the instantaneous frequency $\omega(n)$. The gradient signal $s_{\hat{\omega}}(n)$ is the output of the gradient filter defined by (6.9) with $\hat{\omega}_0$ replaced by $\hat{\omega}(n)$. Hence, $s_{\hat{\omega}}(n)$ can be obtained by (6.10) with $\hat{\omega}_0$ replaced by $\hat{\omega}(n)$. Its steady-state properties with constant input frequency ω_0 have been analyzed in the previous section.

Now, let us consider its tracking properties. Firstly, we define the following tracking error:

$$\varepsilon(n) \triangleq \hat{\omega}(n) - \omega(n). \tag{6.30}$$

6.3.1 Asymptotic error signal and gradient signal

At the asymptotic state of the notch filter, the filter coefficient $\hat{\omega}(n)$ will be close to its true value $\omega(n)$. In what follows, we will analyze the performance of the proposed PG-based direct frequency estimation algorithm (6.29) by studying the asymptotic estimation bias and MSE when the input signal is given by (6.27).

Using the Taylor series expansion, the transfer function of $H(e^{-j\omega(n)}, \hat{\omega}(n))$ of (6.7) in the vicinity of $\hat{\omega}(n) = \omega(n)$ can be approximated as follows:

$$H(e^{-j\omega(n)}, \hat{\omega}(n)) \approx \{B_1[\omega(n)]\varepsilon(n) + B_2[\omega(n)]\varepsilon^2(n)\}e^{-j\vartheta[\omega(n)]}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$$+B_3[\omega(n)]\varepsilon^2(n)e^{-j2\vartheta[\omega(n)]}, \tag{6.31}$$

where $B_1[\omega(n)]$, $B_2[\omega(n)]$, $B_3[\omega(n)]$, $\vartheta[\omega(n)]$ are given by (6.13) with ω replaced by $\omega(n)$.

When the input signal $u(n)$ is given by (6.27), the output of the notch filter $\hat{v}(n)$ and the gradient signal $s_{\hat{\omega}}(n)$ at their asymptotic state are the same as (6.14) and (6.15) with $\delta(\omega_0)$ replaced by $\varepsilon(n)$ and ω_0 replaced by $\omega(n)$, respectively.

For convenience, we rewrite their expressions:

$$\begin{aligned}\hat{v}(n) &= M\{B_1[\omega(n)]\varepsilon(n) + B_2[\omega(n)]\varepsilon^2(n)\}\cos[\varphi(n) - \vartheta] \\ &\quad + MB_3[\omega(n)]\varepsilon^2(n)\cos[\varphi(n) - 2\vartheta] + v_1(n)\end{aligned} \tag{6.32}$$

where $\vartheta = \vartheta[\omega(n)]$.

$$\begin{aligned}s_{\hat{\omega}}(n) &= -2\rho\sin\omega(n)MB_1[\omega(n)]\cos[\varphi^\star(n) - \vartheta]\varepsilon(n) + 2M\cos[\omega(n)]\cos[\varphi^\star(n)]\varepsilon(n) \\ &\quad -\rho\cos\omega(n)MB_1[\omega(n)]\cos[\varphi^\star(n) - \vartheta]\varepsilon^2(n) + 2\sin\omega(n)M\cos[\varphi^\star(n)] + v_2(n) \\ &\quad -M\sin\omega(n)\cos[\varphi^\star(n)]\varepsilon^2(n) - 2\rho MB_1[\omega(n)]\cos\omega(n)\cos[\varphi^\star(n) - \vartheta]\varepsilon^2(n) \\ &\quad -2\rho\sin\omega(n)MB_3[\omega(n)]\cos[\varphi^\star(n) - 2\vartheta]\varepsilon^2(n),\end{aligned} \tag{6.33}$$

where $\varphi^\star(n) \stackrel{\triangle}{=} \varphi(n) - \omega(n)$.

6.3.2 Tracking error and MSE

In this section, first the tracking error and, then, the tracking MSE of the PG-based algorithm (6.29) at their asymptotic states will be derived in closed-form. The development is based on the approximate analytical expressions derived before for the error and gradient signals at their asymptotic states. Noting that they are

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

derived from Taylor series expansion of the notch filter transfer function in the vicinity of the instantaneous frequency of the chirped signal, these expressions hold with reasonable accuracy if $\varepsilon(n)$ is very small at its asymptotic state. In the sequel, this condition is assumed. Furthermore, the chirp rate of the sinusoid and the step size μ in (6.29) are assumed small enough such that the asymptotical stationarility of the signals is held.

The recursion (6.29) can be rewritten as

$$\varepsilon(n+1) = \varepsilon(n) - \mu\hat{v}(n)s_{\hat{\omega}}(n) - \alpha. \tag{6.34}$$

Ensemble averaging (6.34) and using (6.32) and (6.33) with the simplification rules mentioned in Section 6.2.2 yields

$$E[\varepsilon(n+1)] = (1-\mu\zeta_{11})E[\varepsilon(n)] + \mu\zeta_{12}E[\varepsilon^2(n)] - \alpha - \mu R_t \tag{6.35}$$

where ζ_{11}, ζ_{12} and R_t are the same as ξ_{11}, ξ_{12} and $R_{1,2}$ given by (6.18) with ω_0 replaced with $\omega(n)$, respectively.

Noting

$$E[\varepsilon(n+1)]|_{n\to\infty} = E[\varepsilon(n)]|_{n\to\infty} = E[\varepsilon(\infty)], \tag{6.36}$$

one reaches a closed-form expression for the asymptotic tracking error as follows:

$$E[\varepsilon(\infty)] = \frac{\zeta_{12}}{\zeta_{11}}E[\varepsilon^2(\infty)] - \frac{R_t}{\zeta_{11}} - \frac{\alpha}{\mu\zeta_{11}} \tag{6.37}$$

where $E[\varepsilon^2(\infty)]$ is the asymptotic tracking MSE at $n \to \infty$.

Squaring both sides of (6.34) and then taking ensemble averaging, we readily have

$$E[\varepsilon^2(n+1)] \;=\; E[\varepsilon^2(n)] + \mu^2 E[\hat{v}^2(n)s^2_{\hat{\omega}}(n)] - 2\mu E[\varepsilon(n)\hat{v}(n)s_{\hat{\omega}}(n)]$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

$$+\alpha^2 - 2\alpha E[\varepsilon(n)] + 2\mu\alpha E[\hat{v}(n)s_{\hat{\omega}}(n)]. \tag{6.38}$$

With some manipulations and the simplification conditions in Section 6.2.2 and (6.25), the following difference equation can be obtained:

$$\begin{aligned} E[\varepsilon^2(n+1)] &= (1 - 2\mu\zeta_{11} - 2\mu\alpha\zeta_{12} + \mu^2\zeta_{22})E[\varepsilon^2(n)] + \mu^2\zeta_{23} + \alpha^2 + 2\mu\alpha R_t \\ &\quad -(2\mu R_t + \mu^2\zeta_{21} + 2\alpha - 2\mu\alpha\zeta_{11})E[\varepsilon(n)]. \end{aligned} \tag{6.39}$$

It follows from (6.39) with (6.36) and (6.37) that

$$E[\varepsilon^2(\infty)] = \frac{\frac{2\mu R_t^2}{\zeta_{11}} + \mu^2(\frac{\zeta_{21}R_t}{\zeta_{11}} + \zeta_{23}) + 4\alpha\frac{R_t}{\zeta_{11}} - \alpha^2 + \mu\alpha\frac{\zeta_{21}}{\zeta_{11}} + \frac{2\alpha^2}{\mu\zeta_{11}}}{2\mu(\zeta_{11} + \frac{\zeta_{12}R_t}{\zeta_{11}}) - \mu^2(\zeta_{22} - \frac{\zeta_{12}\zeta_{21}}{\zeta_{11}}) + 2\alpha\frac{\zeta_{12}}{\zeta_{11}}}, \tag{6.40}$$

where ζ_{21}, ζ_{22}, ζ_{23} are the same as ξ_{21}, ξ_{22}, ξ_{23} given in (6.23) with ω_0 replaced by $\omega(n)$, respectively. The corresponding σ_1^2 and σ_2^2 that will be used to compute ζ_{21}, ζ_{22} and ζ_{23} are given by (6.24) with ω_0 replaced by $\omega(n)$.

Clearly, the analytical expression for the tracking MSE is complicated and we cannot explain directly how it is related to the pole radius, noise variance, and so on. It is also noted here that the asymptotic tracking MSE is slowly time varying. Furthermore, if the chirp rate is zero, the tracking MSE reduces exactly to its steady-state MSE (6.26).

6.4 An alternative direct frequency estimation algorithm

In the previous sections, a PG-based algorithm has been proposed to solve such a frequency estimation problem. In what follows, a new direct frequency estimation

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

algorithm, which is also very simple, will be derived.

A direct frequency estimation algorithm, which is based on the linear prediction property of the sinusoid signal, was developed by So in [129], in which the auto-regressive (AR) prediction error function $e(n)$ was expressed as:

$$e(n) = u(n) - 2\cos(\hat{\omega}_0)u(n-1) + u(n-2) \tag{6.41}$$

where $u(n)$ is given by (6.1) and $\hat{\omega}_0$ represents an estimate version of ω_0. To minimize the mean square error function $E[e^2(n)]$ with respect to $\hat{\omega}_0$, a stochastic gradient estimate algorithm was used in [129], where the estimated frequency is updated iteratively with the following equation:

$$\begin{aligned} \hat{\omega}_0(n+1) &= \hat{\omega}_0(n) - \mu[e(n)u(n-1)\sin(\hat{\omega}_0(n)) \\ &\quad +2\sigma_v^2 \sin(\hat{\omega}_0(n))\cos(\hat{\omega}_0(n))]. \end{aligned} \tag{6.42}$$

This adaptive algorithm is computationally efficient, but it needs to know the noise variance σ_v^2 to update the estimated frequency at each iteration, which is not practical in many applications. Based on the same AR model of the sinusoid signal as in [129], Sheu *et al.* derived an adaptive algorithm for single frequency estimation with even lower computational cost [130]. But Sheu's algorithm is an indirect frequency estimation algorithm, which means that the frequency, if needed, has to be converted from the updated parameter at each iteration. Such kind of conversion will cost additional computation and more seriously, may cause instability to the bandpass filter used in Sheu's algorithm.

Look at the frequency parametrized 2nd-order notch filter of (6.7), the frequency response of such filter is almost equal to one for all frequencies except for $\omega = \hat{\omega}_0$, at which the response is nil. When $\hat{\omega}_0$ is very close to ω_0, the output of the adaptive

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

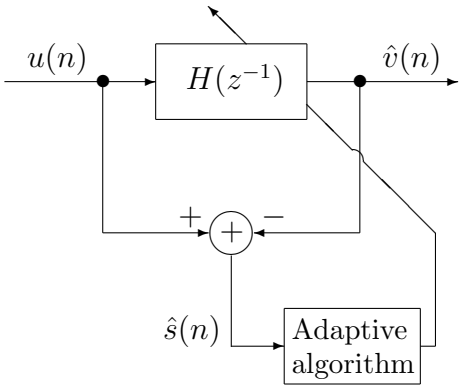


Figure 6.1: Block diagram of the proposed frequency estimation algorithm.

notch filter $\hat{v}(n)$ when excited with the measurable signal $u(n)$ given by (6.1), is almost equal to $v(n)$ and $\hat{s}(n)$, obtained by subtracting $\hat{v}(n)$ from $u(n)$, is then a very good estimate of $s(n)$. See Figure 6.1. Using the linear prediction property of the estimated sinusoid signal $\hat{s}(n) = u(n) - \hat{v}(n)$, a new error signal, which is very similar to the auto-regressive prediction error function (6.41), is defined:

$$\hat{e}(n) = \hat{s}(n) - 2\cos(\hat{\omega}_0)\hat{s}(n-1) + \hat{s}(n-2). \tag{6.43}$$

The frequency can be estimated by minimizing the mean square error function, $E[\hat{e}^2(n)]$, with respect to $\hat{\omega}_0$. Differentiating $\hat{e}^2(n)$ with respect to the instantaneous frequency $\hat{\omega}_0(n)$, one can derive the following algorithm to estimate the true frequency ω_0:

$$\hat{\omega}_0(n+1) = \hat{\omega}_0(n) - \mu\hat{e}(n)\hat{s}(n-1)\sin(\hat{\omega}_0(n)), \tag{6.44}$$

where $\hat{e}(n)$ can be computed by (6.43) with $\hat{\omega}_0$ replace by the instantaneous frequency $\hat{\omega}_0(n)$.

Taking the expected value from equation (6.44), the learning behavior of the fre-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

quency estimate can be evaluated as

$$\begin{aligned} E[\hat{\omega}_0(n+1)] &= E[\hat{\omega}_0(n)] - \mu E\{M^2\cos(\omega_0 n-\phi)\cos[\omega_0(n-1)-\phi]\sin[\hat{\omega}_0(n)] \\ &\quad -2M^2\cos[\hat{\omega}_0(n)]\cos^2[\omega_0(n-1)-\phi]\sin[\hat{\omega}_0(n)] \\ &\quad +M^2\cos[\omega_0(n-2)-\phi]\cos[\omega_0(n-1)-\phi]\sin[\hat{\omega}_0(n)]\} \\ &= E[\hat{\omega}_0(n)] - \mu\frac{M^2}{2}E\{2\cos\omega_0\sin[\hat{\omega}_0(n)] - \sin[2\hat{\omega}_0(n)]\} \\ &\approx E[\hat{\omega}_0(n)] - \mu\frac{M^2}{2}E\{[\hat{\omega}_0(n)-\omega_0][1-\cos\frac{3\hat{\omega}_0(n)+\omega_0}{2}]\} \end{aligned} \quad (6.45)$$

Noting that $\hat{\omega}_0(n)-\omega_0 \approx 0$ and

$$\cos\frac{3\hat{\omega}_0(n)+\omega_0}{2} = \cos[2\omega_0+\frac{3}{2}\delta_\omega(n)] \approx \cos(2\omega_0) - \sin(2\omega_0)\frac{3}{2}\delta_\omega(n),$$

where $\delta_\omega(n)$ is defined in (6.16).

Keep this in mind, we can obtain the following equation from (6.45),

$$\begin{aligned} E[\delta_\omega(n+1)] &\approx E[\delta_\omega(n)] - \frac{\mu M^2}{2}E[\delta_\omega(n)]\{1-\cos(2\omega_0)+\sin(2\omega_0)\frac{3}{2}E[\delta_\omega(n)]\} \\ &\quad \{1-\mu M^2\sin^2\omega_0\}E[\delta_\omega(n)] - \frac{3\mu M^2}{4}\sin(2\omega_0)E[\delta_\omega^2(n)] \end{aligned} \quad (6.46)$$

The convergence trajectory of the frequency estimation can be studied though equation (6.46).

6.5 Simulation results

In this section, we present some numerical examples and the corresponding simulations to show the accuracy of the theoretical steady-state MSE and asymptotic tracking MSE of the proposed PG-based algorithm and also we will evaluate the

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

performance of the alternative algorithm proposed in Section 6.4. The performance of the proposed algorithms will be compared with that of the indirect frequency estimation algorithms in terms of stability.

For convenience, we denote Alg_a as the indirect frequency estimation algorithm in [66], which consists of (6.3) with a replaced by $a(n)$, (6.4) and (6.6), and Alg_ω as our proposed one in Section 6.1.

Example 6.1: In this example, $u(n)$ is generated using (6.1) with, $\omega_0 = 0.2\pi$, $M = \sqrt{2}$ and $\sigma_v^2 = 0.1$, which yields a signal-to-noise ratio (SNR) of 10dB. The number of samples is $300,000$. We run the two algorithms Alg_a and Alg_ω with the same $\hat{\omega}_0(0) = 0.6599$ (equivalently, $a(0) = -1.5801$), $\mu = 5 \times 10^{-6}$ and $\rho = 0.9$.

Figure 6.2 shows the evolution of the theoretical MSE and the simulated MSE. 10 independent runs were ensemble averaged to produce the simulated MSE. In Figure 6.2, the solid line is the theoretical MSE obtained using (6.22) with $E[\delta_\omega^2(0)] = 0.001$ for the proposed adaptive algorithm, the dash-dotted line is the equivalent (frequency) MSE to the theoretical MSE given in [66], while the +-line and the diamond line show the simulated MSE for Alg_ω and the Alg_a, respectively.

We observe that for the proposed algorithm Alg_ω, the theoretical MSE is very close to the simulated one, and by comparing the two theoretical MSE curves, one can see that the proposed algorithm Alg_ω converges faster than Alg_a.

Example 6.2: The conditions and initial values for this example are exactly the same as the Example 6.1 except that the input frequency ω_0 is varying from 0 to $\pi/2$. The dash-dotted line is obtained by (6.26) and the solid line is the equivalent frequency counterpart of the theoretical steady-state MSE given in [66]. The

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

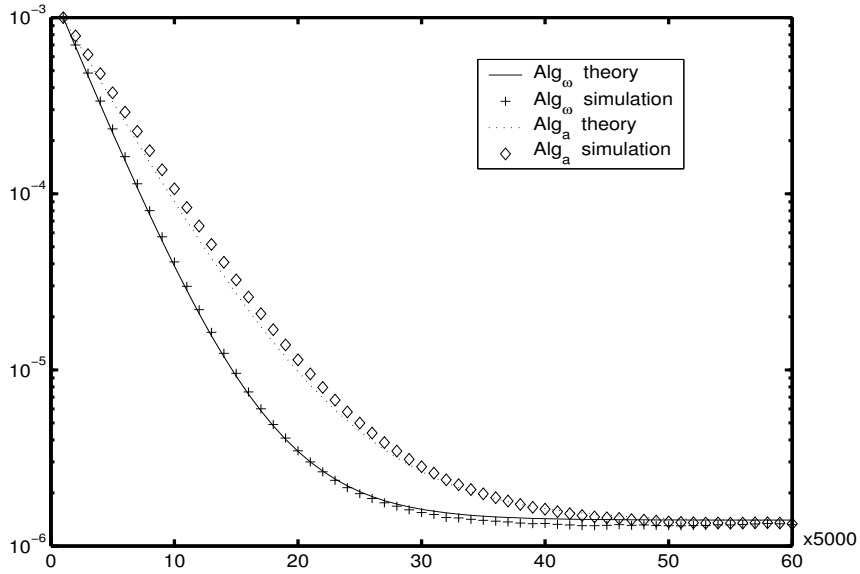


Figure 6.2: Comparison between the theoretical and simulated MSE of Alg_ω and Alg_a.

squared line and the $\times$-line denote the simulated steady-state MSE of Alg_ω and Alg_a, respectively. It can be seen from Figure 6.3 that the theoretical curves of Alg_ω and Alg_a are very similar and the simulated curves indicate a good agreement with the theoretical ones expect the input frequency is very low.

Example 6.3: In this example, we choose $\omega_0 = 0.1$, $\rho = 0.95$, $M = 3$, $\phi = \pi/6$ and $SNR = 10$dB. The number of samples is 5,000. The initial frequency $\hat{\omega}_0(0) = 0.4$, and the corresponding $a(0) = -1.8421$. This example is used to show the instability problem of the indirect frequency estimation algorithm Alg_a. Under the above initial conditions, according to [66], the range for the adaptation step size μ is

$$0 < \mu < 0.00288.$$

Here, we take $\mu = 0.002$.

Figure 6.4 gives the evolution of the estimated frequency $\hat{\omega}_0(n)$ by Alg_ω, which shows

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

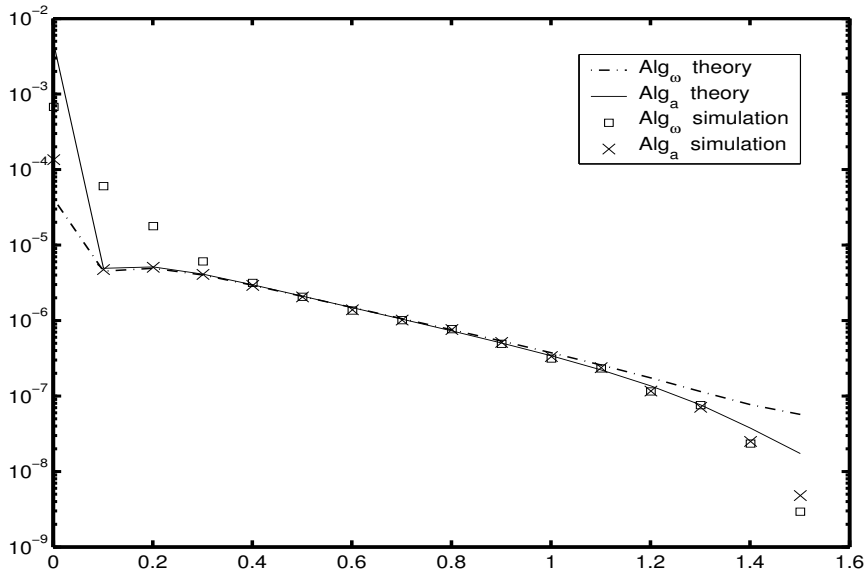


Figure 6.3: Comparison between the theoretical and simulated steady-state MSE of Alg_ω and Alg_a versus the input frequency ω_0, where the x-axis denotes the frequency.

that Alg_ω can converge to the true frequency easily. With the same conditions, $a(n)$ by Alg_a goes to infinity. From this example, one can see that the direct frequency estimation algorithm Alg_ω has a better stability behavior than the indirect one Alg_a. It should be pointed out that in this example, Alg_a can converge to its true parameter $a_0 = -1.9900$ when a much smaller step size (say $\mu = 0.0005$) is used but the absolute value of $a(n)$ is larger than 2 during the adaptation. This means that no frequency can be found from $a(n) = -2\cos\hat{\omega}_0(n)$ at these iteration points. The whole adaptive system has the potential to become unstable under such a situation. While to our proposed algorithm Alg_ω, it can estimate the true frequency quickly and smoothly.

Another experiment is provided to compare the performance of Alg_ω and the existing direct frequency estimation algorithm. The difference between Alg_ω and the existing direct frequency estimation algorithm is the gradient signal. The gradient signal of the existing direct frequency estimation algorithm, which is denoted as $\psi_{df}(n)$, can

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

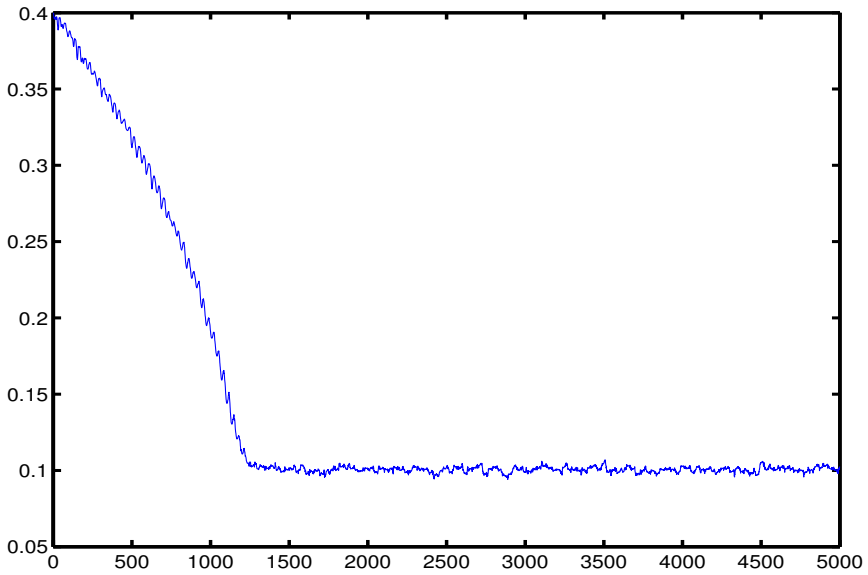


Figure 6.4: Evolution of $\hat{\omega}_0(n)$ for Example 6.3, where the x-axis denotes the iteration number.

be computed by

$$\psi_{df}(n) = \begin{bmatrix} 2\rho\cos\hat{\omega}_0 & -\rho^2 \end{bmatrix} \begin{bmatrix} \psi_{df}(n-1) \\ \psi_{df}(n-2) \end{bmatrix} + 2\sin\hat{\omega}_0 u(n-1) - 2\rho\sin\hat{\omega}_0\hat{v}(n-1)$$

without any approximation, where $u(n)$ and $\hat{v}(n)$ are given by (6.1) and (6.3), respectively. The same LMS like algorithm

$$\hat{\omega}_0(n+1) = \hat{\omega}_0(n) - \mu\hat{v}(n)\psi_{df}(n) \tag{6.47}$$

is used. The input signal is still generated by (6.1) with $\omega_0 = 0.2\pi$, $M = 2$, $\phi = \pi/6$ and $SNR = 10$dB. We run Alg$_\omega$ and the algorithm of (6.47) with $\rho = 0.95$, $\hat{\omega}_0(0) = 0.4$ and $\mu = 0.0002$. The learning curve of the estimated frequency for both algorithms are given in Figure 6.5, where the solid line is for Alg$_\omega$ and the dotted line is for (6.47). The proposed Alg$_\omega$ can achieve a better accuracy under the same initial conditions. Moreover, by comparing the computational complexity of the two algorithms, one can see that the computation of $s_{\hat{\omega}_0}(n)$ only needs two

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

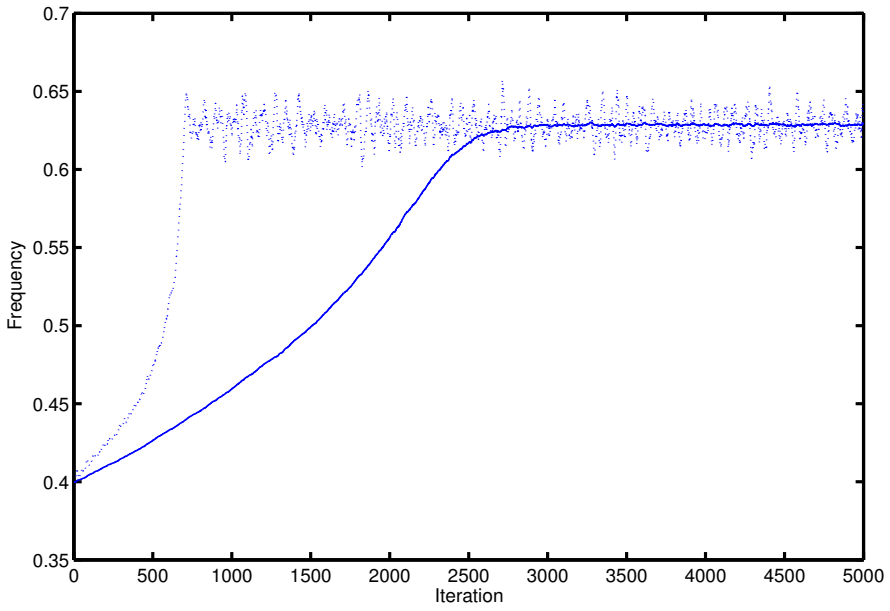


Figure 6.5: Comparison between Alg_ω and (6.47), where the x-axis denotes the iteration number.

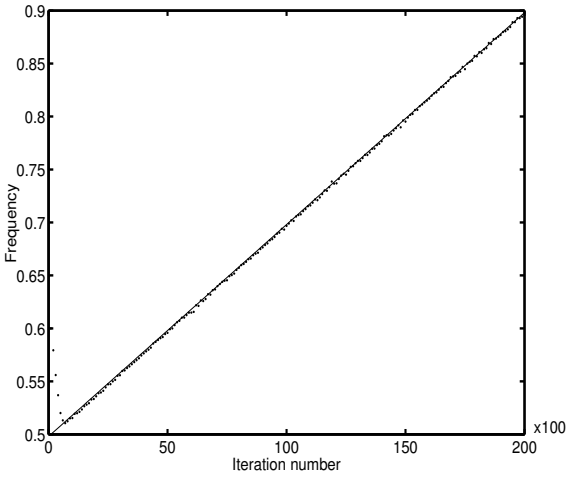
multiplication and one add, which is much simpler than that of $\psi_{df}(n)$.

Example 6.4: In this example, some typical comparisons of tracking MSE between the theoretical analysis in Section 6.3 and the simulated results for both positive and negative chirp rates will be provided.

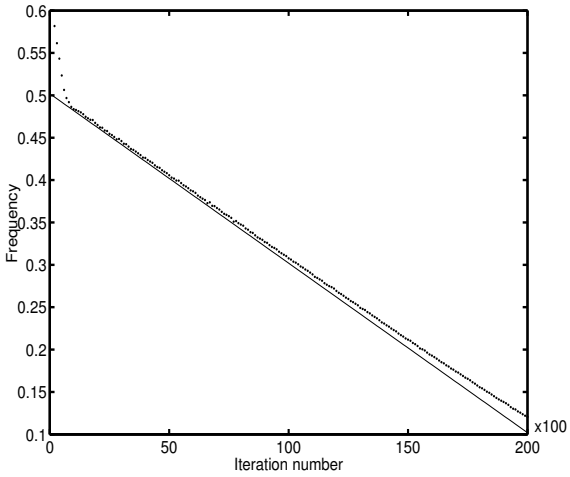
$u(n)$ is generated using (6.27) with $\omega_0 = 0.5$, $M = \sqrt{2}$ and $\sigma_v^2 = 0.1$, which yields a signal-to-noise ratio (SNR) of 10dB. The number of samples is $20,000$. We run the algorithm (6.29) with $\hat{\omega}(0) = 0.6$ (for both positive and negative chirp rates), $\mu = 8 \times 10^{-4}$ and $\rho = 0.95$.

Figure 6.6 shows the estimated $\hat{\omega}(n)$ and the true frequency $\omega(n)$ versus iteration number by a single run. The solid line is for $\omega(n)$ and the pointed line is for $\hat{\omega}(n)$. It can be seen that our proposed algorithm (6.29) can track the time-varying frequency well.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library



(a) $\hat{\omega}(n)$ (pointed line) and $\omega(n)$ (solid line) versus iteration number where chirp rate $\alpha = 2 \times 10^{-5}$.



(b) $\hat{\omega}(n)$ (pointed line) and $\omega(n)$ (solid line) versus iteration number where chirp rate $\alpha = -2 \times 10^{-5}$.

Figure 6.6: Estimation of $\omega(n)$ with both positive and negative chirp rates.

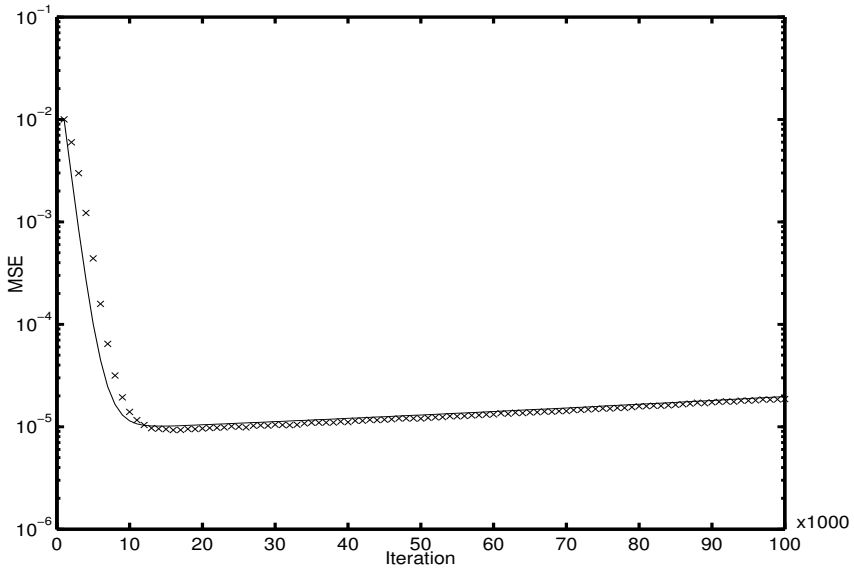


Figure 6.7: Comparison between the theoretical and simulated tracking MSEs versus the iteration number, where the solid line is for the theoretical one obtained from (6.39) and the $\times$-line is for the simulated one.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Figure 6.7 presents an example, where the theoretical tracking MSE is compared with the simulated one which is obtained by averaging the results of 20 independent runs. The input $u(n)$ is the same as the one above except $\omega_0 = 0.6$ and $\alpha = -1.5 \times 10^{-6}$. We run (6.29) with $\hat{\omega}(0) = 0.7$, which equals to $E[\varepsilon^2(0)] = 0.01$ in (6.39), $\mu = 6.5 \times 10^{-5}$ and $\rho = 0.93$. The theoretical curve is then obtained from (6.39) with the same initial conditions. It is seen that the theoretical curve indicates a good agreement with the simulated one. Furthermore, the tracking MSE is slowly time-varying, even at its asymptotic state, which coincides with the results in [67] (see Figure 2 in [67]).

We now present an example to demonstrate the performance of our proposed alternative algorithm (6.44) in Section 6.4. For convenience, the alternative algorithm derived in this chapter, the algorithm based on (6.42) proposed in [129] and the Sheu's algorithm in [130] are denoted as Alg_s, Alg_{ar} and Alg_{bpf}, respectively.

Example 6.5: The signal $u(n)$ is generated with

$$u(n) = 2\sin(\omega_0 n - \phi) + \sigma_0 v_0(n), \tag{6.48}$$

where ϕ is generated randomly and $v_0(n)$ is a white noise with zero-mean and unit variance, while σ_0 is chosen to control the signal-to-noise ratio (SNR). We choose $\mu = 0.001$ and $\rho = 0.95$. In what follows, we consider two situations: $\omega_0 = 0.2$ and $\omega_0 = 1.6$.

Case I: $\omega_0 = 0.2$ and $SNR = 10$dB.

A set of initial values of $\hat{\omega}_0(n)$ for Alg_s and Alg_{ar} is randomly generated and their equivalent ones are then computed with $\alpha(0) = 2\cos[\hat{\omega}_0(0)]$ for Alg_{bpf}. It is observed that Alg_{ar} shows a very large ripple after convergence, while Alg_s can converge

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

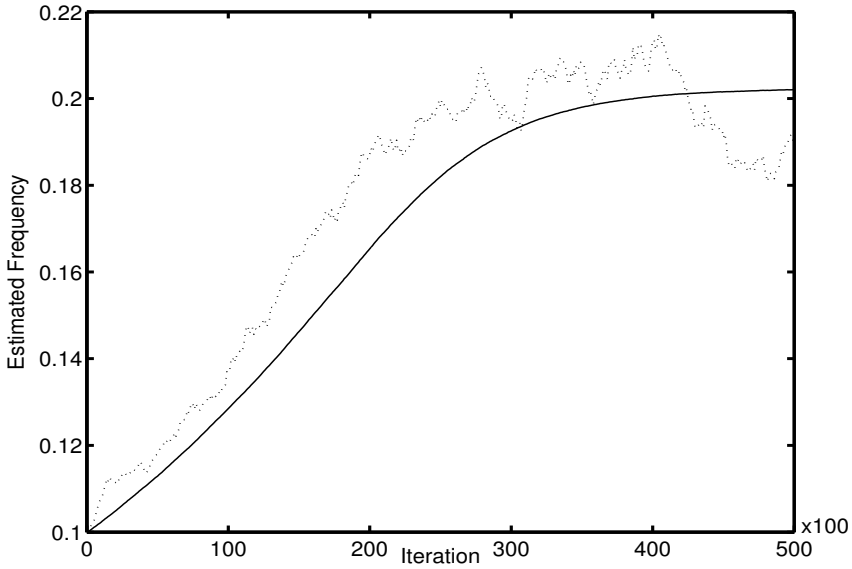


Figure 6.8: Trajectories of $\hat{\omega}_0(n)$ by a single run of Alg_{ar} and Alg_s when $\omega_0 = 0.2$, where the dotted line is the trajectory obtained by Alg_{ar} and the solid line is the one obtained by Alg_s.

smoothly. The trajectories of $\hat{\omega}_0(n)$ by a single run of Alg_{ar} and Alg_s with the same initial condition can be seen in Figure 6.8. It is found that Alg_{bpf} can yield almost the same performance as that by Alg_s for some initial conditions but it can become unstable during the adaptation for the absolute initial values of $\alpha(0)$ very near to 2, which is an undesired property in real-time applications. While to Alg_s, as expected, no instability problem is found in all the trials.

Case II: $\omega_0 = 1.6$ and $SNR = 3$dB.

As pointed out in [129], Alg_{ar} yields the most accurate estimation when $\omega_0 = 1.6$. 100 independent sequences of $u(n)$ are generated. With the same initial conditions, we run the three algorithms for each sequence. The estimation MSE of frequency is obtained by averaging the 100 runs for each of the three algorithms, where the equivalent frequency for Alg_{bpf} is obtained with $\hat{\omega}_0(n) = \cos^{-1}[0.5\alpha(n)]$. Figure 6.9 shows the trajectories of $\hat{\omega}_0(n)$ obtained by a single run of the three algorithms and Figure 6.10 depicts the evolution of the MSE for each algorithm. It is observed that

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

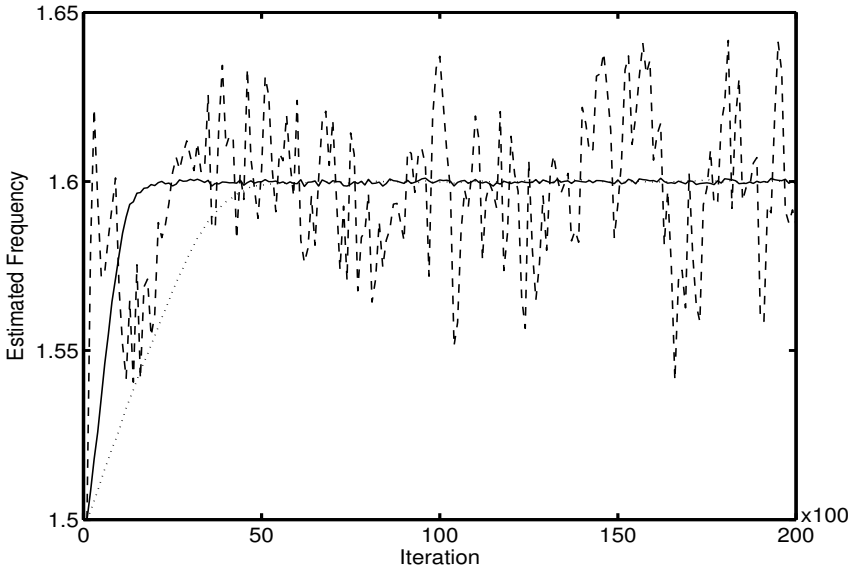


Figure 6.9: Trajectories of $\hat{\omega}_0(n)$ by a single run of Alg_{ar}, Alg_s and Alg_{bpf} when $\omega_0 = 1.6$ and $SNR = 3\text{dB}$, where the dashed line is the trajectory obtained by Alg_{ar}, the solid line is the one obtained by Alg_s and the dotted line is the one obtained by Alg_{bpf}.

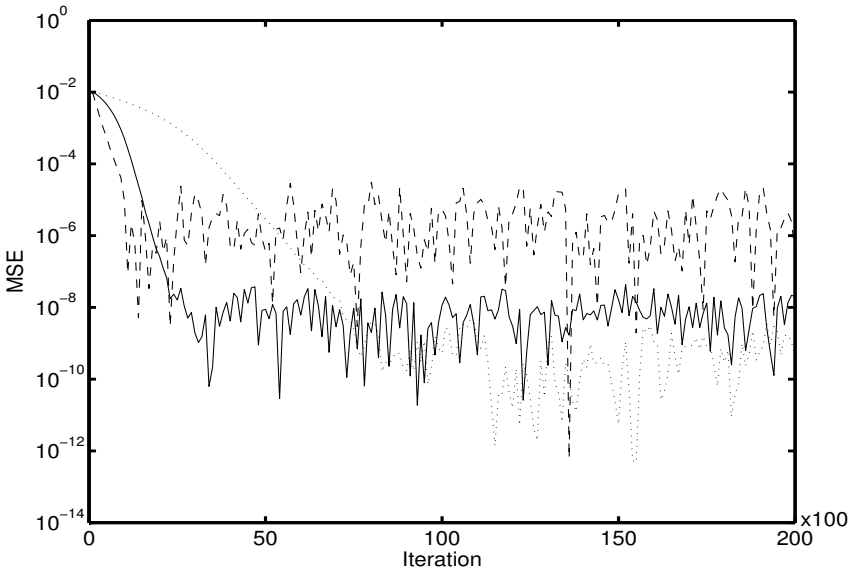


Figure 6.10: The estimation MSE of the three algorithms when $\omega_0 = 1.6$, where the dashed line is the trajectory obtained by Alg_{ar}, the solid line is the one obtained by Alg_s and the dotted line is the one obtained by AL_{bpf}.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

our proposed Alg_s has a faster convergence speed than Alg_{bpf} and its estimation error is much smaller than Alg_{ar}.

6.6 Conclusions

We have studied the problem of frequency estimation using a 2nd-order constrained notch filter. Noting some demerits of the traditional indirect frequency estimation algorithm, we have proposed a PG-based direct frequency estimation algorithm. The advantages of this algorithm include avoiding the conversion from $a(n)$ to $\hat{\omega}_0(n)$, which exists in the traditional algorithms, and having nice stability property. The performance of this PG-based algorithm has been analyzed by deriving the expressions for the steady-state estimation bias and MSE as well as the asymptotic tracking MSE. The theoretical MSE has been confirmed with simulations and the stability comparison has also been illustrated by a numerical example.

Noting some weak points in the frequency estimation algorithms proposed in [129] and [130], we have derived an alternative algorithm for direct frequency estimation. The proposed alternative algorithm has a low computational complexity which makes it suitable for real-time applications. Its problem is the slow convergence. However, in some applications, such as electrocardiogram sinusoidal interference cancellation [131] and sinusoidal engine noise reduction, a coarse estimate for the frequency is available in advance, and the notch filter just needs to adjust its coefficient to reduce the small mismatch between the estimated frequency and the true frequency. In such applications, the alternative algorithm is a good option in terms of both computational cost and performance.

It should be pointed out that the constrained notch filters used in the two algorithms proposed in this chapter can be implemented in many different structures, includ-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

ing the normal state-space realization which has been investigated in Chapter 4. When the adaptive notch filter is implemented in a normal state-space realization, according to the Theorem 2 and Theorem 3, the BIBO stability in the time-varying environment can be ensured if $0 < \rho \leq \gamma < 1$ with a pre-determined constant γ given by the designer, which is very easy to implement in real-time applications.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Chapter 7

Conclusions and Directions for Future Work

We have presented the research work and results achieved during the last three years in the previous chapters. In this chapter, we will give some concluding remarks and possible directions to which future research may go.

7.1 Conclusions

This thesis has addressed the parametrization and structure issue for adaptive IIR filter design and examined how a well-designed adaptive algorithm can be degraded when implemented with finite precision. The important elements of the adaptive algorithms and some possible applications of the proposed adaptive filtering algorithms have also been discussed. There are two fundamental issues associated with adaptive IIR filters: convergence and stability. As we have seen, it is very difficult to predict the behavior of an adaptive IIR algorithm in a general way.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

After the preliminary description of adaptive IIR filters, we emphasize our work on the FWL effects in adaptive digital filter implementation. In Chapter 3, an efficient adaptive state-space filter has been derived. This structure is computationally efficient and it is shown that such an adaptive filter has much better performance than the classical adaptive state-space filters in terms of stability and FWL behavior.

In Chapter 4, a normal state-space realization has been proposed for adaptive IIR filter implementation. It is shown that when an adaptive filter is implemented in a normal state-space realization, the corresponding adaptive algorithm is BIBO stable if the instantaneous poles of the filter are all inside the circle $|z| < \gamma$ with γ a constant satisfying $0 < \gamma < 1$. As one of the applications of this new result, a novel adaptive constrained IIR notch filter has been derived based on the work in [19], where the filter is parametrized using the frequency vector and implemented in a cascade form with each subfilter realized using a normal state-space realization. The convergence properties are studied and it has been shown that the proposed algorithm converges to the desired notch filter whose notch frequencies are those underlying the input signal. Comparison between the proposed normal realization and the lattice notch filter shows that normal realization has less computational cost than the lattice one. Application of this new algorithm to speech signal processing has been also discussed.

Classically, the adaptive IIR algorithms are derived from the direct-form structures. However, some disadvantages of these structures such as finite-precision effects and the complexity of stability monitoring have led to the development of adaptive algorithms using alternative structures. It is well known that for any (linear) digital filter there exists a class of state-space realizations, called input balanced realizations. These realizations have many nice numerical properties, among which is the inherent dynamic scaling that makes the state variables be constrained within a cer-

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

tain range. In Chapter 5, we have derived a new adaptive IIR filter which is based on an input balanced state-space realization. Besides other nice properties, such as small parametrization sensitivity, it has been proved that the proposed algorithm using IB realization is BIBO stable under a set of constrains. Simulations have shown that it over-performs the DFII-based adaptive IIR filters when implemented in finite precision. An application of the proposed IB-based algorithm in echo cancellation has also been given in this chapter, where the IB-based algorithm showed a better ERLE result than some of the existing IIR echo cancellers.

In Chapter 6, we have investigated the problem of direct frequency estimation. Two new adaptive algorithms, the PG-based algorithm and the alternative algorithm, have been proposed for direct frequency estimation, where the constrained notch filter is parametrized using its notch frequency. Due to the IIR nature of the IIR notch filters, their performance analysis is generally very difficult. The steady-state performance of the PG-based algorithm has been analyzed in detail as well as its asymptotic tracking behavior. The most attractive advantage of the alternative algorithm is that it requires very low computational cost comes with a satisfactory performance. Simulations have been given to confirm our theoretical analysis and comparison is made between our propose algorithms and the traditional indirect frequency estimation algorithm.

7.2 Directions of future work

A. Analyzing the sign algorithms

Assessment of the performance of the frequency estimation algorithms has been an important topic of research. So far, the steady-state and tracking performance of the RPE algorithm for the 2nd-order constrained adaptive notch filter were analyzed

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

using the ODE or the MSE techniques [22], [124]. An effective technique to analyze the performance of the PG algorithm for a 2nd-order IIR notch filter which is similar to the lattice notch filter was proposed in [125]. The technique used in [125] was extended to gradient non-linearization for the analysis of the steady-state and the tracking performance of the PG algorithm for the constrained notch filter [66], [67]. However, the analysis procedures taken there can not be applied to the *sign* algorithm. This is because of the presence of a sign function in the *sign* algorithm, which makes the derivation intractable. To solve this problem, Xiao proposed to use the probability distribution of the notch filter output in [120]. That is, by assuming the Gaussianity of the filer output, this analysis of the *sign* algorithm can be performed. In [120], a detailed analysis of the steady-state performance of a sign algorithm (SA) has been given by Xiao.

It should be pointed out that PG algorithms in [66], [67] and the SA algorithm in [120] are all indirect frequency estimation algorithms. The drawbacks of such kind of algorithms have been discussed in Chapter 6. One of the future work is to modify our proposed PG-based algorithm in Section 6.1 into a corresponding *sign* algorithm which is very simple and hence has great merit in hardware implementation. Moreover, its steady-state analysis can be studied using the same approach in [120].

To further reduce the computational burden of (6.44) in Section 6.4, we can adopt the following *sign* algorithm, noting $\sin[\hat{\omega}_0(n)] > 0$, $\hat{\omega}_0(n) \in (0, \pi)$:

$$\hat{\omega}_0(n+1) = \hat{\omega}_0(n) - \mu sign[\hat{s}(n-1)]\hat{e}(n), \tag{7.1}$$

where *sign* $[c]$ is equal to 1 and -1, depending on $c > 0$ and $c < 0$, and nil if $c = 0$. In this *sign* algorithm, the notch filter $H(z^{-1})$ is updated with $\hat{\omega}_0$ replaced by the instantaneous frequency $\hat{\omega}_0(n)$ in (7.1). Motivated by Xiao's new approach

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

to the SA in [120], the steady-state performance of (7.1) can be investigated and the closed-form of estimation bias and MSE can be derived.

B. Adaptive IIR filtering with cascade structure

Stability monitoring is computational expensive and is often required in the adaptive IIR filters using direct-form-based structures. To overcome this problem, various researchers have recommended alternative IIR structures like the parallel form, cascade form, lattice form [45], [46], [60], [141]. Motivated by the notch filter structure in Chapter 4, the cascade form provides an attractive realization due to the easily controlled stability of $H(z^{-1})$ and $H_g(z^{-1})$ and the reduced computational complexity of calculating the gradient signal.

One direction of the future work is to study the use of cascade form filter structures. In Chapter 4, the structure for constrained IIR notch filter is in a cascade form which can reduce the complexity of computation and moreover, the stability of the notch filter can be guaranteed. From the signal processing point of view, a decomposed filter structure consisting of cascaded 2nd-order sections is superior for coefficient sensitivities [142].

For an IIR filter $H(z^{-1})$ of order $K = 2p$, one can implement it with the following cascade form

$$H(z^{-1}) = \frac{N(z^{-1})}{D(z^{-1})} \triangleq b_0 \prod_{k=1}^{p} \frac{1 + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \tag{7.2}$$

with $N(z^{-1})$, $D(z^{-1})$ defined in (2.1). The typical cascade form filter is implemented in 2nd-order sections and each section has two zeros and two poles, as shown in Figure 7.1, together with an overall gain parameter b_0. This structure was considered for adaptation in [46]. We refer to this cascade form as the *standard cascade* (SC).

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

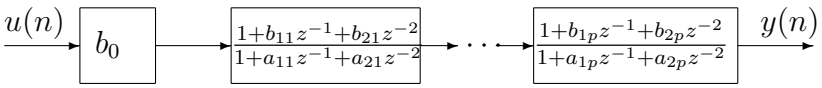


Figure 7.1: Block diagram of the standard cascade IIR filter structure.

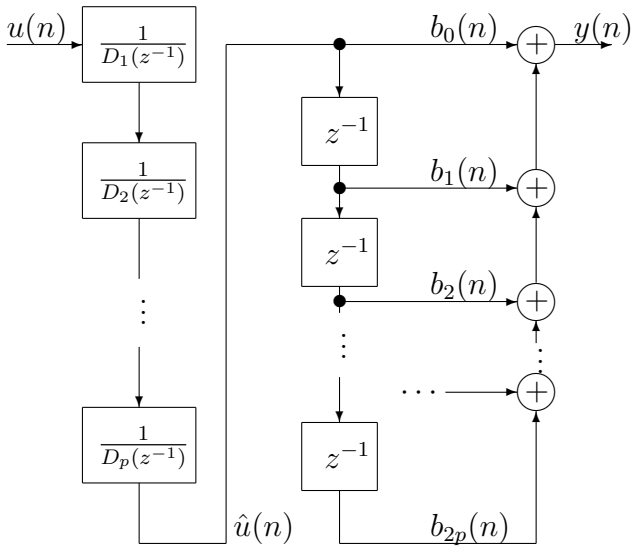


Figure 7.2: Block diagram of the all-pole-based cascade IIR filter structure.

The filter was cascaded as $H(z^{-1}) = N(z^{-1})D^{-1}(z^{-1})$ in [10], where $D^{-1}(z^{-1})$ is implemented in the all-pole normalized lattice structure (of order K). This structure demonstrates a good convergence and stability behavior but the computational complexity is very high. A different cascade structure, considered in [143], realizes the filter numerator as an all-zero tapped delay line section, with an all-pole cascade of 2nd-order section following in series. In [47], the reverse is suggested: an all-pole cascade followed by a tapped delay line to realize numerator. We refer to these structures as *all-pole-based cascades* (APC).

In the future work, we propose the following APC structure, which is shown in Figure 7.2, for adaptive IIR filters

$$H(z^{-1}) = N(z^{-1}) \prod_{k=1}^{p} \frac{1}{D_k(z^{-1})}, \tag{7.3}$$

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

where $\prod_{k=1}^{p} D_k(z^{-1}) = D(z^{-1})$ with $D_k(z^{-1}) = 1 + a_{1k}z^{-1} + a_{2k}z^{-2}$. The basic idea is to implement each $\frac{1}{D^k(z^{-1})}$ with the normal realization and the input balanced realization, proposed in Chapters 4 and 5, respectively. By doing so, the corresponding adaptive algorithms are expected to have a better stability behavior and to be more efficient in terms of implementation.

C. Applications

Adaptive IIR filtering, like other parameter estimation techniques, has been used in many applications. As general application examples, of potential interest are short training channel estimation-based equalization techniques for wired or even wireless communications. It would be interesting to look at how the adaptive algorithms proposed in this thesis perform in these systems.

Mandarin Chinese is a tone language, there are four tones for stressed syllables, the formal reading of a monosyllable being always considered stressed. The four tones in isolation are traditionally understood as follows. Tones correspond to fundamental frequency or pitch. The first one, high level, starts near the top of the speaker's pitch range and continues on that level to the end. The second tone, high rising, starts at mid-range and rises to the top of the range. The third tone, low falling, starts below mid-range, dips to the lowest pitch, and rises above mid-range. The fourth tone, high falling, starts near the top of the range and falls toward the bottom. It is known that tone of a syllable in Mandarin Chinese is basically the reflection of the variation of the fundamental frequency of the articulated syllable. So, it would be interesting to use the algorithms proposed in Chapters 4 and 6 to accurately determine the (time-varying) fundamental frequency of the voiced part of speech signal and then utilize it for the Mandarin Chinese recognition system.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Author's Publications

[1] J. Zhou, "Simplified adaptive algorithm for constrained notch filters with guaranteed stability," *IEE Proceedings of Vision, Image & Signal processing*, accepted with minor revision, Apr. 2006.

[2] J. Zhou and G. Li, "A structural view of stability in adaptive IIR filters," *IEEE Trans. Signal Processing*, accepted, Jan. 2006.

[3] J. Zhou and G. Li, "An improved adaptive algorithm for constrained notch filters with guaranteed stability," *Int. Journal of Circuits, Systems and Signal Processing*, vol. 25, no. 3, pp. 361-380, 2006.

[4] J. Zhou and G. Li, "Plain gradient based direct frequency estimation using second-order constrained adaptive IIR notch filter," *Electron. Lett.*, vol. 40, no. 5, pp. 351-352, Mar. 2004.

[5] J. Zhou and G. Li, "An efficient adaptive direct frequency estimation algorithm using a constrained IIR notch filter with guaranteed stability," in *Proc. Int. Conf. on Information, Communications and Signal Processing* (ICICS'05), Bangkok, Thailand, Dec. 2005, pp. 905-908.

[6] J. Zhou and G. Li, "Input balanced realization based adaptive IIR filters with guaranteed stability," in *Proc. Int. Conf. on Information, Communications*

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

and Signal Processing (ICICS'05), Bangkok, Thailand, Dec. 2005, pp. 913-916.

[7] J. Zhou and G. Li, "A new structure for adaptive IIR filters with guaranteed stability," in *Proc. IEEE Int. Conf. on Communications, Circuits and Systems (ICCCAS'05)*, HKUST, China, May 2005, pp. 687-690.

[8] Z. X. Zhao, G. Li, and J. Zhou, "Efficient digital filter structures with minimum roundoff noise gain," in *Proc. IEEE Int. Symp. on Circuits and Systems (ISCAS'05)*, Kobe, Japan, May 2005, pp. 2587-2590.

[9] Z. X. Zhao, G. Li, and J. Zhou "Roundoff noise analysis of two efficient digital filter structures," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP'05)*, Philadelphia, USA, vol. 4, Apr. 2005, pp. 333-336.

[10] J. Zhou and G. Li, "Adaptive frequency estimation based on normal realizations and its application in speech signal processing," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing (ICASSP'03)*, Hong Kong, China, vol. 6, Apr. 2003, pp. 201-204.

[11] J. Zhou and G. Li, "Input balanced state-space realization based adaptive recursive filters," in *Proc. IEEE Int. Symp. on Circuits and Systems (ISCAS'03)*, Bangkok, Thailand, vol. 4, May 2003, pp. 420-423.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

Bibliography

[1] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[2] S. S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[3] P. A. Regalia, *Adaptive IIR Filtering in Signal Processing and Control*. New York: Marcel Dekker, 1995.

[4] A. Gray, "Passive cascaded lattice digital filters," *IEEE Trans. on Circuits and Systems*, vol. CAS-27, pp. 337-344, May 1980.

[5] D. Parikh, N. Ahmed, and S. D. Stearns, "An adaptive lattice algorithm for recursive filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, pp. 110-112, 1980.

[6] P. A. Regalia, "Stable and efficient lattice algorithms for adaptive IIR filtering," *IEEE Trans. Signal Processing*, vol. 40, no. 2, pp. 375-388, Feb. 1992.

[7] J. E. Cousseau, "Adaptive IIR filtering: available results," *IEEE Circuits and Systems - News Letter*, vol. 10, no. 3, Sept/Oct. 1999.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[8] J. Cousseau, P. S. R. Diniz, G. Sentoni and O. Agamennoni, "On orthogonal realizations for adaptive IIR filters," *Int. J. Circ. Theor. Appl.*, no. 28, pp. 481-500, 2000.

[9] H. Fan, "A structural view of asymptotic convergence speed of adaptive IIR filtering algorithms: Part I - infinite precision implementation," *IEEE Trans. Signal Processing*, vol. 41, no. 4, pp. 1493-1517, Apr. 1993.

[10] K. X. Miao, H. Fan, and M. Doroslovacki, "Cascade lattice IIR adaptive filters," *IEEE Trans. Signal Processing*, vol. 42, no. 4, pp. 721-741, Apr. 1994.

[11] S. M. Kay, *Modern Spectral Estimation: Theory and Application*. Englewood Cliffs, NJ: Prentice-Hall, 1988.

[12] P. T. Gough, "A fast spectral estimation algorithm based on the FFT," *IEEE Trans. Signal Processing*, vol. 42, pp. 1317-1322, Jun. 1994.

[13] S. M. Kay and A. K. Shaw, "Frequency estimation by principal component AR spectral estimation method without eigen decomposition," *IEEE Trans. ASSP*, vol. 36, pp. 95-101, Jan. 1988.

[14] J. T. Karhunen and J. Joutsensalo, "Sinusoidal frequency estimation by signal subspace approximation," *Signal Processing*, vol. 40, no. 12, pp. 2961-2972, 1992.

[15] D. V. Rao and S. Y. Kung, "Adaptive notch filtering for the retrieval of sinusoids in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 791-802, Aug. 1984.

[16] T. S. Ng, "Some aspects of an adaptive digital notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 158-161, 1987.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[17] A. Nehorai, "A minimal parameter adaptive notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 983-996, 1985.

[18] B. S. Chen, T. Y. Yang, and B. H. Lin, "Adaptive notch filter by direct frequency estimation," *Signal Processing*, vol. 27, pp. 161-176, 1992.

[19] G. Li, "A stable and efficient adaptive notch filter for direct frequency estimation," *IEEE Trans. Signal Processing*, vol. 45, no. 8, pp. 2001-2009, Aug. 1997.

[20] L. Ljung, "Analysis of a recursive prediction error identification algorithm," *Automatica*, vol. 17, no. 1, pp. 89-100, Jan. 1981.

[21] L. Ljung and T.Söderstörm, *Theory and Practice of Recursive Identification*. Cambridge, MA: MIT Press, 1983.

[22] P. Stoica and A. Nehorai, "Performance analysis of an adaptive notch filter with constrained poles and zeros," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-36, pp. 911-919, June 1988.

[23] G. C. Goodwin and K. J. Sin, *Adaptive Filtering, Prediction and Control*. Englewood Cliffs, N.J: Prentice-Hall, 1984.

[24] G. C. Goodwin, "Some observations on robust stochastic estimation," in *Proc. of 8th IFAC/IFORS Symposium on Identification and System Parameter Estimation*, Beijing, 1988, pp. 22-32.

[25] G. Li and M. Gevers, "Data filtering, representation, and the numerical accuracy of parameter estimator," in *Proc. of the 31st IEEE Conf. Decision and Control*, vol. 4, 1992, pp. 3692-3697.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[26] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge University Express, 1985.

[27] K. J. Astrom, *Introduction to Stochastic Control Theory*. New York: Academic Press, 1970.

[28] T. Kailath, A. Viera, and M. Morf, "Inverses of Toeplitz operator, innovations, and orthogonal polynomials," *SIAM Rev.*, vol. 20, pp. 106-110, Jan. 1978.

[29] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs, N.J: Prentice-Hall, 1978.

[30] J. R. Deller, J. G. Proakis, and J. H. L. Hansen, *Discrete-Time Processing of Speech Signals*. Prentice-Hall, 1993.

[31] J. M. Mendel, *Discrete Techniques of Parameter Estimation: The Equation Error Formulation*. New York: Marcel Dekker, 1973.

[32] K. Steiglitz and L. F. McBride, "A Technique for the Identification of Linear Systems," *IEEE Trans. Automat. Contr.*, vol. AC-10, pp. 461-464, Oct. 1965.

[33] L. Ljung, "Analysis of recursive stochastic algorithms," *IEEE Trans. Automat. Contr.*, vol. 22, pp. 551-575, Aug. 1977.

[34] L. Ljung, "On positive real transfer functions and the convergence of some recursive schemes," *IEEE Trans. Automat. Contr.*, vol. 22, pp. 539-551, Aug. 1977.

[35] B. D. O. Adnerson and C. R. Johnson Jr., "Exponential convergence of adaptive identification and control algorithms," *Automatica*, vol. 18, pp. 1-13, 1982.

[36] R. Bellman, *Stability Theory of Differential Equations*. New York: McGraw-Hill, 1953.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[37] J. E. Cousseau and P. S. R. Diniz, "Alternative parallel realization for adaptive IIR filters," in *Proc. Int. Symp. Circuits Syst.(ISCAS'90)*, vol. 3, New Orleans, May 1990, pp. 1927-1930.

[38] D. A. Johns, W. M. Snelgrove, and A. S. Sedra, "Orthonormal ladder filters," *IEEE Trans. Circuits Syst.*, vol. 36, no. 3, pp. 337-343, Mar. 1989.

[39] D. S. Humpherys, *The Analysis, Design, and Synthesis of Electrical Filters.* Englewood Cliffs, NJ: Prentice-Hall, 1970.

[40] C. R. Johnson Jr., "Adaptive IIR filtering: Current results and open issues," *IEEE Trans. Inform. Theory*, vol. IT-30, no. 2, pp. 237-250, Mar. 1984.

[41] S. Gee and M. Rupp "An comparison of adaptive IIR echo canceller hybrids," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP'91)*, Toronto, Canada, May 1991, pp. 1541-1544.

[42] E. R. Ferrara, "Fast implementation of LMS adaptive filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, no. 4, pp. 474-475, Apr. 1980.

[43] D. F. Marshall, W. K. Jenkins, and J. J. Murphy, "The use of orthogonal transforms for improving performance of adaptive filers," *IEEE Trans. Circuits Syst.*, vol. 36, no. 4, pp. 474-484, Apr. 1989.

[44] S. A. White, "An adaptive recursive digital filter," in *Proc. 9th Asilomar Conf. Circuits, Syst., Comput.*, 1975, pp. 21-25.

[45] M. Nayeri and W. K. Jenkins, "Analysis of alternative realizations of adaptive IIR filters," in *Proc. Int. Symp. on Circuits, Syst. (ISCAS'88)*, vol. 3, Espoo, Finland, Jun. 1988, pp. 2157-2160.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[46] M. Nayeri and W. K. Jenkins, "Alternate realizations to adaptive IIR filters and properties of their performance surfaces," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 485-496, Apr. 1989.

[47] B. D. Rao, "Adaptive IIR filtering using cascade structures," in *Conference Record of the Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, vol. 1, Pacific Grove, CA, Nov. 1993, pp. 194-198.

[48] N. I. Cho, C. H. Choi, and S. U. Lee, "Adaptive line enhancement by using an IIR lattice notch filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, pp. 585-589, Apr. 1989.

[49] A. Jr. Gray and J. D Markel, "A normalized digital filter structure," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 268-277, Jun. 1975.

[50] G. Li and L. J. Qiu, "A new parametrization for designing stable adaptive IIR filters," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP'02)*, Orlando, Florida, USA, May 2002, 1333-1336.

[51] G. Li, M. Gevers, and Y. X Sun, "Performance analysis of a new structure for digital filter implementation," *IEEE Trans. Circuits Syst. I*, vol. 47, no. 4, pp. 474-482, Apr. 2000.

[52] D. A. Johns, W. M. Snelgrove, and A. S. Sedra, "Adaptive recursive state-space filters using a gradient-based algorithm," *IEEE Trans. Circuits Syst.*, vol. 37, no. 6, pp. 673-684, Jun. 1990.

[53] P. Stoica and T. Söderström, "The Steiglitz-McBride algorithm revised - convergence analysis and accuracy aspects," *IEEE Trans. Automat. Contr.*, vol. 26, no. 6, pp. 712-717, Jun. 1981.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[54] D. E. Johnson, *Introduction to Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1976.

[55] C. F. N. Cowan and P. M. Grant, *Adaptive Filters*. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[56] H. Fan and W. K. Jenkins, "A new adaptive IIR filter," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 939-947, Oct. 1986.

[57] H. Fan and M. Nayeri, "On reduced order identification: Revisiting 'On some system identification techniques for adaptive filtering'," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 1144-1151, Sept. 1990.

[58] V. E. DeBrunner and A. A. Beex, "An information approach to the convergence of output error adaptive IIR filter structures," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing (ICASSP'90)*, Albuquerque, NM, Apr. 1990, pp. 1261-1264.

[59] S. L. Netto, P. S. R. Diniz, and P. Agathoklis, "Adaptive IIR filtering algorithms for system identification: A general framework," *IEEE Trans. Education*, vol. 38, pp. 54-66, Feb. 1995.

[60] J. J. Shynk, "Adaptive IIR filtering," *IEEE ASSP Mag*, vol. ASSP-6, pp. 4-21, Apr. 1989.

[61] K. J. Aström and P. Eykhoff, "System identification - A survey," *Automatica*, vol. 7, no. 2, pp. 123-162, Mar. 1971.

[62] B. Friedlander, "System identification techniques for adaptive signal processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, no. 2, pp. 240-246, Apr. 1982.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[63] L. Ljung, *System Identification: Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[64] T. Söderström, L. Ljung, and I. Gustavsson, "A theoretical analysis of recursive identification methods," *Automatica*, vol. 14, no. 3, pp. 231-244, May 1978.

[65] J. F. Chicharo and T. S. Ng, "Gradient-based adaptive IIR notch filtering for frequency estimation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 769-777, May 1990.

[66] Y. Xiao, Y. Takeshita, and K. Shida, "Steady-state analysis of a plain gradient algorithm for a second-order adaptive IIR notch filter with constrained poles and zeros," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 7, pp. 733-740, Jul. 2001.

[67] Y. G. Xiao, Y. Takeshita, and K. Shida, " Tracking properties of a gradient-based second-order adaptive IIR notch filter with constrained poles and zeros," *IEEE Trans. Signal Processing*, vol. 50, no. 4, pp. 878-888, Apr. 2002.

[68] J. Zhou and G. Li, "Adaptive frequency estimation based on normal realizations and its application in speech signal processing," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing(ICASSP'03)*, Hong Kong, China, vol. 6, Apr. 2003, pp. 201-204.

[69] J. Zhou and G. Li, "Input balanced state-space realization based adaptive recursive filters," in *Proc. IEEE Int. Symposium on Circuits and Systems(ISCAS'03)*, Bangkok, Thailand, vol. 4, May 2003, pp. 420-423.

[70] J. Zhou and G. Li, "Plain gradient based direct frequency estimation suing second-order constrained adaptive IIR notch filter", *Electron. Lett.*, vol. 40, no. 5, pp. 351-352, Mar. 2004.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[71] A. Antoniou, *Digital Filters: Analysis and Design*. New York: McGraw-Hill, 1993.

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

[73] W. S. Lu and A. Antoniou, *Two-Dimensional Digital Filters*. New York: Marcel Dekker, 1992.

[74] C. T. Mullis and R. A. Roberts, "Synthesis of minimum roundoff noise fixed-point digital filters," *IEEE Trans. Circuits and Systems*, vol. CAS-23, pp. 551-562, Sept. 1976.

[75] M. Gevers and G. Li, *Parametrizations in Control, Estimation and Filtering Problems: Accuracy Aspects*. London: Springer Verlag, in Communications and Control Engineering Series, 1993.

[76] S. Y. Hwang, "Minimum uncorrelated unit noise in state-space digital filtering," *IEEE Trans. on Acoust., Speech, and Signal Processing*, vol. ASSP-25, pp. 273-281, Aug. 1977.

[77] D. V. B. Rao, "Analysis of coefficient quantization errors in state-space digital filters," *IEEE on Trans. on Acoust., Speech, and Signal Processing*, vol. ASSP-34, pp. 131-139, Feb. 1986.

[78] L. Thiele, "On the sensitivity of linear state-space systems," *IEEE Trans. on Circuits and Systems*, vol. CAS-33, pp. 502-510, May 1986.

[79] T. Hinamoto, S. Karino, N. Kuroda, and T. Kuma, "Error spectrum shaping in two-dimensional recursive digital filters," *IEEE Trans. Circuits Syst.*, vol. 46, pp. 1203-1215, Oct. 1999.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[80] T. Hinamoto, H. Ohnishi, and W. S. Lu, "Roundoff noise minimization of state-space digital filters using separate and joint error feedback/coordinate transformation optimization," *IEEE Trans. Circuits Syst. I*, vol. 50, pp. 23-33, Jan. 2003.

[81] G. Amit and U. Shaked, "Small roundoff realization of fixed-point digital filters and controllers," *IEEE Trans. Acoust., Speech, and Signal Processing*, vol. ASSP-36, pp. 880-891, Jun. 1988.

[82] G. Li and M. Gevers, "Roundoff noise minimization using delta-operator realizations," *IEEE Trans. Signal Processing*, vol. 41, pp. 629-637, Feb. 1993.

[83] N. Wong and T. S. Ng, "A generalized direct-form delta operator-based IIR filter with minimum noise gain and sensitivity," *IEEE Trans. Circuits and Systems - II*, vol. 48, pp. 425-431, Apr. 2001.

[84] N. Wong and T. S. Ng, "Roundoff noise minimization in a modified direct form delta operator IIR structure," *IEEE Trans. on Circuits and Systems - II*, vol. 47, pp. 1533-1536, Dec. 2000.

[85] J. Kauraniemi and T. I. Laakso, "Roundoff noise analysis of modified delta operator direct form structures," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1997.

[86] J. Kauraniemi, T.I. Laakso, I. Harttimo, and S.J. Ovaska , "Delta operator realizations of direct-form IIR filters," *IEEE Trans. Circuits and Systems-II*, vol. 45, pp. 41-52, Jan. 1998.

[87] G. Li and Z. X. Zhao, "On the generalized DFIIt structure and its state-space realization in digital filter implementation," *IEEE Trans. CAS-I*, vol. 51, no. 4, pp. 769-778 Apr. 2004.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[88] J. B. Knowles and E. M. Olcayto, "Coefficient accuracy and digital filter response," *IEEE Trans. Circuits Theory*, vol. CT-15, pp. 31-41, Mar. 1968.

[89] R. E. Crochiere, "A new statistical approach to the coefficient word length problem for digital filters," *IEEE Trans. Circuits and Systems*, vol. CAS-22, pp. 190-196, Mar. 1975.

[90] G. Li, "Two dimensional system optimal realizations with L_2-sensitivity minimization," *IEEE Trans. Signal Processing*, vol. 46, pp. 809-693, Mar. 1998.

[91] C. Xiao, "Improved L_2-sensitivity for state-space digital system,' *IEEE Trans. Signal Processing*, vol. 45, pp. 837-840, Apr. 1997.

[92] W. Y. Yan and J. B. Moore, "On L^2-sensitivity minimization of linear state-space systems," *IEEE Trans. Circuits and Systems I*, vol. 39, pp. 641-648, Aug. 1992.

[93] R. A. Roberts and C. T. Mullis, *Digital Signal Processing*. Reading, MA: Addison Wesley, 1987.

[94] C. Kenney and A. J. Laub, "Controllability and stability radii for companion form systems," *Mathematics of Control, Signals, and Systems,* vol. 1, pp. 239-256, 1988.

[95] S. Dasgupta, M. Y. Fu, and C. Schwarz, "Robust relative stability of time-invariant and time-varying lattice filters," *IEEE Trans. Signal Processing*, vol. 46, pp. 2088-2100, Aug. 1998.

[96] K. X. Miao and H. Fan, "BIBO stability of adaptive normalized lattice IIR system," in *Proc. of the 29th Conf. on Decision and Control*, Hawaii, USA, 1990, pp. 3563-3564.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[97] H. Fan, "A structural view of asymptotic convergence speed of adaptive IIR filtering algorithms: part II - finite precision implementation," *IEEE Trans. Signal Processing*, vol. 45, pp. 1458-1472, Jun. 1997.

[98] V. E. DeBrunner, *Parameter Sensitivity, Estimation and Convergence - An Information Approach*, Ph.D. Dissertation, Virginia Tech., Blacksburg, VA, Apr. 1990.

[99] V. E. DeBrunner and A. A. (Louis) Beex, "Sensitivity analysis of digital filter structures," *SIAM Journal on Matrix Analysis and Application*, vol. 9, no. 1, pp. 106-125, Jan. 1988.

[100] S. G. Sankanran and A. A. Beex, "Balanced-realization based adaptive IIR filtering," *in Proc. IEEE Int. Conf. Acous., Speech, and Signal Processing (ICASSP'99)*, vol. 4, Phoenix Arizona, Mar. 1999, pp. 1837-1840.

[101] R. J. Ober, "Balanced realizations: canonical form, parametrization, model reduction," *International Journal of Control*, vol. 46, pp. 643-670, 1987.

[102] L. M. Silveman and B. D. O. Anderson, "Controllability, observability, and stability of linear systems," *SIAM J. Contr.*, vol. 6, pp. 121-130, 1968.

[103] G. Li, "A polynomial-operator-based DFIIt structure for IIR filters," *IEEE Trans. Circuits and Systems - II*, vol. 51, pp. 147-151, Mar. 2004.

[104] R. Bellman, *Introduction to Matrix Analysis*. 2nd Edition, New York: McGraw-Hill, 1970.

[105] P. A. Regalia, "An improved lattice-based adaptive IIR notch filter," *IEEE Trans. Signal Processing*, vol. 39, pp. 2124-2128, Sept. 1991.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[106] P. A. Regalia and M. Mboup, "Undermodeled adaptive filtering: an *a priori* error bound for the Steiglitz-McBride method," *IEEE Trans. Circuits Syst. II*, vol. 43, pp. 105-116, Feb. 1996.

[107] B. W. Bomar and J. C. Hung, "Minimum roundoff noise digital filters with some power-of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. CAS-31, pp. 833-840, Oct. 1984.

[108] F. F. Yassa, "Optimality in the choice of the convergence factor for gradient-based adaptive algorithms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 35, pp. 48-59, Jan. 1987.

[109] J. E. Volder, "The CORDIC trigonometric computing technique," *IEEE Trans. Electron. Comput.*, vol. EC-8, pp. 330-334, Sep. 1959.

[110] J. S. Walter, "A unified algorithm for elementary functions," in *Proc. 1971 AFIPS Spring Joint Comput. Conf.*, Atlantic City, NJ, May 1971, pp. 379-385.

[111] G. A. Williamson, "Implementation of adaptive IIR filters with lowest complexity," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 673-678, Aug. 1997.

[112] D. Hush and N. Ahmed, "Detection and identification of sinusoids in broadband noise via a parallel recursive ALE," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP'85)*, vol. 10, Tampa, Florida, Mar, 1985, pp. 1193-1196.

[113] N. Ahmed *et al.*, "Detection of multiple sinusoids using an adaptive cascade structure," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP'84)*, vol. 9, San Diego, Mar. 1984, pp. 199-202.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[114] R. A. David, "Detection of multiple sinusoid using a parallel ALE," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP'84)*, vol. 9, San Diego, Mar. 1984, pp. 195-198.

[115] A. Carini, V. J. Mathews, and G. L. Sicuranza, "Sufficient stability bounds for slowly varying direct-form recursive linear filters and their applications in adaptive IIR filters," *IEEE Trans. Signal Processing*, vol 47, pp. 2561-2567, Sep. 1999.

[116] S. D. Stearns, "Error surfaces of recursive adaptive filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp. 763-766, Jun, 1981.

[117] M. C. Hall and P. M. Hughes, "R-cosθ adaptive IIR filter algorithm," in *Proc. IEE Saraga Colloquium on Electronic Filters*, May 1988, pp. 7/1-7/5.

[118] A. David and T. Aboulnasr, "A globally convergent adaptive IIR filter," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS'00)*, vol. 3, Geneva, Switzerland, May 2000, pp. 531-534.

[119] H. Liu, R. Ortega, and G. Damm, "A globally convergent frequency estimator," *IEEE Trans. Automat. Contr.*, vol. 44, no. 4, pp. 698-713, Apr. 1999.

[120] Y. G. Xiao, R. K. Ward, and A. Ikira, " Performance analysis of the sign algorithm for a constrained adaptive IIR notch filter", *IEEE Trans., Signal Processing*, vol. 51, pp. 1846-1858, Jul. 2003.

[121] N. I. Cho and S. U. Lee, "On the adaptive lattice notch filter for the detection of sinusoids," *IEEE Trans. Circuits Syst. II.*, vol. 40, pp. 405-416, Jul. 1993.

[122] B. D. Rao and R. Peng, "Tracking characteristics of the constrained IIR adaptive notch filter," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1466-1479, Sept. 1988.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[123] N. I. Cho and S. U. Lee, "Tracking analysis of an adaptive lattice notch filter," *IEEE Trans. Circuits Syst. II.*, vol. 42, pp. 186-195, Mar. 1995.

[124] P. Händel and A. Nehorai, "Tracking analysis of an adaptive notch filter with constrained poles and zeros," *IEEE Trans. Signal Processing*, vol. 42, pp. 281-291, Feb. 1994.

[125] S. Nishimura, J. K. Kim, and K. Hirano, "Mean-squared error analysis of an adaptive notch filter," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS'89)*, May 1989, pp. 732-735.

[126] T. Kwan and K. Martin, "Adaptive detection and enhancement of multiple sinusoids using a cascade IIR filter," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 937-947, Jul. 1989.

[127] R. Lopez-Valcarce and F. Perez-Gonzalez, "Some remarks on the lattice form of the Steiglitz-McBride iteration," *IEEE Signal Processing Lett.*, vol. 6, pp. 290-292, Nov. 1999.

[128] R. Lopez-Valcarce and F. Perez-Gonzalez, "Adaptive lattice IIR filtering revisited: convergence issue and new algorithms with improved stability properties," *IEEE Trans. Signal Processing*, vol. 49, pp. 811-821, Apr. 2001.

[129] H. C. So, "Adaptive algorithm for direct estimation of sinusoidal frequency," *Electron. Lett.,* vol. 36, no.8, pp. 759-760, 2000.

[130] M. Sheu, H. Liao, S. Kan, and M. Shieh, "A novel adaptive algorithm and VLSI design for frequency detection in noisy environment based on adaptive IIR filter," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS'01)*, vol. 4, Sydney, Australia, May 2001, pp. 446-449.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[131] S.-C. Pei and C.-C. Tseng, "Adaptive IIR notch filter based on least mean p-power error criterion," *IEEE Trans. Circuits Syst.,* vol. 40, pp. 525-529, Aug. 1993.

[132] A. V. Zitzewitz, "Considerations on acoustic echo cancellation based on real time experiments," in *Proc. EUSIPCO-90*, Barcelona, Spain, 1990, pp. 1987-1990.

[133] S. J. Flockton, S. Gudvangen, "Pole-zero modelling of room transfer functions," *2nd Int. Workshop on Acoust. Echo Cancellation*, L'Aquila, Italy, Sept. 1991.

[134] H. Fan and W. K. Jekins, "An investigation of an adaptive IIR echo canceller: Advantages and problems," *IEEE Trans. Acoust., Speech, Signal processing*, vol. 36, no. 12, pp. 1819-1834, Dec. 1988.

[135] J. A. B. Gerald, N. L. Esteves, and M. M. Silva, "A new IIR echo canceller structure," *IEEE Trans. Circuits Syst.-II*, vol. 42, no. 12, pp. 818-821, Dec. 1995.

[136] R. D. Gitlin and J. S. Thompson, "A phase adaptive structure for echo cancellation," *IEEE Trans. Commun.*, vol. COM-26, pp. 1211-1220, Aug. 1978.

[137] R. Wehrmann, J. V. List, and P. Meissner, "A noise-insensitive compromise gradient method for the adjustment of adaptive echo canceller," *IEEE Trans. Commun.*, vol. COM-28, pp. 753-759, May 1980.

[138] R. D. Gitlin and J. S. Thompson, "A new structure for adaptive digital echo cancellation," in *Proc. Nat. Telecommun., Conf.*, Dallas, TX, Dec. 1976.

[139] Technical Staff of Bell Laboratories, Transmission Systems for Communications, 5th ed. Bell Telephone Laboratories, 1982.

ATTENTION: The Singapore Copyright Act applies to the use of this document. Nanyang Technological University Library

[140] D. D. Falconer, "Adaptive reference echo cancellation," *IEEE Trans. Commun.*, vol. COM-30, pp. 2083-2094, Sept. 1982.

[141] G. A. Williamson, J. P. Ashley, and M. Nayeri, "Structural issue in cascade-form adaptive IIR filters," in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing (ICASSP'95)*, vol. 2, Detroit, Michigan, May 1995, pp. 1436-1439.

[142] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1974.

[143] F. X. Y. Gao and W. M. Snelgrove, "An adaptive backpropagation cascade IIR filter," *IEEE Trans. Circuits Syst. II*, vol. 39, pp. 606-610, Sept. 1992.

[144] L. B. Jackson and S. L. Wood, "Linear prediction in cascade form," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 26, pp. 518-528, Dec. 1978.

[145] M. R. Petraglia, J. J. Shynk, and S. K. Mitra, "Stability bounds and steady-state coefficient variance for a second-order adaptive IIR notch filter," *IEEE Trans. Signal Processing*, vol. 42, pp. 1841-1845, Jul. 1994.