Weighted Balanced Model Reduction Methods for 2-D Discrete Systems and Related Techniques

by

Hong Luo

M.Sc. (Eng.), Shanghai Jiao Tong University, China, 1986
B.Sc. (Eng.), Shanghai Jiao Tong University, China, 1983

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of
Electrical and Computer Engineering

We accept this dissertation as conforming
to the required standard

Dr. W.-S. Lu, Co-supervisor (Electrical and Computer Engineering)

Dr. A. Antoniou, Co-supervisor (Electrical and Computer Engineering)

Dr. P. Agathoklis, Member (Electrical and Computer Engineering)

Dr. Z. Dong, Outside Member (Mechanical Engineering)

Dr. K. Zhou, External Examiner (Electrical and Computer Engineering)
Louisiana State University

Abstract

Two new reliable algorithms are developed for the computation of structured controllability and observability gramians based on the mathematical solution of the Lyapunov inequalities for two-dimensional (2-D) discrete systems. The resulting improved structurally balanced realization (ISBR) and model reduction (ISBMR) methods lead to a reduced-order system that is guaranteed to be stable while maintaining small approximation error.

The ISBR and ISBMR are subsequently extended to the case of 2-D discrete systems with input and output weights. The proposed methods known as the *weighted structurally balanced realization* (WSBR) and *model reduction* (WSBMR) methods are shown to yield stable reduced-order systems with small approximation errors in *specified* frequency ranges.

New algorithms are developed for the determination of *transfer-function matrices* from the Roesser and Fornasini-Marchesini state-space models, whose computational efficiency is superior to that of the existing algorithms.

The dissertation concludes with a general-purpose *design environment* for various recursive and nonrecursive 2-D digital filters. The design environment integrates the singular-value decomposition design method with the algorithms developed, and is expected to be useful to engineers and researchers in the areas of 2-D digital filter and digital signal processing.

Numerous design examples are provided throughout the dissertation to demonstrate the efficiency of the proposed algorithms and the flexibility of the design environment developed.

Examiners:

Dr. W.-S. Lu, Co-supervisor (Electrical and Computer Engineering)

Dr. A. Antoniou, Co-supervisor (Electrical and Computer Engineering)

Dr. P. Agathoklis, Member (Electrical and Computer Engineering)

Dr. P. Agathoklis, Member (Electrical and Computer Engineering)

Dr. Z. Dong, Outside Member (Mechanical Engineering)

Dr. K. Zhou, External Examiner (Electrical and Computer Engineering)
Louisiana State University

To my parents and sisters

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Professors W.-S. Lu and A. Antoniou, for their valuable guidance and support in this endeavour.

I would also like to acknowledge my fellow graduate students in the Micronet Centre at the University of Victoria for the friendly working environment.

This research was supported in part by Micronet (Networks of Centres of Excellence Program) and the award of a University of Victoria Fellowship.

Lastly, I would like to express my heartfelt appreciation to my family for their understanding and encouragement throughout.

Contents

List of Figures

List of Tables

List of Abbreviations

1-D	one-dimensional
2-D	two-dimensional
BIBO	bounded-input bounded-output
DFT	discrete Fourier transform
DSP	digital signal processing
F-M	Fornasini-Marchesini
MIMO	multi-input multi-output
PBR	pseudo-balanced realization
PBMR	pseudo-balanced model-reduction
QBR	quasi-balanced realization
QBMR	quasi-balanced model-reduction
Q-stable	quadratically stable
Q-stability	quadratic stability
SBR	structurally balanced realization
SBMR	structurally balanced model-reduction
ISBR	improved structurally balanced realization
ISBMR	improved structurally balanced model-reduction
SISO	single-input single-output
SVD	singular-value decomposition
VLSI	very large scale integration
WSBR	weighted structurally balanced realization
WSBMR	weighted structurally balanced model reduction

Chapter 1

Introduction

1.1 2-D Digital Filters and 2-D Discrete Systems

Two-dimensional (2-D) digital signal processing (DSP) is primarily concerned with the representation, transformation, and manipulation of signals that can be represented as 2-D arrays. Typical examples of 2-D signals that might need to be processed are images such as satellite photographs, radar and sonar maps, medical X-ray pictures, and data from seismic and geophysical records [25, 40, 51]. In many cases, the central part of a 2-D DSP system is a specific piece of software or a dedicated hardware board implementing an algorithm that can process signals received, and is referred to, in general, as a *2-D digital filter.*

2-D digital filters can be classified as *recursive* or *nonrecursive*, based upon whether the output of the filter depends on previous values of the output. Nonrecursive filters have the advantage that they are free of stability

problems while recursive filters have the advantage of modest requirements on computation and computer memory. In addition, 2-D digital filters are single-input single-output (SISO) discrete systems. 2-D discrete systems can be characterized in terms of difference equations or state-space models in two independent variables and in terms of transfer functions or matrices of transfer functions, which are rational functions of polynomials in two variables. The mathematical theory developed for the analysis of 2-D discrete systems provides the necessary framework for the study of 2-D digital filters [9, 10, 16, 25].

At a conceptual level, a great deal of similarity exits between 1-D and 2-D discrete systems. However, at a more detailed level, considerable differences exist between the two types of systems. One major difference is the amount of data involved in typical applications. As a result, the computational efficiency of an algorithm plays a much more important role in 2-D systems. Another major difference is that the mathematics used in 2-D systems is often more complex than in 1-D systems, as may be expected. For example, the fundamental theorem of algebra states that any 1-D polynomial can be factored as a product of lower-order polynomials while a 2-D polynomial cannot generally be factored as a product of lower-order polynomials. Therefore, the study of stability in 2-D discrete systems is much more complicated [25].

1.2 Previous Work

Extensive research conducted over the past decade on 2-D discrete systems has resulted in several useful methods for the analysis and design of 2-D systems [9, 10, 25, 40, 51]. These include methods for the analysis of stability [2, 28, 29, 30], for the study of finite-wordlength effects [12, 18, 21, 26, 28, 31], for balanced realization [32, 53, 56, 62], and for design and implementation [3, 11, 25, 27, 38].

1.2.1 2-D Balanced Realization

Balanced model-reduction method is an effective and numerically economical technique to obtain a reduced-order system by directly truncating the balanced realization for a original full-order system. Balanced model-reduction method, which has been applied to 1-D systems in [52] and to 2-D systems in [32, 56, 62], has several desirable properties such as a bounded approximation error and preservation of the stability [52, 56] of the original system. To obtain the balanced realization for a 1-D or 2-D system, one needs to compute the controllability and observability gramians of the system. These gramians are equivalent to those used extensively in the analysis of finite wordlength effects [21, 27, 31] and the synthesis of linear dynamic systems [20].

In 1-D systems, the gramians are uniquely defined by a 1-D Lyapunov equation [52]. However, in 2-D systems the gramians are divided into three

categories, namely, *pseudo-gramians* [32], *quasi-gramians* [62], and *structured gramians* [56]. Balanced realization methods based on the pseudo-, quasi-, and structured gramians result in *pseudo-*, *quasi-*, and *structurally balanced realization* methods [32, 62], respectively. Pseudo-gramians were originally proposed in [32] and have since found applications in model-reduction [32], round-off noise minimization [31, 34], and filter design [37, 38]. The computation of pseudo-gramians is rather expensive for 2-D systems of order greater than 10. Quasi-gramians were originally proposed in [62] and subsequently applied to model-reduction and filter design. To the author's knowledge, the necessary and sufficient conditions for the existence of the quasi-gramians has not been addressed in the literature.

Structured gramians, which can be considered as a direct extension of 1-D gramians, are defined as the solution of two 2-D Lyapunov *inequalities* for 2-D discrete systems [56]. 1-D gramians and 2-D structured gramians play an important role in the analysis of stability in 1-D and 2-D discrete systems [20, 56], respectively. When a pseudo- or quasi-balanced model-reduction method is used to reduce the order of a 2-D system, the reduced-order system may be unstable even if the original system is stable [53, 62]. A structurally balanced model-reduction method, on the other hand, always leads to a stable reduced-order system. However, no algorithms have been given in the literature for the computation of structured gramians.

1.2.2 1-D Weighted Balanced Realization

For a 1-D system with input and output weights, a so-called *weighted reduced-order system* can be obtained by applying a *weighted balanced model-reduction* method for the system [1, 13, 61]. Such applications may include the design of feedback control systems where the reduced-order system needs to be an accurate approximation to the full-order system at the crossover region [1], and the design of digital filters where the approximation error must satisfy prescribed specifications only in the passband and stopband. Although several studies on weighted balanced realization and model-reduction methods for 1-D continuous and discrete systems have been reported in the literature [1, 13, 61], the extension to 2-D systems has not to date been addressed.

1.2.3 2-D Transfer-Function Matrices

The full-order and reduced-order 2-D systems referred to in Sections 1.2.1 and 1.2.2 are often represented by state-space models. However, many of the available analysis and design methods are applicable only to the direct forms of 2-D systems, that is, the transfer-function matrices that represent the systems [25]. Therefore, it is often necessary to determine the transfer-function matrices from the state-space description of a 2-D system.

Several state-space models have been proposed for 2-D discrete systems. Among these models, the Roesser model [54] is the most commonly used. Some algorithms for the determination of 2-D transfer-function matrices from the Roesser model have been proposed in [6, 7, 22, 49, 50, 59]. The algo-

rithms in [22, 49, 50] are basically extensions of the well-known Fadeeva (1-D) algorithm [60] to the 2-D case while the algorithms in [6, 7, 59] are based on the discrete Fourier transform (DFT). In either case, to obtain the optimal structure from the Roesser state-space model, the 2-D similarity transformation matrix is restricted to be block-diagonal so that the transfer-function matrix is invariant [31]. Another popular state-space representation for 2-D discrete systems is in terms of the Fornasini-Marchesini model [15] in which the 2-D similarity transformation matrix is not required to be block-diagonal and can be utilized without affecting the input-output relation. To date, no efficient algorithm for the determination of the 2-D transfer-function matrices from the Fornasini-Marchesini state-space representation has been given in the literature.

1.2.4 2-D Digital Filters

Many useful algorithms for the analysis, design, and realization of 2-D digital filters exist today [25, 40]. The design methods include the window method [11, 19], the McClellan transformation [46, 47, 48], several optimization methods [44, 58], and the singular-value decomposition (SVD) design method [4, 33, 37, 38]. The SVD design method, which is based on a numerically reliable matrix decomposition named SVD [17], can be used to design 2-D digital filters with arbitrary amplitude and phase responses [37]. The SVD design method has several advantages. First, the design of a 2-D digital filter can be accomplished by designing a set of 1-D sub-filters and, there-

fore, many well-established techniques for the design of 1-D filters can be employed. Second, the 2-D filter obtained is always stable. Third, the 1-D sub-filters form a parallel structure which allows extensive parallel processing, hence the structure obtained is suitable for very large scale integration (VLSI) implementation [38]. Although there exist many design methods, no general-purpose design environment is available for 2-D digital filters.

1.3 Contributions of the Dissertation

The stability of 2-D systems and the computational efficiency of algorithms for 2-D systems are, in general, of major concern. Hence, it is desirable to develop mathematical models and computationally efficient algorithms that guarantee stability. The main contributions of this dissertation can be summarized as follows:

- A new sufficient condition for the existence of quasi-gramians is presented. A computationally efficient iterative algorithm for computing quasi-gramians is introduced.

- Two new reliable algorithms are developed for the computation of structured gramians. The second algorithm results in improved structurally balanced realization (ISBR) and model-reduction (ISBMR) methods which lead to reduced-order systems that are guaranteed to be stable, and has small approximation error.

- Innovative *weighted structurally balanced realization* (WSBR) and *model-reduction* (WSBMR) methods are developed for 2-D discrete systems with input and output weights. To assist in the definition of the *weighted* structured controllability and observability gramians for the 2-D systems with input and output weights, the so-called *weighted-input-to-state* and *state-to-weighted-output* auxiliary transfer-function matrices are introduced. The proposed WSBMR yields a stable reduced-order system that approximates the original full-order system with a small error in *specified* frequency regions.

- Computationally efficient algorithms for the determination of transfer-function matrices from the Roesser and Fornasini-Marchesini state-space models of multi-input multi-output (MIMO) 2-D discrete systems are developed. The computational efficiency of the new algorithms for the case of the Roesser state-space model has been found to be superior to that of the existing algorithms [7, 50].

- A general-purpose design environment is developed for a variety of recursive and nonrecursive 2-D digital filters. The design environment consists of two independent modules: one is a *user-interface* design software, which assists the novice to design 2-D digital filters; the other is a *design toolbox*, which consists of a library of design functions to assist an expert to design highly specialized 2-D digital filters. The design environment can be used not only for the design of standard 2-

D filters, such as *circularly symmetric* or *quadrantally symmetric* filters, but also for the design of non-standard *user-defined* filters.

1.4 Organization of the Dissertation

This dissertation is organized as follows:

In Chapter 2, the definitions for the pseudo-, quasi-, and structured gramians and the corresponding balanced realization methods reported in [32, 56, 62] are summarized. A sufficient condition for the existence of quasi-gramians is presented. This is followed by efficient and reliable algorithms to compute the quasi- and structured gramians. The ISBR and ISBMR methods are obtained using the new algorithm for computing structured gramians. The performance of the proposed algorithms is evaluated through two examples.

In Chapter 3, two *auxiliary* transfer-function matrices of a 2-D discrete system with input and output weights are first introduced. 2-D weighted structured gramians are defined based on the new definitions for the auxiliary transfer-function matrices. The WSBR and WSBMR methods are subsequently developed. A lowpass 2-D filter is then designed to demonstrate the proposed WSBMR method.

In Chapter 4, new algorithms for the determination of transfer-function matrices from the Roesser and Fornasini-Marchesini state-space models for 2-D discrete systems are proposed. The computational efficiency of the pro-

posed algorithms is examined through numerical computations and compared with that of the existing algorithms.

In Chapter 5, a general-purpose design environment for the design of 2-D digital filters is described. The structure of the design environment developed is first discussed followed by a step-by-step description of the software routines that compose the design environment. The use and usefulness of the design environment are demonstrated through a number of examples.

In Chapter 6, conclusions of the dissertation are given, and several research topics that could be undertaken in the future are described.

Chapter 2

2-D Balanced Realization and Model Reduction

2.1 Introduction

The computation of the controllability and observability gramians is essential in the balanced realization method. In this chapter, a review of the three known balanced realizations for 2-D discrete systems, namely, the pseudo-, quasi-, and structurally balanced realizations, is given in Section 2.2. Motivated by the lack of an efficient and reliable algorithm for computing quasi-gramians, a new algorithm for the computation of the quasi-gramians is developed in Section 2.3. The existence of quasi-gramians is addressed in Section 2.3.

It is shown in Section 2.2 that the computation of the structured gramians, which are defined as the solution of two 2-D Lyapunov inequalities [56], amounts to solving two *constrained* optimization problems. In Section 2.4, these optimization problems are reformulated as *unconstrained* minimization

problems, and a new algorithm for the computation of structured gramians is then proposed. In order to ensure that the minimizations can be carried out effectively, a procedure for determining a good initial point is also proposed.

In Section 2.5, the algorithm developed in Section 2.4 is modified to take into account both system stability and approximation error. The improved structurally balanced realization (ISBR) and model-reduction (ISBMR) methods are obtained using the modified algorithm. The resulting reduced-order system guarantees stability and approximates the original system with small approximation error. The chapter concludes with examples to illustrate the proposed algorithms and to evaluate their performances.

2.2 Review of 2-D Balanced Realization

Consider a linear, shift-invariant, multi-input multi-output (MIMO) 2-D discrete system of order (m, n) represented by the block diagram in Figure 2.1, where $\mathbf{H}(z_1, z_2) \in \Re^{(s\times t)}$ denotes the transfer-function matrix of the system. The Roesser state-space model [54] for the system can be written as

$$\begin{bmatrix} \mathbf{x}^h(k+1, l) \\ \mathbf{x}^v(k, l+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \begin{bmatrix} \mathbf{x}^h(k, l) \\ \mathbf{x}^v(k, l) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix} \mathbf{u}(k, l)$$

$$= \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u} \tag{2.1a}$$

$$\mathbf{y}(k, l) = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}^h(k, l) \\ \mathbf{x}^v(k, l) \end{bmatrix} + \mathbf{D}\,\mathbf{u}(k, l)$$

$$= \mathbf{C}\,\mathbf{x} + \mathbf{D}\,\mathbf{u} \tag{2.1b}$$

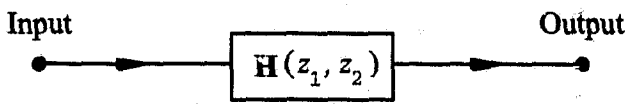

Figure 2.1: A 2-D discrete system.

where $\mathbf{x}^h \in \Re^m$ and $\mathbf{x}^v \in \Re^n$ are the horizontal and vertical state-space vectors, respectively; and $\mathbf{u} \in \Re^t$ and $\mathbf{y} \in \Re^s$ are the input and output vectors, respectively. For simplicity, the 2-D discrete system is represented by the set $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ where

$$\mathbf{A} \in \Re^{(m+n)\times(m+n)}, \quad \mathbf{B} \in \Re^{(m+n)\times t}, \quad \mathbf{C} \in \Re^{s\times(m+n)}, \quad \mathbf{D} \in \Re^{(s\times t)}$$

By taking the 2-D z transform of the system of equations in (2.1a, b) and defining

$$\mathbf{I}(z_1, z_2) = z_1 \mathbf{I} \oplus z_2 \mathbf{I}$$

where $\mathbf{I}$ denotes the identity matrix, (z_1, z_2) is a pair of complex variables, and $\oplus$ denotes the direct sum of matrices, the transfer-function matrix of the system in Figure 2.1 is obtained as

$$\mathbf{H}(z_1, z_2) = \mathbf{C}\,[\,\mathbf{I}(z_1, z_2) - \mathbf{A}\,]^{-1}\,\mathbf{B} + \mathbf{D} \tag{2.2}$$

2.2.1 Pseudo-Balanced Realization

Based on the Roesser state-space model given in (2.1a, b), the 2-D *pseudo-controllability and observability gramians* are first generalized from the 1-D

gramians. The following definitions provide the necessary background for the implementation of the pseudo-balanced realization (PBR) method.

Definition 2.1 [32]

The *pseudo-controllability and observability gramians* of the system in Figure 2.1, denoted by $\mathbf{P}^p$ and $\mathbf{Q}^p$, are defined in terms of the integrals[1]

$$\mathbf{P}^p = \frac{1}{(2\pi j)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} \mathcal{K}(z_1, z_2)\, \mathcal{K}^*(z_1, z_2)\, \frac{dz_2}{z_2}\, \frac{dz_1}{z_1}$$

and

$$\mathbf{Q}^p = \frac{1}{(2\pi j)^2} \oint_{|z_1|=1} \oint_{|z_2|=1} \mathcal{L}(z_1, z_2)\, \mathcal{L}^*(z_1, z_2)\, \frac{dz_2}{z_2}\, \frac{dz_1}{z_1}$$

respectively, where

$$\mathcal{K} = [\, \mathbf{I}(z_1, z_2) - \mathbf{A}\,]^{-1}\, \mathbf{B}$$

$$\mathcal{L} = \mathbf{C}\, [\, \mathbf{I}(z_1, z_2) - \mathbf{A}\,]^{-1}$$

Straightforward analysis shows that

$$\mathbf{P}^p = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} \mathcal{M}^{(k,l)}\, \mathcal{M}^{(k,l)^T} \tag{2.3a}$$

$$\mathbf{Q}^p = \sum_{k=0}^{\infty} \sum_{l=0}^{\infty} \begin{bmatrix} [\mathcal{A}^{(k,l)^T} \mathbf{C}^T \mathbf{C} \mathcal{A}^{(k,l)}]_1 & [\mathcal{A}^{(k,l)^T} \mathbf{C}^T \mathbf{C} \mathcal{A}^{(k+1,l-1)}]_2 \\ [\mathcal{A}^{(k,l)^T} \mathbf{C}^T \mathbf{C} \mathcal{A}^{(k-1,l+1)}]_3 & [\mathcal{A}^{(k,l)^T} \mathbf{C}^T \mathbf{C} \mathcal{A}^{(k,l)}]_4 \end{bmatrix} \tag{2.3b}$$

where the individual matrices $\mathcal{A}^{(k,l)}$ and $\mathcal{M}^{(k,l)}$ are computed using the

[1]The italic superscript p is reserved for matrices directly related to the PBR.

iterative formulas[2]

$$\mathcal{A}^{(0,0)} = \mathbf{I}, \quad \mathcal{A}^{(-1,l)} = \mathbf{0}, \quad \mathcal{A}^{(k,-1)} = \mathbf{0}$$

$$\mathcal{A}^{(1,0)} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \mathcal{A}^{(0,1)} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix}$$

$$\mathcal{A}^{(k,l)} = \mathcal{A}^{(1,0)}\,\mathcal{A}^{(k-1,l)} + \mathcal{A}^{(0,1)}\,\mathcal{A}^{(k,l-1)}$$

$$\mathcal{M}^{(k,l)} = \mathcal{A}^{(k-1,l)} \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \end{bmatrix} + \mathcal{A}^{(k,l-1)} \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_2 \end{bmatrix}$$

for $k, l = \{0, 1, 2, \ldots, \infty\}$.

Matrices $\mathbf{P}^p$ and $\mathbf{Q}^p$ are often used in block form as

$$\mathbf{P}^p = \begin{bmatrix} \mathbf{P}_1^p & \mathbf{P}_2^p \\ \mathbf{P}_2^{pT} & \mathbf{P}_4^p \end{bmatrix}, \qquad \mathbf{Q}^p = \begin{bmatrix} \mathbf{Q}_1^p & \mathbf{Q}_2^p \\ \mathbf{Q}_2^{pT} & \mathbf{Q}_4^p \end{bmatrix}$$

where

$$\mathbf{P}_1^p \in \Re^{(m\times m)}, \quad \mathbf{P}_4^p \in \Re^{(n\times n)}, \quad \mathbf{Q}_1^p \in \Re^{(m\times m)}, \quad \mathbf{Q}_4^p \in \Re^{(n\times n)}$$

Definition 2.2 [32]

A 2-D system represented by the Roesser state-space model in (2.1a, b) is said to be *pseudo-balanced* if the pseudo-gramians satisfy

$$\mathbf{P}_1^p = \mathbf{Q}_1^p = \mathbf{\Sigma}_1^p = \text{diag}(\sigma_1^p, \ldots, \sigma_m^p) \tag{2.4a}$$

$$\mathbf{P}_2^p = \mathbf{Q}_2^p = \mathbf{\Sigma}_2^p = \text{diag}(\mu_1^p, \ldots, \mu_n^p) \tag{2.4b}$$

[2]The italic superscript (k, l) denotes iteration k, l in an iterative formula, and the lower-subscript notation $[\,]_1$ denotes the upper-left $m \times m$, $[\,]_2$ the upper-right $m \times n$, $[\,]_3$ the lower-left $n \times m$, and $[\,]_4$ the lower-right $n \times n$ sub-matrices.

where

$$\sigma_1^p \geq \sigma_2^p \geq \cdots \geq \sigma_m^p \geq 0, \qquad \mu_1^p \geq \mu_2^p \geq \cdots \geq \mu_n^p \geq 0$$

are called the *pseudo-Hankel singular values* of the system.

A PBR can be obtained by finding a nonsingular transformation matrix

$$\mathbf{T}^p = \mathbf{T}_1^p \oplus \mathbf{T}_2^p$$

such that the equivalent system

$$\{\mathbf{A}^p, \mathbf{B}^p, \mathbf{C}^p, \mathbf{D}\} = \{ (\mathbf{T}^p)^{-1} \mathbf{A} \mathbf{T}^p, (\mathbf{T}^p)^{-1} \mathbf{B}, \mathbf{C} \mathbf{T}^p, \mathbf{D} \} \qquad (2.5)$$

is pseudo-balanced, i.e.,

$$(\mathbf{T}_1^p)^{-1} \mathbf{P}_1^p (\mathbf{T}_1^p)^{-T} = \mathbf{T}_1^{pT} \mathbf{Q}_1^p \mathbf{T}_1^p = \mathbf{\Sigma}_1^p = \text{diag}(\sigma_1^p, \ldots, \sigma_m^p) \quad (2.6a)$$

$$(\mathbf{T}_2^p)^{-1} \mathbf{P}_2^p (\mathbf{T}_2^p)^{-T} = \mathbf{T}_2^{pT} \mathbf{Q}_2^p \mathbf{T}_2^p = \mathbf{\Sigma}_2^p = \text{diag}(\mu_1^p, \ldots, \mu_n^p) \quad (2.6b)$$

One approach to find a transformation matrix $\mathbf{T}^p$ is to apply the well-known algorithm proposed by Laub in [23] to $\{\mathbf{P}_1^p, \mathbf{Q}_1^p\}$ and $\{\mathbf{P}_4^p, \mathbf{Q}_4^p\}$, respectively, to find nonsingular matrices $\mathbf{T}_1^p$ and $\mathbf{T}_2^p$, respectively. For the sake of completeness, the algorithm described by Laub is listed below

Algorithm 2.1 [23]

Step 1: Use the Cholesky factorization to decompose $\mathbf{P}_1^p$ as

$$\mathbf{P}_1^p = \mathbf{L}_{\mathrm{P}1}^p \mathbf{L}_{\mathrm{P}1}^{pT}$$

where $\mathbf{L}^p_{P1}$ is a lower-triangular matrix.

Step 2: Obtain the eigenvalue and eigenvector matrices, denoted by $\mathbf{\Lambda}_{P1}$ and $\mathbf{U}_{P1}$, respectively, of the matrix $\mathbf{L}^p_{P1}\,\mathbf{P}^p_2\,\mathbf{L}^{pT}_{P1}$ as

$$\mathbf{L}^p_{P1}\,\mathbf{P}^p_2\,\mathbf{L}^{pT}_{P1} = \mathbf{U}_{P1}\,\mathbf{\Lambda}_{P1}\,\mathbf{U}^T_{P1}$$

Step 3: Form the nonsingular transformation matrix $\mathbf{T}^p_1$

$$\mathbf{T}^p_1 = \mathbf{L}^p_{P1}\,\mathbf{U}_{P1}\,\mathbf{\Lambda}_{P1}^{-1/4}$$

2.2.2 Quasi-Balanced Realization

In the study of 2-D discrete systems, the inequalities

$$\mathbf{A}\mathbf{P}^s\mathbf{A}^T - \mathbf{P}^s + \mathbf{B}\mathbf{B}^T < \mathbf{0} \tag{2.7a}$$

$$\mathbf{A}^T\mathbf{Q}^s\mathbf{A} - \mathbf{Q}^s + \mathbf{C}^T\mathbf{C} < \mathbf{0} \tag{2.7b}$$

play an important role. In general, the solutions of these inequalities, if they exist, can only be found numerically and involve a substantial amount of computation. As an alternative, the so-called *quasi-balanced realization* (QBR) [62] may be considered. To define this realization, the concept of *quasi-controllability* and *observability gramians* is first introduced.

Definition 2.3 [62]

The quasi-controllability and observability gramians for a 2-D system represented by the Roesser state-space model in (2.1a, b) are

$$\mathbf{P}^q = \mathbf{P}^q_1 \oplus \mathbf{P}^q_2 \quad \text{and} \quad \mathbf{Q}^q = \mathbf{Q}^q_1 \oplus \mathbf{Q}^q_2$$

such that[3]

$$\mathbf{A}_1\mathbf{P}_1^q\mathbf{A}_1^T - \mathbf{P}_1^q + \mathbf{B}_1\mathbf{B}_1^T + \mathbf{A}_2\mathbf{P}_2^q\mathbf{A}_2^T = \mathbf{0} \tag{2.8a}$$

$$\mathbf{A}_4\mathbf{P}_2^q\mathbf{A}_4^T - \mathbf{P}_2^q + \mathbf{B}_2\mathbf{B}_2^T + \mathbf{A}_3\mathbf{P}_1^q\mathbf{A}_3^T = \mathbf{0} \tag{2.8b}$$

$$\mathbf{A}_1^T\mathbf{Q}_1^q\mathbf{A}_1 - \mathbf{Q}_1^q + \mathbf{C}_1^T\mathbf{C}_1 + \mathbf{A}_3^T\mathbf{Q}_2^q\mathbf{A}_3 = \mathbf{0} \tag{2.8c}$$

$$\mathbf{A}_4^T\mathbf{Q}_2^q\mathbf{A}_4 - \mathbf{Q}_2^q + \mathbf{C}_2^T\mathbf{C}_2 + \mathbf{A}_2^T\mathbf{Q}_1^q\mathbf{A}_2 = \mathbf{0} \tag{2.8d}$$

As will be shown in Section 2.5, solving (2.8a-d) requires much less computation than solving (2.7a, b).

Definition 2.4 [62]

A 2-D system represented by the Roesser state-space model in (2.1a, b) is said to be *quasi-balanced* if the quasi-gramians satisfy

$$\mathbf{P}_1^q = \mathbf{Q}_1^q = \mathbf{\Sigma}_1^q = \text{diag}(\sigma_1^q, \ldots, \sigma_m^q) \tag{2.9a}$$

$$\mathbf{P}_2^q = \mathbf{Q}_2^q = \mathbf{\Sigma}_2^q = \text{diag}(\mu_1^q, \ldots, \mu_n^q) \tag{2.9b}$$

where

$$\sigma_1^q \geq \sigma_2^q \geq \cdots \geq \sigma_m^q \geq 0, \qquad \mu_1^q \geq \mu_2^q \geq \cdots \geq \mu_n^q \geq 0$$

are called the *quasi-Hankel singular values* of the system.

If the 2-D system represented by the set $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ is not quasi-balanced and if the equations in (2.8a-d) have a positive definite solution $\mathbf{P}^q$ and $\mathbf{Q}^q$, then Algorithm 2.1 can be applied to $\{\mathbf{P}_1^q, \mathbf{Q}_1^q\}$ and $\{\mathbf{P}_2^q, \mathbf{Q}_2^q\}$ to

[3]The italic superscript q is reserved for matrices directly related to QBR.

find nonsingular matrices $\mathbf{T}_1^q$ and $\mathbf{T}_2^q$, respectively, such that

$$\{\mathbf{A}^q, \mathbf{B}^q, \mathbf{C}^q, \mathbf{D}\} = \{ (\mathbf{T}^q)^{-1}\mathbf{A}\mathbf{T}^q, (\mathbf{T}^q)^{-1}\mathbf{B}, \mathbf{C}\mathbf{T}^q, \mathbf{D} \} \quad (2.10)$$

is quasi-balanced, where

$$\mathbf{T}^q = \mathbf{T}_1^q \oplus \mathbf{T}_2^q$$

is the transformation matrix.

Corollary 2.1: [62]

If the denominator of the transfer-function matrix of a system represented by the Roesser state-space model in (2.1a, b) is separable, that is,

$$\mathbf{A}_2 = \mathbf{0} \quad \text{or} \quad \mathbf{A}_3 = \mathbf{0}$$

then, the QBR and PBR become identical.

2.2.3 Structurally Balanced Realization

When the PBR or QBR is used to reduce the order of a 2-D discrete system, the reduced-order system may be unstable even if the original system is stable [53, 62]. As will be shown later, the *structurally balanced realization* (SBR) [56] assures the stability of the reduced-order system if the original system is quadratically stable (Q-stable).

Definition 2.5 [56]

A 2-D system represented by the Roesser state-space model in (2.1a, b) is said to be *Q-stable* if there exists a nonsingular matrix

$$\mathbf{T}^s = \mathbf{T}_1^s \oplus \mathbf{T}_2^s$$

such that[4]

$$\gamma_{\max}\left[(\mathbf{T}^s)^{-1}\mathbf{A}\mathbf{T}^s\right] < 1$$

where $\gamma_{\max}(\mathbf{X})$ denotes the largest eigenvalue of matrix $\mathbf{X}$.

Definition 2.6 [56]

A 2-D system represented by the Roesser state-space model in (2.1a, b) is said to be *structurally balanced* if there exist positive definite matrices

$$\mathbf{P}^s = \mathbf{P}_1^s \oplus \mathbf{P}_2^s > \mathbf{0} \quad \text{and} \quad \mathbf{Q}^s = \mathbf{Q}_1^s \oplus \mathbf{Q}_2^s > \mathbf{0}$$

where

$$\mathbf{P}_1^s,\ \mathbf{Q}_1^s \in \Re^{(m\times m)} \quad \text{and} \quad \mathbf{P}_2^s,\ \mathbf{Q}_2^s \in \Re^{(n\times n)}$$

such that the *2-D Lyapunov inequalities*

$$\mathbf{A}\mathbf{P}^s\mathbf{A}^T - \mathbf{P}^s + \mathbf{B}\mathbf{B}^T < \mathbf{0} \tag{2.11a}$$

$$\mathbf{A}^T\mathbf{Q}^s\mathbf{A} - \mathbf{Q}^s + \mathbf{C}^T\mathbf{C} < \mathbf{0} \tag{2.11b}$$

are satisfied, and

$$\mathbf{P}_1^s = \mathbf{Q}_1^s = \mathbf{\Sigma}_1^s = \text{diag}\,(\sigma_1^s, \ldots, \sigma_m^s) \tag{2.12a}$$

$$\mathbf{P}_2^s = \mathbf{Q}_2^s = \mathbf{\Sigma}_2^s = \text{diag}\,(\mu_1^s, \ldots, \mu_n^s) \tag{2.12b}$$

$\mathbf{P}^s$ and $\mathbf{Q}^s$ are called the *structured controllability* and *observability gramians* of the system, respectively, and

$$\sigma_1^s \geq \sigma_2^s \geq \cdots \geq \sigma_m^s \geq 0, \qquad \mu_1^s \geq \mu_2^s \geq \cdots \geq \mu_n^s \geq 0$$

[4]The italic superscript s is reserved for matrices directly related to SBR.

are called the *structured Hankel singular values* of the system.

The quadratic stability (Q-stability) of the 2-D system is considered to be a stronger statement than the bounded-input bounded-output (BIBO) stability [25]. The following lemma proves that the Q-stability of the system guarantees the existence of a SBR.

Lemma 2.1 [56]

A 2-D system represented by the Roesser state-space model in terms of the set $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ is Q-stable if and only if there exist

$$\mathbf{P}^s = \mathbf{P}_1^s \oplus \mathbf{P}_2^s > \mathbf{0} \quad \text{and} \quad \mathbf{Q}^s = \mathbf{Q}_1^s \oplus \mathbf{Q}_2^s > \mathbf{0}$$

such that the 2-D Lyapunov inequalities (2.11a, b) hold.

In principle, the $\mathbf{P}^s$ and $\mathbf{Q}^s$ that satisfy (2.11a, b) can be obtained by solving the *constrained* minimization problems [62]

$$\underset{\mathbf{P}^s>0}{\text{minimize}} \ \gamma_{\max}(\mathbf{A}\mathbf{P}^s\mathbf{A}^T - \mathbf{P}^s + \mathbf{B}\mathbf{B}^T) \tag{2.13a}$$

$$\underset{\mathbf{Q}^s>0}{\text{minimize}} \ \gamma_{\max}(\mathbf{A}^T\mathbf{Q}^s\mathbf{A} - \mathbf{Q}^s + \mathbf{C}^T\mathbf{C}) \tag{2.13b}$$

provided that the solutions $\mathbf{P}^s$ and $\mathbf{Q}^s$ satisfy

$$\gamma_{\max}(\mathbf{A}\mathbf{P}^s\mathbf{A}^T - \mathbf{P}^s + \mathbf{B}\mathbf{B}^T) < 0$$

$$\gamma_{\max}(\mathbf{A}^T\mathbf{Q}^s\mathbf{A} - \mathbf{Q}^s + \mathbf{C}^T\mathbf{C}) < 0$$

A feature of the solution for the inequalities is that once a solution is found, there exist infinitely many solutions. It follows that as long as the

system is Q-stable, one can find positive definite structured gramians $\mathbf{P}^s$ and $\mathbf{Q}^s$, and apply Algorithm 2.1 to $\{\mathbf{P}_1^s,\ \mathbf{Q}_1^s\}$ and $\{\mathbf{P}_2^s,\ \mathbf{Q}_2^s\}$ to find nonsingular matrices $\mathbf{T}_1^s$ and $\mathbf{T}_2^s$, respectively, such that

$$\{\mathbf{A}^s, \mathbf{B}^s, \mathbf{C}^s, \mathbf{D}\} = \{\ (\mathbf{T}^s)^{-1}\ \mathbf{A}\ \mathbf{T}^s,\ (\mathbf{T}^s)^{-1}\ \mathbf{B},\ \mathbf{C}\ \mathbf{T}^s,\ \mathbf{D}\ \} \qquad (2.14)$$

is structurally balanced.

Note that the pseudo-, quasi-, and structurally balanced realizations are equivalent to the original state-space representation in the sense that they do not change the transfer-function matrix of the original system, i.e.,

$$\begin{aligned}
\mathbf{H}(z_1, z_2) &= \mathbf{C}\ [\ \mathbf{I}(z_1, z_2) - \mathbf{A}\ ]^{-1}\ \mathbf{B} + \mathbf{D} \\
&= \mathbf{C}^p\ [\ \mathbf{I}(z_1, z_2) - \mathbf{A}^p\ ]^{-1}\ \mathbf{B}^p + \mathbf{D} \\
&= \mathbf{C}^q\ [\ \mathbf{I}(z_1, z_2) - \mathbf{A}^q\ ]^{-1}\ \mathbf{B}^q + \mathbf{D} \\
&= \mathbf{C}^s\ [\ \mathbf{I}(z_1, z_2) - \mathbf{A}^s\ ]^{-1}\ \mathbf{B}^s + \mathbf{D}
\end{aligned}$$

2.3 Quasi-Gramians

To the author's knowledge, the question of whether a unique positive definite solution exists for (2.8a-d) has not been addressed in the literature. In what follows, an easy-to-apply sufficient condition for the existence of a solution of (2.8a-d) is given.

2.3.1 Existence of Quasi-Gramians

Theorem 2.1

If a 2-D system represented by $\{\mathbf{A},\ \mathbf{B},\ \mathbf{C},\ \mathbf{D}\}$ is Q-stable and both $(\mathbf{A}_1,$

$\mathbf{B}_1$) and ($\mathbf{A}_4$, $\mathbf{B}_2$) are controllable, then equations in (2.8a, b) have a unique positive definite solution $\mathbf{P}^q = \mathbf{P}_1^q \oplus \mathbf{P}_2^q$.

Proof

For the sake of simplicity, it is assumed that

$$\| \mathbf{A} \| = \delta < 1, \qquad \mathbf{B}_1\mathbf{B}_1^T = \mathbf{I}, \qquad \mathbf{B}_2\mathbf{B}_2^T = \mathbf{I}$$

Then, (2.8a, b) can be written as

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \end{bmatrix} \begin{bmatrix} \mathbf{P}_1^q & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_2^q \end{bmatrix} \begin{bmatrix} \mathbf{A}_1^T \\ \mathbf{A}_2^T \end{bmatrix} - \mathbf{P}_1^q = -\mathbf{I} \tag{2.15a}$$

$$\begin{bmatrix} \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \begin{bmatrix} \mathbf{P}_1^q & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_2^q \end{bmatrix} \begin{bmatrix} \mathbf{A}_3^T \\ \mathbf{A}_4^T \end{bmatrix} - \mathbf{P}_2^q = -\mathbf{I} \tag{2.15b}$$

respectively. Matrix sequences $\{\mathbf{P}_1^{q(k)}\}$ and $\{\mathbf{P}_2^{q(k)}\}$ can be constructed by the iterative formulas[5]

$$\mathbf{P}_1^{q(k)} = \mathbf{I} + \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \end{bmatrix} \begin{bmatrix} \mathbf{P}_1^{q(k-1)} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_2^{q(k-1)} \end{bmatrix} \begin{bmatrix} \mathbf{A}_1^T \\ \mathbf{A}_2^T \end{bmatrix} \tag{2.16a}$$

$$\mathbf{P}_2^{q(k)} = \mathbf{I} + \begin{bmatrix} \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \begin{bmatrix} \mathbf{P}_1^{q(k-1)} & \mathbf{0} \\ \mathbf{0} & \mathbf{P}_2^{q(k-1)} \end{bmatrix} \begin{bmatrix} \mathbf{A}_3^T \\ \mathbf{A}_4^T \end{bmatrix} \tag{2.16b}$$

for $k = 1, 2, \ldots$ with initial conditions

$$\mathbf{P}_1^{q(0)} = \mathbf{0} \quad \text{and} \quad \mathbf{P}_2^{q(0)} = \mathbf{0}$$

Note that $\mathbf{P}_1^{q(k)}$ and $\mathbf{P}_2^{q(k)}$ can be expressed as

$$\mathbf{P}_1^{q(k)} = \sum_{l=0}^{k} \mathcal{N}_1^{(l)} \quad \text{and} \quad \mathbf{P}_2^{q(k)} = \sum_{l=0}^{k} \mathcal{N}_2^{(l)} \tag{2.17}$$

[5]The italic superscript (k) denotes iteration k in an iterative formula.

where

$$\mathcal{N}_1^{(l)} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \end{bmatrix} \begin{bmatrix} \mathcal{N}_1^{(l-1)} & \mathbf{0} \\ \mathbf{0} & \mathcal{N}_2^{(l-1)} \end{bmatrix} \begin{bmatrix} \mathbf{A}_1^T \\ \mathbf{A}_2^T \end{bmatrix} \tag{2.18a}$$

$$\mathcal{N}_2^{(l)} = \begin{bmatrix} \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \begin{bmatrix} \mathcal{N}_1^{(l-1)} & \mathbf{0} \\ \mathbf{0} & \mathcal{N}_2^{(l-1)} \end{bmatrix} \begin{bmatrix} \mathbf{A}_3^T \\ \mathbf{A}_4^T \end{bmatrix} \tag{2.18b}$$

for $l = 1, 2, \ldots$ with initial conditions

$$\mathcal{N}_1^{(0)} = \mathbf{0} \quad \text{and} \quad \mathcal{N}_2^{(0)} = \mathbf{0}$$

Since

$$\left\| \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \end{bmatrix} \right\| \leq \| \mathbf{A} \| = \delta < 1$$

$$\left\| \begin{bmatrix} \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \right\| \leq \| \mathbf{A} \| = \delta < 1$$

it follows that each term $\mathcal{N}_1^{(l)}$ and $\mathcal{N}_2^{(l)}$ of (2.18a, b) is positive semi-definite, and

$$\left\| \mathcal{N}_1^{(l)} \right\| \leq \delta^{2l} \quad \text{and} \quad \left\| \mathcal{N}_2^{(l)} \right\| \leq \delta^{2l}$$

Hence,

$$\left\| \mathbf{P}_1^{q(k)} \right\| \leq \frac{1}{1-\delta^2} \quad \text{and} \quad \left\| \mathbf{P}_2^{q(k)} \right\| \leq \frac{1}{1-\delta^2} \tag{2.19}$$

and the corresponding terms $\mathbf{P}_1^{q(k)}$ and $\mathbf{P}_2^{q(k)}$ are positive definite. Furthermore, the two sequences are monotonically increasing, that is,

$$\mathbf{P}_1^{q(0)} < \mathbf{P}_1^{q(1)} \leq \cdots \leq \mathbf{P}_1^{q(k)} \leq \cdots$$

$$\mathbf{P}_2^{q(0)} < \mathbf{P}_2^{q(1)} \leq \cdots \leq \mathbf{P}_2^{q(k)} \leq \cdots$$

and bounded because of (2.19). Finally, from (2.16a, b) and (2.17), the limits of the above two sequences, which are given by

$$\mathbf{P}_1^q = \sum_{l=0} \mathcal{N}_1^{(l)} \quad \text{and} \quad \mathbf{P}_2^q = \sum_{l=0} \mathcal{N}_2^{(l)}$$

satisfy (2.8a, b). □

2.3.2 Algorithm for Obtaining a Quasi-Balanced Realization

Based on the proof of Theorem 2.1, the following new algorithm for obtaining a QBR is proposed.

Algorithm 2.2

Step 1: Set $\mathbf{P}_2^{q(0)} = \mathbf{Q}_2^{q(0)} = \mathbf{0}$ and $k = 1$.

Step 2: Solve the 1-D Lyapunov equations

$$\mathbf{A}_1 \mathbf{P}_1^{q(k)} \mathbf{A}_1^T - \mathbf{P}_1^{q(k)} + \mathcal{B}_1 = \mathbf{0} \tag{2.20a}$$

$$\mathbf{A}_1^T \mathbf{Q}_1^{q(k)} \mathbf{A}_1 - \mathbf{Q}_1^{q(k)} + \mathcal{C}_1 = \mathbf{0} \tag{2.20b}$$

for $\mathbf{P}_1^{q(k)}$ and $\mathbf{Q}_1^{q(k)}$, where

$$\mathcal{B}_1 = \mathbf{B}_1 \mathbf{B}_1^T + \mathbf{A}_2 \mathbf{P}_2^{q(k-1)} \mathbf{A}_2^T$$

$$\mathcal{C}_1 = \mathbf{C}_1^T \mathbf{C}_1 + \mathbf{A}_3^T \mathbf{Q}_2^{q(k-1)} \mathbf{A}_3$$

Step 3: Solve the 1-D Lyapunov equations

$$\mathbf{A}_4 \mathbf{P}_2^{q(k)} \mathbf{A}_4^T - \mathbf{P}_2^{q(k)} + \mathcal{B}_2 = \mathbf{0} \tag{2.21a}$$

$$\mathbf{A}_4^T \mathbf{Q}_2^{q(k)} \mathbf{A}_4 - \mathbf{Q}_2^{q(k)} + \mathcal{C}_2 = \mathbf{0} \tag{2.21b}$$

for $\mathbf{P}_2^{q(k)}$ and $\mathbf{Q}_2^{q(k)}$, where

$$\mathcal{B}_2 = \mathbf{B}_2\mathbf{B}_2^T + \mathbf{A}_3\mathbf{P}_1^{q(k)}\mathbf{A}_3^T$$

$$\mathcal{C}_2 = \mathbf{C}_2^T\mathbf{C}_2 + \mathbf{A}_2^T\mathbf{Q}_1^{q(k)}\mathbf{A}_2$$

Step 4: Set $k = k+1$ and repeat Steps 2 and 3 until

$$\left\| \mathbf{P}_l^{q(k)} - \mathbf{P}_l^{q(k-1)} \right\| < \epsilon \quad \text{for} \quad l = 1, 2$$

$$\left\| \mathbf{Q}_l^{q(k)} - \mathbf{Q}_l^{q(k-1)} \right\| < \epsilon \quad \text{for} \quad l = 1, 2$$

where ϵ is a prescribed tolerance. Then set

$$\mathbf{P}_1^q = \mathbf{P}_1^{q(k)}, \qquad \mathbf{P}_2^q = \mathbf{P}_2^{q(k)}$$

$$\mathbf{Q}_1^q = \mathbf{Q}_1^{q(k)}, \qquad \mathbf{Q}_2^q = \mathbf{Q}_2^{q(k)}$$

Step 5: Apply Algorithm 2.1 to $\{\mathbf{P}_1^q,\ \mathbf{Q}_1^q\}$ and $\{\mathbf{P}_2^q,\ \mathbf{Q}_2^q\}$ to find nonsingular matrices $\mathbf{T}_1^q$ and $\mathbf{T}_2^q$, respectively. Then, construct the balancing transformation matrix

$$\mathbf{T}^q = \mathbf{T}_1^q \oplus \mathbf{T}_2^q$$

Step 6: Obtain a QBR $\{\mathbf{A}^q, \mathbf{B}^q, \mathbf{C}^q, \mathbf{D}\}$ as

$$\{\mathbf{A}^q, \mathbf{B}^q, \mathbf{C}^q, \mathbf{D}\} = \{\, (\mathbf{T}^q)^{-1}\,\mathbf{A}\,\mathbf{T}^q,\ (\mathbf{T}^q)^{-1}\,\mathbf{B},\ \mathbf{C}\,\mathbf{T}^q,\ \mathbf{D}\,\}$$

2.4 Structured Gramians

As was mentioned in Section 2.2.3, the structured gramians, denoted by $\mathbf{P}^s$ and $\mathbf{Q}^s$, can be obtained by solving the *constrained* minimization problems in

(2.13a, b). In this section, we show that these problems can be reformulated as *unconstrained* minimization problems.

2.4.1 Problem Formulation

Since the structured gramians are positive definite matrices, the Cholesky factorization can be used to decompose $\mathbf{P}^s$ and $\mathbf{Q}^s$ as

$$\mathbf{P}^s = \mathbf{L}_{\mathrm{P}}^s \, \mathbf{L}_{\mathrm{P}}^{s\,T} \quad \text{and} \quad \mathbf{Q}^s = \mathbf{L}_{\mathrm{Q}}^s \, \mathbf{L}_{\mathrm{Q}}^{s\,T} \tag{2.22}$$

respectively, where

$$\mathbf{L}_{\mathrm{P}}^s = \mathbf{L}_{\mathrm{P}1}^s \oplus \mathbf{L}_{\mathrm{P}2}^s \quad \text{and} \quad \mathbf{L}_{\mathrm{Q}}^s = \mathbf{L}_{\mathrm{Q}1}^s \oplus \mathbf{L}_{\mathrm{Q}2}^s$$

are block-diagonal lower-triangular matrices. Hence, the inequalities in (2.11a, b) become

$$\begin{bmatrix} (\mathbf{L}_{\mathrm{P}}^s)^{-1}\mathbf{A}\,\mathbf{L}_{\mathrm{P}}^s & (\mathbf{L}_{\mathrm{P}}^s)^{-1}\mathbf{B} \end{bmatrix} \begin{bmatrix} (\mathbf{L}_{\mathrm{P}}^s)^{-1}\mathbf{A}\,\mathbf{L}_{\mathrm{P}}^s & (\mathbf{L}_{\mathrm{P}}^s)^{-1}\mathbf{B} \end{bmatrix}^T < \mathbf{I}$$

$$\begin{bmatrix} (\mathbf{L}_{\mathrm{Q}}^s)^{-1}\mathbf{A}^T\mathbf{L}_{\mathrm{Q}}^s & (\mathbf{L}_{\mathrm{Q}}^s)^{-1}\mathbf{C}^T \end{bmatrix} \begin{bmatrix} (\mathbf{L}_{\mathrm{Q}}^s)^{-1}\mathbf{A}^T\mathbf{L}_{\mathrm{Q}}^s & (\mathbf{L}_{\mathrm{Q}}^s)^{-1}\mathbf{C}^T \end{bmatrix}^T < \mathbf{I}$$

Hence, instead of solving the *constrained* minimization problems in (2.13a, b), the *unconstrained* optimization problems

$$\underset{\mathbf{L}_{\mathrm{P}}}{\text{minimize}} \left\| (\mathbf{L}_{\mathrm{P}}^s)^{-1} \begin{bmatrix} \mathbf{A}\mathbf{L}_{\mathrm{P}}^s & \mathbf{B} \end{bmatrix} \right\| \tag{2.23a}$$

$$\underset{\mathbf{L}_{\mathrm{Q}}}{\text{minimize}} \left\| (\mathbf{L}_{\mathrm{Q}}^s)^{-1} \begin{bmatrix} \mathbf{A}^T\mathbf{L}_{\mathrm{Q}}^s & \mathbf{C}^T \end{bmatrix} \right\| \tag{2.23b}$$

can now be considered. Obviously, the local minimum points obtained from the minimization problems in (2.23a, b) are acceptable if and only if

$$\left\| (\mathbf{L}_{\mathrm{P}}^{s})^{-1} \left[\begin{array}{cc} \mathbf{A}\mathbf{L}_{\mathrm{P}}^{s} & \mathbf{B} \end{array} \right] \right\| < 1 \qquad (2.24a)$$

$$\left\| (\mathbf{L}_{\mathrm{Q}}^{s})^{-1} \left[\begin{array}{cc} \mathbf{A}^{T}\mathbf{L}_{\mathrm{Q}}^{s} & \mathbf{C}^{T} \end{array} \right] \right\| < 1 \qquad (2.24b)$$

This is because only then matrices $\mathbf{P}^{s}$ and $\mathbf{Q}^{s}$ formed using (2.22) will satisfy the 2-D Lyapunov inequalities in (2.11a, b).

Determination of Initial Points

Even for a 2-D system of modest order, for example $m = n = 15$, the number of parameters involved in each of the above minimization problems, i.e.,

$$[\, m(m+1) + n(n+1) \,]/2 = 240$$

is quite large. From a computational view point it would, therefore, be beneficial to use a "good" initial point before beginning minimizing the norms in (2.23a, b).

Consider the 2-D Lyapunov-like equations in (2.8a, b) which are linear in $\mathbf{P}_1^{q}$ and $\mathbf{P}_2^{q}$. From Lemma 2.3.1, if a 2-D discrete system $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ is Q-stable and both $(\mathbf{A}_1, \mathbf{B}_1)$ and $(\mathbf{A}_4, \mathbf{B}_2)$ are controllable, then there exist two unique positive definite matrices $\mathbf{P}_1^{q}$ and $\mathbf{P}_2^{q}$ that satisfy (2.8a, b). Having obtained $\mathbf{P}_1^{q}$ and $\mathbf{P}_2^{q}$, the Cholesky decompositions can be applied to yield

$$\mathbf{P}_1^{q} = \mathbf{L}_{\mathrm{P}1}^{q}\, \mathbf{L}_{\mathrm{P}1}^{q\,T} \quad \text{and} \quad \mathbf{P}_2^{q} = \mathbf{L}_{\mathrm{P}2}^{q}\, \mathbf{L}_{\mathrm{P}2}^{q\,T}$$

and the relations in (2.8a, b) imply that

$$\left\| (\mathbf{L}^q_{P1})^{-1} \begin{bmatrix} \mathbf{A}_1\mathbf{L}^q_{P1} & \mathbf{A}_2\mathbf{L}^q_{P2} & \mathbf{B}_1 \end{bmatrix} \right\| = 1$$

$$\left\| (\mathbf{L}^q_{P2})^{-1} \begin{bmatrix} \mathbf{A}_3\mathbf{L}^q_{P1} & \mathbf{A}_4\mathbf{L}^q_{P2} & \mathbf{B}_2 \end{bmatrix} \right\| = 1$$

Hence

$$\left\| (\mathbf{L}^q_{P})^{-1} \begin{bmatrix} \mathbf{A}\mathbf{L}^q_{P} & \mathbf{B} \end{bmatrix} \right\| \leq 2 \tag{2.25}$$

where

$$\mathbf{L}^q_P = \mathbf{L}^q_{P1} \oplus \mathbf{L}^q_{P2}$$

The matrix $\mathbf{L}^q_P$ obtained from (2.25) provides a "good" initial point for $\mathbf{L}^s_P$ in the minimization problems in (2.23a). A good initial point for the problems in (2.23b) can be found in the same manner.

2.4.2 Algorithm for Obtaining a Structurally Balanced Realization

In general, it is difficult to find the local minimum points, $\mathbf{L}^s_P$ and $\mathbf{L}^s_Q$, of the minimization problems in (2.23a, b) such that the inequalities in (2.24a, b) are satisfied. A new algorithm that can alleviate this problem [43] is as follows:

Algorithm 2.3

Step 1: Find $\mathbf{L}^s_P$ and $\mathbf{L}^s_Q$ by solving the minimization problems given by

$$\underset{\mathbf{L}^s_P}{\text{minimize}} \left\| (\mathbf{L}^s_P)^{-1}\mathbf{A}\,\mathbf{L}^s_P \right\| \tag{2.26a}$$

$$\underset{\mathbf{L}_Q^s}{\text{minimize}} \left\| (\mathbf{L}_Q^s)^{-1} \mathbf{A}^T \mathbf{L}_Q^s \right\| \tag{2.26b}$$

The optimization problems are unconstrained and can be carried out using established numerical optimization techniques, such as the quasi-Newton methods [14].

Step 2: Let

$$\mathbf{P}^s = \mathbf{L}_P^s \, \mathbf{L}_P^{s\,T} \quad \text{and} \quad \mathbf{Q}^s = \mathbf{L}_Q^s \, \mathbf{L}_Q^{s\,T}$$

and compute

$$\mathbf{\Phi}_P = \mathbf{P}^s - \mathbf{A}\mathbf{P}^s\mathbf{A}^T$$

$$\mathbf{\Phi}_Q = \mathbf{Q}^s - \mathbf{A}^T\mathbf{Q}^s\mathbf{A}$$

Step 3: Obtain the singular-value decompositions of $\mathbf{\Phi}_P$ and $\mathbf{\Phi}_Q$ as

$$\mathbf{\Phi}_P = \mathbf{U}_P \, \mathbf{S}_P \, \mathbf{U}_P^T = \tilde{\mathbf{U}}_P \, \tilde{\mathbf{U}}_P^T \tag{2.27a}$$

$$\mathbf{\Phi}_Q = \mathbf{U}_Q \, \mathbf{S}_Q \, \mathbf{U}_Q^T = \tilde{\mathbf{U}}_Q \, \tilde{\mathbf{U}}_Q^T \tag{2.27b}$$

where $\mathbf{S}_P$ and $\mathbf{S}_Q$ are diagonal matrices, $\mathbf{U}_P$ and $\mathbf{U}_Q$ are orthogonal matrices, and

$$\tilde{\mathbf{U}}_P = \mathbf{U}_P \, \mathbf{S}_P^{1/2} \quad \text{and} \quad \tilde{\mathbf{U}}_Q = \mathbf{U}_Q \, \mathbf{S}_Q^{1/2}$$

Step 4: Find scaling factors, k_P and k_Q, by finding the largest eigenvalues defined by

$$k_P^2 = \gamma_{\max} \left[(\tilde{\mathbf{U}}_P)^{-1} \, \mathbf{B} \, \mathbf{B}^T \, (\tilde{\mathbf{U}}_P)^{-T} \right] \tag{2.28a}$$

$$k_Q^2 = \gamma_{\max} \left[(\tilde{\mathbf{U}}_Q)^{-1} \, \mathbf{C} \, \mathbf{C}^T \, (\tilde{\mathbf{U}}_Q)^{-T} \right] \tag{2.28b}$$

Step 5: Find a local minimum point $(\mathbf{L}_{\mathrm{P}}^s, \mathbf{L}_{\mathrm{Q}}^s)$ of the minimization problems

$$\underset{\mathbf{L}_{\mathrm{P}}^s}{\text{minimize}} \left\| (\mathbf{L}_{\mathrm{P}}^s)^{-1} \left[\mathbf{A}\, \mathbf{L}_{\mathrm{P}}^s \qquad \frac{1}{k_{\mathrm{P}}} \mathbf{B} \right] \right\| \tag{2.29a}$$

$$\underset{\mathbf{L}_{\mathrm{Q}}^s}{\text{minimize}} \left\| (\mathbf{L}_{\mathrm{Q}}^s)^{-1} \left[\mathbf{A}^T \mathbf{L}_{\mathrm{Q}}^s \qquad \frac{1}{k_{\mathrm{Q}}} \mathbf{C}^T \right] \right\| \tag{2.29b}$$

Step 6: Form structured gramians

$$\mathbf{P}^s = \mathbf{L}_{\mathrm{P}}^s \, \mathbf{L}_{\mathrm{P}}^{s\,T} \quad \text{and} \quad \mathbf{Q}^s = \mathbf{L}_{\mathrm{Q}}^s \, \mathbf{L}_{\mathrm{Q}}^{s\,T}$$

Step 7: Apply Algorithm 2.1 to $\{\mathbf{P}_1^s,\ \mathbf{Q}_1^s\}$ and $\{\mathbf{P}_2^s,\ \mathbf{Q}_2^s\}$ to find nonsingular matrices $\mathbf{T}_1^s$ and $\mathbf{T}_2^s$, respectively, and form the balancing transformation matrix

$$\mathbf{T}^s = \mathbf{T}_1^s \oplus \mathbf{T}_2^s.$$

Step 8: Obtain the SBR $\{\mathbf{A}^s, \mathbf{B}^s, \mathbf{C}^s, \mathbf{D}\}$ as

$$\{\mathbf{A}^s, \mathbf{B}^s, \mathbf{C}^s, \mathbf{D}\} = \{\, (\mathbf{T}^s)^{-1}\, \mathbf{A}\, \mathbf{T}^s,\ (\mathbf{T}^s)^{-1}\, \mathbf{B},\ \mathbf{C}\, \mathbf{T}^s,\ \mathbf{D}\,\}$$

2.5 2-D Balanced Model-Reduction

An important application of the balanced realization methods discussed in Section 2.2 is to reduce the order of a 2-D discrete system. These methods will be referred to as *balanced model-reduction* methods since they are based on the balanced realizations addressed in Section 2.2. The transfer-function matrix of a reduced-order system of order (r_1, r_2) is denoted as $\mathbf{H}^r(z_1, z_2)$,

and the approximation error introduced by the reduction in the L_∞-norm is defined by

$$\begin{aligned} e_\infty &= \| \mathbf{H}(z_1, z_2) - \mathbf{H}^r(z_1, z_2) \|_\infty \\ &= \max_{\substack{0 \le \omega_1 < 1 \\ 0 \le \omega_2 < 1}} \left| \mathbf{H}(e^{j2\pi\omega_1}, e^{j2\pi\omega_2}) - \mathbf{H}^r(e^{j2\pi\omega_1}, e^{j2\pi\omega_2}) \right| \end{aligned} \tag{2.30}$$

where $\omega_1 \in [0, 1]$ and $\omega_2 \in [0, 1]$ denote the normalized frequencies.

Let $\{\mathbf{A}^p, \mathbf{B}^p, \mathbf{C}^p, \mathbf{D}\}$ be the PBR of the original system. To find a reduced-order system, we partition matrices in the set $\{\mathbf{A}^p, \mathbf{B}^p, \mathbf{C}^p, \mathbf{D}\}$ as

$$\mathbf{A}^p = \left[\begin{array}{cc|cc} \mathbf{A}_1^{pr} & \mathbf{A}_{12}^p & \mathbf{A}_2^{pr} & \mathbf{A}_{22}^p \\ \mathbf{A}_{13}^p & \mathbf{A}_{14}^p & \mathbf{A}_{23}^p & \mathbf{A}_{24}^p \\ \hline \mathbf{A}_3^{pr} & \mathbf{A}_{32}^p & \mathbf{A}_4^{pr} & \mathbf{A}_{42}^p \\ \mathbf{A}_{33}^p & \mathbf{A}_{34}^p & \mathbf{A}_{43}^p & \mathbf{A}_{44}^p \end{array}\right], \quad \mathbf{B}^p = \left[\begin{array}{c} \mathbf{B}_1^{pr} \\ \mathbf{B}_{12}^p \\ \hline \mathbf{B}_2^{pr} \\ \mathbf{B}_{22}^p \end{array}\right], \quad \mathbf{C}^p = \left[\begin{array}{c} \mathbf{C}_1^{prT} \\ \mathbf{C}_{12}^{p\,T} \\ \hline \mathbf{C}_2^{prT} \\ \mathbf{C}_{22}^{p\,T} \end{array}\right]^T$$

with

$$\mathbf{A}_1^{pr} \in \Re^{r_1 \times r_1}, \quad \mathbf{A}_2^{pr} \in \Re^{r_1 \times r_2}, \quad \mathbf{B}_1^{pr} \in \Re^{r_1 \times t}, \quad \mathbf{C}_1^{pr} \in \Re^{s \times r_1}$$

$$\mathbf{A}_3^{pr} \in \Re^{r_2 \times r_1}, \quad \mathbf{A}_4^{pr} \in \Re^{r_2 \times r_2}, \quad \mathbf{B}_2^{pr} \in \Re^{r_2 \times t}, \quad \mathbf{C}_2^{pr} \in \Re^{s \times r_2}$$

and form the reduced-order system of order (r_1, r_2) as

$$\mathbf{A}^{pr} = \begin{bmatrix} \mathbf{A}_1^{pr} & \mathbf{A}_2^{pr} \\ \mathbf{A}_3^{pr} & \mathbf{A}_4^{pr} \end{bmatrix}, \quad \mathbf{B}^{pr} = \begin{bmatrix} \mathbf{B}_1^{pr} \\ \mathbf{B}_2^{pr} \end{bmatrix}, \quad \mathbf{C}^{pr} = \begin{bmatrix} \mathbf{C}_1^{prT} \\ \mathbf{C}_2^{prT} \end{bmatrix}^T$$

The transfer-function matrix of the reduced-order system is given by

$$\mathbf{H}^r(z_1, z_2) = \mathbf{C}^{pr} \left[\mathbf{I}(z_1, z_2) - \mathbf{A}^{pr} \right]^{-1} \mathbf{B}^{pr} + \mathbf{D} \tag{2.31}$$

This method for obtaining a reduced-order system is called a *pseudo-balanced model-reduction* (PBMR) method. Similarly, QBR and SBR can be used to

obtain *quasi-* and *structurally balanced model-reduction* (QBMR and SBMR) methods, respectively. The corresponding reduced-order systems are represented by $\{\mathbf{A}^{qr}, \mathbf{B}^{qr}, \mathbf{C}^{qr}, \mathbf{D}\}$ and $\{\mathbf{A}^{sr}, \mathbf{B}^{sr}, \mathbf{C}^{sr}, \mathbf{D}\}$, respectively.

Although it has been shown in [56] that the SBMR defined above always leads to stable reduced-order systems, the approximation error is often unsatisfactory when the structured gramians are computed using Algorithm 2.3. This will be demonstrated in Section 2.5.2 through two examples. It has also been shown in [56] that the approximation error introduced by the SBR is bounded by twice the sum of the discarded structured Hankel singular values

$$\left\{\sigma^s_{r_1+1},\ \sigma^s_{r_1+2},\ \ldots,\ \sigma^s_m,\ \mu^s_{r_2+1},\ \mu^s_{r_2+2},\ \ldots,\ \mu^s_n\right\}$$

For 1-D discrete systems, the Hankel singular values are defined as the square roots of the eigenvalues of $\mathbf{PQ}$, where $\mathbf{P}$ and $\mathbf{Q}$ denote the controllability and observability gramians of the 1-D system. It is known that the Hankel singular values are invariant under state-variable transformation. Similarly, for a 2-D discrete system, the *structured Hankel singular values* can be defined as the square roots of the eigenvalues of $\mathbf{P}^s\mathbf{Q}^s$, where $\mathbf{P}^s$ and $\mathbf{Q}^s$ are the structured controllability and observability gramians of the 2-D system. It can be readily verified that the non-negative scalars

$$\{\sigma^s_1,\ \sigma^s_2,\ \ldots,\ \sigma^s_m,\ \mu^s_1,\ \mu^s_2,\ \ldots,\ \mu^s_n\}$$

in (2.12a, b) are indeed such defined structured Hankel singular values.

It is important to note that the structured Hankel singular values are $\mathbf{P}^s$ and $\mathbf{Q}^s$ dependent. $\mathbf{P}^s$ and $\mathbf{Q}^s$, which are solutions of the 2-D Lyapunov

inequalities in (2.11a, b), are not unique. In fact, any $\alpha\mathbf{P}^s$ and $\beta\mathbf{Q}^s$ with $\alpha \geq 1$ and $\beta \geq 1$ still satisfy (2.11a, b). Consequently, a certain degree of freedom is available which can be used to improve Algorithm 2.3. In other word, this dependence provides an approach to obtain a suitable solution of (2.11a, b) so as to achieve small approximation error. In what follows, a new improved algorithm for computing structured gramians is developed.

2.5.1 Improved Structurally Balanced Model Reduction

Instead of solving the constrained optimization problems in (2.13a, b) or the unconstrained optimization problems in (2.23a, b), we consider the minimization problem

$$\underset{\mathbf{L}_{\mathrm{P}}^s, \mathbf{L}_{\mathrm{Q}}^s}{\text{minimize}}\ f(\mathbf{L}_{\mathrm{P}}^s, \mathbf{L}_{\mathrm{Q}}^s) \tag{2.32}$$

where

$$f(\mathbf{L}_{\mathrm{P}}^s, \mathbf{L}_{\mathrm{Q}}^s) = \underbrace{k_0\, f_1(\mathbf{L}_{\mathrm{P}}^s) + k_1\, f_2(\mathbf{L}_{\mathrm{Q}}^s)}_{\text{stability control}} + \underbrace{k_2\, f_e(\mathbf{L}_{\mathrm{P}}^s, \mathbf{L}_{\mathrm{Q}}^s)}_{\text{error control}}$$

$$f_1(\mathbf{L}_{\mathrm{P}}^s) = \left\| (\mathbf{L}_{\mathrm{P}}^s)^{-1} \begin{bmatrix} \mathbf{A}\mathbf{L}_{\mathrm{P}}^s & \mathbf{B} \end{bmatrix} \right\|$$

$$f_2(\mathbf{L}_{\mathrm{Q}}^s) = \left\| (\mathbf{L}_{\mathrm{Q}}^s)^{-1} \begin{bmatrix} \mathbf{A}^T\mathbf{L}_{\mathrm{Q}}^s & \mathbf{C}^T \end{bmatrix} \right\|$$

$$f_e(\mathbf{L}_{\mathrm{P}}^s, \mathbf{L}_{\mathrm{Q}}^s) = \sum_{k=r_1+1}^{m} \sigma_k^s + \sum_{l=r_2+1}^{n} \mu_l^s$$

with scalars k_0, k_1, $k_2 > 0$, which can be appropriately adjusted such that $f_2(\mathbf{L}_{\mathrm{P}}^s)$ and $f_2(\mathbf{L}_{\mathrm{Q}}^s)$ are as close to unity as possible and the inequalities

$$f_1(\mathbf{L}_{\mathrm{P}}^s) < 1 \quad \text{and} \quad f_2(\mathbf{L}_{\mathrm{Q}}^s) < 1 \tag{2.33}$$

are satisfied. Therefore, the Q-stability of the reduced-order system is guaranteed and the approximation error is expected to be minimized as well. The following is a step-by-step summary of the proposed algorithm which results in ISBR and ISBMR methods

Algorithm 2.4

Step 1: Find a local minimum point $(\mathbf{L}_{\mathrm{P}}^s, \mathbf{L}_{\mathrm{Q}}^s)$ by solving the minimization problem in (2.32). The optimization problem is unconstrained and can be carried out using established numerical optimization techniques [14]. If necessary, the constants k_0, k_1 and k_2 can be adjusted to satisfy (2.33).

Step 2: Compute the structured gramians

$$\mathbf{P}^s = \mathbf{L}_{\mathrm{P}}^s \mathbf{L}_{\mathrm{P}}^{s\,T}, \qquad \mathbf{Q}^s = \mathbf{L}_{\mathrm{Q}}^s \mathbf{L}_{\mathrm{Q}}^{s\,T} \tag{2.34}$$

Step 3: Apply Algorithm 2.1 to $\{\mathbf{P}_1^s, \mathbf{Q}_1^s\}$ and $\{\mathbf{P}_2^s, \mathbf{Q}_2^s\}$ to find nonsingular matrices $\mathbf{T}_1^s$ and $\mathbf{T}_2^s$, respectively, and form the balancing transformation matrix

$$\mathbf{T}^s = \mathbf{T}_1^s \oplus \mathbf{T}_2^s$$

Step 4: Obtain the ISBR $\{\mathbf{A}^s, \mathbf{B}^s, \mathbf{C}^s, \mathbf{D}\}$ as

$$\{\mathbf{A}^s, \mathbf{B}^s, \mathbf{C}^s, \mathbf{D}\} = \{(\mathbf{T}^s)^{-1}\mathbf{A}\,\mathbf{T}^s, (\mathbf{T}^s)^{-1}\mathbf{B}, \mathbf{C}'\mathbf{T}^s, \mathbf{D}\}$$

Step 5: Partition matrices $\{\mathbf{A}^s, \mathbf{B}^s, \mathbf{C}^s, \mathbf{D}\}$ as

$$\mathbf{A}^s = \left[\begin{array}{cc|cc} \mathbf{A}^{sr}_1 & \mathbf{A}^s_{12} & \mathbf{A}^{sr}_2 & \mathbf{A}^s_{22} \\ \mathbf{A}^s_{13} & \mathbf{A}^s_{14} & \mathbf{A}^s_{23} & \mathbf{A}^s_{24} \\ \hline \mathbf{A}^{sr}_3 & \mathbf{A}^s_{32} & \mathbf{A}^{sr}_4 & \mathbf{A}^s_{42} \\ \mathbf{A}^s_{33} & \mathbf{A}^s_{34} & \mathbf{A}^s_{43} & \mathbf{A}^s_{44} \end{array}\right], \quad \mathbf{B}^s = \left[\begin{array}{c} \mathbf{B}^{sr}_1 \\ \mathbf{B}^s_{12} \\ \hline \mathbf{B}^{sr}_2 \\ \mathbf{B}^s_{22} \end{array}\right], \quad \mathbf{C}^s = \left[\begin{array}{c} \mathbf{C}^{srT}_1 \\ \mathbf{C}^{s\,T}_{12} \\ \hline \mathbf{C}^{srT}_2 \\ \mathbf{C}^{s\,T}_{22} \end{array}\right]^T$$

where

$$\mathbf{A}^{sr}_1 \in \Re^{r_1 \times r_1}, \quad \mathbf{A}^{sr}_2 \in \Re^{r_1 \times r_2}, \quad \mathbf{B}^{sr}_1 \in \Re^{r_1 \times t}, \quad \mathbf{C}^{sr}_1 \in \Re^{s \times r_1}$$

$$\mathbf{A}^{sr}_3 \in \Re^{r_2 \times r_1}, \quad \mathbf{A}^{sr}_4 \in \Re^{r_2 \times r_2}, \quad \mathbf{B}^{sr}_2 \in \Re^{r_2 \times t}, \quad \mathbf{C}^{sr}_2 \in \Re^{s \times r_2}$$

and form the reduced-order system of order (r_1, r_2) as

$$\mathbf{A}^{sr} = \begin{bmatrix} \mathbf{A}^{sr}_1 & \mathbf{A}^{sr}_2 \\ \mathbf{A}^{sr}_3 & \mathbf{A}^{sr}_4 \end{bmatrix}, \quad \mathbf{B}^{sr} = \begin{bmatrix} \mathbf{B}^{sr}_1 \\ \mathbf{B}^{sr}_2 \end{bmatrix}, \quad \mathbf{C}^{sr} = \begin{bmatrix} \mathbf{C}^{srT}_1 \\ \mathbf{C}^{srT}_2 \end{bmatrix}^T$$

The transfer-function matrix of the reduced-order system is given by

$$\mathbf{H}^r(z_1, z_2) = \mathbf{C}^{sr} \left[\mathbf{I}(z_1, z_2) - \mathbf{A}^{sr} \right]^{-1} \mathbf{B}^{sr} + \mathbf{D} \tag{2.35}$$

2.5.2 Performance Evaluation of Balanced Model Reduction Methods

In this section, the performance of reduced-order systems obtained using the pseudo-, quasi-, and structurally balanced model-reduction methods is assessed. Two 2-D systems are examined. The first example involves a system of order (2, 2). The second example is a lowpass filter of order (4, 8). In both examples, the pseudo-, quasi-, and structured gramians have been

computed for the 2-D systems by applying equations (2.3a, b), Algorithm 2.2, and Algorithms 2.3 and 2.4, respectively.

Example 2.1

Consider the 2-D discrete system of order (2, 2) given in [53], which is pseudo-balanced. It was shown in [53] that the reduced-order system obtained using the PBMR method is unstable. The system is represented by the Roesser state-space model in (2.1a, b) with

$$\mathbf{A} = \begin{bmatrix} 0.9399204 & 0.0682773 & -0.1221131 & 0.2383674 \\ -0.0233407 & 0.8600796 & 0.0713971 & -0.1393688 \\ 0.4183718 & 0.2402084 & 0.9874241 & 0.0672975 \\ -0.1313337 & 0.3928846 & -0.1135699 & 0.8125759 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} -0.3142416 & -0.3879902 & -0.0303866 & -0.0008281 \end{bmatrix}^T$$

$$\mathbf{C} = \begin{bmatrix} -0.3528145 & 0.2728651 & -0.1139450 & 0.1961035 \end{bmatrix}$$

$$\mathbf{D} = 0$$

The state-space model of the reduced-order systems of order (2, 1) obtained using PBMR, QBMR, SBMR, and ISBMR, denoted by $\{\mathbf{A}^{pr}, \mathbf{B}^{pr}, \mathbf{C}^{pr}, \mathbf{D}\}$, $\{\mathbf{A}^{qr}, \mathbf{B}^{qr}, \mathbf{C}^{qr}, \mathbf{D}\}$, $\{\mathbf{A}^{sr}, \mathbf{B}^{sr}, \mathbf{C}^{sr}, \mathbf{D}\}$, and $\{\tilde{\mathbf{A}}^{sr}, \tilde{\mathbf{B}}^{sr}, \tilde{\mathbf{C}}^{sr}, \mathbf{D}\}$, respectively, are given by (2.1a, b) with

$$\mathbf{A}^{pr} = \begin{bmatrix} 0.9399204 & 0.0682773 & -0.1221131 \\ -0.0233407 & 0.8600796 & 0.0713971 \\ 0.4183718 & 0.2402084 & 0.9874241 \end{bmatrix}$$

$$\mathbf{B}^{pr} = \begin{bmatrix} -0.3142416 & -0.3879902 & -0.0303866 \end{bmatrix}^T$$

$$\mathbf{C}^{pr} = \begin{bmatrix} -0.3528145 & 0.2728651 & -0.1139450 \end{bmatrix}$$

$$\mathbf{A}^{qr} = \begin{bmatrix} 0.9266884 & 0.0418817 & -0.2615219 \\ -0.0170068 & 0.8733116 & 0.1666507 \\ 0.3130809 & 0.0063373 & 0.9514318 \end{bmatrix}$$

$$\mathbf{B}^{qr} = \begin{bmatrix} -0.4836842 & -0.4330236 & -0.0224213 \end{bmatrix}^T$$

$$\mathbf{C}^{qr} = \begin{bmatrix} -0.3806516 & 0.4136384 & -0.2872360 \end{bmatrix}$$

$$\mathbf{A}^{sr} = \begin{bmatrix} 0.9001254 & 0.0026952 & -0.3484015 \\ -0.0000064 & 0.8998746 & 0.0162201 \\ 0.3413570 & -0.0298503 & 0.9015208 \end{bmatrix}$$

$$\mathbf{B}^{sr} = \begin{bmatrix} -0.3839145 & -3.4950792 & -0.0050412 \end{bmatrix}^T$$

$$\mathbf{C}^{sr} = \begin{bmatrix} -2.1016847 & 0.2294275 & -1.2271479 \end{bmatrix}$$

$$\tilde{\mathbf{A}}^{sr} = \begin{bmatrix} 0.9042282 & 0.0013707 & 0.0132593 \\ -0.0130439 & 0.8957718 & -0.0409043 \\ -0.1467671 & -0.0153027 & 0.9068744 \end{bmatrix}$$

$$\tilde{\mathbf{B}}^{sr} = \begin{bmatrix} 1.4433097 & -0.2014683 & -0.0212982 \end{bmatrix}^T$$

$$\tilde{\mathbf{C}}^{sr} = \begin{bmatrix} -0.1590501 & -1.1642451 & -0.0900774 \end{bmatrix}$$

The performance of the reduced-order systems of order (2, 1) obtained using PBMR, QBMR, SBMR, and ISBMR is summarized in Table 2.1.

Table 2.1: Performance of the reduced-order system of order (2, 1).

Method $r_1=2, r_2=1$	Stability	Flops	Error e_∞
PBMR	unstable	2.8348×10^5	9.2272
QBMR	stable	6.5677×10^4	3.8951
SBMR	stable	2.1946×10^6	3.4502
ISBMR*	stable	1.7234×10^7	2.6947

ISBMR*: $k_0 = 1$, $k_1 = 1$, and $k_2 = 8 \times 10^{-5}$.

Example 2.2

Consider a 2-D lowpass filter of order (4, 8) used in [62]. The Roesser state-space model of the filter is given by (2.1a, b) with

$$\mathbf{A}_1 = \begin{bmatrix} 0.537000 & -0.068810 & 0.985510 & 0.503880 \\ 1.000000 & 0 & 0 & 0 \\ 0 & 0 & 0.538819 & -0.066576 \\ 0 & 0 & 1.000000 & 0 \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A}_3 = \begin{bmatrix} -0.390721 & 0.244967 & -0.483603 & -0.247260 \\ 0.251223 & -0.145125 & 0.026974 & 0.013792 \\ 1.270478 & 1.106765 & 0.198140 & 0.101307 \\ 1.796448 & 0.421991 & 0.592117 & 0.302742 \\ 0 & 0 & -0.393352 & 0.242461 \\ 0 & 0 & 0.252014 & -0.145254 \\ 0 & 0 & 1.270845 & 1.107215 \\ 0 & 0 & 1.797547 & 0.423333 \end{bmatrix}$$

$$\mathbf{A}_4 = \begin{bmatrix} 0.490714 & 1 & 0 & 0 & 0.490714 & 0 & -0.490714 & 0 \\ -0.027371 & 0 & 0 & 0 & -0.027371 & 0 & 0.027371 & 0 \\ -0.201054 & 0 & 0 & 1 & -0.201054 & 0 & 0.201054 & 0 \\ -0.600823 & 0 & 0 & 0 & -0.600823 & 0 & 0.600823 & 0 \\ 0 & 0 & 0 & 0 & 0.491231 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.028179 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.201054 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -0.600823 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B}_1 = \begin{bmatrix} 1.34044 & 0 & 1.34044 & 0 \end{bmatrix}^T \times 10^{-3}$$

$$\mathbf{B}_{21} = \begin{bmatrix} -0.657773 & 0.036689 & 0.269501 & 0.805367 \end{bmatrix}$$

$$\mathbf{B}_{22} = \begin{bmatrix} -0.658465 & 0.037772 & 0.269501 & 0.805367 \end{bmatrix}$$

$$\mathbf{B}_2 = \begin{bmatrix} \mathbf{B}_{21} & \mathbf{B}_{22} \end{bmatrix}^T \times 10^{-3}$$

$$\mathbf{C}_1 = \begin{bmatrix} 0.983681 & 0.501644 & 0.985510 & 0.503879 \end{bmatrix}$$

$$\mathbf{C}_2 = \begin{bmatrix} -1 & 0 & 1 & 0 & -1 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{D} = 1.34044 \times 10^{-3}$$

The amplitude response of the original filter of order (4, 8) is depicted in Figure 2.2. The state-space model of the reduced-order filters of order (4, 4) obtained using PBMR, QBMR, SBMR, and ISBMR, denoted by $\{\mathbf{A}^{pr}, \mathbf{B}^{pr}, \mathbf{C}^{pr}, \mathbf{D}\}$, $\{\mathbf{A}^{qr}, \mathbf{B}^{qr}, \mathbf{C}^{qr}, \mathbf{D}\}$, $\{\mathbf{A}^{sr}, \mathbf{B}^{sr}, \mathbf{C}^{sr}, \mathbf{D}\}$, and $\{\tilde{\mathbf{A}}^{sr}, \tilde{\mathbf{B}}^{sr}, \tilde{\mathbf{C}}^{sr}, \mathbf{D}\}$, respectively, are given by (2.1a, b) with

$$\mathbf{A}_1^{rr} = \begin{bmatrix} 0.476575 & 0.071504 & -0.008308 & 0.006999 \\ -0.199883 & 0.448935 & -0.059115 & 0.132773 \\ 0.017593 & 0.457132 & 0.122131 & 0.227868 \\ 0.322202 & -0.096415 & -0.071173 & 0.028179 \end{bmatrix}$$

$$\mathbf{A}_2^{pr} = \begin{bmatrix} -0.406289 & 0.404954 & -0.071654 & -0.187262 \\ 0.132303 & 0.232817 & 0.373998 & -0.383797 \\ -0.104233 & -0.075402 & -0.190783 & 0.170627 \\ 0.269909 & -0.070960 & 0.238050 & -0.117157 \end{bmatrix}$$

$$\mathbf{A}_3^{pr} = \begin{bmatrix} -0.591179 & 0.213756 & 0.054254 & -0.058509 \\ 0.142647 & 0.552924 & 0.016455 & -0.103587 \\ -0.103675 & -0.134178 & -0.448242 & -0.006527 \\ 0.048576 & -0.370375 & 0.399663 & 0.101127 \end{bmatrix}$$

$$\mathbf{A}_4^{pr} = \begin{bmatrix} 0.424425 & -0.095418 & 0.051227 & -0.189558 \\ 0.099973 & 0.373724 & 0.159001 & 0.276695 \\ -0.019707 & -0.171204 & 0.481402 & 0.435481 \\ 0.174433 & 0.191557 & -0.084246 & 0.013259 \end{bmatrix}$$

$$\mathbf{B}_1^{pr} = \begin{bmatrix} 0.215478 & 0.189703 & -0.072481 & -0.002011 \end{bmatrix}^T$$

$$\mathbf{B}_2^{pr} = \begin{bmatrix} -0.091674 & 0.226184 & 0.158677 & 0.028007 \end{bmatrix}^T$$

$$\mathbf{C}_1^{pr} = \begin{bmatrix} 0.082786 & -0.092497 & -0.033056 & 0.023888 \end{bmatrix}$$

$$\mathbf{C}_2^{pr} = \begin{bmatrix} -0.330487 & -0.181282 & 0.082261 & -0.029086 \end{bmatrix}$$

$$\mathbf{A}_1^{qr} = \begin{bmatrix} 0.612849 & 0.141803 & 0.047366 & 0.025964 \\ -0.228874 & 0.482604 & -0.187528 & 0.173616 \\ -0.304246 & -0.001961 & -0.011828 & -0.097158 \\ -0.049942 & -0.465894 & 0.223375 & -0.007806 \end{bmatrix}$$

$$\mathbf{A}_2^{qr} = \begin{bmatrix} -0.602419 & 0.344197 & 0.013783 & 0.087485 \\ 0.182553 & 0.231221 & 0.591259 & 0.130874 \\ -0.290191 & 0.042176 & -0.210903 & -0.015847 \\ 0.037937 & 0.089470 & 0.196562 & 0.046626 \end{bmatrix}$$

$$\mathbf{A}_3^{qr} = \begin{bmatrix} -0.751363 & -0.029996 & 0.048884 & 0.033235 \\ -0.053691 & 0.717473 & 0.164217 & -0.069002 \\ -0.035735 & -0.045099 & -0.016739 & 0.633445 \\ 0.173899 & -0.263968 & 0.474349 & 0.213684 \end{bmatrix}$$

$$\mathbf{A}_4^{qr} = \begin{bmatrix} 0.550314 & -0.146292 & 0.039558 & 0.007343 \\ 0.146330 & 0.396116 & 0.171277 & -0.448576 \\ 0.033982 & -0.240615 & 0.250198 & 0.428946 \\ 0.239385 & 0.131979 & -0.018291 & -0.238276 \end{bmatrix}$$

$$\mathbf{B}_1^{qr} = \begin{bmatrix} 0.133295 & 0.111795 & 0.008135 & 0.042019 \end{bmatrix}^T$$

$$\mathbf{B}_2^{qr} = \begin{bmatrix} -0.076113 & 0.089793 & 0.076410 & 0.004162 \end{bmatrix}^T$$

$$\mathbf{C}_1^{qr} = \begin{bmatrix} 0.058379 & -0.051749 & -0.019926 & 0.019167 \end{bmatrix}$$

$$\mathbf{C}_2^{qr} = \begin{bmatrix} -0.143384 & -0.140029 & 0.052143 & -0.046524 \end{bmatrix}$$

$$\mathbf{A}_1^{sr} = \begin{bmatrix} 0.204028 & -0.478494 & -0.039806 & 0.110344 \\ 0.008351 & 0.322376 & -0.015419 & -0.072203 \\ -0.007354 & 0.046330 & 0.321571 & 0.494798 \\ -0.005895 & -0.015251 & 0.015140 & 0.227844 \end{bmatrix}$$

$$\mathbf{A}_2^{sr} = \begin{bmatrix} -0.014757 & 0.004628 & 0.322581 & -0.237760 \\ -0.015998 & -0.003952 & 0.460555 & -0.335131 \\ -0.010118 & 0.013917 & 0.088398 & -0.070334 \\ 0.022123 & -0.036190 & -0.122059 & 0.104066 \end{bmatrix}$$

$$\mathbf{A}_3^{sr} = \begin{bmatrix} 0.003209 & -0.011068 & -0.134203 & -0.026011 \\ -0.015853 & -0.033533 & -0.069078 & -0.027601 \\ 0.463531 & 0.356722 & 0.161862 & -0.095306 \\ -0.263790 & 0.096745 & 0.198288 & 0.161307 \end{bmatrix}$$

$$\mathbf{A}_4^{sr} = \begin{bmatrix} -0.156506 & -0.050858 & 0.062349 & 0.124685 \\ 0.379723 & 0.164662 & 0.099129 & 0.032810 \\ -0.033946 & -0.014127 & 0.020646 & -0.042906 \\ -0.039028 & 0.036357 & 0.359830 & 0.463635 \end{bmatrix}$$

$$\mathbf{B}_1^{sr} = \begin{bmatrix} 0.529674 & 1.888615 & -1.211267 & 3.492757 \end{bmatrix}^T$$

$$\mathbf{B}_2^{sr} = \begin{bmatrix} 0.051378 & -0.231813 & 0.679364 & -0.296411 \end{bmatrix}^T$$

$$\mathbf{C}_1^{sr} = \begin{bmatrix} 0.030463 & -0.044590 & 0.010074 & 0.023741 \end{bmatrix}$$

$$\mathbf{C}_2^{sr} = \begin{bmatrix} 0.005591 & -0.009694 & -0.072331 & -0.077022 \end{bmatrix}$$

$$\tilde{\mathbf{A}}_1^{sr} = \begin{bmatrix} 0.398395 & 0.067647 & -0.041205 & -0.047699 \\ -0.165376 & 0.097911 & 0.056528 & 0.057307 \\ 0.054464 & -0.386181 & 0.439015 & 0.198796 \\ -0.300009 & 0.004979 & -0.079289 & 0.140498 \end{bmatrix}$$

$$\tilde{\mathbf{A}}_2^{sr} = \begin{bmatrix} -0.493470 & 0.111113 & 0.433265 & -0.097438 \\ 0.164658 & -0.074752 & 0.539074 & 0.478618 \\ -0.106099 & 0.031564 & -0.046091 & -0.111813 \\ -0.328834 & 0.085950 & 0.072648 & -0.205922 \end{bmatrix}$$

$$\tilde{\mathbf{A}}_3^{sr} = \begin{bmatrix} -0.503071 & 0.027229 & 0.054018 & -0.078483 \\ 0.034454 & -0.185564 & 0.310070 & 0.350054 \\ 0.042530 & 0.446818 & -0.018492 & 0.407874 \\ 0.027084 & 0.270648 & 0.381488 & -0.371722 \end{bmatrix}$$

$$\tilde{\mathbf{A}}_4^{sr} = \begin{bmatrix} 0.340191 & 0.384383 & 0.158443 & -0.101164 \\ 0.000524 & 0.366455 & 0.024461 & -0.015399 \\ 0.113994 & -0.139845 & 0.034981 & -0.044875 \\ -0.045414 & 0.299108 & -0.065030 & 0.087856 \end{bmatrix}$$

$$\tilde{\mathbf{B}}_1^{sr} = \begin{bmatrix} 0.304264 & 0.359372 & -0.028458 & 0.057085 \end{bmatrix}^T$$

$$\tilde{\mathbf{B}}_2^{sr} = \begin{bmatrix} -0.150370 & -0.189039 & 0.230790 & 0.157800 \end{bmatrix}^T$$

$$\tilde{\mathbf{C}}_1^{sr} = \begin{bmatrix} 0.100857 & -0.057272 & 0.132552 & -0.064706 \end{bmatrix}$$

$$\tilde{\mathbf{C}}_2^{sr} = \begin{bmatrix} -0.398981 & 0.357827 & -0.336220 & 0.456789 \end{bmatrix}$$

The amplitude responses of the reduced-order filters of order (4, 4) obtained using PBMR, QBMR, SBMR, and ISBMR are shown in Figures 2.3-2.5, respectively. The performance of the reduced-order filters obtained using the

Table 2.2: Performance of the reduced-order filter of order (4, 4).

Method $r_1=4,\ r_2=4$	Stability	Flops $\times 10^6$	Error e_∞
PBMR	stable	5.7232×10^6	0.1107
QBMR	stable	1.8451×10^6	0.2783
SBMR	stable	1.6860×10^9	1.3323
ISBMR*	stable	6.1558×10^9	0.1595

ISBMR*: $k_0 = 1$, $k_1 = 1$, and $k_2 = 0.3$.

different model-reduction methods is summarized in Table 2.2.

On comparing Tables 2.1 and 2.2, some observations can be made. The PBMR method produces a stable reduced-order system of order (4, 4) for Example 2.2, but an unstable reduced-order system of order (2, 1) for Example 2.1. The QBMR method results in stable reduced-order systems for both examples. (It has been found that the QBMR leads to an unstable reduced-order system in another example). The QBMR method is the most computationally efficient model-reduction method. The SBMR and ISBMR methods produce stable reduced-order systems for both examples. Although Algorithm 2.3 finds local minimum points $\mathbf{L}_P^i$ and $\mathbf{L}_Q^i$ of the minimization problems in (2.23a, b) such that the inequalities in (2.24a, b) are satisfied, the approximation error in Example 2.2 is not considered acceptable. Algorithm 2.4, on the other hand, results in reduced-order systems with small approximation error for both examples.

2.6 Conclusions

In this chapter, a sufficient condition for the existence of quasi-gramians was presented. A new algorithm, Algorithm 2.2, was proposed for computing quasi-gramians. The algorithm involves solving two 1-D Lyapunov equations at each iteration. The resulting QBMR method is the most computationally efficient model-reduction method.

Two new reliable algorithms, Algorithms 2.3 and 2.4, were developed for the computation of structured gramians. Both algorithms amount to solving *unconstrained* optimization problems. Algorithm 2.4 takes into account both system stability and approximation error. The ISBMR method obtained using Algorithm 2.4 is the only balanced model-reduction method that leads to a stable reduced-order system with small approximation error.

Algorithm 2.4 is used in Chapter 3 for the development of weighted balanced realization and model-reduction methods for 2-D discrete systems with input and output weights.

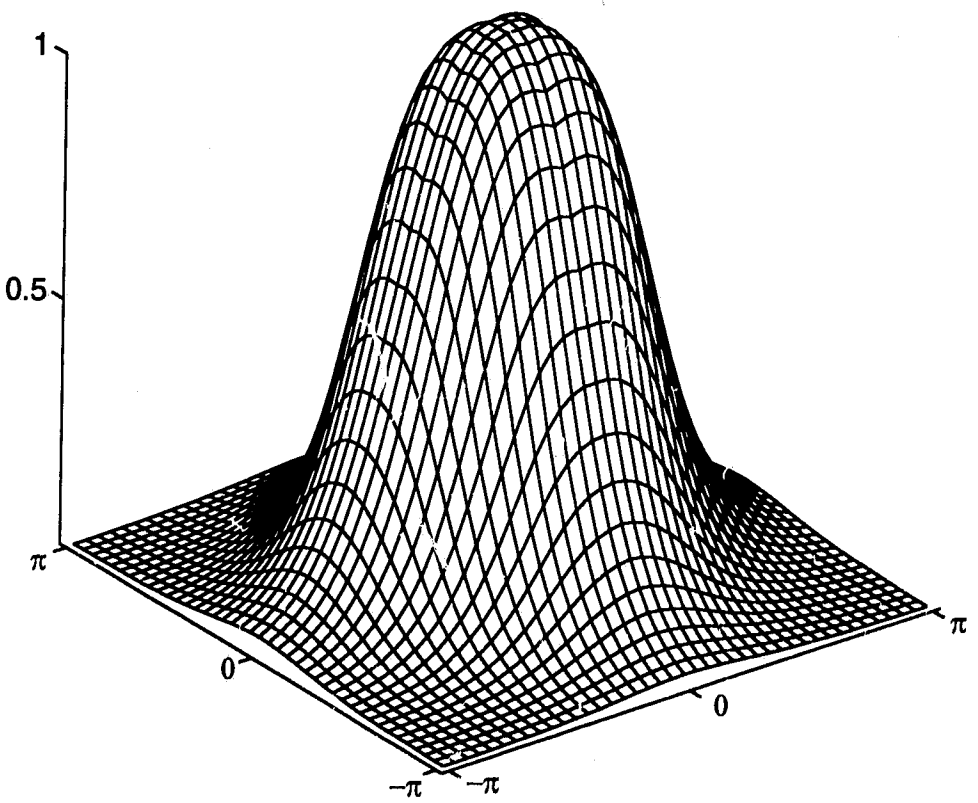

Figure 2.2: Amplitude response of the original filter of order (4, 8).

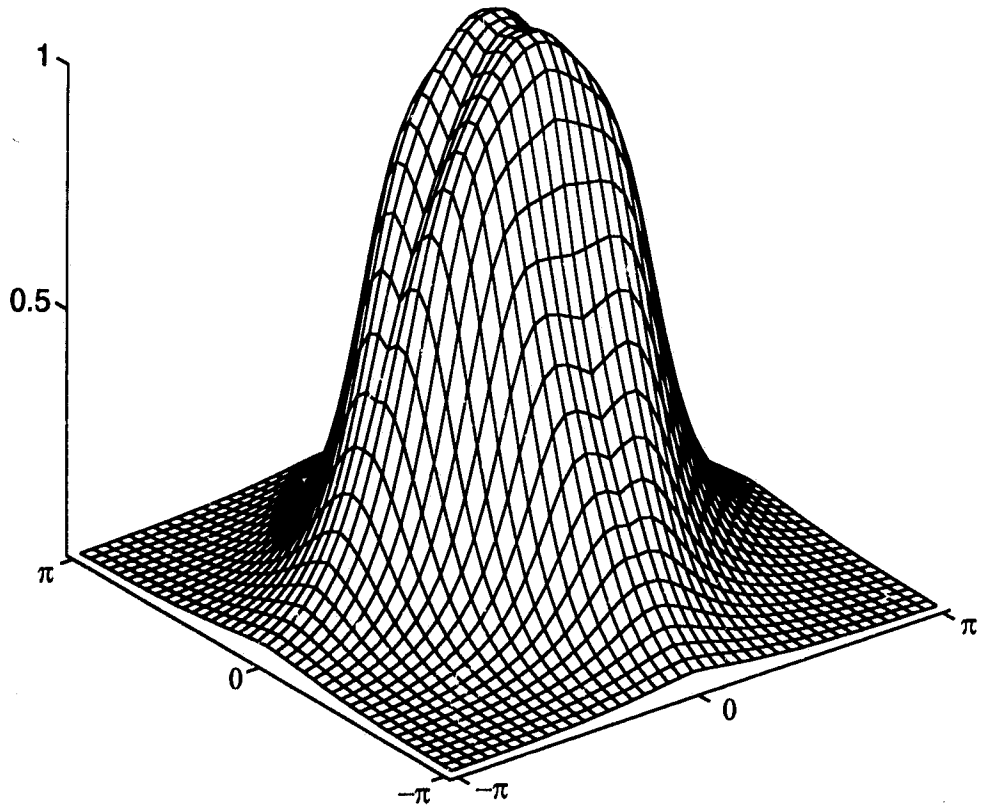

Figure 2.3: Amplitude response of the filter of order (4, 4) from PBMR.

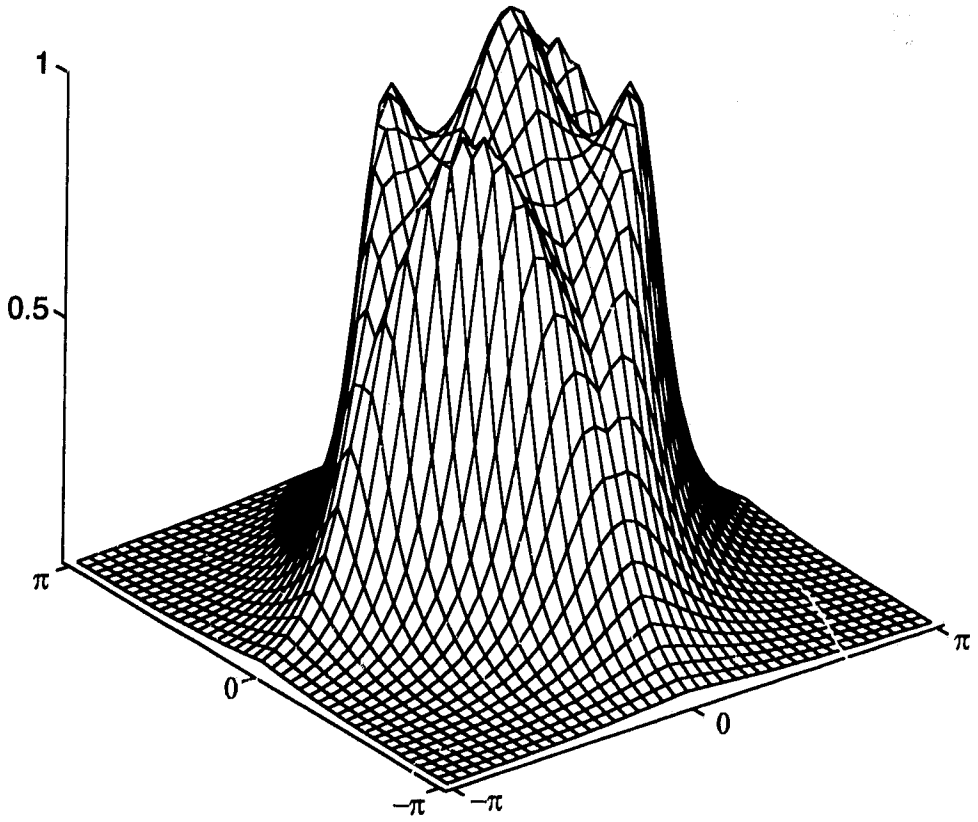

Figure 2.4: Amplitude response of the filter of order (4, 4) from QBMR.

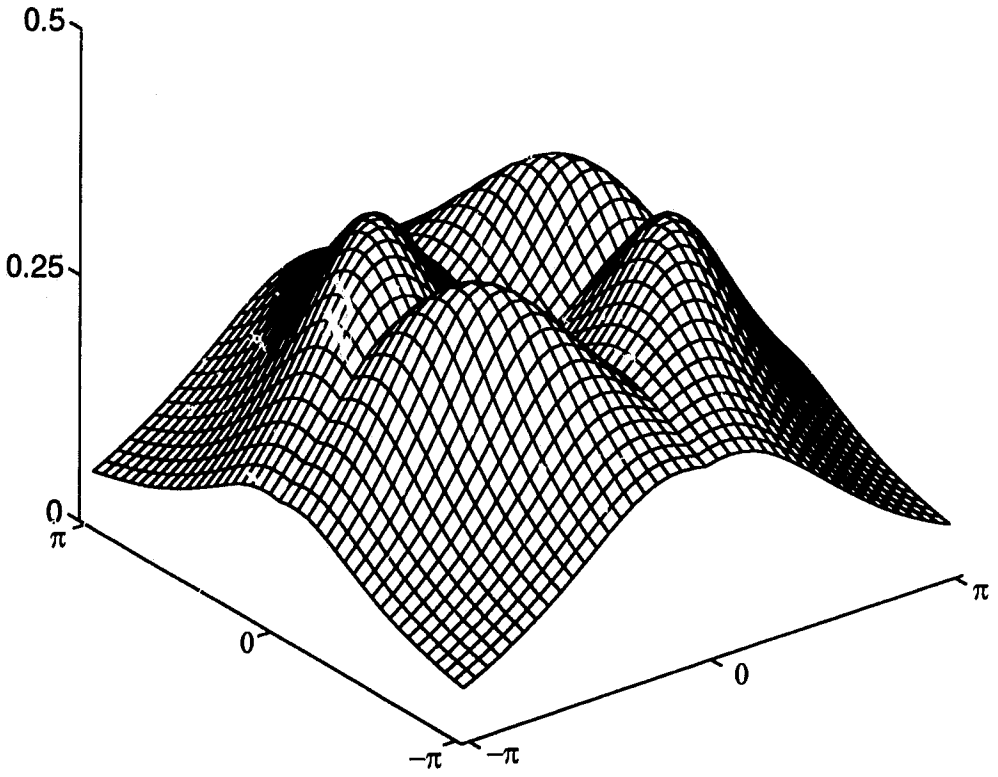

Figure 2.5: Amplitude response of the filter of order (4, 4) from SBMR.

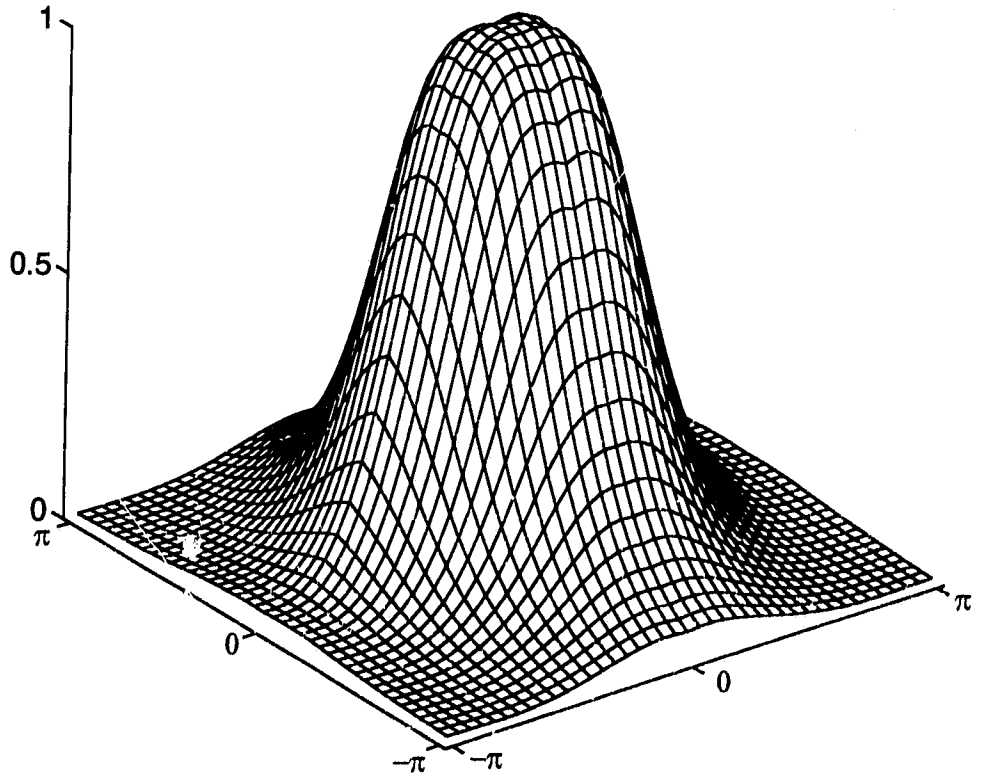

Figure 2.6: Amplitude response of the filter of order (4, 4) from ISBMR.

Chapter 3

2-D Weighted Balanced Realization and Model-Reduction

3.1 Introduction

The structurally balanced realization method discussed in Chapter 2 is an effective technique for the reduction of system order in 2-D discrete systems [32, 56, 62]. However, for special cases where the requirements for approximation error between the full-order and reduced-order systems are different in different frequency regions, an improved reduced-order system can be obtained by applying the *weighted structurally balanced realization* (WSBR) and *model-reduction* (WSBMR) methods for 2-D discrete systems with input and output weights proposed in this chapter.

There are three crucial stages in the development of the WSBR and WSBMR. The first stage is the introduction of two auxiliary transfer-function matrices called *weighted-input-to-state* and *state-to-weighted-output* transfer-function matrices (see Section 3.2). The second stage is the definition of

the 2-D *weighted structured controllability and observability gramians* (see Section 3.3). The third stage is formulating an unconstrained minimization problem to obtain the WSBMR method that takes into account both system stability and approximation error (see Section 3.4).

It is shown that the auxiliary transfer-function matrices are Q-stable provided that the input/output weights and the original system are Q-stable. The 2-D weighted gramians are defined as the solution to the 2-D Lyapunov inequalities in Section 3.3 and the existence of the weighted gramians is justified in Section 3.2.2. These gramians result in the WSBR method. It is shown in Section 3.3.3 that the SBR (or ISBR) discussed in Chapter 2 is a special case of the WSBR. Eventually, the WSBMR presented in Section 3.4 leads to a Q-stable reduced-order weighted system that approximates the full-order weighted system to within a small error in desired frequency regions. The proposed WSBMR method is illustrated and compared with ISBMR by an example.

3.2 Auxiliary Transfer-Function Matrices of 2-D Weighted Systems

The system configuration to be considered in this chapter is shown in Figure 3.1. For simplicity, the system with input and output weights in Figure 3.1 is referred as 2-D *weighted discrete system.* First, the *input* and *output weights* are treated as 2-D transfer-function matrices, i.e., $\mathbf{W}^i(z_1, z_2) \in \Re^{(l \times v)}$

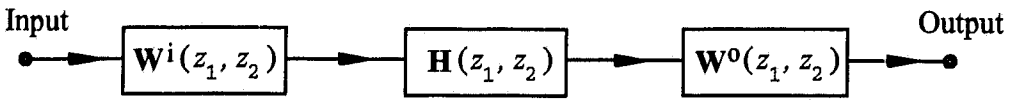

Figure 3.1: A 2-D weighted discrete system.

and $\mathbf{W}^o(z_1, z_2) \in \Re^{(u\times s)}$, respectively, and then both the transfer-function matrix of the (unweighted) system in Figure 2.1, $\mathbf{H}(z_1, z_2) \in \Re^{(s\times t)}$, and the transfer-function matrices of the weights are realized in extended 2-D state-space models.

If the Roesser state-space model given by (2.1a, b) is used to describe $\mathbf{H}(z_1, z_2)$, $\mathbf{W}^i(z_1, z_2)$, and $\mathbf{W}^o(z_1, z_2)$, then

$$\mathbf{W}^i(z_1, z_2) = \mathbf{C}^i [\, \mathbf{I}(z_1, z_2) - \mathbf{A}^i \,]^{-1} \mathbf{B}^i + \mathbf{D}^i \tag{3.1a}$$

$$\mathbf{W}^o(z_1, z_2) = \mathbf{C}^o [\, \mathbf{I}(z_1, z_2) - \mathbf{A}^o \,]^{-1} \mathbf{B}^o + \mathbf{D}^o \tag{3.1b}$$

and the expression for $\mathbf{H}(z_1, z_2)$ is given in (2.2). The matrices involved can be partitioned as

$$\mathbf{A}^i = \begin{bmatrix} \mathbf{A}^i_1 & \mathbf{A}^i_2 \\ \mathbf{A}^i_3 & \mathbf{A}^i_4 \end{bmatrix}_{(m_i+n_i)\times(m_i+n_i)} \qquad \mathbf{B}^i = \begin{bmatrix} \mathbf{B}^i_1 \\ \mathbf{B}^i_2 \end{bmatrix}_{(m_i+n_i)\times v}$$

$$\mathbf{A}^o = \begin{bmatrix} \mathbf{A}^o_1 & \mathbf{A}^o_2 \\ \mathbf{A}^o_3 & \mathbf{A}^o_4 \end{bmatrix}_{(m_o+n_o)\times(m_o+n_o)} \qquad \mathbf{B}^o = \begin{bmatrix} \mathbf{B}^o_1 \\ \mathbf{B}^o_2 \end{bmatrix}_{(m_o+n_o)\times s}$$

$$\mathbf{C}^i = \begin{bmatrix} \mathbf{C}^i_1 & \mathbf{C}^i_2 \end{bmatrix}_{t\times(m_i+n_i)} \qquad \mathbf{C}^o = \begin{bmatrix} \mathbf{C}^o_1 & \mathbf{C}^o_2 \end{bmatrix}_{u\times(m_o+n_o)}$$

where (m_i, n_i) and (m_o, n_o) are the orders of the input and output weights, respectively. Henceforth, it is assumed that $\mathbf{H}(z_1, z_2)$, $\mathbf{W}^i(z_1, z_2)$, and $\mathbf{W}^o(z_1, z_2)$ are 2-D Q-stable. By Lemma 2.1, this is true if and only if there exist block-diagonal positive definite matrices given by

$$\mathbf{Y} = \mathbf{Y}_1 \oplus \mathbf{Y}_2, \qquad \mathbf{Y}^i = \mathbf{Y}_1^i \oplus \mathbf{Y}_2^i, \qquad \mathbf{Y}^o = \mathbf{Y}_1^o \oplus \mathbf{Y}_2^o$$

where

$$\mathbf{Y}_1 \in \Re^{(m\times m)}, \qquad \mathbf{Y}_2 \in \Re^{(n\times n)}, \qquad \mathbf{Y}_1^i \in \Re^{(m_i\times m_i)}$$
$$\mathbf{Y}_2^i \in \Re^{(n_i\times n_i)}, \qquad \mathbf{Y}_1^o \in \Re^{(m_o\times m_o)}, \qquad \mathbf{Y}_2^o \in \Re^{(n_o\times n_o)}$$

such that

$$\mathbf{\Psi} = \mathbf{AYA}^T - \mathbf{Y} + \mathbf{BB}^T < \mathbf{0} \tag{3.2a}$$

$$\mathbf{\Psi}^i = \mathbf{A}^i\mathbf{Y}^i\mathbf{A}^{iT} - \mathbf{Y}^i + \mathbf{B}^i\mathbf{B}^{iT} < \mathbf{0} \tag{3.2b}$$

$$\mathbf{\Psi}^o = \mathbf{A}^o\mathbf{Y}^o\mathbf{A}^{oT} - \mathbf{Y}^o + \mathbf{B}^o\mathbf{B}^{oT} < \mathbf{0} \tag{3.2c}$$

In order to define the WSBR in Section 3.3, two auxiliary transfer-function matrices need to be defined first based on the configuration in Figure 3.1.

3.2.1 Definition of Auxiliary Transfer-Function Matrices

Definition 3.1

The *weighted-input-to-state* transfer-function matrix, $\mathbf{H}^i(z_1, z_2) \in \Re^{(s\times v)}$, and *state-to-weighted-output* transfer-function matrix, $\mathbf{H}^o(z_1, z_2) \in \Re^{(u\times t)}$,

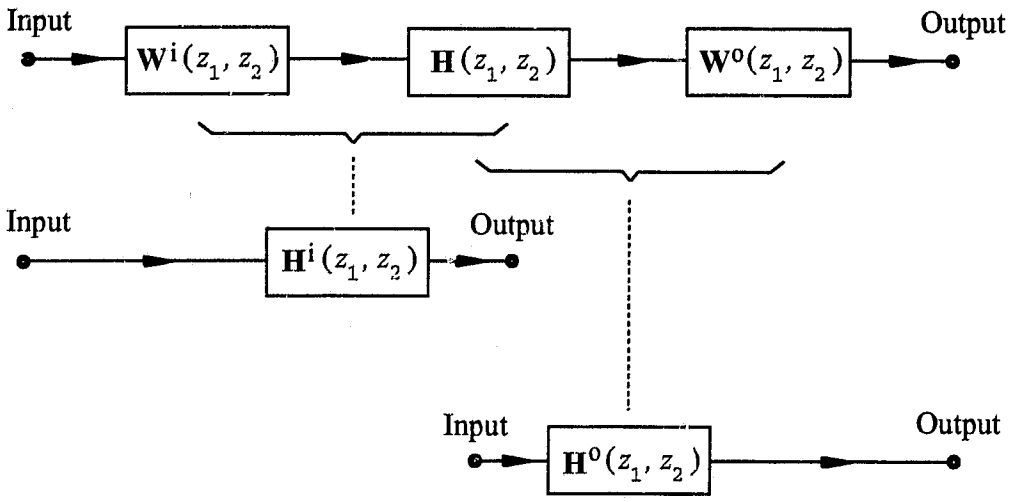

Figure 3.2: Auxiliary transfer-function matrices $\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$.

of the weighted system in Figure 3.1 are defined as

$$\mathbf{H}^i(z_1, z_2) = [\, \mathbf{I}(z_1, z_2) - \mathbf{A} \,]^{-1} \mathbf{B}\, \mathbf{W}^i(z_1, z_2) \tag{3.3a}$$

$$\mathbf{H}^o(z_1, z_2) = \mathbf{W}^o(z_1, z_2)\, \mathbf{C}\, [\, \mathbf{I}(z_1, z_2) - \mathbf{A} \,]^{-1} \tag{3.3b}$$

These definitions are illustrated in Figure 3.2.

By introducing permutation matrix, $\hat{\mathbf{I}}$, given by

$$\hat{\mathbf{I}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

and performing permutations for certain blocks of the matrices in (3.3a, b), the auxiliary transfer-function matrices $\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$ can be

rewritten, respectively, as

$$\mathbf{H}^i(z_1, z_2) = \hat{\mathbf{C}}^i \left[\mathbf{I}(z_1, z_2) - \hat{\mathbf{A}}^i \right]^{-1} \hat{\mathbf{B}}^i \tag{3.4a}$$

and

$$\mathbf{H}^o(z_1, z_2) = \hat{\mathbf{C}}^o \left[\mathbf{I}(z_1, z_2) - \hat{\mathbf{A}}^o \right]^{-1} \hat{\mathbf{B}}^o \tag{3.4b}$$

where the associated *weighted-input-to-state* matrices $\{\hat{\mathbf{A}}^i, \hat{\mathbf{B}}^i, \hat{\mathbf{C}}^i, \mathbf{0}\}$ are given by

$$\hat{\mathbf{A}}^i = \begin{bmatrix} \hat{\mathbf{A}}^i_1 & \hat{\mathbf{A}}^i_2 \\ \hat{\mathbf{A}}^i_3 & \hat{\mathbf{A}}^i_4 \end{bmatrix} = \left[\begin{array}{cc|cc} \mathbf{A}_1 & \mathbf{B}_1\mathbf{C}^i_1 & \mathbf{A}_2 & \mathbf{B}_1\mathbf{C}^i_2 \\ \mathbf{0} & \mathbf{A}^i_1 & \mathbf{0} & \mathbf{A}^i_2 \\ \hline \mathbf{A}_3 & \mathbf{B}_2\mathbf{C}^i_1 & \mathbf{A}_4 & \mathbf{B}_2\mathbf{C}^i_2 \\ \mathbf{0} & \mathbf{A}^i_3 & \mathbf{0} & \mathbf{A}^i_4 \end{array}\right] \tag{3.5a}$$

$$\hat{\mathbf{B}}^i = \left[\hat{\mathbf{B}}^{i\,T}_1 \quad \hat{\mathbf{B}}^{i\,T}_2 \right]^T = \left[\mathbf{D}^{i\,T}\mathbf{B}^T_1 \quad \mathbf{B}^{i\,T}_1 \;\middle|\; \mathbf{D}^{i\,T}\mathbf{B}^T_2 \quad \mathbf{B}^{i\,T}_2 \right]^T \tag{3.5b}$$

$$\hat{\mathbf{C}}^i = \left[\hat{\mathbf{C}}^i_1 \quad \hat{\mathbf{C}}^i_2 \right] = \left[\begin{array}{cc|cc} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{array}\right] \tag{3.5c}$$

and *state-to-weighted-output* matrices $\{\hat{\mathbf{A}}^o, \hat{\mathbf{B}}^o, \hat{\mathbf{C}}^o, \mathbf{0}\}$ by

$$\hat{\mathbf{A}}^o = \begin{bmatrix} \hat{\mathbf{A}}^o_1 & \hat{\mathbf{A}}^o_2 \\ \hat{\mathbf{A}}^o_3 & \hat{\mathbf{A}}^o_4 \end{bmatrix} = \left[\begin{array}{cc|cc} \mathbf{A}_1 & \mathbf{0} & \mathbf{A}_2 & \mathbf{0} \\ \mathbf{B}^o_1\mathbf{C}_1 & \mathbf{A}^o_1 & \mathbf{B}^o_1\mathbf{C}_2 & \mathbf{A}^o_2 \\ \hline \mathbf{A}_3 & \mathbf{0} & \mathbf{A}_4 & \mathbf{0} \\ \mathbf{B}^o_2\mathbf{C}_1 & \mathbf{A}^o_3 & \mathbf{B}^o_2\mathbf{C}_2 & \mathbf{A}^o_4 \end{array}\right] \tag{3.5d}$$

$$\hat{\mathbf{B}}^o = \left[\hat{\mathbf{B}}^{o\,T}_1 \quad \hat{\mathbf{B}}^{o\,T}_2 \right]^T = \left[\begin{array}{cc|cc} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{array}\right]^T \tag{3.5e}$$

$$\hat{\mathbf{C}}^o = \left[\hat{\mathbf{C}}^o_1 \quad \hat{\mathbf{C}}^o_2 \right] = \left[\mathbf{D}^o\mathbf{C}_1 \quad \mathbf{C}^o_1 \;\middle|\; \mathbf{D}^o\mathbf{C}_2 \quad \mathbf{C}^o_2 \right] \tag{3.5f}$$

As indicated in Figure 3.2, transfer-function matrix $\mathbf{H}^i(z_1, z_2)$ in (3.3a) relates a weighted input signal to a state of the weighted system in Figure 3.1, which takes the input weight $\mathbf{W}^i(z_1, z_2)$ into account. Similarly, transfer-function matrix $\mathbf{H}^o(z_1, z_2)$ in (3.3b) relates a state of the weighted system in Figure 3.1 to the weighted output, which takes the output weight $\mathbf{W}^o(z_1, z_2)$ into account. As will be seen in Section 3.3, these auxiliary transfer-function matrices serve as the starting point for the definition of the WSBR. From (3.4a, b), the extended Roesser state-space models of $\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$ are given by

$$\begin{bmatrix} \mathbf{x}^{h_i}(k+1, l) \\ \mathbf{x}^{v_i}(k, l+1) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{A}}^i_1 & \hat{\mathbf{A}}^i_2 \\ \hat{\mathbf{A}}^i_3 & \hat{\mathbf{A}}^i_4 \end{bmatrix} \begin{bmatrix} \mathbf{x}^{h_i}(k, l) \\ \mathbf{x}^{v_i}(k, l) \end{bmatrix} + \begin{bmatrix} \hat{\mathbf{B}}^i_1 \\ \hat{\mathbf{B}}^i_2 \end{bmatrix} \mathbf{u}^i(k, l)$$

$$= \hat{\mathbf{A}}^i \, \mathbf{x}^i + \hat{\mathbf{B}}^i \, \mathbf{u}^i \tag{3.6a}$$

$$\mathbf{y}^i(k, l) = \begin{bmatrix} \hat{\mathbf{C}}^i_1 & \hat{\mathbf{C}}^i_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}^{h_i}(k, l) \\ \mathbf{x}^{v_i}(k, l) \end{bmatrix} = \hat{\mathbf{C}}^i \, \mathbf{x}^i \tag{3.6b}$$

and

$$\begin{bmatrix} \mathbf{x}^{h_o}(k+1, l) \\ \mathbf{x}^{v_o}(k, l+1) \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{A}}^o_1 & \hat{\mathbf{A}}^o_2 \\ \hat{\mathbf{A}}^o_3 & \hat{\mathbf{A}}^o_4 \end{bmatrix} \begin{bmatrix} \mathbf{x}^{h_o}(k, l) \\ \mathbf{x}^{v_o}(k, l) \end{bmatrix} + \begin{bmatrix} \hat{\mathbf{B}}^o_1 \\ \hat{\mathbf{B}}^o_2 \end{bmatrix} \mathbf{u}^o(k, l)$$

$$= \hat{\mathbf{A}}^o \, \mathbf{x}^o + \hat{\mathbf{B}}^o \, \mathbf{u}^o \tag{3.6c}$$

$$\mathbf{y}^o(k, l) = \begin{bmatrix} \hat{\mathbf{C}}^o_1 & \hat{\mathbf{C}}^o_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}^{h_o}(k, l) \\ \mathbf{x}^{v_o}(k, l) \end{bmatrix} = \hat{\mathbf{C}}^o \, \mathbf{x}^o \tag{3.6d}$$

where

$$\mathbf{x}^{h_i} \in \Re^{(m+m_i)}, \quad \mathbf{x}^{v_i} \in \Re^{(n+n_i)}, \quad \text{and} \quad \mathbf{x}^{h_o} \in \Re^{(m+m_o)}, \quad \mathbf{x}^{v_o} \in \Re^{(n+n_o)}$$

are the *horizontal* and *vertical* state-space vectors of the $\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$, respectively; and

$$\mathbf{u}^i \in \Re^v, \quad \mathbf{y}^i \in \Re^{(m+n)}, \quad \text{and} \quad \mathbf{u}^o \in \Re^{(m+n)}, \quad \mathbf{y}^o \in \Re^u$$

are the *input* vector for the weighted system depicted in Figure 3.1, the *output* vector for $\mathbf{H}^i(z_1, z_2)$, and the *input* and *output* vectors for $\mathbf{H}^o(z_1, z_2)$, respectively.

As the structurally balanced realization is only defined for Q-stable 2-D systems, the Q-stability of auxiliary transfer-function matrices, $\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$, is discussed in the following lemma.

3.2.2 Q-Stability of the Auxiliary Transfer-Function Matrices

Theorem 3.1

If the (unweighted) system depicted in Figure 2.1, and input and output weights $\mathbf{W}^i(z_1, z_2)$ and $\mathbf{W}^o(z_1, z_2)$, are all 2-D Q-stable, then the weighted-input-to-state and state-to-weighted-output transfer-function matrices, $\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$, defined by (3.3a) and (3.3b), respectively, are also 2-D *Q-stable.*

Proof

The Q-stability of the (unweighted) system and input weight implies that $\mathbf{H}(z_1, z_2)$ and $\mathbf{W}^i(z_1, z_2)$ satisfy the 2-D Lyapunov inequalities (3.2a) and

(3.2b), respectively. Define

$$\hat{\mathbf{Y}} = \left[\begin{array}{cc|cc} \mathbf{Y}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}_1^i & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{Y}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Y}_2^i \end{array}\right] = \hat{\mathbf{Y}}_1 \oplus \hat{\mathbf{Y}}_2 \tag{3.7}$$

and

$$\hat{\mathbf{\Psi}} = \hat{\mathbf{A}}^i \hat{\mathbf{Y}} \hat{\mathbf{A}}^{iT} - \hat{\mathbf{Y}} + \hat{\mathbf{B}}^i \hat{\mathbf{B}}^{iT} \tag{3.8}$$

Premultiplying (3.8) by the permutation matrix, $\hat{\mathbf{I}}$, and postmultiplying by $\hat{\mathbf{I}}^T$, we have

$$\hat{\mathbf{I}}\,\hat{\mathbf{\Psi}}\,\hat{\mathbf{I}}^T = \tilde{\mathbf{A}}^i \tilde{\mathbf{Y}} \tilde{\mathbf{A}}^{iT} - \tilde{\mathbf{Y}} + \tilde{\mathbf{B}}^i \tilde{\mathbf{B}}^{iT} \tag{3.9}$$

where

$$\tilde{\mathbf{A}}^i = \left[\begin{array}{cc|cc} \mathbf{A}_1 & \mathbf{A}_2 & \mathbf{B}_1\mathbf{C}_1^i & \mathbf{B}_1\mathbf{C}_2^i \\ \mathbf{A}_3 & \mathbf{A}_4 & \mathbf{B}_2\mathbf{C}_1^i & \mathbf{B}_2\mathbf{C}_2^i \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{A}_1^i & \mathbf{A}_2^i \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_3^i & \mathbf{A}_4^i \end{array}\right] = \left[\begin{array}{cc} \mathbf{A} & \mathbf{B}\mathbf{C}^i \\ \mathbf{0} & \mathbf{A}^i \end{array}\right]$$

$$\tilde{\mathbf{B}}^i = \left[\begin{array}{cc|cc} \mathbf{D}^{iT}\mathbf{B}_1^T & \mathbf{D}^{iT}\mathbf{B}_2^T & \mathbf{B}_1^{iT} & \mathbf{B}_2^{iT} \end{array}\right]^T = \left[\begin{array}{c} \mathbf{B}\mathbf{D}^i \\ \mathbf{B}^i \end{array}\right]$$

and

$$\tilde{\mathbf{Y}} = \left[\begin{array}{cc|cc} \mathbf{Y}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}_2 & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{Y}_1^i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Y}_2^i \end{array}\right] = \mathbf{Y} \oplus \mathbf{Y}^i$$

Now expand (3.9) as

$$\hat{\mathbf{I}}\,\hat{\mathbf{\Psi}}\,\hat{\mathbf{I}}^T = \begin{bmatrix} \mathbf{AYA}^T - \mathbf{Y} + \mathbf{B}(\mathbf{C}^i\mathbf{Y}^i\mathbf{C}^{iT} + \mathbf{D}^i\mathbf{D}^{iT})\mathbf{B}^T & \mathbf{B}(\mathbf{C}^i\mathbf{Y}^i\mathbf{A}^{iT} + \mathbf{D}^i\mathbf{B}^{iT}) \\ (\mathbf{A}^i\mathbf{Y}^i\mathbf{C}^{iT} + \mathbf{B}^i\mathbf{D}^{iT})\mathbf{B}^T & \mathbf{\Psi}^i \end{bmatrix}$$

where $\mathbf{\Psi}^i$ is defined by (3.2b). As $\mathbf{\Psi}^i$ is nonsingular, $\hat{\mathbf{I}}\,\hat{\mathbf{\Psi}}\,\hat{\mathbf{I}}^T$ can be block diagonalized by premultiplying with

$$\mathbf{X} = \begin{bmatrix} \mathbf{I} & -\mathbf{B}(\mathbf{C}^i\mathbf{Y}^i\mathbf{A}^{iT} + \mathbf{D}^i\mathbf{B}^{iT})(\mathbf{\Psi}^i)^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$$

and then postmultiplying with $\mathbf{X}^T$. This leads to

$$\mathbf{X}(\hat{\mathbf{I}}\,\hat{\mathbf{\Psi}}\,\hat{\mathbf{I}}^T)\,\mathbf{X}^T = \begin{bmatrix} \mathbf{AYA}^T - \mathbf{Y} + \mathbf{B\Theta B}^T & \mathbf{0} \\ \mathbf{0} & \mathbf{\Psi}^i \end{bmatrix}$$

where

$$\mathbf{\Theta} = \mathbf{C}^i\mathbf{Y}^i\mathbf{C}^{iT} + \mathbf{D}^i\mathbf{D}^{iT} - (\mathbf{C}^i\mathbf{Y}^i\mathbf{A}^{iT} + \mathbf{D}^i\mathbf{B}^{iT})(\mathbf{\Psi}^i)^{-1}(\mathbf{A}^i\mathbf{Y}^i\mathbf{C}^{iT} + \mathbf{B}^i\mathbf{D}^{iT}) \tag{3.10}$$

As $\mathbf{\Psi}^i < \mathbf{0}$, matrix $\mathbf{X}(\hat{\mathbf{I}}\,\hat{\mathbf{\Psi}}\,\hat{\mathbf{I}}^T)\,\mathbf{X}^T < \mathbf{0}$ (and hence $\hat{\mathbf{\Psi}} < \mathbf{0}$) if and only if

$$\mathbf{AYA}^T - \mathbf{Y} + \mathbf{B\Theta B}^T < \mathbf{0} \tag{3.11}$$

With a fixed $\mathbf{Y}^i$ that satisfies (3.2b), $\mathbf{\Theta}$ in (3.10) is a known matrix, and so

$$\|\mathbf{\Theta}\| \le \beta \quad \text{for some} \quad \beta > \mathbf{0}$$

From (3.2a) it follows that for any scalar $\alpha > 1$

$$\begin{aligned} \mathbf{A}(\alpha\mathbf{Y})\mathbf{A}^T - \alpha\mathbf{Y} + \mathbf{B\Theta B}^T &= \alpha\mathbf{\Psi} + \mathbf{B}(\mathbf{\Theta} - \alpha\mathbf{I})\mathbf{B}^T \\ &\le \alpha\mathbf{\Psi} + \mathbf{B\Theta B}^T \end{aligned}$$

Since $\mathbf{\Psi} < \mathbf{0}$, if α is chosen such that

$$\alpha > \frac{\beta \|\mathbf{B}\|^2}{|\gamma_{\max}(\mathbf{\Psi}|} \tag{3.12}$$

then

$$\mathbf{A}(\alpha\mathbf{Y})\mathbf{A}^T - \alpha\mathbf{Y} + \mathbf{B}\mathbf{\Theta}\mathbf{B}^T < \mathbf{0} \tag{3.13}$$

In other words, (3.11) holds if $\mathbf{Y}$ is scaled to $\alpha\mathbf{Y}$ as is seen in (3.13). We conclude that

$$\hat{\mathbf{A}}^i\hat{\mathbf{P}}\hat{\mathbf{A}}^{iT} - \hat{\mathbf{P}} + \hat{\mathbf{B}}^i\hat{\mathbf{B}}^{iT} < 0$$

where

$$\hat{\mathbf{P}} = \left[\begin{array}{cc|cc} \alpha\mathbf{Y}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}_1^i & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \alpha\mathbf{Y}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Y}_2^i \end{array}\right] \tag{3.14}$$

with α satisfying (3.12) and, therefore, $\mathbf{H}^i(z_1, z_2)$ is 2-D Q-stable. Likewise, by assuming that the unweighted system and output weight are 2-D Q-stable, and using (3.2a, c) in conjunction with an argument similar to the above, one can show that $\mathbf{H}^o(z_1, z_2)$ is also a Q-stable 2-D transfer-function matrix. □

Based on $\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$, the WSBR is developed in the following section.

3.3 A Weighted Structurally Balanced Realization

The WSBR that will be defined in this section is essentially an extension of the structurally balanced realization method for 2-D (unweighted) systems proposed in [56].

3.3.1 Definitions

$\mathbf{H}^i(z_1, z_2)$ and $\mathbf{H}^o(z_1, z_2)$ are 2-D Q-stable, consequently, there exist block-diagonal positive definite matrices

$$\hat{\mathbf{P}} = \hat{\mathbf{P}}_1 \oplus \hat{\mathbf{P}}_2 = \left[\begin{array}{cc|cc} \hat{\mathbf{P}}_{11} & \hat{\mathbf{P}}_{21} & \mathbf{0} & \mathbf{0} \\ \hat{\mathbf{P}}_{21}^T & \hat{\mathbf{P}}_{31} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \hat{\mathbf{P}}_{12} & \hat{\mathbf{P}}_{22} \\ \mathbf{0} & \mathbf{0} & \hat{\mathbf{P}}_{22}^T & \hat{\mathbf{P}}_{32} \end{array}\right] \tag{3.15a}$$

$$\hat{\mathbf{Q}} = \hat{\mathbf{Q}}_1 \oplus \hat{\mathbf{Q}}_2 = \left[\begin{array}{cc|cc} \hat{\mathbf{Q}}_{11} & \hat{\mathbf{Q}}_{21} & \mathbf{0} & \mathbf{0} \\ \hat{\mathbf{Q}}_{21}^T & \hat{\mathbf{Q}}_{31} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \hat{\mathbf{Q}}_{12} & \hat{\mathbf{Q}}_{22} \\ \mathbf{0} & \mathbf{0} & \hat{\mathbf{Q}}_{22}^T & \hat{\mathbf{Q}}_{32} \end{array}\right] \tag{3.15b}$$

where

$$\hat{\mathbf{P}}_1 \in \Re^{(m+m_i)\times(m+m_i)}, \qquad \hat{\mathbf{P}}_2 \in \Re^{(n+n_i)\times(n+n_i)}$$

$$\hat{\mathbf{Q}}_1 \in \Re^{(m+m_o)\times(m+m_o)}, \qquad \hat{\mathbf{Q}}_2 \in \Re^{(n+n_o)\times(n+n_o)}$$

that satisfy two 2-D *Lyapunov inequalities* given by

$$\hat{\mathbf{A}}^i\hat{\mathbf{P}}\hat{\mathbf{A}}^{iT} - \hat{\mathbf{P}} + \hat{\mathbf{B}}^i\hat{\mathbf{B}}^{iT} < \mathbf{0} \tag{3.16a}$$

$$\hat{\mathbf{A}}^{o^T}\hat{\mathbf{Q}}\hat{\mathbf{A}}^{o} - \hat{\mathbf{Q}} + \hat{\mathbf{C}}^{o^T}\hat{\mathbf{C}}^{o} < \mathbf{0} \tag{3.16b}$$

These inequalities enable the WSBR defined in this section to be applied to *MIMO* 2-D weighted discrete systems depicted in Figure 3.1.

From (3.5a-d), it is observed that the four blocks of the system matrix of $\mathbf{H}(z_1, z_2)$, namely, $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$, and $\mathbf{A}_4$ defined in (2.1a), occur in both $\hat{\mathbf{A}}^i$ and $\hat{\mathbf{A}}^o$ as the (1, 1), (1, 3), (3, 1), and (3, 3) blocks. From (3.16a, b), the following definition can be formulated.

Definition 3.2

The *weighted structured controllability and observability gramians* of the 2-D weighted discrete system depicted in Figure 3.1 are denoted by

$$\underline{\mathbf{P}} = \underline{\mathbf{P}}_1 \oplus \underline{\mathbf{P}}_2 \quad \text{and} \quad \underline{\mathbf{Q}} = \underline{\mathbf{Q}}_1 \oplus \underline{\mathbf{Q}}_2$$

and defined by[1]

$$\underline{\mathbf{P}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{bmatrix} \hat{\mathbf{P}} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{P}}_{11} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{P}}_{12} \end{bmatrix} \tag{3.17a}$$

$$\underline{\mathbf{Q}} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \end{bmatrix} \hat{\mathbf{Q}} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{Q}}_{11} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{Q}}_{12} \end{bmatrix} \tag{3.17b}$$

where $\hat{\mathbf{P}}$ and $\hat{\mathbf{Q}}$ are solutions of the 2-D Lyapunov inequalities (3.16a, b).

[1]The underline is herein reserved for matrices which are directly related to the WSBR.

Definition 3.3

The 2-D weighted discrete system depicted in Figure 3.1 is said to be *weighted structurally balanced* if the weighted structured gramians satisfy

$$\underline{\mathbf{P}}_1 = \underline{\mathbf{Q}}_1 = \underline{\mathbf{\Sigma}}_1 = \text{diag}\,(\sigma_1^w, \sigma_2^w, \ldots, \sigma_m^w) \tag{3.18a}$$

$$\underline{\mathbf{P}}_2 = \underline{\mathbf{Q}}_2 = \underline{\mathbf{\Sigma}}_2 = \text{diag}\,(\mu_1^w, \mu_2^w, \ldots, \mu_n^w) \tag{3.18b}$$

where

$$\sigma_1^w \geq \sigma_2^w \geq \cdots \geq \sigma_m^w \geq 0$$

$$\mu_1^w \geq \mu_2^w \geq \cdots \geq \mu_n^w \geq 0$$

are said to be the *weighted structured Hankel singular values* of the system.

If the 2-D Lyapunov inequalities (3.16a, b) have solutions $\hat{\mathbf{P}}$ and $\hat{\mathbf{Q}}$, the Algorithm 2.1 can be applied to $\{\hat{\mathbf{P}}_{11}, \hat{\mathbf{Q}}_{11}\}$ and $\{\hat{\mathbf{P}}_{12}, \hat{\mathbf{Q}}_{12}\}$ in order to find nonsingular matrices $\underline{\mathbf{T}}_1$ and $\underline{\mathbf{T}}_2$, respectively.

Having found a nonsingular block-diagonal *weighted balancing transformation matrix*

$$\underline{\mathbf{T}} = \underline{\mathbf{T}}_1 \oplus \underline{\mathbf{T}}_2$$

the 2-D weighted structurally balanced system can be characterized by the weighted structurally balanced matrices (in the Roesser state-space model) $\{\underline{\mathbf{A}}, \underline{\mathbf{B}}, \underline{\mathbf{C}}, \mathbf{D}\}$ with

$$\underline{\mathbf{A}} = (\underline{\mathbf{T}})^{-1}\mathbf{A}\underline{\mathbf{T}}, \qquad \underline{\mathbf{B}} = (\underline{\mathbf{T}})^{-1}\mathbf{B}, \qquad \underline{\mathbf{C}} = \mathbf{C}\,\underline{\mathbf{T}} \tag{3.19}$$

where the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ are given by (2.1a, b).

3.3.2 Computation of Weighted Structured Gramians

As discussed in Section 2.2.3, matrices $\hat{\mathbf{P}}$ and $\hat{\mathbf{Q}}$ that satisfy (3.16a, b) can be obtained by solving the following *constrained* convex minimization problems

$$\underset{\hat{\mathbf{P}}>\mathbf{0}}{\text{minimize}}\ \gamma_{\max}(\hat{\mathbf{A}}^i\hat{\mathbf{P}}\hat{\mathbf{A}}^{iT} - \hat{\mathbf{P}} + \hat{\mathbf{B}}^i\hat{\mathbf{B}}^{iT}) < \mathbf{0} \tag{3.20a}$$

$$\underset{\hat{\mathbf{Q}}>\mathbf{0}}{\text{minimize}}\ \gamma_{\max}(\hat{\mathbf{A}}^{oT}\hat{\mathbf{Q}}\hat{\mathbf{A}}^o - \hat{\mathbf{Q}} + \hat{\mathbf{C}}^{oT}\hat{\mathbf{C}}^o) < \mathbf{0} \tag{3.20b}$$

or the following *unconstrained* optimization problems as discussed in Section 2.4.1

$$\underset{\mathbf{L}_{\hat{\mathbf{P}}}}{\text{minimize}} \left\| (\mathbf{L}_{\hat{\mathbf{P}}})^{-1} \begin{bmatrix} \hat{\mathbf{A}}^i\mathbf{L}_{\hat{\mathbf{P}}} & \hat{\mathbf{B}}^i \end{bmatrix} \right\| < 1 \tag{3.21a}$$

$$\underset{\mathbf{L}_{\hat{\mathbf{Q}}}}{\text{minimize}} \left\| (\mathbf{L}_{\hat{\mathbf{Q}}})^{-1} \begin{bmatrix} \hat{\mathbf{A}}^{oT}\mathbf{L}_{\hat{\mathbf{Q}}} & \hat{\mathbf{C}}^{oT} \end{bmatrix} \right\| < 1 \tag{3.21b}$$

where block-diagonal lower-triangular matrices

$$\mathbf{L}_{\hat{\mathbf{P}}} = \mathbf{L}_{\hat{\mathbf{P}}1} \oplus \mathbf{L}_{\hat{\mathbf{P}}2} \quad \text{and} \quad \mathbf{L}_{\hat{\mathbf{Q}}} = \mathbf{L}_{\hat{\mathbf{Q}}2} \oplus \mathbf{L}_{\hat{\mathbf{Q}}2}$$

are obtained using the Cholesky factorizations of $\hat{\mathbf{P}}$ and $\hat{\mathbf{Q}}$ as

$$\hat{\mathbf{P}} = \mathbf{L}_{\hat{\mathbf{P}}}\,\mathbf{L}_{\hat{\mathbf{P}}}{}^T \quad \text{and} \quad \hat{\mathbf{Q}} = \mathbf{L}_{\hat{\mathbf{Q}}}\,\mathbf{L}_{\hat{\mathbf{Q}}}{}^T \tag{3.22}$$

Note that even for a 2-D weighted system of modest order, say

$$m = n = 15, \quad m_i = n_i = 10, \quad m_o = n_o = 10$$

the number of parameters involved in each of the minimization problems of (3.21a) and (3.21b) is given by

$$[\,(m+m_i)(m+m_i+1) + (n+n_i)(n+n_i+1)\,]/2 = 650$$

and is much larger than that involved in each of (2.23a) and (2.23b). Therefore, it is very helpful if a "good" initial point can be deduced before one starts minimizing the norms in (3.21a, b). Hence, the technique discussed in Section 2.4.1 is also applied here.

3.3.3 Unit Weights

In this section, the SBR (or ISBR), which was discussed in Section 2.2.3, is shown to be a special case of the WSBR defined in Section 3.3.1.

The (unweighted) 2-D system depicted in Figure 2.1 is equivalent to the weighted 2-D system shown in Figure 3.1 if the input and output weights are unity, that is,

$$\mathbf{W}^i(z_1, z_2) = \mathbf{W}^o(z_1, z_2) = \mathbf{I}$$

which implies that

$$\mathbf{A}^i = \mathbf{B}^i = \mathbf{C}^i = \mathbf{0}, \qquad \mathbf{D}^i = \mathbf{I}$$

$$\mathbf{A}^o = \mathbf{B}^o = \mathbf{C}^o = \mathbf{0}, \qquad \mathbf{D}^o = \mathbf{I}$$

and (3.3a, b) become

$$\mathbf{H}^i(z_1, z_2) = [\,\mathbf{I}(z_1, z_2) - \mathbf{A}\,]^{-1}\,\mathbf{B} \tag{3.23a}$$

$$\mathbf{H}^o(z_1, z_2) = \mathbf{C}\,[\,\mathbf{I}(z_1, z_2) - \mathbf{A}\,]^{-1} \tag{3.23b}$$

that is,

$$\hat{\mathbf{A}}^i = \hat{\mathbf{A}}^o = \mathbf{A}, \qquad \hat{\mathbf{B}}^i = \mathbf{B}$$

$$\hat{\mathbf{C}}^i = \hat{\mathbf{B}}^o = \mathbf{I}, \qquad \hat{\mathbf{C}}^o = \mathbf{C}$$

and

$$\hat{\mathbf{P}} = \mathbf{P}^s, \qquad \hat{\mathbf{Q}} = \mathbf{Q}^s$$

Therefore, the 2-D Lyapunov inequalities (3.16a, b) become

$$\mathbf{A}\mathbf{P}^s\mathbf{A}^T - \mathbf{P}^s + \mathbf{B}\mathbf{B}^T < \mathbf{0}$$

$$\mathbf{A}^T\mathbf{Q}^s\mathbf{A} - \mathbf{Q}^s + \mathbf{C}^T\mathbf{C} < \mathbf{0}$$

which are the 2-D Lyapunov inequalities given by (2.11a, b) that define 2-D (unweighted) structured gramians [56] and the structurally balanced realization.

It is well known that the (unweighted) balanced realization is an effective method for the reduction of system order in 1-D and 2-D systems [32, 43, 52, 56, 62]. In the next section, we show that this is also true for the WSBR defined in this section.

3.4 A Weighted Structurally Balanced Model-Reduction

The WSBMR entails finding a Q-stable reduced-order system of order (r_1, r_2) whose transfer-function matrix is denoted by $\underline{\mathbf{H}}^r(z_1, z_2)$, such that for given Q-stable $\mathbf{H}(z_1, z_2)$, $\mathbf{W}^i(z_1, z_2)$, and $\mathbf{W}^o(z_1, z_2)$, *weighted approximation error* [2]

$$\left\| \mathbf{W}^o(z_1, z_2) \left[\mathbf{H}(z_1, z_2) - \underline{\mathbf{H}}^r(z_1, z_2) \right] \mathbf{W}^i(z_1, z_2) \right\|_\infty \tag{3.24}$$

[2]The term *weighted* refers to any error measure involving $\mathbf{W}^i(z_1, z_2)$, $\mathbf{W}^o(z_1, z_2)$.

is minimized with respect to all Q-stable transfer-function matrices of order (r_1, r_2).

Unlike the 1-D case, the reduced-order 2-D system obtained by truncating a full-order 2-D *weighted structurally balanced* system using the WSBR method may not be satisfactory in terms of the weighted approximation error. This is mainly due to the fact that there are infinitely many solutions to the 2-D Lyapunov inequalities (3.16a, b) and, further, the analytic relation of these solutions to the approximation error introduced by a simple truncation remains unclear. In what follows, a WSBMR that takes both the stability issue and approximation error into account is proposed. The Q-stability of the resulting reduced-order weighted system is then demonstrated.

3.4.1 The Method

The *weighted structured Hankel singular values* are defined as the square roots of the eigenvalues of $\underline{\mathbf{P}}\,\underline{\mathbf{Q}}$, where $\underline{\mathbf{P}}$ and $\underline{\mathbf{Q}}$ are the weighted structured gramians of the weighted system depicted in Figure 3.1. It can be readily verified that this definition is equivalent to that defined in (3.18a, b).

The weighted structured Hankel singular values so defined are not, in general, invariant under state-variable transformation since the solutions of Lyapunov inequalities (3.16a, b) are not unique. In other words, the weighted structured Hankel singular values are $\underline{\mathbf{P}}$ and $\underline{\mathbf{Q}}$ dependent. As discussed in Section 2.5.1, this dependence provides an approach to specify a suitable solution of (3.16a, b) so as to achieve small approximation error. The modified

unconstrained minimization problem is

$$\underset{\mathbf{L}_{\hat{\mathrm{P}}},\mathbf{L}_{\hat{\mathrm{Q}}}}{\text{minimize}}\ f(\mathbf{L}_{\hat{\mathrm{P}}},\mathbf{L}_{\hat{\mathrm{Q}}}) \tag{3.25}$$

where

$$f(\mathbf{L}_{\hat{\mathrm{P}}},\mathbf{L}_{\hat{\mathrm{Q}}}) = k_0\, f_1(\mathbf{L}_{\hat{\mathrm{P}}}) + k_1\, f_2(\mathbf{L}_{\hat{\mathrm{Q}}}) + k_2\, f_e(\mathbf{L}_{\hat{\mathrm{P}}},\mathbf{L}_{\hat{\mathrm{Q}}})$$

$$f_1(\mathbf{L}_{\hat{\mathrm{P}}}) = \left\| (\mathbf{L}_{\hat{\mathrm{P}}})^{-1}\left[\begin{array}{cc} \hat{\mathbf{A}}^i\mathbf{L}_{\hat{\mathrm{P}}} & \hat{\mathbf{B}}^i \end{array}\right]\right\|$$

$$f_2(\mathbf{L}_{\hat{\mathrm{Q}}}) = \left\| (\mathbf{L}_{\hat{\mathrm{Q}}})^{-1}\left[\begin{array}{cc} \hat{\mathbf{A}}^{oT}\mathbf{L}_{\hat{\mathrm{Q}}} & \hat{\mathbf{C}}^{oT} \end{array}\right]\right\|$$

$$f_e(\mathbf{L}_{\hat{\mathrm{P}}},\mathbf{L}_{\hat{\mathrm{Q}}}) = \sum_{j=r_1+1}^{m}\sigma_j^w + \sum_{\ell=r_2+1}^{n}\mu_\ell^w$$

and $k_0,\ k_1,\ k_2 > 0$ are weighted scalars, which can be appropriately selected such that $f_2(\mathbf{L}_{\hat{\mathrm{P}}})$ and $f_2(\mathbf{L}_{\hat{\mathrm{Q}}})$ are as close to unity as possible and the inequalities

$$f_1(\mathbf{L}_{\hat{\mathrm{Q}}}) < 1 \quad \text{and} \quad f_2(\mathbf{L}_{\hat{\mathrm{Q}}}) < 1 \tag{3.26}$$

are satisfied. With the appropriate selection of k_0, k_1 and k_2, this optimization problem suppresses the least weighted structured Hankel singular values

$$\{\sigma_{r_1+1}^w,\ \sigma_{r_1+2}^w,\ \ldots,\ \sigma_m^w\} \quad \text{and} \quad \{\mu_{r_2+1}^w,\ \mu_{r_2+2}^w,\ \ldots,\ \mu_n^w\}$$

so as to reduce the approximation error. The following is a step-by-step summary of the proposed WSBMR algorithm.

Algorithm 3.1

Step 1: Find a local minimum of $f(\mathbf{L}_{\hat{\mathrm{P}}},\mathbf{L}_{\hat{\mathrm{Q}}})$ in (3.25) such that the in-

equalities (3.26) are satisfied. The optimization is unconstrained and can be carried out using established numerical optimization techniques [14]. The weighted scalars k_0, k_1 and k_2 can be adjusted to find the local minimum points $(\mathbf{L}_{\hat{\mathbf{P}}}, \mathbf{L}_{\hat{\mathbf{Q}}})$ such that the constraints in (3.26) are satisfied.

Step 2: Compute

$$\hat{\mathbf{P}} = \mathbf{L}_{\hat{\mathbf{P}}}\,\mathbf{L}_{\hat{\mathbf{P}}}^{T} \quad \text{and} \quad \hat{\mathbf{Q}} = \mathbf{L}_{\hat{\mathbf{Q}}}\,\mathbf{L}_{\hat{\mathbf{Q}}}^{T}$$

Step 3: Obtain the weighted structured gramians, $\underline{\mathbf{P}}$ and $\underline{\mathbf{Q}}$, using the relations

$$\underline{\mathbf{P}} = \begin{bmatrix} \underline{\mathbf{P}}_1 & \mathbf{0} \\ \mathbf{0} & \underline{\mathbf{P}}_2 \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{P}}_{11} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{P}}_{12} \end{bmatrix}$$

$$\underline{\mathbf{Q}} = \begin{bmatrix} \underline{\mathbf{Q}}_1 & \mathbf{0} \\ \mathbf{0} & \underline{\mathbf{Q}}_2 \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{Q}}_{11} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{Q}}_{12} \end{bmatrix}$$

Step 4: Apply Algorithm 2.1 to $\{\underline{\mathbf{P}}_1,\ \underline{\mathbf{Q}}_1\}$ and $\{\underline{\mathbf{P}}_2,\ \underline{\mathbf{Q}}_2\}$ in order to find nonsingular matrices $\underline{\mathbf{T}}_1$ and $\underline{\mathbf{T}}_2$, respectively. Then, construct a weighted balancing transformation matrix

$$\underline{\mathbf{T}} = \underline{\mathbf{T}}_1 \oplus \underline{\mathbf{T}}_2$$

Step 5: Obtain the *weighted structurally balanced realization* $\{\underline{\mathbf{A}}, \underline{\mathbf{B}}, \underline{\mathbf{C}}, \mathbf{D}\}$ with

$$\underline{\mathbf{A}} = \underline{\mathbf{T}}^{-1}\,\mathbf{A}\,\underline{\mathbf{T}}, \qquad \underline{\mathbf{B}} = \underline{\mathbf{T}}^{-1}\,\mathbf{B}, \qquad \underline{\mathbf{C}} = \mathbf{C}\,\underline{\mathbf{T}}$$

Step 6: Partition the matrices $\underline{\mathbf{A}}$, $\underline{\mathbf{B}}$, and $\underline{\mathbf{C}}$ as

$$\underline{\mathbf{A}} = \left[\begin{array}{cc|cc} \underline{\mathbf{A}}_1^r & \underline{\mathbf{A}}_{12} & \underline{\mathbf{A}}_2^r & \underline{\mathbf{A}}_{22} \\ \underline{\mathbf{A}}_{13} & \underline{\mathbf{A}}_{14} & \underline{\mathbf{A}}_{23} & \underline{\mathbf{A}}_{24} \\ \hline \underline{\mathbf{A}}_3^r & \underline{\mathbf{A}}_{32} & \underline{\mathbf{A}}_4^r & \underline{\mathbf{A}}_{42} \\ \underline{\mathbf{A}}_{33} & \underline{\mathbf{A}}_{34} & \underline{\mathbf{A}}_{43} & \underline{\mathbf{A}}_{44} \end{array}\right], \quad \underline{\mathbf{B}} = \left[\begin{array}{c} \underline{\mathbf{B}}_1^r \\ \underline{\mathbf{B}}_{12} \\ \hline \underline{\mathbf{B}}_2^r \\ \underline{\mathbf{B}}_{22} \end{array}\right], \quad \underline{\mathbf{C}} = \left[\begin{array}{c} \underline{\mathbf{C}}_1^{rT} \\ \underline{\mathbf{C}}_{12}{}^T \\ \hline \underline{\mathbf{C}}_2^{rT} \\ \underline{\mathbf{C}}_{22}{}^T \end{array}\right]^T$$

where

$$\underline{\mathbf{A}}_1^r \in \Re^{r_1 \times r_1}, \quad \underline{\mathbf{A}}_2^r \in \Re^{r_1 \times r_2}, \quad \underline{\mathbf{B}}_1^r \in \Re^{r_1 \times t}, \quad \underline{\mathbf{C}}_1^r \in \Re^{s \times r_1}$$

$$\underline{\mathbf{A}}_3^r \in \Re^{r_2 \times r_1}, \quad \underline{\mathbf{A}}_4^r \in \Re^{r_2 \times r_2}, \quad \underline{\mathbf{B}}_2^r \in \Re^{r_2 \times t}, \quad \underline{\mathbf{C}}_2^r \in \Re^{s \times r_2}$$

Then, form the reduced-order weighted 2-D system of order (r_1, r_2) as $\{\underline{\mathbf{A}}^r, \underline{\mathbf{B}}^r, \underline{\mathbf{C}}^r, \mathbf{D}\}$ with

$$\underline{\mathbf{A}}^r = \begin{bmatrix} \underline{\mathbf{A}}_1^r & \underline{\mathbf{A}}_2^r \\ \underline{\mathbf{A}}_3^r & \underline{\mathbf{A}}_4^r \end{bmatrix}, \quad \underline{\mathbf{B}}^r = \begin{bmatrix} \underline{\mathbf{B}}_1^r \\ \underline{\mathbf{B}}_2^r \end{bmatrix}, \quad \underline{\mathbf{C}}^r = \begin{bmatrix} \underline{\mathbf{C}}_1^r & \underline{\mathbf{C}}_2^r \end{bmatrix}$$

Step 7: Obtain the transfer-function matrix of the reduced-order weighted system as

$$\underline{\mathbf{H}}^r(z_1, z_2) = \underline{\mathbf{C}}^r \left[\mathbf{I}(z_1, z_2) - \underline{\mathbf{A}}^r \right]^{-1} \underline{\mathbf{B}}^r + \mathbf{D} \tag{3.27}$$

The reduced-order weighted system obtained from the above WSBMR method is shown to be Q-stable in the following theorem.

3.4.2 Q-Stability of the Reduced-Order Weighted System

In this section, it is shown that for the single-input single-output (SISO) case, the reduced-order systems are Q-stable for any Q-stable input and output

weights $\mathbf{W}^i(z_1, z_2)$ and $\mathbf{W}^o(z_1, z_2)$; and that for the MIMO case, the Q-stability of the reduced-order system holds under some additional conditions.

Theorem 3.2

If the (unweighted) 2-D system in Figure 2.1 and output weight are Q-stable, and the input weight is unity (or if the unweighted 2-D system in Figure 2.1 and input weight are Q-stable, and the output weight is unity), then the reduced-order weighted system obtained from the WSBMR method in Section 3.4.1 is also Q-stable.

Proof

If the (unweighted) 2-D system in Figure 2.1 and output weights are Q-stable with unit input weight, that is,

$$\mathbf{W}^i(z_1, z_2) = \mathbf{I}$$

then (3.3a) becomes

$$\mathbf{H}^i(z_1, z_2) = [\, \mathbf{I}(z_1, z_2) - \mathbf{A} \,]^{-1} \mathbf{B} \tag{3.28}$$

Comparison of (3.28) with (3.4a) leads to

$$\hat{\mathbf{A}}^i = \mathbf{A} \quad \text{and} \quad \hat{\mathbf{B}}^i = \mathbf{B}$$

and (3.16a) becomes

$$\mathbf{A}\mathbf{P}^s\mathbf{A}^T - \mathbf{P}^s + \mathbf{B}\mathbf{B}^T < \mathbf{0} \tag{3.29}$$

Premultiplying (3.29) by $\underline{\mathbf{T}}^{-1}$ and postmultiplying by $\underline{\mathbf{T}}^{-T}$ leads to

$$\underline{\mathbf{A}}\underline{\mathbf{\Sigma}}\underline{\mathbf{A}}^T - \underline{\mathbf{\Sigma}} + \underline{\mathbf{B}}\underline{\mathbf{B}}^T < \mathbf{0} \tag{3.30}$$

where

$$\underline{\Sigma} = \underline{\Sigma}_1 \oplus \underline{\Sigma}_2$$

Premultiplying inequality (3.30) by $\hat{\mathbf{I}}$ and postmultiplying by $\hat{\mathbf{I}}^T$, then

$$\tilde{\underline{A}}\tilde{\underline{\Sigma}}\tilde{\underline{A}}^T - \tilde{\underline{\Sigma}} + \tilde{\underline{B}}\tilde{\underline{B}}^T < 0 \tag{3.31}$$

where

$$\tilde{\underline{A}} = \left[\begin{array}{cc|cc} \underline{A}^r{}_1 & \underline{A}^r{}_2 & \underline{A}_{12} & \underline{A}_{22} \\ \underline{A}^r{}_3 & \underline{A}^r{}_4 & \underline{A}_{32} & \underline{A}_{42} \\ \hline \underline{A}_{13} & \underline{A}_{23} & \underline{A}_{14} & \underline{A}_{24} \\ \underline{A}_{33} & \underline{A}_{43} & \underline{A}_{34} & \underline{A}_{44} \end{array}\right] = \left[\begin{array}{cc} \underline{A}^r & \tilde{\underline{A}}_2 \\ \tilde{\underline{A}}_3 & \tilde{\underline{A}}_4 \end{array}\right]$$

$$\tilde{\underline{B}} = \left[\begin{array}{cc|cc} \underline{B}_1^{rT} & \underline{B}_2^{rT} & \underline{B}_{12}^T & \underline{B}_{22}^T \end{array}\right]^T = \left[\begin{array}{cc} \underline{B}^{rT} & \tilde{\underline{B}}_2^T \end{array}\right]^T$$

$$\tilde{\underline{\Sigma}} = \hat{\mathbf{I}}\underline{\Sigma}\hat{\mathbf{I}}^T = \underline{\Sigma}^r \oplus \tilde{\underline{\Sigma}}_2$$

and

$$\underline{\Sigma}^r = \text{diag}\,(\sigma_1^w, \ldots, \sigma_{r_1}^w, \mu_1^w, \ldots, \mu_{r_2}^w)$$

$$\tilde{\underline{\Sigma}}_2 = \text{diag}\,(\sigma_{r_1+1}^w, \ldots, \sigma_m^w, \mu_{r_2+1}^w, \ldots, \mu_n^w)$$

The inequality (3.31) can be expressed in partitioned form as

$$\left[\begin{array}{cc} \underline{A}^r\underline{\Sigma}^r\underline{A}^{rT} + \tilde{\underline{A}}_2\tilde{\underline{\Sigma}}_2\tilde{\underline{A}}_2^T - \underline{\Sigma}^r + \underline{B}^r\underline{B}^{rT} & \underline{A}^r\underline{\Sigma}^r\tilde{\underline{A}}_3^T + \tilde{\underline{A}}_2\tilde{\underline{\Sigma}}_2\tilde{\underline{A}}_4^T + \underline{B}^r\tilde{\underline{B}}_2^T \\ \tilde{\underline{A}}_3\underline{\Sigma}^r\underline{A}^{rT} + \tilde{\underline{A}}_4\tilde{\underline{\Sigma}}_2\tilde{\underline{A}}_2^T + \tilde{\underline{B}}_2\underline{B}^{rT} & \tilde{\underline{A}}_3\underline{\Sigma}^r\tilde{\underline{A}}_3^T + \tilde{\underline{A}}_4\tilde{\underline{\Sigma}}_2\tilde{\underline{A}}_4^T - \tilde{\underline{\Sigma}}_2 + \tilde{\underline{B}}_2\tilde{\underline{B}}_2^T \end{array}\right] < 0$$

Therefore,

$$\underline{\mathbf{A}}^r \underline{\mathbf{\Sigma}}^r \underline{\mathbf{A}}^{rT} + \tilde{\underline{\mathbf{A}}}_2 \tilde{\underline{\mathbf{\Sigma}}}_2 \tilde{\underline{\mathbf{A}}}_2^T - \underline{\mathbf{\Sigma}}^r + \underline{\mathbf{B}}^r \underline{\mathbf{B}}^{rT} < \mathbf{0}$$

Since $\underline{\mathbf{\Sigma}} \geq \mathbf{0}$,

$$\tilde{\underline{\mathbf{A}}}_2 \tilde{\underline{\mathbf{\Sigma}}}_2 \tilde{\underline{\mathbf{A}}}_2^T \geq \mathbf{0}$$

we conclude that

$$\underline{\mathbf{A}}^r \underline{\mathbf{\Sigma}}^r \underline{\mathbf{A}}^{rT} - \underline{\mathbf{\Sigma}}^r + \underline{\mathbf{B}}^r \underline{\mathbf{B}}^{rT} < \mathbf{0} \tag{3.32}$$

which implies the Q-stability of the reduced-order weighted system $\{\underline{\mathbf{A}}^r, \underline{\mathbf{B}}^r, \underline{\mathbf{C}}^r, \mathbf{D}\}$. □

An immediate consequence of Theorem 3.2 is the Q-stability of the reduced-order systems for the SISO case.

Lemma 3.1

For the SISO case, if the (unweighted) 2-D system in Figure 2.1 and the input and output weights, $\mathbf{W}^i(z_1, z_2)$ and $\mathbf{W}^o(z_1, z_2)$, are all Q-stable, then the reduced-order system obtained using the WSBMR method in Section 3.4.1 is also Q-stable.

Theorem 3.3

If the (unweighted) 2-D system in Figure 2.1 and the input and output weights, $\mathbf{W}^i(z_1, z_2)$ and $\mathbf{W}^o(z_1, z_2)$, are Q-stable; and if $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$ in (3.15a) are block-diagonal (or $\hat{\mathbf{Q}}_1$ and $\hat{\mathbf{Q}}_2$ in (3.15b) are block-diagonal), then the

reduced-order weighted system of order (r_1, r_2) obtained from the WSBMR method in Section 3.4.1 is also Q-stable.

Proof

If the (unweighted) 2-D system in Figure 2.1 and input weight are Q-stable, then from (3.14) there exist block-diagonal matrices $\hat{\mathbf{P}}_1$ and $\hat{\mathbf{P}}_2$ such that

$$\hat{\mathbf{P}} = \begin{bmatrix} \hat{\mathbf{P}}_1 & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{P}}_2 \end{bmatrix} = \left[\begin{array}{cc|cc} \alpha\mathbf{Y}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}_1^i & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \alpha\mathbf{Y}_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Y}_2^i \end{array}\right]$$

satisfies the 2-D Lyapunov inequality (3.16a). Premultiplying the Lyapunov inequality (3.16a) by $\hat{\mathbf{I}}$ and postmultiplying by $\hat{\mathbf{I}}^T$, then,

$$\tilde{\mathbf{A}}^i\tilde{\mathbf{P}}\tilde{\mathbf{A}}^{iT} - \tilde{\mathbf{P}} + \tilde{\mathbf{B}}^i\tilde{\mathbf{B}}^{iT} < \mathbf{0} \tag{3.33}$$

where $\tilde{\mathbf{A}}^i$ and $\tilde{\mathbf{B}}^i$ are given in (3.9), and

$$\tilde{\mathbf{P}} = \left[\begin{array}{cc|cc} \alpha\mathbf{Y}_1 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \alpha\mathbf{Y}_2 & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{Y}_1^i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{Y}_2^i \end{array}\right] = \begin{bmatrix} \underline{\mathbf{P}} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}^i \end{bmatrix}$$

Define

$$\mathbf{J} = \begin{bmatrix} \underline{\mathbf{T}} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$$

Premultiplying inequality by $\mathbf{J}^{-1}$ and postmultiplying by $\mathbf{J}^{-T}$, inequality (3.33) can be rewritten as

$$\check{\mathbf{A}}^i\check{\mathbf{P}}\check{\mathbf{A}}^{iT} - \check{\mathbf{P}} + \check{\mathbf{B}}^i\check{\mathbf{B}}^{iT} < \mathbf{0} \tag{3.34}$$

where

$$\check{\mathbf{A}}^i = \begin{bmatrix} \underline{\mathbf{A}} & \underline{\mathbf{B}}\mathbf{C}^i \\ \mathbf{0} & \mathbf{A}^i \end{bmatrix}, \qquad \check{\mathbf{B}}^i = \begin{bmatrix} \underline{\mathbf{B}}\mathbf{D}^i \\ \mathbf{B}^i \end{bmatrix}, \qquad \check{\mathbf{P}} = \begin{bmatrix} \underline{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{Y}^i \end{bmatrix}$$

The partitioned form of inequality (3.34) is given by

$$\begin{bmatrix} \Gamma_1 & \Gamma_2 \\ \Gamma_2^T & \Psi^i \end{bmatrix} < 0 \tag{3.35}$$

where

$$\Gamma_1 = \underline{\mathbf{A}}\underline{\Sigma}\underline{\mathbf{A}}^T - \underline{\Sigma} + \underline{\mathbf{B}}(\mathbf{C}^i\mathbf{Y}^i\mathbf{C}^{iT} + \mathbf{D}^i\mathbf{D}^{iT})\underline{\mathbf{B}}^T$$

$$\Gamma_2 = \underline{\mathbf{B}}(\mathbf{C}^i\mathbf{Y}^i\mathbf{A}^{iT} + \mathbf{D}^i\mathbf{B}^{iT})$$

Premultiplying (3.35) by

$$\hat{\mathbf{X}} = \begin{bmatrix} \mathbf{I} & -\Gamma_2(\Psi^i)^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$$

and postmultiplying by $\hat{\mathbf{X}}^T$, leads to

$$\begin{bmatrix} \Gamma_1 - \Gamma_2(\Psi^i)^{-1}\Gamma_2^T & \mathbf{0} \\ \mathbf{0} & \Psi^i \end{bmatrix} < 0$$

Since $\Psi^i < \mathbf{0}$,

$$\Gamma_1 \leq \Gamma_1 - \Gamma_2(\Psi^i)^{-1}\Gamma_2^T < 0$$

Also $\mathbf{Y}^i > \mathbf{0}$, and thus

$$\underline{\mathbf{A}}\underline{\boldsymbol{\Sigma}}\underline{\mathbf{A}}^T - \underline{\boldsymbol{\Sigma}} \leq \boldsymbol{\Gamma}_1 < \mathbf{0} \tag{3.36}$$

Premultiplying inequality (3.36) by $\hat{\mathbf{I}}$ and postmultiplying by $\hat{\mathbf{I}}^T$, then

$$\tilde{\underline{\mathbf{A}}}\tilde{\underline{\boldsymbol{\Sigma}}}\tilde{\underline{\mathbf{A}}}^T - \tilde{\underline{\boldsymbol{\Sigma}}} < \mathbf{0} \tag{3.37}$$

Inequality (3.37) can be expressed in partitioned form as

$$\begin{bmatrix} \underline{\mathbf{A}}^r\underline{\boldsymbol{\Sigma}}^r\underline{\mathbf{A}}^{rT} + \tilde{\underline{\mathbf{A}}}_2\tilde{\underline{\boldsymbol{\Sigma}}}_2\tilde{\underline{\mathbf{A}}}_2^T - \underline{\boldsymbol{\Sigma}}^r & \underline{\mathbf{A}}^r\underline{\boldsymbol{\Sigma}}^r\tilde{\underline{\mathbf{A}}}_3^T + \tilde{\underline{\mathbf{A}}}_2\tilde{\underline{\boldsymbol{\Sigma}}}_2\tilde{\underline{\mathbf{A}}}_4^T \\ \tilde{\underline{\mathbf{A}}}_3\underline{\boldsymbol{\Sigma}}^r\underline{\mathbf{A}}^{rT} + \tilde{\underline{\mathbf{A}}}_4\tilde{\underline{\boldsymbol{\Sigma}}}_2\tilde{\underline{\mathbf{A}}}_2^T & \tilde{\underline{\mathbf{A}}}_3\underline{\boldsymbol{\Sigma}}^r\tilde{\underline{\mathbf{A}}}_3^T + \tilde{\underline{\mathbf{A}}}_4\tilde{\underline{\boldsymbol{\Sigma}}}_2\tilde{\underline{\mathbf{A}}}_4^T - \tilde{\underline{\boldsymbol{\Sigma}}}_2 \end{bmatrix} < \mathbf{0}$$

Hence

$$\underline{\mathbf{A}}^r\underline{\boldsymbol{\Sigma}}^r\underline{\mathbf{A}}^{rT} + \tilde{\underline{\mathbf{A}}}_2\tilde{\underline{\boldsymbol{\Sigma}}}_2\tilde{\underline{\mathbf{A}}}_2^T - \underline{\boldsymbol{\Sigma}}^r < \mathbf{0}$$

and

$$\underline{\mathbf{A}}^r\underline{\boldsymbol{\Sigma}}^r\underline{\mathbf{A}}^{rT} - \underline{\boldsymbol{\Sigma}}^r < \mathbf{0} \tag{3.38}$$

which implies the Q-stability of the reduced-order weighted system $\{\underline{\mathbf{A}}^r, \underline{\mathbf{B}}^r, \underline{\mathbf{C}}^r, \mathbf{D}\}$. □

3.4.3 Performance Evaluation

The Example 2.2 is used to demonstrate the proposed WSBMR algorithm. The data is given in Section 2.5.2. The amplitude response of the original filter of order (4, 8) is depicted in Figure 2.2. A Q-stable lowpass filter of

order (4, 4) is used as the input weight $\mathbf{W}^i(z_1, z_2)$ to emphasize the reduced-order weighted system over the low frequency region. This input weight is modeled by the Roesser state-space model in (2.1a, b) with

$$\mathbf{A}_1^i = \begin{bmatrix} 0.865266 & -0.320202 & -0.038859 & -0.029287 \\ 0.320207 & 0.669571 & -0.345662 & -0.047159 \\ -0.038856 & 0.345681 & 0.488474 & -0.405720 \\ 0.029288 & -0.047162 & 0.405698 & 0.359184 \end{bmatrix}$$

$$\mathbf{A}_2^i = \begin{bmatrix} 0.376026 & 0.327476 & 0.168483 & 0.057372 \\ -0.329114 & -0.286621 & -0.147464 & -0.050214 \\ 0.168784 & 0.146991 & 0.075626 & 0.025752 \\ -0.057005 & -0.049645 & -0.025542 & -0.008697 \end{bmatrix}$$

$$\mathbf{A}_3^i = \begin{bmatrix} 0.002027 & 0.002826 & 0.003092 & 0.002087 \\ -0.002002 & -0.003350 & -0.004149 & -0.002547 \\ 0.000960 & 0.002303 & 0.003304 & 0.001774 \\ -0.001168 & -0.002485 & -0.002790 & -0.000828 \end{bmatrix}$$

$$\mathbf{A}_4^i = \begin{bmatrix} 0.865687 & -0.320519 & -0.039609 & -0.029425 \\ 0.320210 & 0.669343 & -0.346448 & -0.046520 \\ -0.038457 & 0.345556 & 0.488348 & -0.403728 \\ 0.029377 & -0.047308 & 0.407151 & 0.359117 \end{bmatrix}$$

$$\mathbf{B}_1^i = \begin{bmatrix} -0.014868 & 0.013013 & -0.006674 & 0.002254 \end{bmatrix}^T$$

$$\mathbf{B}_2^i = \begin{bmatrix} -0.263117 & 0.228922 & -0.118278 & 0.039798 \end{bmatrix}^T$$

$$\mathbf{C}_1^i = \begin{bmatrix} -0.263607 & -0.228362 & -0.118098 & -0.039582 \end{bmatrix}$$

$$\mathbf{C}_2^i = \begin{bmatrix} -0.014877 & -0.012956 & -0.006666 & -0.002270 \end{bmatrix}$$

$$\mathbf{D}^i = 5.8826 \times 10^{-4}$$

The amplitude response of the input weight is depicted in Figure 3.3. The WSBMR with $\mathbf{W}^i(z_1, z_2)$,

$$\mathbf{W}^o(z_1, z_2) = \mathbf{I}, \qquad k_0 = k_1 = 1, \qquad \text{and} \qquad k_2 = 0.27$$

leads to a *weighted* reduced-order filter of order (4, 4) characterized by $\mathbf{H}^w(z_1, z_2)$. The state-space model of the filter is given by (2.1a, b) with

$$\underline{\mathbf{A}}_1^r = \begin{bmatrix} 0.377205 & -0.029050 & -0.031851 & -0.020335 \\ -0.263009 & 0.233722 & 0.303009 & 0.169255 \\ 0.172187 & 0.021959 & 0.430882 & 0.129212 \\ -0.180399 & -0.077882 & -0.287881 & 0.034010 \end{bmatrix}$$

$$\underline{\mathbf{A}}_2^r = \begin{bmatrix} -0.479988 & 0.458503 & 0.293809 & -0.119313 \\ -0.100826 & 0.195997 & 0.250964 & 0.148250 \\ -0.093182 & -0.143636 & -0.384634 & -0.427647 \\ -0.229037 & 0.021276 & -0.234765 & -0.400324 \end{bmatrix}$$

$$\underline{\mathbf{A}}_3^r = \begin{bmatrix} -0.486739 & 0.068631 & 0.073475 & -0.035764 \\ 0.174899 & 0.439104 & 0.094213 & 0.276255 \\ -0.054904 & 0.241124 & -0.424258 & 0.004268 \\ 0.031853 & 0.135682 & 0.104322 & -0.530429 \end{bmatrix}$$

$$\underline{\mathbf{A}}_4^r = \begin{bmatrix} 0.400167 & 0.430839 & -0.045702 & -0.098613 \\ 0.051712 & 0.162037 & -0.060519 & -0.023039 \\ 0.095767 & -0.273725 & 0.226859 & -0.057916 \\ 0.040648 & 0.232954 & -0.170824 & 0.078143 \end{bmatrix}$$

$$\underline{\mathbf{B}}_1^r = \begin{bmatrix} 0.277210 & 0.201779 & -0.281204 & -0.152142 \end{bmatrix}^T$$

$$\underline{\mathbf{B}}_2^r = \begin{bmatrix} -0.157826 & -0.010694 & 0.263263 & 0.142137 \end{bmatrix}^T$$

$$\underline{\mathbf{C}}_1^r = \begin{bmatrix} 0.129210 & -0.026848 & 0.146844 & -0.088941 \end{bmatrix}^T$$

$$\underline{\mathbf{C}}_2^r = \begin{bmatrix} -0.419206 & 0.103558 & -0.591489 & 0.448008 \end{bmatrix}^T$$

$$\mathbf{D} = 1.34044 \times 10^{-3}$$

This weighted reduced-order filter of order (4, 4) is Q-stable since

$$\left\| \underline{\mathbf{A}}^r \right\| = 0.9387 < 1$$

Its amplitude response is shown in Figure 3.4. To compare $\mathbf{H}^w(z_1, z_2)$ with the transfer function $\mathbf{H}^u(z_1, z_2)$ of the (unweighted) reduced-order filter of order (4, 4) obtained from ISBMR in Section 2.5.1 whose amplitude response is shown in Figure 2.6, the contour plots for the original filter, the (unweighted) reduced-order filter of order (4,4) by the ISBMR (corresponding to Figure 2.6), the input weight, and the weighted reduced-order filter of order (4,4) using WSBMR are given in Figures 3.5 – 3.8, respectively.

The approximation-error matrices of the ISBMR and WSBMR can be defined as

$$\mathbf{E}^u = (E_{kl}^u) \quad \text{and} \quad \mathbf{E}^w = (E_{kl}^w)$$

where

$$E_{kl}^u = \left| H(e^{j2\pi k/K}, e^{j2\pi l/L}) - H^u(e^{j2\pi k/K}, e^{j2\pi l/L}) \right| \qquad (3.39a)$$

$$E_{kl}^w = \left| H(e^{j2\pi k/K}, e^{j2\pi l/L}) - H^w(e^{j2\pi k/K}, e^{j2\pi l/L}) \right| \qquad (3.39b)$$

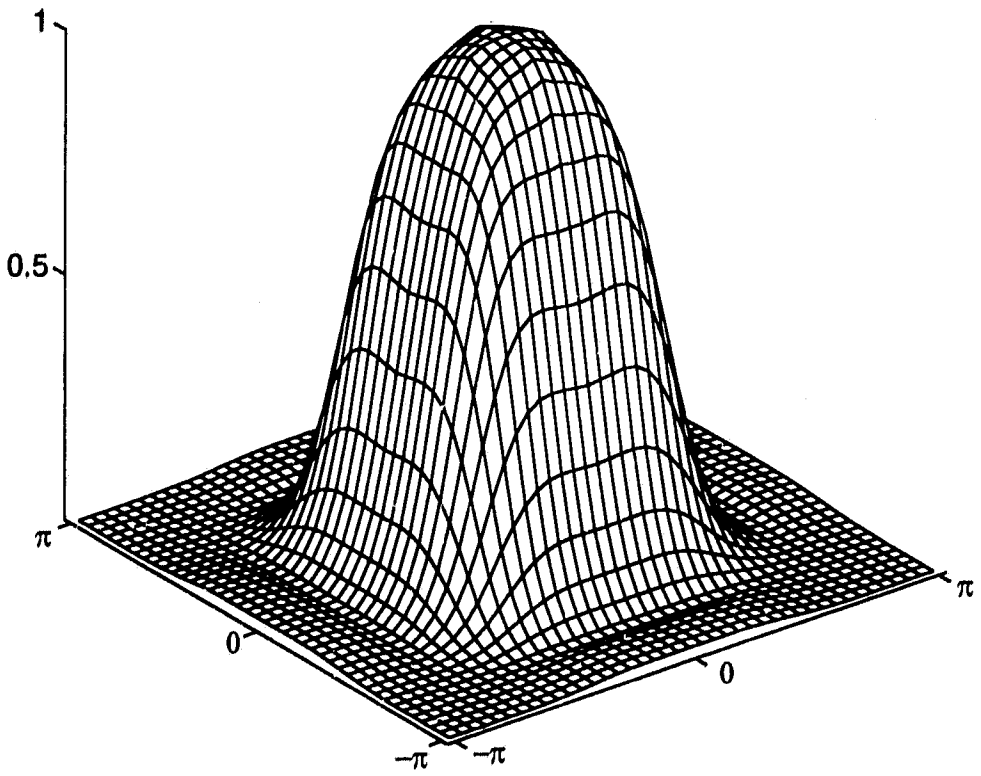

Figure 3.3: Amplitude response of the input weight.

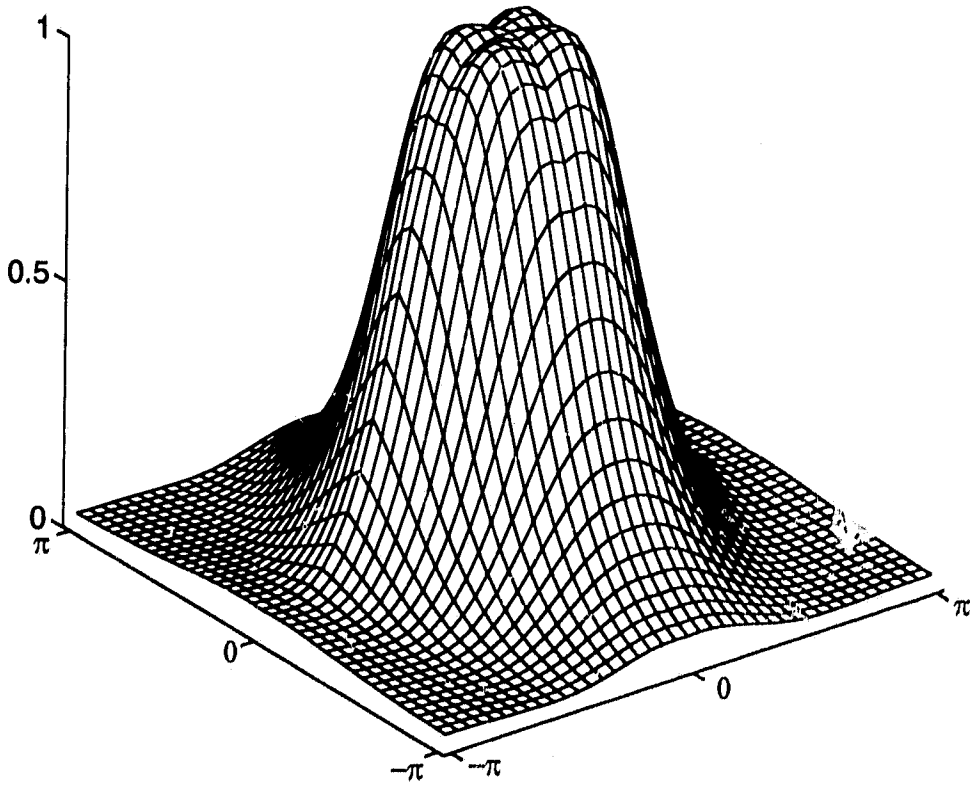

Figure 3.4: Amplitude response of the filter of order (4, 4) from WSBMR.

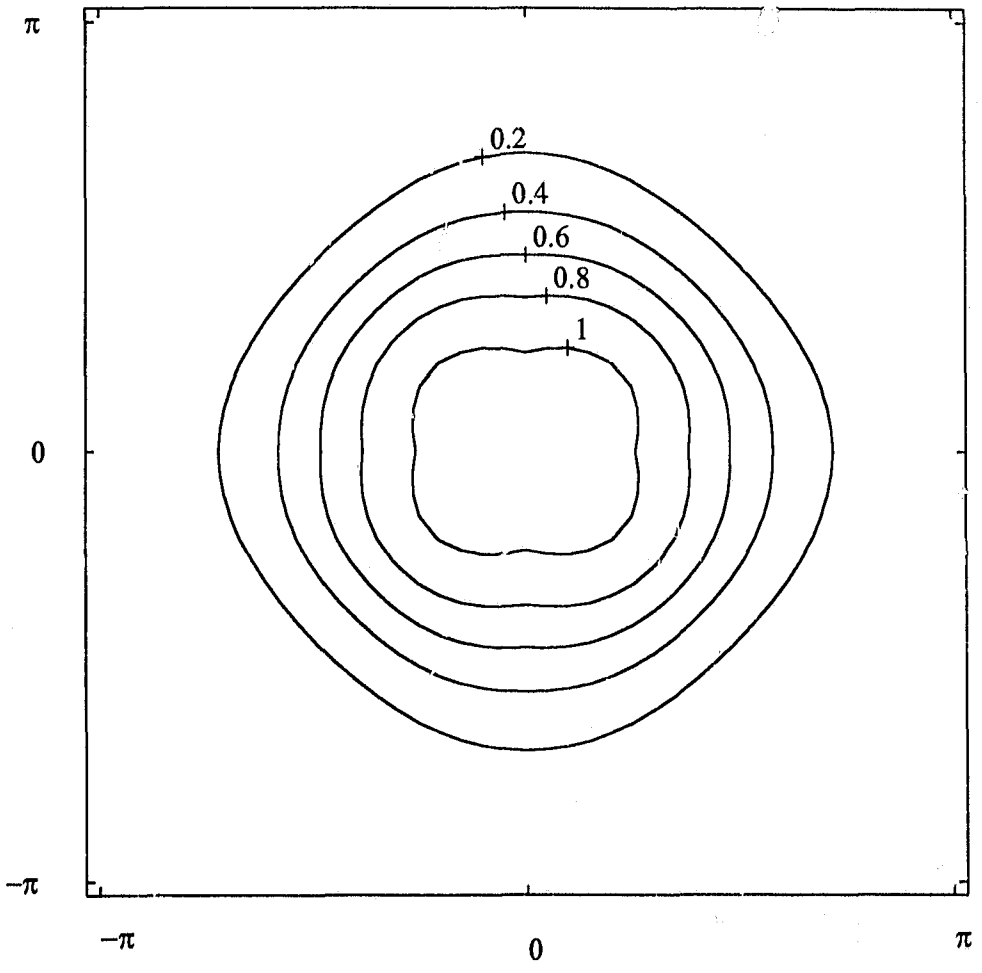

Figure 3.5: Contour of the original filter of order (4, 8).

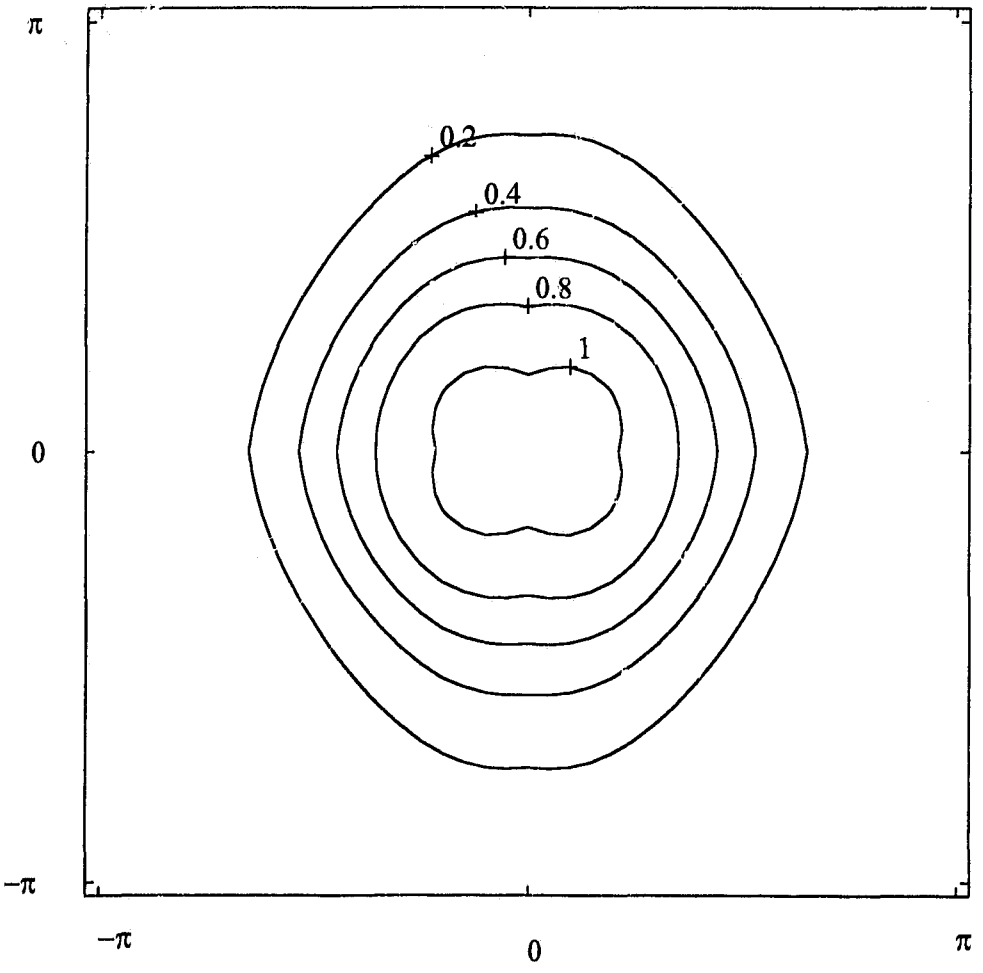

Figure 3.6: Contour of the filter of order (4, 4) from ISBMR.

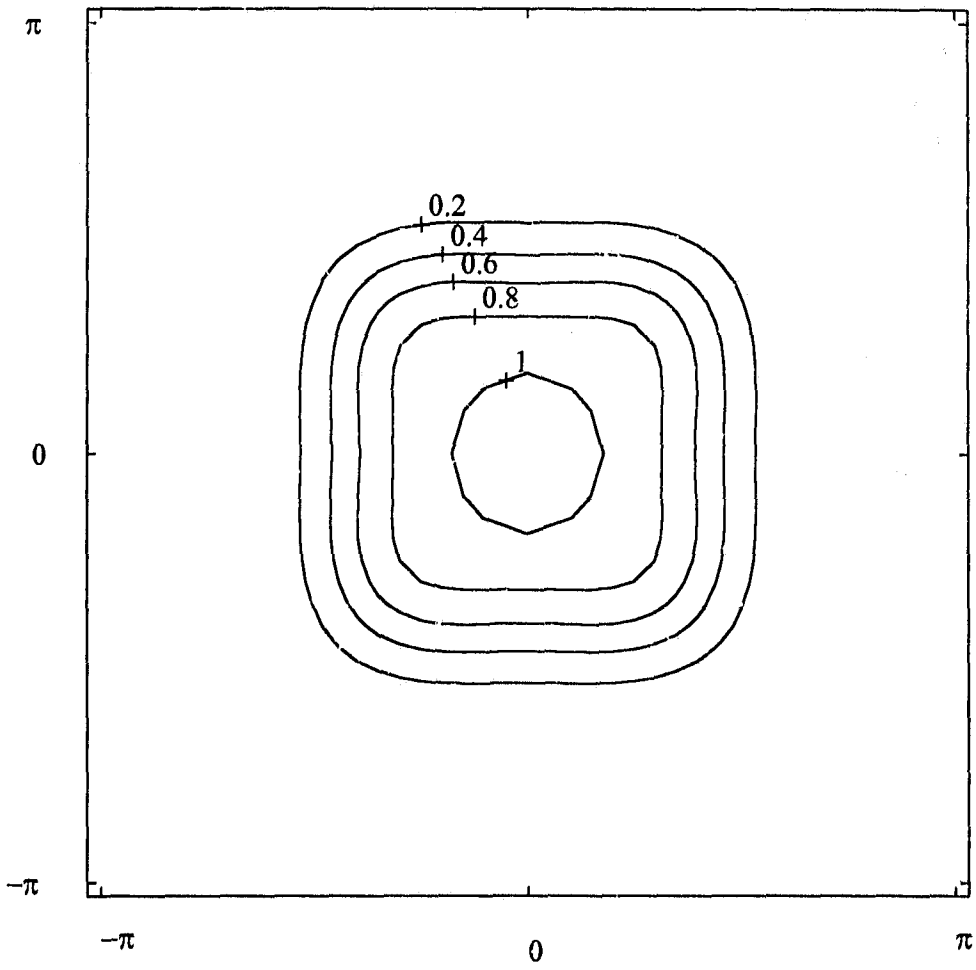

Figure 3.7: Contour of the input weight.

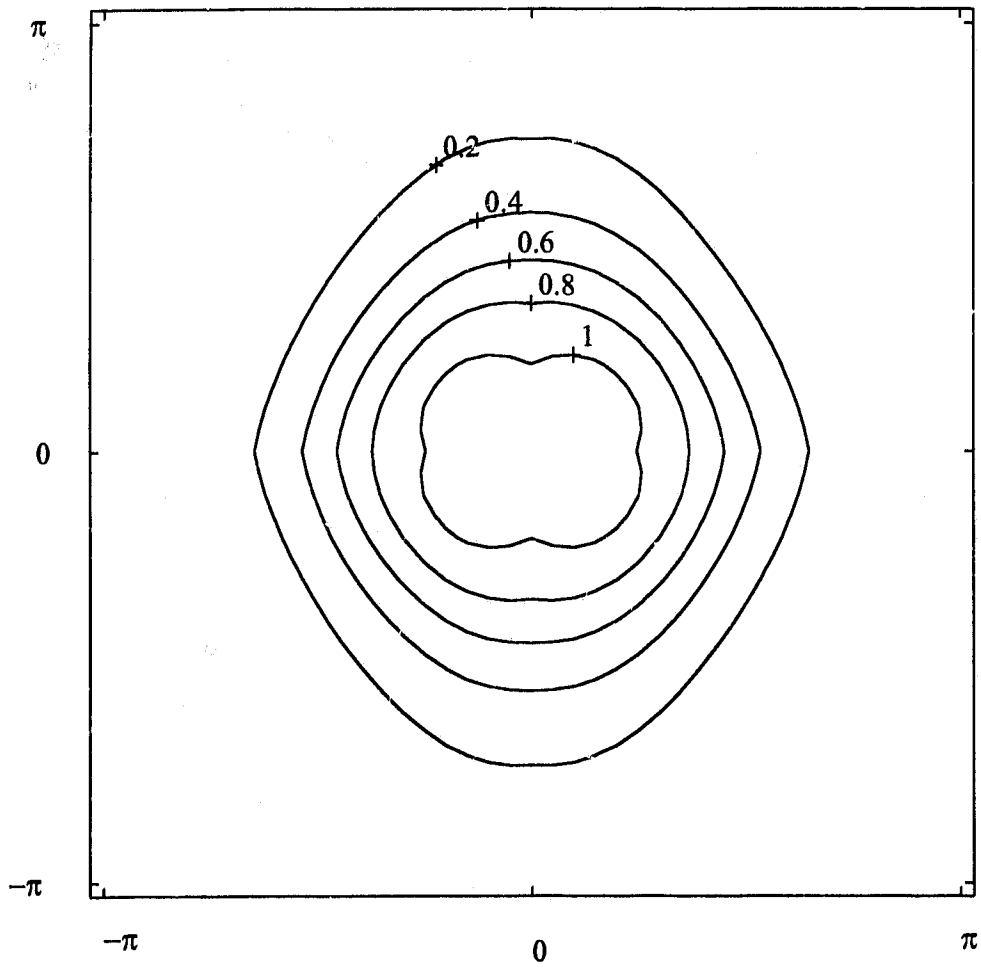

Figure 3.8: Contour of the filter of order (4, 4) from WSBMR.

with $k = \{0, 1, \ldots, K\}$ and $l = \{0, 1, \ldots, L\}$, where K and L are the numbers of frequency-response samples in the normalized frequency regions $\omega_1, \omega_2 \in [0, 1.0]$ ($\omega_1 = k/K$ and $\omega_2 = l/L$). The Frobenius and l_∞ norms of $\mathbf{E}^u$ and $\mathbf{E}^w$ are given by

$$e_F = \| \mathbf{E}^u \|_{\mathrm{F}} = \left[\sum_{k=0}^{K} \sum_{l=0}^{L} (E_{kl}^u)^2 \right]^{1/2}$$

$$e_F^w = \| \mathbf{E}^w \|_{\mathrm{F}} = \left[\sum_{k=0}^{K} \sum_{l=0}^{L} (E_{kl}^w)^2 \right]^{1/2}$$

$$e_\infty = \| \mathbf{E}^u \|_\infty = \max_{\substack{0 \le k \le K \\ 0 \le l \le L}} E_{kl}^u$$

$$e_\infty^w = \| \mathbf{E}^w \|_\infty = \max_{\substack{0 \le k \le K \\ 0 \le l \le L}} E_{kl}^w$$

The approximation errors of the ISBMR and WSBMR are computed in four normalized frequency regions and summarized in Table 3.1. As is anticipated, the approximation error for the WSBMR, measured in e_F^w, is smaller than the approximation error for the ISBMR, measured in e_F, in the low and intermediate frequency regions, but, it is larger than the error for the ISBMR in the high frequency region due to the use of the lowpass input weight.

If the weighted approximation-error matrices of the ISBMR and WSBMR are defined by

$$\hat{\mathbf{E}}^u = (\hat{E}_{kl}^u) \quad \text{and} \quad \hat{\mathbf{E}}^w = (\hat{E}_{kl}^w)$$

where

Table 3.1: Approximation errors for the ISBMR and WSBMR

Error	Normalized Frequency, $\omega = \omega_1 = \omega_2$			
$r_1 = r_2 = 4$	$\omega \in [0, 0.30]$	$\omega \in [0.30, 0.55]$	$\omega \in [0.55, 1.0]$	$\omega \in [0, 1.0]$
e_F	0.4214	1.2775	3.0288	3.3141
e_F^w	0.2684	1.2470	3.6109	3.8295
e_∞	0.0562	0.1440	0.1595	0.1595
e_∞^w	0.0551	0.1662	0.1891	0.1891

$$\hat{E}^u_{kl} = \left| W^o(e^{j2\pi k/K}, e^{j2\pi l/L}) \left[H(e^{j2\pi k/K}, e^{j2\pi l/L}) - H^u(e^{j2\pi k/K}, e^{j2\pi l/L}) \right] W^i(e^{j2\pi k/K}, e^{j2\pi l/L}) \right| \tag{3.40a}$$

$$\hat{E}^w_{kl} = \left| W^o(e^{j2\pi k/K}, e^{j2\pi l/L}) \left[H(e^{j2\pi k/K}, e^{j2\pi l/L}) - H^w(e^{j2\pi k/K}, e^{j2\pi l/L}) \right] W^i(e^{j2\pi k/K}, e^{j2\pi l/L}) \right| \tag{3.40b}$$

then, the Frobenius and l_∞ norms of $\hat{\mathbf{E}}^u$ and $\hat{\mathbf{E}}^w$ are given by

$$\epsilon_F = \left\| \hat{\mathbf{E}}^u \right\|_{\mathrm{F}} = \left[\sum_{k=0}^{K} \sum_{l=0}^{L} (\hat{E}^u_{kl})^2 \right]^{1/2}$$

$$\epsilon_F^w = \left\| \hat{\mathbf{E}}^w \right\|_{\mathrm{F}} = \left[\sum_{k=0}^{K} \sum_{l=0}^{L} (\hat{E}^w_{kl})^2 \right]^{1/2}$$

$$\epsilon_\infty = \left\| \hat{\mathbf{E}}^u \right\|_{\infty} = \max_{\substack{0 \le k \le K \\ 0 \le l \le L}} \hat{E}^u_{kl}$$

Table 3.2: Weighted approximation errors for the ISBMR and WSBMR

Error	Normalized Frequency, $\omega = \omega_1 = \omega_2$			
$r_1 = r_2 = 4$	$\omega \in [0, 0.30]$	$\omega \in [0.30, 0.55]$	$\omega \in [0.55, 1.0]$	$\omega \in [0, 1.0]$
ϵ_F	0.4248	0.8793	0.5751	1.1333
ϵ_F^w	0.2744	0.8221	0.6713	1.0963
ϵ_∞	0.0582	0.0911	0.0839	0.0911
ϵ_∞^w	0.0572	0.1057	0.0990	0.1057

$$\epsilon_\infty^w = \left\| \widehat{\mathbf{E}}^w \right\|_\infty = \max_{\substack{0 \le k \le K \\ 0 \le l \le L}} \widehat{E}_{kl}^w$$

The weighted approximation errors for the ISBMR and WSBMR in four normalized frequency regions in this case are given in Table 3.2.

From Table 3.2, it is noted that the weighted approximation error for the WSBMR, ϵ_F^w, is smaller than the weighted approximation errors for the ISBMR, ϵ_F, in the low and intermediate frequency regions, but it is larger than the weighted error for the ISBMR in the high frequency region.

3.5 Conclusions

In Chapters 2 and 3, the balanced model-reduction methods for both unweighted and weighted 2-D discrete systems have been studied. The ISBMR and WSBMR for 2-D unweighted and weighted discrete systems, respectively, lead to 2-D unweighted and weighted reduced-order systems, respectively.

The resulting reduced-order systems satisfy both stability and approximation error criteria. A lowpass filter of order (4, 8) has been used to demonstrate that the approximation error of the weighted reduced-order filter can be reduced relative to the corresponding unweighted reduced-order filter in desirable frequency ranges.

In Chapters 2 and 3, the 2-D discrete systems are represented by the Roesser state-space model. However, the direct forms of the systems (that is, the coefficient matrices of the numerator and denominator polynomials of the transfer-function matrices that represent the systems) are often required. Efficient algorithms are developed in Chapter 4 for the determination of the coefficient matrices of the numerator and denominator polynomials of the transfer-function matrices that represent a 2-D discrete system from its state-space representation.

Chapter 4

Determination of 2-D Transfer-Function Matrices

4.1 Introduction

The representation of the transfer-function matrix of a 2-D discrete system by a state-space model and vice versa are two basic problems of importance in system analysis and design [7]. Two commonly used state-space models for 2-D discrete systems are the Roesser [54] and Fornasini-Marchesini (F-M) [15] models. To date, no efficient algorithm for the determination of 2-D transfer-function matrices from the F-M state-space representation of the system has been reported.

In this chapter, new efficient algorithms for the determination of the 2-D transfer-function matrices from the Roesser and F-M state-space models are presented in Sections 4.2 and 4.3, respectively. The development of the algorithms involves three major steps. First, the determination of the transfer-function matrix of a MIMO 2-D discrete system is reformulated as

the evaluation of the transfer functions of a series of SISO systems. Second, the 2-D transfer-function matrices are reformulated in terms of the characteristic polynomials of several matrices involved that depend on one complex variable (see Section 4.2.1). Third, algorithms that determine the coefficients of a 1-D polynomial of order n when its values at $(n+1)$ points on the unit circle are known are proposed. The algorithms entail solving a system of linear equations whose coefficient matrix is a unitary Vandermonde matrix (see Section 4.2.2). In Section 4.4, examples are given to illustrate the computational efficiency of the algorithms proposed to compare them with the existing algorithms.

4.2 Determination of the Transfer-Function Matrices from the Roesser State-Space Model

In this section, two new algorithms for the determination of the transfer-function matrix of a linear, shift-invariant, MIMO 2-D discrete system depicted in Figure 2.1 from the Roesser state-space model are developed [41].

4.2.1 Determination of the Transfer Function of SISO Systems

Consider the Roesser state-space model of a SISO 2-D discrete system [54] given by

$$\begin{bmatrix} \mathbf{x}^h(k+1,\, l) \\ \mathbf{x}^v(k,\, l+1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \begin{bmatrix} \mathbf{x}^h(k,\, l) \\ \mathbf{x}^v(k,\, l) \end{bmatrix} + \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} u(k,\, l)$$

$$= \mathbf{A}\,\mathbf{x} + \mathbf{b}\,u \tag{4.1a}$$

$$y(k,\, l) = \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix} \begin{bmatrix} \mathbf{x}^h(k,\, l) \\ \mathbf{x}^v(k,\, l) \end{bmatrix} + d\,u(k,\, l)$$

$$= \mathbf{c}\,\mathbf{x} + d\,u \tag{4.1b}$$

where $\mathbf{x}^h \in \Re^m$ and $\mathbf{x}^v \in \Re^n$ are the horizontal and vertical state-space vectors, respectively. The quantities u and y are the input and output, respectively. The 2-D discrete system can be represented by the set $\{\mathbf{A},\, \mathbf{b},\, \mathbf{c},\, d\}$ where

$$\mathbf{A} \in \Re^{(m+n)\times(m+n)}, \qquad \mathbf{b} \in \Re^{(m+n)\times 1}, \qquad \mathbf{c} \in \Re^{1\times(m+n)}$$

and d is scalar. Taking the 2-D z transform of the system given by (4.1a, b), the transfer function of the SISO system is obtained as

$$H(z_1,\, z_2) = \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix} \begin{bmatrix} z_1\mathbf{I} - \mathbf{A}_1 & -\mathbf{A}_2 \\ -\mathbf{A}_3 & z_2\mathbf{I} - \mathbf{A}_4 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} + d$$

$$= \mathbf{c}\,[\,\mathbf{I}(z_1,\, z_2) - \mathbf{A}\,]^{-1}\,\mathbf{b} + d \tag{4.2}$$

Alternatively, the transfer function can be rewritten as

$$H(z_1,\, z_2) = \frac{\sum_{k=0}^{m} q_k(z_2)\, z_1^k}{\sum_{k=0}^{m} p_k(z_2)\, z_1^k} \tag{4.3}$$

where $p_k(z_2)$ and $q_k(z_2)$ are polynomials (of order not greater than n) in z_2, defined by

$$\sum_{k=0}^{m} p_k(z_2)\, z_1^k \;=\; \det\Big[\mathbf{I}(z_1,\, z_2) - \mathbf{A}\Big] \tag{4.4a}$$

$$\sum_{k=0}^{m} q_k(z_2)\, z_1^k \;=\; \mathbf{c}\,\mathrm{adj}\Big[\mathbf{I}(z_1,\, z_2) - \mathbf{A}\Big]\,\mathbf{b} + d\,\det\Big[\mathbf{I}(z_1,\, z_2) - \mathbf{A}\Big] \tag{4.4b}$$

In these formulas, adj denotes the adjoint and det the determinant of a matrix. From (4.4a) it follows that

$$p_m(z_2) \;=\; P(z_2,\, \mathbf{A}_4) \;=\; \det\big[\, z_2\,\mathbf{I} - \mathbf{A}_4\,\big] \tag{4.5}$$

where $P(z_2,\, \mathbf{A}_4)$ denotes the characteristic polynomial of $\mathbf{A}_4$ in variable z_2. Further, from (4.2) and the formula of matrix inversion [20], $H(z_1,\, z_2)$ can be expressed as

$$H(z_1,\, z_2) \;=\; h(z_2) + \mathbf{g}(z_2)\,[\, z_1\,\mathbf{I} - \mathbf{E}(z_2)\,]^{-1}\,\mathbf{f}(z_2) \tag{4.6}$$

where

$$\mathbf{E}(z_2) \;=\; \mathbf{A}_1 + \mathbf{A}_2\,[\, z_2\,\mathbf{I} - \mathbf{A}_4\,]^{-1}\,\mathbf{A}_3 \tag{4.7a}$$

$$\mathbf{f}(z_2) \;=\; \mathbf{b}_1 + \mathbf{A}_2\,[\, z_2\,\mathbf{I} - \mathbf{A}_4\,]^{-1}\,\mathbf{b}_2 \tag{4.7b}$$

$$\mathbf{g}(z_2) \;=\; \mathbf{c}_1 + \mathbf{c}_2\,[\, z_2\,\mathbf{I} - \mathbf{A}_4\,]^{-1}\,\mathbf{A}_3 \tag{4.7c}$$

$$h(z_2) \;=\; d + \mathbf{c}_2\,[\, z_2\,\mathbf{I} - \mathbf{A}_4\,]^{-1}\,\mathbf{b}_2 \tag{4.7d}$$

For efficient computation of the matrices $\mathbf{E}(z_2)$, $\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $h(z_2)$, (4.7a–d) can be expressed in compact matrix form as

$$\begin{bmatrix} \mathbf{E}(z_2) & \mathbf{f}(z_2) \\ \mathbf{g}(z_2) & h(z_2) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{b}_1 \\ \mathbf{c}_1 & d \end{bmatrix} + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{c}_2 \end{bmatrix} \left[z_2 \mathbf{I} - \mathbf{A}_4 \right]^{-1} \begin{bmatrix} \mathbf{A}_3 & \mathbf{b}_2 \end{bmatrix} \quad (4.8)$$

By using a well-known formula for the transfer function of a 1-D SISO state-space model (see Appendix A.13 of [20]), (4.6) can be rewritten as

$$\begin{aligned} H(z_1, z_2) &= \frac{\det \left[z_1 \mathbf{I} - \mathbf{E}(z_2) + \mathbf{f}(z_2)\, \mathbf{g}(z_2) \right] - \det \left[z_1 \mathbf{I} - \mathbf{E}(z_2) \right]}{\det \left[z_1 \mathbf{I} - \mathbf{E}(z_2) \right]} + h(z_2) \\ &= \frac{\det \left[z_1 \mathbf{I} - \mathbf{E}(z_2) + \mathbf{f}(z_2)\, \mathbf{g}(z_2) \right]}{\det \left[z_1 \mathbf{I} - \mathbf{E}(z_2) \right]} + h(z_2) - 1 \\ &= \frac{P\left[z_1, \mathbf{E}(z_2) - \mathbf{f}(z_2)\, \mathbf{g}(z_2) \right]}{P\left[z_1, \mathbf{E}(z_2) \right]} + h(z_2) - 1 \end{aligned} \quad (4.9)$$

where $P[\, z_1, \mathbf{E}(z_2)\,]$ and $P[\, z_1, \mathbf{E}(z_2) - \mathbf{f}(z_2)\, \mathbf{g}(z_2)\,]$ are the characteristic polynomials of $\mathbf{E}(z_2)$ and $\mathbf{E}(z_2) - \mathbf{f}(z_2)\, \mathbf{g}(z_2)$, respectively. Also, one may note that the denominator in (4.9) is a monic polynomial in z_1; the denominator in (4.3) is a polynomial in z_1 with $p_m(z_2)$ as the coefficient of z_1^m. This observation in conjunction with (4.5) leads to

$$\sum_{k=0}^{m} p_k(z_2)\, z_1^k = P(z_2, \mathbf{A}_4)\, P[\, z_1, \mathbf{E}(z_2)\,] \quad (4.10a)$$

$$\begin{aligned} \sum_{k=0}^{m} q_k(z_2)\, z_1^k = P(z_2, \mathbf{A}_4) \Big\{ & P[\, z_1, \mathbf{E}(z_2) - \mathbf{f}(z_2)\, \mathbf{g}(z_2)\,] \\ & + [\, h(z_2) - 1\,]\, P[\, z_1, \mathbf{E}(z_2)\,] \Big\} \end{aligned} \quad (4.10b)$$

4.2.2 Algorithm for the SISO Case

A new algorithm for the determination of the transfer function of SISO 2-D discrete systems is now presented based on (4.10a, b). The algorithm utilizes an effective method for the determination of a 1-D polynomial and hence it is both computationally efficient and reliable.

A. Determination of the Coefficients of a 1-D Polynomial

Let

$$p(z_2) = \alpha_n z_2^n + \cdots + \alpha_1 z_2 + \alpha_0$$

be a polynomial of order n with coefficients $\{\alpha_n, \ldots, \alpha_1, \alpha_0\}$. Also, let $\{z_2(l),\ 0 \le l \le n\}$ denote $(n+1)$ points that are uniformly distributed on the unit circle of the complex z_2 plane, such that

$$z_2(l) = e^{j2\pi l/(n+1)} \qquad \text{for} \quad l = \{0, 1, \ldots, n\} \tag{4.11}$$

If the values of $\{p_l = p[\, z_2(l)\,],\ 0 \le l \le n\}$ are known, then the corresponding coefficients, denoted by $\{\alpha_l,\ 0 \le l \le n\}$, can be determined as

$$\boldsymbol{\alpha} = \mathbf{V}^{-1}(\mathbf{z}_2)\, \mathbf{q} \tag{4.12}$$

where

$$\boldsymbol{\alpha} = [\alpha_n \ \cdots \ \alpha_1 \ \alpha_0]^T, \qquad \mathbf{q} = [p_0 \ \ p_1 \ \cdots \ p_n]^T$$

and $\mathbf{V}(\mathbf{z}_2)$ is the $(n+1) \times (n+1)$ Vandermonde matrix whose second to last column is

$$\mathbf{z}_2 = \left[\, z_2(0) \ \ z_2(1) \ \cdots \ z_2(n) \,\right]^T$$

that is,

$$\mathbf{V}(\mathbf{z}_2) = \begin{bmatrix} z_2(0)^n & \cdots & z_2(0) & 1 \\ z_2(1)^n & \cdots & z_2(1) & 1 \\ \vdots & & \vdots & \vdots \\ z_2(n)^n & \cdots & z_2(n) & 1 \end{bmatrix} \tag{4.13}$$

Since the points on the unit circle given by $\{z_2(l),\ 0 \leq l \leq n\}$ are distinct, $\mathbf{V}(\mathbf{z}_2)$ is always nonsingular. More importantly, it follows from (4.11) that

$$\mathbf{V}^H(\mathbf{z}_2)\,\mathbf{V}(\mathbf{z}_2) = (n+1)\,\mathbf{I} \tag{4.14}$$

where $\mathbf{V}^H(\mathbf{z}_2)$ is the complex-conjugate transpose of $\mathbf{V}(\mathbf{z}_2)$. Therefore,

$$\mathbf{V}(\mathbf{z}_2)/\sqrt{n+1}$$

is an unitary matrix and (4.12) can be rewritten as

$$\boldsymbol{\alpha} = \frac{1}{n+1}\,\mathbf{V}^H(\mathbf{z}_2)\,\mathbf{q} \tag{4.15}$$

Equation (4.15) provides an effective formula for the determination of 1-D polynomial $p(z_2)$.

B. Determination of the Coefficients of $p_k(z_2)$ and $q_k(z_2)$

For convenience, assume that matrix $\mathbf{A}_4$ has no eigenvalues on the unit circle, that is, the SISO 2-D system represented by $\{\mathbf{A}, \mathbf{b}, \mathbf{c}, d\}$ is stable [29]. If $\mathbf{A}_4$ has eigenvalues on the unit circle, the 2-D system is unstable. Since unstable systems are not useful, this case will not be considered here.

Given a point z_2 on the unit circle, it follows from (4.8) that terms $\mathbf{E}(z_2)$, $\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $h(z_2)$ can be easily evaluated. Hence, (4.10a, b) can be used to obtain the values of $p_k(z_2)$ and $q_k(z_2)$ for $0 \leq k \leq m$ at the given point z_2. If this procedure is repeatedly applied to each point $z_2(l)$ defined by (4.11), then the values of every $p_k(z_2)$ and $q_k(z_2)$ of the set $\{z_2(l),\ 0 \leq l \leq n\}$ can be obtained. Therefore, an efficient method exists to determine all the coefficients of the polynomials $p_k(z_2)$ and $q_k(z_2)$.

The transfer function of an SISO 2-D system can be obtained by using the following algorithm:

Algorithm 4.1

Step 1: Use (4.8) to evaluate $\mathbf{E}(z_2)$, $\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $h(z_2)$ over the set of points defined by (4.11).

Step 2: Compute the determinant of $[\ z_2\,\mathbf{I} - \mathbf{A}_4\ ]$ and form the characteristic equations of $\mathbf{E}(z_2)$ and $\mathbf{E}(z_2) - \mathbf{f}(z_2)\,\mathbf{g}(z_2)$, where $z_2 = z_2(l)$ for $0 \leq l \leq n$.

Step 3: Use (4.10a, b) to obtain the values of $p_k[\,z_2(l)\,]$ and $q_k[\,z_2(l)\,]$ for $0 \leq k \leq m, \quad 0 \leq l \leq n$.

Step 4: For each k, $0 \leq k \leq m$, form vector $\mathbf{q} = [\,p_0 \quad p_1 \quad \cdots \quad p_n\,]^T$ and determine the coefficients of polynomial $p_k(z_2)$ using (4.15). Also, replace $\boldsymbol{\alpha}$ with

$$\mathbf{D}_k = [\ D_{nk} \quad \cdots \quad D_{1k} \quad D_{0k}\ ]^T$$

Similarly, form vector $\mathbf{q} = [\,q_0 \quad q_1 \quad \cdots \quad q_n\,]^T$, determine the coefficients of

polynomial $q_k(z_2)$ using (4.15), and replace $\boldsymbol{\alpha}$ with

$$\mathbf{N}_k = [\, N_{nk} \;\; \cdots \;\; N_{1k} \;\; N_{0k} \,]^T$$

Step 5: Construct the transfer function of the SISO 2-D system as

$$H(z_1, z_2) = \frac{\begin{bmatrix} 1 & z_2^{-1} & \cdots & z_2^{-n} \end{bmatrix} \begin{bmatrix} N_{nm} & \cdots & N_{n1} & N_{n0} \\ \vdots & & \vdots & \vdots \\ N_{1m} & \cdots & N_{11} & N_{10} \\ N_{0m} & \cdots & N_{01} & N_{00} \end{bmatrix} \begin{bmatrix} 1 \\ z_1^{-1} \\ \vdots \\ z_1^{-m} \end{bmatrix}}{\begin{bmatrix} 1 & z_2^{-1} & \cdots & z_2^{-n} \end{bmatrix} \begin{bmatrix} D_{nm} & \cdots & D_{n1} & D_{n0} \\ \vdots & & \vdots & \vdots \\ D_{1m} & \cdots & D_{11} & D_{10} \\ D_{0m} & \cdots & D_{01} & D_{00} \end{bmatrix} \begin{bmatrix} 1 \\ z_1^{-1} \\ \vdots \\ z_1^{-m} \end{bmatrix}}$$

$$= \frac{\hat{\mathbf{z}}_2^T \, \mathbf{N}_R \, \hat{\mathbf{z}}_1}{\hat{\mathbf{z}}_2^T \, \mathbf{D}_R \, \hat{\mathbf{z}}_1} \tag{4.16}$$

where

$$\hat{\mathbf{z}}_1 = \begin{bmatrix} 1 & z_1^{-1} & \cdots & z_1^{-m} \end{bmatrix}^T \quad \text{and} \quad \hat{\mathbf{z}}_2 = \begin{bmatrix} 1 & z_2^{-1} & \cdots & z_2^{-n} \end{bmatrix}^T$$

4.2.3 Dual Algorithm

A dual algorithm to Algorithm 4.1 can be obtained if the roles of variables z_1 and z_2 are interchanged. It is important to note that $\mathbf{A}$, $\mathbf{b}$, and $\mathbf{c}$ are not symmetric respect to the variables z_1 and z_2. Therefore, it is not trivial to develop a dual algorithm by interchanging the roles of variables z_1 and z_2.

By representing $H(z_1, z_2)$ in (4.3) as

$$H(z_1, z_2) = \frac{\sum_{l=0}^{n} \hat{q}_l(z_1)\, z_2^l}{\sum_{l=0}^{n} \hat{p}_l(z_1)\, z_2^l} \tag{4.17}$$

where $\hat{p}_l(z_1)$ and $\hat{q}_l(z_1)$ are polynomials (of order not greater than m) in z_1 given by

$$\sum_{l=0}^{n} \hat{p}_l(z_1)\, z_2^l = P(z_1, \mathbf{A}_1)\, P\left[\, z_2, \hat{\mathbf{E}}(z_1)\,\right] \tag{4.18a}$$

$$\sum_{l=0}^{n} \hat{q}_l(z_1)\, z_2^l = P(z_1, \mathbf{A}_1)\left\{P\left[\, z_2, \hat{\mathbf{E}}(z_1) - \hat{\mathbf{f}}(z_1)\,\hat{\mathbf{g}}(z_1)\,\right] + \left[\,\hat{h}(z_1) - 1\,\right] P\left[\, z_2, \hat{\mathbf{E}}(z_1)\,\right]\right\} \tag{4.18b}$$

where $\hat{\mathbf{E}}(z_1)$, $\hat{\mathbf{f}}(z_1)$, $\hat{\mathbf{g}}(z_1)$, and $\hat{h}(z_1)$ can be obtained from the following relation

$$\begin{bmatrix} \hat{\mathbf{E}}(z_1) & \hat{\mathbf{f}}(z_1) \\ \hat{\mathbf{g}}(z_1) & \hat{h}(z_1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_4 & \mathbf{b}_2 \\ \mathbf{c}_2 & d \end{bmatrix} + \begin{bmatrix} \mathbf{A}_3 \\ \mathbf{c}_1 \end{bmatrix} \left[z_1 \mathbf{I} - \mathbf{A}_1\right]^{-1} \begin{bmatrix} \mathbf{A}_2 & \mathbf{b}_1 \end{bmatrix} \tag{4.19}$$

Further, (4.15) needs to be modified as

$$\boldsymbol{\alpha} = \frac{1}{m+1}\, \mathbf{V}^H(\mathbf{z}_1)\, \mathbf{q} \tag{4.20}$$

where

$$\mathbf{z}_1 = \begin{bmatrix} z_1(0) & z_1(1) & \cdots & z_1(m) \end{bmatrix}^T$$

with

$$z_1(k) = e^{j2\pi k/(m+1)} \qquad \text{for} \quad k = \{0, 1, \ldots, m\} \tag{4.21}$$

The above analysis leads to the following dual algorithm for the determination of the transfer function of an SISO 2-D system.

Algorithm 4.2

Step 1: Use (4.19) to evaluate $\hat{\mathbf{E}}(z_1)$, $\hat{\mathbf{f}}(z_1)$, $\hat{\mathbf{g}}(z_1)$, and $\hat{h}(z_1)$ over the set of points defined by (4.21).

Step 2: Compute the characteristic equations of $\mathbf{A}_1$, $\hat{\mathbf{E}}(z_1)$, and $\hat{\mathbf{E}}(z_1) - \hat{\mathbf{f}}(z_1)\,\hat{\mathbf{g}}(z_1)$ where $z_1 = z_1(k)$ for $0 \leq k \leq m$.

Step 3: Use (4.18a, b) to obtain the values of $\hat{p}_l\,[\,z_1(k)\,]$ and $\hat{q}_l\,[\,z_1(k)\,]$ for $0 \leq k \leq m$, $0 \leq l \leq n$.

Step 4: For each l, $0 \leq l \leq n$, form vector $\mathbf{q} = [\ \hat{p}_0\ \ \hat{p}_1\ \cdots\ \hat{p}_m\ ]^T$ and determine the coefficients of polynomial $\hat{p}_l(z_1)$ using (4.20). Also, replace $\boldsymbol{\alpha}$ with

$$\mathbf{D}_l = [\ D_{ml}\ \cdots\ D_{1l}\ \ D_{0l}\ ]^T$$

Similarly, form vector $\mathbf{q} = [\ \hat{q}_0\ \ \hat{q}_1\ \cdots\ \hat{q}_m\ ]^T$, determine the coefficients of polynomial $\hat{q}_l(z_1)$, and replace $\boldsymbol{\alpha}$ with

$$\mathbf{N}_l = [\ N_{ml}\ \cdots\ N_{1l}\ \ N_{0l}\ ]^T$$

Step 5: Construct the transfer function of the SISO 2-D system as

$$H(z_1, z_2) = \frac{\begin{bmatrix} 1 & z_1^{-1} & \cdots & z_1^{-m} \end{bmatrix} \begin{bmatrix} N_{mn} & \cdots & N_{m1} & N_{m0} \\ \vdots & & \vdots & \vdots \\ N_{1n} & \cdots & N_{11} & N_{10} \\ N_{0n} & \cdots & N_{01} & N_{00} \end{bmatrix} \begin{bmatrix} 1 \\ z_2^{-1} \\ \vdots \\ z_2^{-n} \end{bmatrix}}{\begin{bmatrix} 1 & z_1^{-1} & \cdots & z_1^{-m} \end{bmatrix} \begin{bmatrix} D_{mn} & \cdots & D_{m1} & D_{m0} \\ \vdots & & \vdots & \vdots \\ D_{1n} & \cdots & D_{11} & D_{10} \\ D_{0n} & \cdots & D_{01} & D_{00} \end{bmatrix} \begin{bmatrix} 1 \\ z_2^{-1} \\ \vdots \\ z_2^{-n} \end{bmatrix}}$$

$$= \frac{\hat{\mathbf{z}}_1^T \, \mathbf{N}_{\mathrm{R}}^T \, \hat{\mathbf{z}}_2}{\hat{\mathbf{z}}_1^T \, \mathbf{D}_{\mathrm{R}}^T \, \hat{\mathbf{z}}_2}$$

As previously stated, Algorithm 4.2 can only be used to evaluate $H(z_1, z_2)$ if matrix $\mathbf{A}_1$ has no eigenvalues on the unit circle. If $\mathbf{A}_1$ has eigenvalues on the unit circle, the 2-D system is unstable.

4.2.4 MIMO Case

Consider the Roesser state-space model described in (2.1a, b) for a MIMO 2-D discrete system represented by Figure 2.1. The transfer-function matrix for the system is given by

$$\mathbf{H}(z_1, z_2) = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \end{bmatrix} \begin{bmatrix} z_1 \mathbf{I} - \mathbf{A}_1 & -\mathbf{A}_2 \\ -\mathbf{A}_3 & z_2 \mathbf{I} - \mathbf{A}_4 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix} + \mathbf{D}$$

$$= \mathbf{C} \, [\, \mathbf{I}(z_1, z_2) - \mathbf{A} \,]^{-1} \, \mathbf{B} + \mathbf{D} \tag{4.22}$$

Alternatively, (4.22) can be written in component form as

$$H_{kl}(z_1, z_2) = C_{ki} \, [\, I(z_1, z_2) - A \,]_{ij}^{-1} \, B_{jl} + D_{kl} \tag{4.23}$$

where

$$k \in \{1, 2, \ldots, s\}, \qquad i \in \{1, 2, \ldots, (m+n)\}$$
$$j \in \{1, 2, \ldots, (m+n)\}, \qquad l \in \{1, 2, \ldots, t\}$$

By comparison of (4.23) with (4.2), the MIMO 2-D system can be viewed as a series of SISO 2-D systems by noting the relationships

$$\mathbf{B} = \begin{bmatrix} \mathbf{b}_1 & \mathbf{b}_2 & \cdots & \mathbf{b}_t \end{bmatrix}, \quad \text{and} \quad \mathbf{C} = \begin{bmatrix} \mathbf{c}_1^T & \mathbf{c}_2^T & \cdots & \mathbf{c}_s^T \end{bmatrix}^T$$

where $\mathbf{b}_l \in \Re^{(m+n)\times 1}$ $(l = 1, 2, \ldots, t)$ and $\mathbf{c}_k \in \Re^{1\times(m+n)}$ $(k = 1, 2, \ldots, s)$ correspond to the lth column of $\mathbf{B}$ and kth row of $\mathbf{C}$, respectively. Consequently, transfer-function matrix $\mathbf{H}(z_1, z_2)$ in (4.22) can be evaluated as a series of the transfer functions of SISO systems using Algorithms 4.1 or 4.2. This becomes apparent when $\mathbf{H}(z_1, z_2)$ in (4.22) is rewritten as

$$\mathbf{H}(z_1, z_2) = \mathcal{H}(z_2) + \mathcal{G}(z_2) \left[z_1 \mathbf{I} - \mathcal{E}(z_2) \right]^{-1} \mathcal{F}(z_2) \tag{4.24}$$

where

$$\begin{bmatrix} \mathcal{E}(z_2) & \mathcal{F}(z_2) \\ \mathcal{G}(z_2) & \mathcal{H}(z_2) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1 \\ \mathbf{C}_1 & \mathbf{D} \end{bmatrix} + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{C}_2 \end{bmatrix} \left[z_2 \mathbf{I} - \mathbf{A}_4 \right]^{-1} \begin{bmatrix} \mathbf{A}_3 & \mathbf{B}_2 \end{bmatrix} \tag{4.25}$$

That is, Algorithm 4.1 can deal with a MIMO 2-D system if (4.8) is replaced by (4.25).

As in the dual algorithm for the SISO 2-D systems presented in Section 4.2.3, the role of variables z_1 and z_2 in (4.24) can be interchanged as

$$\mathbf{H}(z_1, z_2) = \hat{\mathcal{H}}(z_1) + \hat{\mathcal{G}}(z_1) \left[z_2 \mathbf{I} - \hat{\mathcal{E}}(z_1) \right]^{-1} \hat{\mathcal{F}}(z_1) \tag{4.26}$$

where

$$\begin{bmatrix} \hat{\mathcal{E}}(z_1) & \hat{\mathcal{F}}(z_1) \\ \hat{\mathcal{G}}(z_1) & \hat{\mathcal{H}}(z_1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_4 & \mathbf{B}_2 \\ \mathbf{C}_2 & \mathbf{D} \end{bmatrix} + \begin{bmatrix} \mathbf{A}_3 \\ \mathbf{C}_1 \end{bmatrix} \left[z_1 \mathbf{I} - \mathbf{A}_1 \right]^{-1} \begin{bmatrix} \mathbf{A}_2 & \mathbf{B}_1 \end{bmatrix} \quad (4.27)$$

Then, Algorithm 4.2 can be applied to a MIMO 2-D system if (4.19) is replaced by (4.27).

4.3 Determination of the Transfer-Function Matrices from the Fornasini-Marchesini State-Space Model

In this section, two new algorithms for the determination of the transfer-function matrix of a linear, shift-invariant, MIMO 2-D discrete system from the F-M state-space description are developed.

4.3.1 Determination of the Transfer Function of SISO Systems

The F-M state-space model of an SISO 2-D discrete system is given by

$$\begin{aligned} \boldsymbol{x}(k+1, l+1) &= \boldsymbol{A}_{\mathrm{I}}\, \boldsymbol{x}(k, l+1) + \boldsymbol{A}_{\mathrm{II}}\, \boldsymbol{x}(k+1, l) \\ &\quad + \boldsymbol{b}_{\mathrm{I}}\, u(k, l+1) + \boldsymbol{b}_{\mathrm{II}}\, u(k+1, l) \qquad (4.28a) \\ y(k, l) &= \boldsymbol{c}\, \boldsymbol{x}(k, l) + d\, u(k, l) \qquad (4.28b) \end{aligned}$$

where $\boldsymbol{x} \in \Re^N$ is the state vector with $N \leq (n+m)$. The terms u, y, and d are as previously defined in (4.1a, b). The transfer function of the system

(4.28a, b) can be expressed in terms of $\{\boldsymbol{A}_{\rm I},\ \boldsymbol{A}_{\rm II},\ \boldsymbol{b}_{\rm I},\ \boldsymbol{b}_{\rm II},\ \boldsymbol{c},\ d\}$ as

$$H(z_1,\ z_2) \ = \ \boldsymbol{c}\,(z_1\,z_2\,\mathbf{I} - z_2\,\boldsymbol{A}_{\rm I} - z_1\,\boldsymbol{A}_{\rm II})^{-1}\,(z_2\,\boldsymbol{b}_{\rm I} + z_1\,\boldsymbol{b}_{\rm II}) + d \qquad (4.29)$$

$$= \frac{\sum\limits_{v=0}^{N} q_v(z_2)\,z_1^v}{\sum\limits_{v=0}^{N} p_v(z_2)\,z_1^v} \qquad (4.30)$$

where $p_v(z_2)$ and $q_v(z_2)$ are polynomials (of order not greater than N) in z_2. As in (4.9), (4.29) can be expressed as

$$H(z_1,\ z_2) \ = \ \frac{\det\left[\,z_1\,z_2\,\mathbf{I} - z_2\,\boldsymbol{A}_{\rm I} - z_1\,\boldsymbol{A}_{\rm II} + z_2\,\boldsymbol{b}_{\rm I}\,\boldsymbol{c} + z_1\,\boldsymbol{b}_{\rm II}\,\boldsymbol{c}\,\right]}{\det\left[\,z_1\,z_2\,\mathbf{I} - z_2\,\boldsymbol{A}_{\rm I} - z_1\,\boldsymbol{A}_{\rm II}\,\right]} + d - 1$$

$$= \frac{\det\left[\,z_2\,\mathbf{I} - \boldsymbol{A}_{\rm II} + \boldsymbol{b}_{\rm II}\,\boldsymbol{c}\,\right]\ \det\left[\,z_1\,\mathbf{I} - \boldsymbol{F}(z_2)\,\right]}{\det\left[\,z_2\,\mathbf{I} - \boldsymbol{A}_{\rm II}\,\right]\ \det\left[\,z_1\,\mathbf{I} - \boldsymbol{E}(z_2)\,\right]} + d - 1$$

$$= \frac{P\,[z_2,\ \boldsymbol{A}_{\rm II} - \boldsymbol{b}_{\rm II}\,\boldsymbol{c}]\,P\,[\,z_1,\ \boldsymbol{F}(z_2)\,]}{P\,[z_2,\ \boldsymbol{A}_{\rm II}]\,P\,[\,z_1,\ \boldsymbol{E}(z_2)\,]} + d - 1 \qquad (4.31)$$

where

$$\boldsymbol{E}(z_2) \ = \ z_2\,\boldsymbol{A}_{\rm I}\,(z_2\,\mathbf{I} - \boldsymbol{A}_{\rm II})^{-1} \qquad (4.32a)$$

$$\boldsymbol{F}(z_2) \ = \ z_2\,(\boldsymbol{A}_{\rm I} - \boldsymbol{b}_{\rm I}\,\boldsymbol{c})\,(z_2\,\mathbf{I} - \boldsymbol{A}_{\rm II} + \boldsymbol{b}_{\rm II}\,\boldsymbol{c})^{-1} \qquad (4.32b)$$

The terms $P\,[z_2,\ \boldsymbol{A}_{\rm II}]$, $P\,[z_2,\ \boldsymbol{A}_{\rm II} - \boldsymbol{b}_{\rm II}\,\boldsymbol{c}]$, $P\,[\,z_1,\ \boldsymbol{E}(z_2)\,]$, and $P\,[\,z_1,\ \boldsymbol{F}(z_2)\,]$ are the characteristic polynomials of $\boldsymbol{A}_{\rm II}$, $\boldsymbol{A}_{\rm II} - \boldsymbol{b}_{\rm II}\,\boldsymbol{c}$, $\boldsymbol{E}(z_2)$, and $\boldsymbol{F}(z_2)$, respectively. From (4.30) and (4.31), it follows that

$$\sum_{v=0}^{N} p_v(z_2)\,z_1^v \ = \ P\,[z_2,\ \boldsymbol{A}_{\rm II}]\,P\,[\,z_1,\ \boldsymbol{E}(z_2)\,] \qquad (4.33a)$$

$$\sum_{v=0}^{N} q_v(z_2)\, z_1^v = P\,[z_2,\, \boldsymbol{A}_{II} - \boldsymbol{b}_{II}\boldsymbol{c}]\, P\,[z_1,\, \boldsymbol{F}(z_2)] + (d-1)\, P\,[z_2,\, \boldsymbol{A}_{II}]\, P\,[z_1,\, \boldsymbol{E}(z_2)] \qquad (4.33b)$$

4.3.2 Algorithm for the SISO Case

A new algorithm for the determination of the transfer function of an SISO 2-D system represented by the F-M model is presented in this subsection. The algorithm is based on (4.33a, b) and assumes that the matrices $\boldsymbol{A}_{II}$ and $\boldsymbol{A}_{II} - \boldsymbol{b}_{II}\,\boldsymbol{c}$ have no eigenvalues on the unit circle. As in Section 4.2.2.A, the method for the determination of a 1-D polynomial can be used with a minor modification; that is, the n and $\mathbf{z}_2$ in (4.15) are replaced by N and

$$\boldsymbol{z}_2 = \begin{bmatrix} z_2(0) & z_2(1) & \cdots & z_2(N) \end{bmatrix}^T$$

where

$$z_2(\mathrm{w}) = e^{j2\pi \mathrm{w}/(N+1)} \qquad \text{for} \quad \mathrm{w} = \{0,\, 1,\, \ldots,\, N\} \qquad (4.34)$$

respectively, yielding

$$\boldsymbol{\alpha} = \frac{1}{N+1}\, \mathbf{V}^H(\boldsymbol{z}_2)\, \mathbf{q} \qquad (4.35)$$

The transfer function of an SISO 2-D discrete system represented by the F-M model can be determined using the following algorithm:

Algorithm 4.3

Step 1: Use (4.32a, b) to evaluate $\boldsymbol{E}(z_2)$ and $\boldsymbol{F}(z_2)$ over the set of points

defined in (4.34).

Step 2: Compute the determinants of $[z_2\,\mathbf{I}-\boldsymbol{A}_{\rm II}]$ and $[z_2\,\mathbf{I}-\boldsymbol{A}_{\rm II}+\boldsymbol{b}_{\rm II}\,\boldsymbol{c}]$. Also, compute the characteristic equations of $\boldsymbol{E}(z_2)$ and $\boldsymbol{F}(z_2)$ where $z_2 = z_2(\mathrm{w})$ for $0 \le \mathrm{w} \le N$.

Step 3: Use (4.33a, b) to obtain the values of $p_v[\,z_2(\mathrm{w})\,]$ and $q_v[\,z_2(\mathrm{w})\,]$ for $0 \le v \le N$, $0 \le \mathrm{w} \le N$.

Step 4: For each v, $0 \le v \le N$, form vector $\mathbf{q} = [\,p_0\ p_1\ \cdots\ p_N\,]^T$, determine the coefficients of polynomial $p_v(z_2)$ using (4.35), and replace $\boldsymbol{\alpha}$ with

$$\mathbf{D}_v \;=\; [\,D_{Nv}\ \cdots\ D_{1v}\ D_{0v}\,]^T$$

Also, form vector $\mathbf{q} = [\,q_0\ q_1\ \cdots\ q_N\,]^T$, determine the coefficients of polynomial $q_v(z_2)$, and replace $\boldsymbol{\alpha}$ with

$$\mathbf{N}_v \;=\; [\,N_{Nv}\ \cdots\ N_{1v}\ N_{0v}\,]^T$$

Step 5: Construct the transfer function for an SISO 2-D discrete system represented by the F-M model as

$$H(z_1,\,z_2) \;=\; \frac{\begin{bmatrix}1 & z_2^{-1} & \cdots & z_2^{-N}\end{bmatrix}\begin{bmatrix}N_{NN} & \cdots & N_{N1} & N_{N0}\\ \vdots & & \vdots & \vdots\\ N_{1N} & \cdots & N_{11} & N_{10}\\ N_{0N} & \cdots & N_{01} & N_{00}\end{bmatrix}\begin{bmatrix}1\\ z_1^{-1}\\ \vdots\\ z_1^{-N}\end{bmatrix}}{\begin{bmatrix}1 & z_2^{-1} & \cdots & z_2^{-N}\end{bmatrix}\begin{bmatrix}D_{NN} & \cdots & D_{N1} & D_{N0}\\ \vdots & & \vdots & \vdots\\ D_{1N} & \cdots & D_{11} & D_{10}\\ D_{0N} & \cdots & D_{01} & D_{00}\end{bmatrix}\begin{bmatrix}1\\ z_1^{-1}\\ \vdots\\ z_1^{-N}\end{bmatrix}}$$

$$= \frac{\tilde{\mathbf{z}}_2^T \, \mathbf{N}_F \, \tilde{\mathbf{z}}_1}{\tilde{\mathbf{z}}_2^T \, \mathbf{D}_F \, \tilde{\mathbf{z}}_1}$$

where

$$\tilde{\mathbf{z}}_1 = \begin{bmatrix} 1 & z_1^{-1} & \cdots & z_1^{-N} \end{bmatrix}^T \quad \text{and} \quad \tilde{\mathbf{z}}_2 = \begin{bmatrix} 1 & z_2^{-1} & \cdots & z_2^{-N} \end{bmatrix}^T$$

If matrix $\boldsymbol{A}_{II}$ has eigenvalues on the unit circle, the system is unstable. For special case where matrix $\boldsymbol{A}_{II} - \boldsymbol{b}_{II}\boldsymbol{c}$ has eigenvalues on the unit circle, the algorithm can be modified as detailed in Section 4.3.4.

4.3.3 Dual Algorithm

A dual algorithm to Algorithm 4.3 can be obtained if the roles of variables z_1 and z_2 are interchanged. By representing $H(z_1, z_2)$ in (4.30) as

$$H(z_1, z_2) = \frac{\sum_{w=0}^{N} \hat{q}_w(z_1) z_2^w}{\sum_{w=0}^{N} \hat{p}_w(z_1) z_2^w} \tag{4.36}$$

where $\hat{p}_w(z_1)$ and $\hat{q}_w(z_1)$ are polynomials (of order not greater than N) in z_1 given by

$$\sum_{w=0}^{N} \hat{p}_w(z_1) z_2^w = P[z_1, \boldsymbol{A}_I] \, P[z_2, \hat{\boldsymbol{E}}(z_1)] \tag{4.37a}$$

$$\sum_{w=0}^{N} \hat{q}_w(z_1) z_2^w = P[z_1, \boldsymbol{A}_I - \boldsymbol{b}_I \boldsymbol{c}] \, P[z_2, \hat{\boldsymbol{F}}(z_1)] + (d - \quad) \, P[z_1, \boldsymbol{A}_I] \, P[z_2, \hat{\boldsymbol{E}}(z_1)] \tag{4.37b}$$

with

$$\hat{\boldsymbol{E}}(z_1) = z_1 \boldsymbol{A}_{II} (z_1 \mathbf{I} - \boldsymbol{A}_I)^{-1} \tag{4.38a}$$

$$\hat{\boldsymbol{F}}(z_1) = z_1 (\boldsymbol{A}_{II} - \boldsymbol{b}_{II}\boldsymbol{c}) (z_1 \mathbf{I} - \boldsymbol{A}_I + \boldsymbol{b}_I \boldsymbol{c})^{-1} \tag{4.38b}$$

The terms $P[z_1, \boldsymbol{A}_I]$, $P[z_1, \boldsymbol{A}_I - \boldsymbol{b}_I \boldsymbol{c}]$, $P[z_2, \hat{\boldsymbol{E}}(z_1)]$, and $P[z_2, \hat{\boldsymbol{F}}(z_1)]$ are the characteristic polynomials of $\boldsymbol{A}_I$, $\boldsymbol{A}_I - \boldsymbol{b}_I \boldsymbol{c}$, $\hat{\boldsymbol{E}}(z_1)$, and $\hat{\boldsymbol{F}}(z_1)$, respectively. Further, (4.20) needs to be modified as

$$\boldsymbol{\alpha} = \frac{1}{N+1} \mathbf{V}^H(\boldsymbol{z}_1)\, \mathbf{q} \tag{4.39}$$

where

$$\boldsymbol{z}_1 = \begin{bmatrix} z_1(0) & z_1(1) & \cdots & z_1(N) \end{bmatrix}^T$$

with

$$z_1(v) = e^{j2\pi v/(N+1)} \qquad \text{for} \quad v = \{0, 1, \ldots, N\} \tag{4.40}$$

The above analysis leads to the dual algorithm of Algorithm 4.3, as follows:

Algorithm 4.4

Step 1: Use (4.38a, b) to evaluate $\hat{\boldsymbol{E}}(z_1)$ and $\hat{\boldsymbol{F}}(z_1)$ over the set of points defined by (4.40).

Step 2: Compute the characteristic equations of $\boldsymbol{A}_I$, $\boldsymbol{A}_I - \boldsymbol{b}_I \boldsymbol{c}$, $\hat{\boldsymbol{E}}(z_1)$, and $\hat{\boldsymbol{F}}(z_1)$ for $z_1 = z_1(v)$, $0 \le v \le N$.

Step 3: Use (4.37a, b) to obtain the values of $\hat{p}_w[z_1(v)]$ and $\hat{q}_w[z_1(v)]$ for $0 \le v \le N$, $0 \le w \le N$.

Step 4: For each w, $0 \le w \le N$, form vector $\mathbf{q} = [\hat{p}_0 \;\; \hat{p}_1 \;\; \cdots \;\; \hat{p}_N]^T$ and determine the coefficients of polynomial $\hat{p}_w(z_1)$ using (4.39). Also, replace $\boldsymbol{\alpha}$

with

$$\mathbf{D}_{\mathrm{w}} = [\, D_{N\mathrm{w}} \;\; \cdots \;\; D_{1\mathrm{w}} \;\; D_{0\mathrm{w}} \,]^T$$

Similarly, form vector $\mathbf{q} = [\, \hat{q}_0 \;\; \hat{q}_1 \;\; \cdots \;\; \hat{q}_N \,]^T$ and determine the coefficients of $\hat{q}_{\mathrm{w}}(z_1)$. Also, replace $\boldsymbol{\alpha}$ with

$$\mathbf{N}_{\mathrm{w}} = [\, N_{N\mathrm{w}} \;\; \cdots \;\; N_{1\mathrm{w}} \;\; N_{0\mathrm{w}} \,]^T$$

Step 5: Construct the transfer function of an SISO 2-D discrete system as

$$H(z_1, z_2) = \frac{\begin{bmatrix} 1 & z_1^{-1} & \cdots & z_1^{-N} \end{bmatrix} \begin{bmatrix} N_{NN} & \cdots & N_{N1} & N_{N0} \\ \vdots & & \vdots & \vdots \\ N_{1N} & \cdots & N_{11} & N_{10} \\ N_{0N} & \cdots & N_{01} & N_{00} \end{bmatrix} \begin{bmatrix} 1 \\ z_2^{-1} \\ \vdots \\ z_2^{-N} \end{bmatrix}}{\begin{bmatrix} 1 & z_1^{-1} & \cdots & z_1^{-N} \end{bmatrix} \begin{bmatrix} D_{NN} & \cdots & D_{N1} & D_{N0} \\ \vdots & & \vdots & \vdots \\ D_{1N} & \cdots & D_{11} & D_{10} \\ D_{0N} & \cdots & D_{01} & D_{00} \end{bmatrix} \begin{bmatrix} 1 \\ z_2^{-1} \\ \vdots \\ z_2^{-N} \end{bmatrix}}$$

$$= \frac{\tilde{\mathbf{z}}_1^T \, \mathbf{N}_{\mathrm{F}}^T \, \tilde{\mathbf{z}}_2}{\tilde{\mathbf{z}}_1^T \, \mathbf{D}_{\mathrm{F}}^T \, \tilde{\mathbf{z}}_2}$$

If matrix $\boldsymbol{A}_{\mathrm{I}}$ has eigenvalues on the unit circle, the system is unstable. For special case where matrix $\boldsymbol{A}_{\mathrm{I}} - \boldsymbol{b}_{\mathrm{I}} \, \boldsymbol{c}$ has eigenvalues on the unit circle, the system can be modified as detailed in Section 4.3.4.

4.3.4 Special Case

For the special case where matrix $\boldsymbol{A}_{\text{II}} - \boldsymbol{b}_{\text{II}}\, \boldsymbol{c}$ (or $\boldsymbol{A}_{\text{I}} - \boldsymbol{b}_{\text{I}}\, \boldsymbol{c}$) has eigenvalues with unity modulus, the $N+1$ points defined by (4.34) can be modified as

$$z_2(\mathrm{w}) \;=\; r\, e^{j2\pi \mathrm{w}/(N+1)} \qquad \text{for} \quad \mathrm{w} = \{0,\, 1,\, \ldots,\, N\}$$

where $r > 0$ is the radius of a circle in which $\boldsymbol{A}_{\text{II}} - \boldsymbol{b}_{\text{II}}\, \boldsymbol{c}$ has no eigenvalues with unity modulus. Form $\mathbf{q} = [\,p_0 \;\; p_1 \;\cdots\; p_N\,]^T$, (4.35) becomes

$$\boldsymbol{\alpha} \;=\; \mathbf{V}_r^{-1}(\boldsymbol{z}_2)\, \mathbf{q}$$

where the Vandermonde matrix of (4.13) is now given by

$$\begin{aligned} \mathbf{V}_r(\boldsymbol{z}_2) \;&=\; \begin{bmatrix} r^N z_2(0)^N & \cdots & r\, z_2(0) & 1 \\ r^N z_2(1)^N & \cdots & r\, z_2(1) & 1 \\ \vdots & & \vdots & \vdots \\ r^N z_2(N)^N & \cdots & r\, z_2(N) & 1 \end{bmatrix} \\ &=\; \mathrm{diag}\,(\,r^N,\, \ldots,\, r,\, 1\,)\, \mathbf{V}(\boldsymbol{z}_2) \end{aligned}$$

and $\mathrm{diag}\,(\,r^N,\, \ldots,\, r,\, 1\,)$ is the diagonal matrix with entries $(r^N,\, \ldots,\, r,\, 1)$ along the main diagonal. Further, (4.14) becomes

$$\mathbf{V}_r(\boldsymbol{z}_2)^H\, \mathbf{V}_r(\boldsymbol{z}_2) \;=\; (N+1)\, \mathrm{diag}\,(\,r^{2N}, \ldots, r^2,\, 1\,)\, \mathbf{I}$$

which implies that

$$\mathbf{V}_r^{-1}(\boldsymbol{z}_2) \;=\; \frac{1}{N+1}\, \mathrm{diag}\,(\,r^{-2N}, \ldots, r^{-2},\, 1\,)\, \mathbf{V}_r^H(\boldsymbol{z}_2)$$

Therefore, (4.35) becomes

$$\boldsymbol{\alpha} = \frac{1}{N+1} \operatorname{diag}(r^{-N}, \ldots, r^{-1}, 1)\, \mathbf{V}^H(\boldsymbol{z}_2)\, \mathbf{q} \tag{4.41}$$

As expected for the case of $r = 1$, (4.41) reduces to (4.35).

4.3.5 MIMO Case

Consider the F-M state-space model of a MIMO 2-D discrete system given by

$$\begin{aligned} \boldsymbol{x}(k+1, l+1) &= \boldsymbol{A}_{\mathrm{I}}\, \boldsymbol{x}(k, l+1) + \boldsymbol{A}_{\mathrm{II}}\, \boldsymbol{x}(k+1, l) \\ &\quad + \boldsymbol{B}_{\mathrm{I}}\, \mathbf{u}(k, l+1) + \boldsymbol{B}_{\mathrm{II}}\, \mathbf{u}(k+1, l) \qquad (4.42\mathrm{a}) \\ \mathbf{y}(k, l) &= \boldsymbol{C}\, \boldsymbol{x}(k, l) + \mathbf{D}\, \mathbf{u}(k, l) \qquad (4.42\mathrm{b}) \end{aligned}$$

where $\mathbf{u}$, $\mathbf{y}$, and $\mathbf{D}$ are as previously defined in (2.1a, b). The transfer-function matrix of system (4.42a, b) can be expressed in terms of $\{\boldsymbol{A}_{\mathrm{I}}, \boldsymbol{A}_{\mathrm{II}}, \boldsymbol{B}_{\mathrm{I}}, \boldsymbol{B}_{\mathrm{II}}, \boldsymbol{C}, \mathbf{D}\}$ as

$$\mathbf{H}(z_1, z_2) = \boldsymbol{C}\,(z_1 z_2 \mathbf{I} - z_2 \boldsymbol{A}_{\mathrm{I}} - z_1 \boldsymbol{A}_{\mathrm{II}})^{-1}\,(z_2 \boldsymbol{B}_{\mathrm{I}} + z_1 \boldsymbol{B}_{\mathrm{II}}) + \mathbf{D} \tag{4.43}$$

Alternatively, the transfer-function matrix given by (4.43) can be written in component form as

$$H_{kl}(z_1, z_2) = C_{ki}\,(z_1 z_2 I - z_2 A_{\mathrm{I}} - z_1 A_{\mathrm{II}})^{-1}_{ij}\,(z_2 B_{\mathrm{I}} + z_1 B_{\mathrm{II}})_{jl} + D_{kl} \tag{4.44}$$

where

$$\begin{aligned} &k \in \{1, 2, \ldots, s\}, && i \in \{1, 2, \ldots, 2N\} \\ &j \in \{1, 2, \ldots, 2N\}, && l \in \{1, 2, \ldots, t\} \end{aligned}$$

By comparison of (4.44) with (4.29), the MIMO 2-D system can be viewed as a series of SISO 2-D systems by noting the relationships

$$\boldsymbol{B}_{\rm I} = \begin{bmatrix} \boldsymbol{b}_{\rm I1} & \boldsymbol{b}_{\rm I2} & \cdots & \boldsymbol{b}_{\rm It} \end{bmatrix}, \qquad \boldsymbol{B}_{\rm II} = \begin{bmatrix} \boldsymbol{b}_{\rm II1} & \boldsymbol{b}_{\rm II2} & \cdots & \boldsymbol{b}_{\rm IIt} \end{bmatrix}$$

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{c}_1^T & \boldsymbol{c}_2^T & \cdots & \boldsymbol{c}_s^T \end{bmatrix}^T$$

where $\boldsymbol{b}_{\rm Il}$ and $\boldsymbol{b}_{\rm IIl} \in \Re^{2N\times 1}$, $(l = 1, 2, \ldots, t)$ and $\boldsymbol{c}_k \in \Re^{1\times 2N}$, $(k = 1, 2, \ldots, s)$ correspond to the lth column of $\boldsymbol{B}_{\rm I}$ and $\boldsymbol{B}_{\rm II}$, and kth row of $\boldsymbol{C}$, respectively. Consequently, the transfer-function matrix, $\mathbf{H}(z_1, z_2)$, given by (4.43) can be evaluated as a series of transfer functions of SISO 2-D systems.

That is, Algorithm 4.3 can deal with a MIMO 2-D system if (4.32b) is replaced by

$$\tilde{\boldsymbol{F}}(z_2) = z_2 \left(\boldsymbol{A}_{\rm I} - \boldsymbol{b}_{\rm Il}\, \boldsymbol{c}_k\right) \left(z_2\, \mathbf{I} - \boldsymbol{A}_{\rm II} + \boldsymbol{b}_{\rm IIl}\, \boldsymbol{c}_k\right)^{-1} \tag{4.45}$$

and (4.33b) is replaced by

$$\begin{aligned} \sum_{v=0}^{N} q_v(z_2)\, z_1^v = {} & P\left[z_2,\, \boldsymbol{A}_{\rm II} - \boldsymbol{b}_{\rm IIl}\, \boldsymbol{c}_k\right] P\left[z_1,\, \tilde{\boldsymbol{F}}(z_2)\right] \\ & + (D_{kl} - 1)\, P\left[z_2,\, \boldsymbol{A}_{\rm II}\right] P\left[z_1,\, \boldsymbol{E}(z_2)\right] \end{aligned} \tag{4.46}$$

for $(k = 1, 2, \ldots, s)$ and $(l = 1, 2, \ldots, t)$, where terms $P\left[z_2,\, \boldsymbol{A}_{\rm II} - \boldsymbol{b}_{\rm IIl}\, \boldsymbol{c}_k\right]$ and $P\left[z_1,\, \tilde{\boldsymbol{F}}(z_2)\right]$ are the characteristic polynomials of $\left(\boldsymbol{A}_{\rm II} - \boldsymbol{b}_{\rm IIl}\, \boldsymbol{c}_k\right)$ and $\tilde{\boldsymbol{F}}(z_2)$, respectively.

By interchanging the role of variables $\boldsymbol{z}_1$ and $\boldsymbol{z}_2$, Algorithm 4.4 of SISO 2-D systems can be extended to deal with MIMO 2-D systems if (4.37b) is

replaced by

$$\sum_{w=0}^{N} \hat{q}_w(z_1)\, z_2^w = P[z_1,\, \boldsymbol{A}_I - \boldsymbol{b}_{Il}\, \boldsymbol{c}_k]\, P[z_2,\, \check{\boldsymbol{F}}(z_1)] + (D_{kl} - 1)\, P[z_1,\, \boldsymbol{A}_I]\, P[z_2,\, \hat{\boldsymbol{E}}(z_2)] \qquad (4.47)$$

and (4.38b) is replaced by

$$\check{\boldsymbol{F}}(z_1) = z_1\, (\boldsymbol{A}_{II} - \boldsymbol{b}_{IIl}\, \boldsymbol{c}_k)\, (z_1\, \mathbf{I} - \boldsymbol{A}_I + \boldsymbol{b}_{Il}\, \boldsymbol{c}_k)^{-1} \qquad (4.48)$$

for $(k = 1, 2, \ldots, s)$ and $(l = 1, 2, \ldots, t)$, where $P[z_1,\, \boldsymbol{A}_I - \boldsymbol{b}_{Il}\, \boldsymbol{c}_k]$ and $P[z_2,\, \check{\boldsymbol{F}}(z_1)]$ are the characteristic polynomials of $(\boldsymbol{A}_I - \boldsymbol{b}_{Il}\, \boldsymbol{c}_k)$ and $\check{\boldsymbol{F}}(z_1)$, respectively.

4.4 Computational Evaluation of the Transfer-Function Matrix Algorithms

In this section, four examples are presented to demonstrate the numerical efficiency of the algorithms proposed in Sections 4.2 and 4.3 for the determination of the transfer functions and the transfer-function matrices for SISO and MIMO 2-D discrete systems, respectively. In addition, these algorithms are compared with the Fadeeva and DFT algorithms given in [7, 50]. As expected, the various algorithms yield an identical transfer function or transfer-function matrix for a given 2-D discrete system.

4.4.1 Examples for the SISO Case

Example 4.1

Consider the example of an SISO 2-D discrete system of order (2, 6) in [2]. The system is represented by the Roesser state-space model given by (4.1a, b) with

$$\mathbf{A}_1 = \begin{bmatrix} 0.500 & 0.007 \\ -0.007 & 0.500 \end{bmatrix}, \quad \mathbf{A}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T$$

$$\mathbf{A}_2 = \begin{bmatrix} 0.012 & -0.008 & 0.028 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.012 & 0.008 & 0.012 \end{bmatrix}$$

$$\mathbf{A}_4 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.845 & -2.657 & 2.810 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -0.845 & -2.657 & -2.810 \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} \mathbf{b}_1^T & \mathbf{b}_2^T \end{bmatrix}^T = \begin{bmatrix} 0.134 & 1 & | & -0.657 & 0.036 & 0.269 & 0.805 & 1 & 2 \end{bmatrix}^T$$

$$\mathbf{c} = \begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 \end{bmatrix} = \begin{bmatrix} 0.983 & 0.500 & | & -1 & 0 & 1 & 2 & 3 & 1 \end{bmatrix}$$

$$d = 0$$

The corresponding F-M state-space model (4.28a, b) of the system can be obtained using the relations given in [15] as [1]

[1]The F-M state-space model obtained in (4.49) may not have minimum order.

$$\begin{aligned} \boldsymbol{A}_{\mathrm{I}} &= \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \boldsymbol{A}_{\mathrm{II}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \\ \boldsymbol{b}_{\mathrm{I}} &= \begin{bmatrix} \mathbf{b}_1^T & \mathbf{0} \end{bmatrix}^T, \quad \boldsymbol{b}_{\mathrm{II}} = \begin{bmatrix} \mathbf{0} & \mathbf{b}_2^T \end{bmatrix}^T, \quad \boldsymbol{c} = \mathbf{c} \end{aligned} \tag{4.49}$$

The transfer function given by the various algorithms is

$$H(z_1, z_2) = \frac{\begin{bmatrix} 1 & z_2^{-1} & \cdots & z_2^{-6} \end{bmatrix} \mathbf{N}_{\mathrm{R}} \begin{bmatrix} 1 & z_1^{-1} & z_1^{-2} \end{bmatrix}^T}{\begin{bmatrix} 1 & z_2^{-1} & \cdots & z_2^{-6} \end{bmatrix} \mathbf{D}_{\mathrm{R}} \begin{bmatrix} 1 & z_1^{-1} & z_1^{-2} \end{bmatrix}^T}$$

where

$$\mathbf{D}_{\mathrm{R}} = \begin{bmatrix} 1.0000 & -1.0000 & 0.2500 \\ 0.0000 & 0.0000 & 0.0001 \\ -2.5821 & 2.5821 & -0.6453 \\ 0.0000 & 0.0000 & 0.0002 \\ 2.3107 & -2.3107 & 0.5778 \\ 0.0000 & 0.0000 & 0.0000 \\ -0.7140 & 0.7140 & -0.1785 \end{bmatrix}$$

$$\mathbf{N}_{\mathrm{R}} = \begin{bmatrix} 0.0000 & 0.6317 & -0.3094 \\ 7.5360 & -6.3818 & 1.3419 \\ -0.8882 & 2.0949 & -0.7442 \\ -23.9776 & 25.1974 & -6.4878 \\ 8.8500 & -9.2265 & 2.5091 \\ 16.5056 & -18.8216 & 5.3540 \\ -7.9463 & 6.2887 & -1.1409 \end{bmatrix}$$

The computational complexity measured in FLOPS for the various algorithms is given in Table 4.1.

Example 4.2

Consider an SISO 2-D discrete system of order (16, 8) represented by the Roesser state-space model given in (4.1a, b), where each entry of $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, as well as d is a random number chosen from a normal distribution with zero mean and unit variance. The corresponding F-M state-space model of the system $\{\boldsymbol{A}_{\mathrm{I}}, \boldsymbol{A}_{\mathrm{II}}, \boldsymbol{b}_{\mathrm{I}}, \boldsymbol{b}_{\mathrm{II}}, \boldsymbol{c}, d\}$ can be obtained using the relations given in (4.49). The computational complexity for the various algorithms is given in Table 4.1.

4.4.2 Examples for the MIMO Case

Example 4.3

Consider the example of a two-input two-output 2-D discrete system represented by the Roesser state-space model of order (2, 2), originally presented in [50] to demonstrate the Fadeeva algorithm. The model is given by (2.1a, b) with

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \left[\begin{array}{cc|cc} 2 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ \hline 1 & 0 & -2 & 0 \\ 0 & 1 & 0 & -2 \end{array}\right]$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_1^T & \mathbf{B}_2^T \end{bmatrix}^T = \left[\begin{array}{cc|cc} 1 & -1 & 2 & 1 \\ 1 & 0 & 1 & 0 \end{array}\right]^T$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_1 & \mathbf{C}_2 \end{bmatrix} = \left[\begin{array}{cc|cc} 1 & 1 & 0 & -1 \\ 0 & -1 & 1 & 1 \end{array}\right]$$

$$\mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The corresponding F-M state-space model (4.42a, b) of the system $\{\boldsymbol{A}_{\mathrm{I}}, \boldsymbol{A}_{\mathrm{II}}, \boldsymbol{B}_{\mathrm{I}}, \boldsymbol{B}_{\mathrm{II}}, \boldsymbol{C}, \mathbf{D}\}$ can be obtained using the relations given in [15] as

$$\begin{aligned} \boldsymbol{A}_{\mathrm{I}} &= \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad \boldsymbol{A}_{\mathrm{II}} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \\ \boldsymbol{B}_{\mathrm{I}} &= \begin{bmatrix} \mathbf{B}_1^T & \mathbf{0} \end{bmatrix}^T, \quad \boldsymbol{B}_{\mathrm{II}} = \begin{bmatrix} \mathbf{0} & \mathbf{B}_2^T \end{bmatrix}^T, \quad \boldsymbol{C} = \mathbf{C} \end{aligned} \tag{4.50}$$

The various algorithms yield the transfer-function matrix

$$\mathbf{H}(z_1, z_2) = \begin{bmatrix} H_{11}(z_1, z_2) & H_{12}(z_1, z_2) \\ H_{21}(z_1, z_2) & H_{22}(z_1, z_2) \end{bmatrix}$$

where each component is given by

$$H_{kl}(z_1, z_2) = \frac{\begin{bmatrix} 1 & z_2^{-1} & z_2^{-2} \end{bmatrix} \mathbf{N}_{kl} \begin{bmatrix} 1 & z_1^{-1} & z_1^{-2} \end{bmatrix}^T}{\begin{bmatrix} 1 & z_2^{-1} & z_2^{-2} \end{bmatrix} \mathbf{D}_{\mathrm{R}} \begin{bmatrix} 1 & z_1^{-1} & z_1^{-2} \end{bmatrix}^T}$$

with

$$\mathbf{D}_{\mathrm{R}} = \begin{bmatrix} 1 & -2 & -1 \\ 4 & -10 & -2 \\ 4 & -12 & 1 \end{bmatrix}$$

and

$$\begin{bmatrix} \mathbf{N}_{11} & \mathbf{N}_{12} \\ \mathbf{N}_{21} & \mathbf{N}_{22} \end{bmatrix} = \left[\begin{array}{ccc|ccc} 0 & 0 & 2 & 0 & 1 & 1 \\ -1 & 6 & 7 & 0 & 5 & 3 \\ -2 & 13 & 0 & 0 & 6 & 0 \\ - & - & - & - & - & - \\ 0 & 1 & -3 & 0 & 0 & -1 \\ 3 & -3 & -14 & 1 & -1 & -5 \\ 6 & -13 & -8 & 2 & -3 & -4 \end{array}\right]$$

The computational complexity for the various algorithms is given in Table 4.1.

Example 4.4

Consider a four-input two-output 2-D discrete system of order (8, 16), represented by the Roesser state-space model given in (2.1a, b), where each entry of **A**, **B**, **C**, and **D** is a random number chosen from a normal distribution with zero mean and unit variance. The corresponding F-M state-space model of the system $\{\boldsymbol{A}_{\rm I}, \boldsymbol{A}_{\rm II}, \boldsymbol{B}_{\rm I}, \boldsymbol{B}_{\rm II}, \boldsymbol{C}, D\}$ can be obtained using the relations given in (4.50). The computational complexity for the various algorithms is given in Table 4.1.

4.4.3 Performance Evaluation of SISO and MIMO Algorithms

From Table 4.1, we note that the determination of the transfer-function matrix of a 2-D discrete system from the Roesser state-space model is consistently computationally less demanding than that from F-M state-space

Table 4.1: Performance of the transfer-function matrix algorithms

Algorithm	Computational Complexity, FLOPS			
	SISO Example		MIMO Example	
	A	B	A	B
RM 4.1*	3.786×10^4	5.375×10^6	5.657×10^3	7.379×10^6
RM 4.2	8.424×10^4	2.591×10^6	5.692×10^3	2.325×10^7
*F-M** 4.3*	2.980×10^5	4.140×10^7	6.195×10^4	1.189×10^8
F-M 4.4	6.123×10^5	2.674×10^7	6.044×10^4	1.877×10^8
Fadeeva [50]	2.431×10^5	1.143×10^8	1.360×10^4	1.440×10^8
DFT [7]	1.815×10^5	2.645×10^7	2.060×10^4	3.180×10^7

* Based on Roesser state-space model.
** Based on F-M state-space model.

model. Table 4.1 also shows that Algorithms 4.1 and 4.2 are more efficient than the Fadeeva and DFT algorithms [7, 50].

One may also note that Algorithms 4.1 and 4.2 (or Algorithms 4.3 and 4.4) require different amounts of computation if $m \neq n$. Extensive results with $1 \leq m \leq 30$ and $1 \leq n \leq 30$ have shown that Algorithm 4.1 (Algorithm 4.3) requires less computation than Algorithm 4.2 (Algorithm 4.4) when $m < n$ (see SISO Example 4.1 and MIMO Example 4.4), and Algorithm 4.2 (Algorithm 4.4) requires less computation when $m > n$ (see SISO Example 4.2).

4.5 Conclusion

New algorithms for the determination of the transfer-function matrix of a 2-D discrete system from the Roesser state-space model have been proposed. The algorithms are based on the determination technique for 1-D polynomials. The computational efficiency of the algorithms has been examined and found to be superior relative to that of the algorithms described in [50, 7]. In addition, two algorithms based on the Fornasini-Marchesini state-space model have been developed.

The theorems and algorithms proposed in Chapters 2, 3 and 4 provide the necessary framework for the analysis and design of 2-D digital filters, which will be discussed in the next chapter.

Chapter 5

Design Environment for 2-D Digital Filters

5.1 Introduction

A general purpose design environment for a variety of recursive and nonrecursive 2-D digital filters has been developed. The state-of-the-art design environment, which is developed within the MATLAB [45] environment, consists of two independent modules. The first module is a *user-interface* software which assists the novice to design 2-D digital filters. The second module is a *design toolbox* which consists of a library of design functions to assist an expert to design highly specialized 2-D digital filters. Each module is composed of software routines which will be detailed in Section 5.3.

The design environment integrates the singular-value decomposition (SVD) design method [4, 33, 38] with the balanced model-reduction algorithms presented in Chapter 2, and the 2-D transfer-function matrix algorithms proposed in Chapter 4. Since the SVD method is suitable for the design of 2-D

filters with arbitrary amplitude and phase responses [37], the design environment can be used for the design of standard 2-D filters such as *circularly symmetric* or *quadrantally symmetric* filters and for the design of nonstandard *user-defined* 2-D filters.

This chapter is organized as follows. The structure of the design environment developed is described in Section 5.2. Section 5.3 gives a step-by-step description of the software routines that are used in the two modules. Section 5.4 defines four groups of 2-D digital filters that one can design using the two software modules. Four design examples that demonstrate the use of the design environment developed are presented in Section 5.5.

5.2 Structure of the Design Environment

The design of a 2-D digital filter entails finding a 2-D transfer function of order (m, n), denoted by $H(z_1, z_2)$, such that the frequency response of the filter approximates a desired frequency response. The frequency response of the filter to be designed can be expressed as

$$H(\omega_1, \omega_2) = H(z_1, z_2)\Big|_{z_1 = e^{j2\pi\omega_1},\ z_2 = e^{j2\pi\omega_2}}$$

where $\omega_1,\ \omega_2 \in [0, 1]$ denote the normalized frequencies with $\omega_1, \omega_2 = 1$ corresponding to the normalized Nyquist frequencies [5]. The two alternative design criteria that are often used are as follows:

(i). Design a filter that meets prescribed maximum approximation errors in the passband and stopband regions. These errors are often referred

to as maximum *passband* and *stopband ripples*, denoted by ξ_P and ξ_S, respectively.

(ii). Design a filter that has a prescribed order, denoted by (m, n).

It was shown in [37] that the SVD design method is valid as long as the frequency response of a desired filter, denoted by $H^d(\omega_1, \omega_2)$, satisfies the conditions

$$|H^d(\omega_1, \omega_2)| = |H^d(-\omega_1, -\omega_2)| \tag{5.1a}$$

$$\angle H^d(\omega_1, \omega_2) = -\angle H^d(-\omega_1, -\omega_2) \tag{5.1b}$$

where $|H^d(\omega_1, \omega_2)|$ is the amplitude and $\angle H^d(\omega_1, \omega_2)$ is the phase responses of the desired 2-D filter. Note that any 2-D rational function with real coefficients satisfies (5.1a, b) and, therefore, the SVD method can be used for the design of 2-D filters with arbitrary amplitude and phase responses.

5.2.1 User-Interface Design Software

One module of the design environment is the *user-interface software*. This module provides step-by-step instructions to assist a user to design a 2-D digital filter properly. Seven types of recursive and nonrecursive 2-D digital filters with prescribed maximum passband and stopband ripples or prescribed filter order, as shown in Table 5.1, can be designed by using the user-interface software. A design example is given in Section 5.5 to illustrate the usage of the module.

Table 5.1: Types of recursive and nonrecursive 2-D digital filters

Type Number	Recursive or Nonrecursive Filters
1	CS* - Lowpass
2	CS - Highpass
3	CS - Bandpass
4	CS - Bandstop
5	Fan - Bandpass
6	Fan - Bandstop
7	User-defined*

CS*: Circularly Symmetric.
User-defined*: A non-standard filter which satisfies relations (5.1a, b).

5.2.2 Design Toolbox

Another module of the design environment is implemented as a *design toolbox* which consists of functions that can be used to design various types of recursive and nonrecursive 2-D filters such as those listed in Tables 5.2 and 5.3. For example, the command for designing a recursive, circularly symmetric, highpass 2-D digital filter with specified maximum passband and stopband ripples is

$$[e_P, e_S, r_1, r_2, r_s, \mathbf{d}_{R1}, \mathbf{d}_{R2}, \mathbf{N}_L, \mathbf{N}_U] = IIR\_CS\_HP\_RS\,(\omega_S, \omega_P, \xi_P, \xi_S, K, L, k_1)$$

Table 5.2: Functions for the design of nonrecursive 2-D digital filters.

Nonrecursive Filters	Ripples Specified	Order Specified
CS - Lowpass	FIR_CS_LP_RS	FIR_CS_LP_OS
CS - Highpass	FIR_CS_HP_RS	FIR_CS_HP_OS
CS - Bandpass	FIR_CS_BP_RS	FIR_CS_BP_OS
CS - Bandstop	FIR_CS_BS_RS	FIR_CS_BS_OS
Fan - Bandpass	FIR_FN_BP_RS	FIR_FN_BP_OS
Fan - Bandstop	FIR_FN_BS_RS	FIR_FN_BS_OS
User-defined	FIR_USER_RS	FIR_USER_OS

Table 5.3: Functions for the design of recursive 2-D digital filters.

Recursive Filters	Ripples Specified	Order Specified
CS - Lowpass	IIR_CS_LP_RS	IIR_CS_LP_OS
CS - Highpass	IIR_CS_HP_RS	IIR_CS_HP_OS
CS - Bandpass	IIR_CS_BP_RS	IIR_CS_BP_OS
CS - Bandstop	IIR_CS_BS_RS	IIR_CS_BS_OS
Fan - Bandpass	IIR_FN_BP_RS	IIR_FN_BP_OS
Fan - Bandstop	IIR_FN_BS_RS	IIR_FN_BS_OS
User-defined	IIR_USER_RS	IIR_USER_OS

where the input parameters for the design are the normalized passband and stopband edges ω_P and ω_S with $0 < \omega_S < \omega_P < 1$; prescribed passband and stopband ripples ξ_P and ξ_S; and the total numbers of *sample points* K and L for the desired frequency response in the normalized frequency regions $\omega_1 \in [0, 1]$, $\omega_2 \in [0, 1]$, respectively. Parameter $k_1 = 1$ or 2 specifies the Kaiser or Hamming window [5] for the design of 1-D nonrecursive digital filters.

The design function returns the maximum design errors in the passband and stopband e_P and e_S; the order of the recursive filter (r_1, r_2); the number of sections of the parallel structure r_s; the coefficient vectors of two separable denominator polynomials $\mathbf{d}_{R1}$ and $\mathbf{d}_{R2}$; and the coefficient matrices of the numerator polynomial $\mathbf{N}_L$ and $\mathbf{N}_U$ where $\mathbf{N}_L\mathbf{N}_U$ is the LU decomposition of the coefficient matrix of the numerator polynomial.

Similarly, the command for designing a recursive circularly symmetric highpass 2-D filter with specified filter order (m, n) is

$$[e_P, e_S, r_1, r_2, r_s, \mathbf{d}_{R1}, \mathbf{d}_{R2}, \mathbf{N}_L, \mathbf{N}_U] = IIR\_CS\_HP\_OS\,(\omega_S, \omega_P, m, n, K, L, k_1)$$

5.3 Design Software Routines

The MATLAB environment has been chosen due to its capability to handle matrix operations and its superb graphics capability. There are seven routines that can be used to support the user interface software or to construct

the functions in the design toolbox. For the sake of completeness, these routines are listed as follows.

SPSVD – Sampling and decomposition of a desired filter.
DSFIR – Design of a linear-phase nonrecursive 2-D filter.
ODFIR – Determination of the order of a nonrecursive 2-D filter.
BAREL – Balanced realization of the 2-D filter.
ORIIR – Order reduction of the recursive 2-D filter.
TRANS – Direct realization of the reduced-order 2-D filter.
SNLUD – Determination of section number and LU decomposition.

Detailed descriptions for each routine are as follows.

SPSVD: Sampling and Decomposition of a Desired Filter

At the beginning of the design process, the amplitude response of a desired filter is sampled and decomposed using SPSVD. A step-by-step description of this routine is given below.

Step 1: Sample the amplitude response of a desired filter as

$$H^d(k, l) = |H^d\left(e^{j(2\pi k/K)}, e^{j(2\pi l/L)}\right)| \tag{5.2}$$

where $k = \{0, 1, \ldots, (K-1)\}$, $l = \{0, 1, \ldots, (L-1)\}$, $\{\omega_1 = k/K,\ \omega_1 \in [0, 1]\}$, and $\{\omega_2 = l/L,\ \omega_2 \in [0, 1]\}$. The sampled amplitude response matrix of the desired filter is denoted by

$$\mathbf{H}^d = [H^d(k, l)]$$

Step 2: Obtain the SVD of the sampled amplitude response matrix of the desired filter as

$$\mathbf{H}^d = \sum_{k=1}^{r_d} s_k \mathbf{u}_k \mathbf{v}_k^T = \sum_{k=1}^{r_d} \tilde{\mathbf{u}}_k \tilde{\mathbf{v}}_k^T \tag{5.3}$$

where

$$s_1 \geq s_2 \geq \cdots \geq s_{r_d} > 0$$

are the nonzero *singular values* of $\mathbf{H}^d$ with rank r_d, $\mathbf{u}_k$ and $\mathbf{v}_k$ are the kth pair of the *left* and *right singular vectors* of $\mathbf{H}^d$, and

$$\tilde{\mathbf{u}}_k = s_k^{1/2} \mathbf{u}_k, \qquad \tilde{\mathbf{v}}_k = s_k^{1/2} \mathbf{v}_k$$

are the *weighted singular vectors* with a weight of $s_k^{1/2}$. For simplicity, the terms $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$ are henceforth called the kth pair of *singular vectors* of $\mathbf{H}^d$.

DSFIR: Design of a Linear-Phase Nonrecursive 2-D Filter

Once the singular vectors $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$ ($k = 1, 2, \ldots, r_d$) of the sampled frequency response matrix are obtained, the design of a linear-phase nonrecursive 2-D filter can be carried out by designing several 1-D sub-filters. The implementation of this routine is outlined as follows.

Step 1: Design two sets of nonrecursive 1-D digital filters. As each singular vector $\tilde{\mathbf{u}}_k$ or $\tilde{\mathbf{v}}_k$ is either mirror-image symmetric or anti-symmetric with respect to its midpoint [33], only one-half of each singular vector is needed for computation.

The window method [35, 57] is used with the Kaiser or Hamming window to design two sets of 1-D digital filters. Let the transfer functions of the 1-D digital filters in the kth section be given by

$$S_k(z_1) = \sum_{l_1=0}^{m} s_k(l_1)\, z_1^{-l_1}, \qquad T_k(z_2) = \sum_{l_2=0}^{n} t_k(l_2)\, z_2^{-l_2}$$

where s_k and t_k are the coefficients of $S_k(z_1)$ and $T_k(z_2)$, respectively. The frequency responses of the 1-D digital filters approximate the singular vectors $\tilde{\mathbf{u}}_k$ and $\tilde{\mathbf{v}}_k$, respectively.

Step 2: Form the transfer function of a nonrecursive 2-D filter as

$$H(z_1, z_2) = \sum_{k=1}^{r_d} S_k(z_1)\, T_k(z_2) = \hat{\mathbf{z}}_1^T\, \mathbf{C}_{\text{H}}\, \hat{\mathbf{z}}_2 \tag{5.4}$$

where

$$C_{\text{H}}(l_1, l_2) = \sum_{k=1}^{r_d} s_k(l_1)\, t_k(l_2), \qquad \text{for} \quad \begin{cases} l_1 = 0, 1, \ldots, m \\ l_2 = 0, 1, \ldots, n \end{cases}$$

is the (l_1, l_2)th element of the coefficient matrix $\mathbf{C}_{\text{H}}$, and $\hat{\mathbf{z}}_1$ and $\hat{\mathbf{z}}_2$ are defined by (4.16).

Step 3: Compute the maximum design errors in the passband and stopband frequency regions.

ODFIR: Determination of the Order of Nonrecursive 2-D Filter

The order of the nonrecursive 2-D filter is determined by a trial-and-error approach to assure that the maximum design errors in the passband and

stopband meet the requirements. Note that this routine only applies when the maximum passband and stopband ripples are specified.

BAREL: Balanced Realization of the 2-D Filter

After the order of the nonrecursive 2-D digital filter is determined, the balanced realization algorithms presented in Chapter 2 can be employed to obtain a recursive 2-D filter. The implementation of this routine is outlined as follows.

Step 1: Once the nonrecursive 2-D digital filter of order (m, n) is obtained, the Roesser state-space model (4.1a, b) of the filter can be obtained with

$$\mathbf{A}_1 = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ 0 & \mathbf{0} \end{bmatrix}, \qquad \mathbf{A}_3 = \mathbf{0}, \qquad \mathbf{A}_4 = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ 0 & \mathbf{0} \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} C_{\mathrm{H}}(1, n) & \cdots & C_{\mathrm{H}}(1, 1) \\ \vdots & & \vdots \\ C_{\mathrm{H}}(m, n) & \cdots & C_{\mathrm{H}}(m, 1) \end{bmatrix}$$

$$\mathbf{b}_1 = \begin{bmatrix} C_{\mathrm{H}}(1, 0) & \cdots & C_{\mathrm{H}}(m, 0) \end{bmatrix}^T$$

$$\mathbf{b}_2 = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix}^T, \qquad \mathbf{c}_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$$

$$\mathbf{c}_2 = \begin{bmatrix} C_{\mathrm{H}}(0, n) & \cdots & C_{\mathrm{H}}(0, 1) \end{bmatrix}, \qquad d = C_{\mathrm{H}}(0, 0)$$

Step 2: Compute the pseudo-gramians of the recursive 2-D filter. Since $\mathbf{A}_3 = \mathbf{0}$, the recursive 2-D filter obtained in Step 1 is separable. According

to Corollary 2.1, the QBMR reduces to the PBMR. Further, since both $\mathbf{A}_1$ and $\mathbf{A}_4$ are nilpotent, that is,

$$\mathbf{A}_1^m = 0 \quad \text{and} \quad \mathbf{A}_4^n = 0$$

the computation of the pseudo-gramians is simplified [38] as

$$\mathbf{P}_1^p = \sum_{k=0}^{m-1} \mathbf{A}_1^k \, \mathbf{P}_{\mathrm{C}} \, \mathbf{P}_{\mathrm{C}}^T \, (\mathbf{A}_1^T)^k, \qquad \mathbf{P}_2^p = \mathbf{I} \tag{5.5a}$$

$$\mathbf{Q}_2^p = \sum_{k=0}^{n-1} (\mathbf{A}_4^T)^k \, \mathbf{Q}_{\mathrm{C}}^T \, \mathbf{Q}_{\mathrm{C}} \, \mathbf{A}_4^k, \qquad \mathbf{Q}_1^p = \mathbf{I} \tag{5.5b}$$

where $\mathbf{P}_{\mathrm{C}}$ and $\mathbf{Q}_{\mathrm{C}}$ are obtained as

$$\mathbf{P}_{\mathrm{C}} = \begin{bmatrix} C_{\mathrm{H}}(1,0) & \cdots & C_{\mathrm{H}}(1,n) \\ \vdots & & \vdots \\ C_{\mathrm{H}}(m,0) & \cdots & C_{\mathrm{H}}(m,n) \end{bmatrix}$$

$$\mathbf{Q}_{\mathrm{C}} = \begin{bmatrix} C_{\mathrm{H}}(0,n) & \cdots & C_{\mathrm{H}}(0,1) \\ \vdots & & \vdots \\ C_{\mathrm{H}}(m,n) & \cdots & C_{\mathrm{H}}(m,1) \end{bmatrix}$$

Step 3: Obtain the PBR for the recursive 2-D filter. Apply Algorithm 2.1 to $\{\mathbf{P}_1^p, \ \mathbf{Q}_1^p\}$ and $\{\mathbf{P}_2^p, \ \mathbf{Q}_2^p\}$ to find nonsingular matrices $\mathbf{T}_1^p$ and $\mathbf{T}_2^p$, respectively. Then, construct the balancing transformation matrix as

$$\mathbf{T}^p = \mathbf{T}_1^p \oplus \mathbf{T}_2^p$$

and the PBR matrices $\{\mathbf{A}^p, \ \mathbf{b}^p, \ \mathbf{c}^p, \ d\}$ as

$$\mathbf{A}^p = \begin{bmatrix} (\mathbf{T}_1^p)^{-1}\mathbf{A}_1\mathbf{T}_1^p & (\mathbf{T}_1^p)^{-1}\mathbf{A}_2\mathbf{T}_2^p \\ \mathbf{0} & (\mathbf{T}_2^p)^{-1}\mathbf{A}_4\mathbf{T}_2^p \end{bmatrix}, \qquad \mathbf{b}^p = \begin{bmatrix} (\mathbf{T}_1^p)^{-1}\,\mathbf{b}_1 \\ (\mathbf{T}_2^p)^{-1}\,\mathbf{b}_2 \end{bmatrix}$$

$$\mathbf{c}^p = \begin{bmatrix} \mathbf{c}_1\mathbf{T}_1^p & \mathbf{c}_2\mathbf{T}_2^p \end{bmatrix}$$

ORIIR: Order Reduction of the Recursive 2-D Filter

After the PBR of the 2-D filter is obtained, the PBMR algorithms presented in Chapter 2 can be employed to obtain a reduced-order recursive 2-D filter. Together, routines BAREL and ORIIR produce a stable, low-order, recursive 2-D digital filter with separable denominator [43]. In addition, the phase response of the filter obtained is approximately linear in the passband [38]. The implementation of this routine is outlined below.

Step 1: Obtain a reduced-order recursive 2-D filter. The Roesser state-space model of the reduced-order filter can be obtained by partitioning the PBR matrices $\mathbf{A}^p$, $\mathbf{b}^p$, and $\mathbf{c}^p$ of the recursive 2-D filter of order (m, n) as

$$\mathbf{A}^p = \left[\begin{array}{cc|cc} \mathbf{A}_1^{pr} & \mathbf{A}_{12}^p & \mathbf{A}_2^{pr} & \mathbf{A}_{22}^p \\ \mathbf{A}_{13}^p & \mathbf{A}_{14}^p & \mathbf{A}_{23}^p & \mathbf{A}_{24}^p \\ \hline \mathbf{0} & \mathbf{0} & \mathbf{A}_4^{pr} & \mathbf{A}_{42}^p \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{43}^p & \mathbf{A}_{44}^p \end{array}\right], \qquad \mathbf{b}^p = \left[\begin{array}{c} \mathbf{b}_1^{pr} \\ \mathbf{b}_{12}^p \\ \hline \mathbf{b}_2^{pr} \\ \mathbf{b}_{22}^p \end{array}\right], \qquad \mathbf{c}^p = \left[\begin{array}{c} \mathbf{c}_1^{pr\,T} \\ \mathbf{c}_{12}^{p\,T} \\ \hline \mathbf{c}_2^{pr\,T} \\ \mathbf{c}_{22}^{p\,T} \end{array}\right]^T$$

with

$$\mathbf{A}_1^{pr} \in \Re^{r_1 \times r_1}, \qquad \mathbf{A}_2^{pr} \in \Re^{r_1 \times r_2}, \qquad \mathbf{b}_1^{pr} \in \Re^{r_1 \times 1}, \qquad \mathbf{c}_1^{pr} \in \Re^{1 \times r_1}$$

$$\mathbf{A}_3^{pr} = \mathbf{0}, \qquad \mathbf{A}_4^{pr} \in \Re^{r_2 \times r_2}, \qquad \mathbf{b}_2^{pr} \in \Re^{r_2 \times 1}, \qquad \mathbf{c}_2^{pr} \in \Re^{1 \times r_2}$$

The reduced-order filter of order (r_1, r_2) is characterized by $\{\mathbf{A}^{pr}, \mathbf{b}^{pr}, \mathbf{c}^{pr}, d\}$ with

$$\mathbf{A}^{pr} = \begin{bmatrix} \mathbf{A}_1^{pr} & \mathbf{A}_2^{pr} \\ \mathbf{0} & \mathbf{A}_4^{pr} \end{bmatrix}, \quad \mathbf{b}^{pr} = \begin{bmatrix} \mathbf{b}_1^{pr} \\ \mathbf{b}_2^{pr} \end{bmatrix}, \quad \mathbf{c}^{pr} = \begin{bmatrix} \mathbf{c}_1^{prT} \\ \mathbf{c}_2^{prT} \end{bmatrix}^T \tag{5.6}$$

Step 2: Obtain the sampled amplitude-response matrix of the reduced-order filter. The transfer function of the reduced-order filter is given by

$$H^r(z_1, z_2) = \mathbf{c}^{pr} \left[\mathbf{I}(z_1, z_2) - \mathbf{A}^{pr}\right]^{-1} \mathbf{b}^{pr} + d \tag{5.7}$$

The sampled amplitude-response matrix of the reduced-order filter can then be obtained as

$$\mathbf{H}^r = [H^r(k, l)] = [\,|H^r\left(e^{j(2\pi k/K)}, e^{j(2\pi l/L)}\right)|\,]$$

Step 3: Compute the maximum design errors of the reduced-order filter in the passband and stopband.

Step 4: Determine the order of the reduced-order filter. The order of the reduced-order filter (r_1, r_2) is increased until the specifications of the maximum passband and stopband ripples of the filter are satisfied.

TRANS: Direct Realization of the Reduced-Order Filter

Once the reduced-order recursive 2-D filter is obtained, the algorithms presented in Chapter 4 can be employed to determine the coefficient matrices

of the 2-D transfer function of the filter. The implementation of this routine is outlined below.

Step 1: Simplify Algorithms 4.1 or 4.2. Since $\mathbf{A}_3^{pr} = \mathbf{0}$, the reduced-order filter is separable. Hence, (4.8) and (4.10a, b) become

$$\begin{bmatrix} \mathbf{E}(z_2) & \mathbf{f}(z_2) \\ \mathbf{g}(z_2) & h(z_2) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1^{pr} & \mathbf{b}_1^{pr} \\ \mathbf{c}_1^{pr} & d \end{bmatrix} + \begin{bmatrix} \mathbf{A}_2^{pr} \\ \mathbf{c}_2^{pr} \end{bmatrix} \left[z_2 \mathbf{I} - \mathbf{A}_4^{pr} \right]^{-1} \begin{bmatrix} \mathbf{0} & \mathbf{b}_2^{pr} \end{bmatrix} \quad (5.8)$$

$$\sum_{k=0}^{m} p_k(z_2)\, z_1^k = P(z_2,\, \mathbf{A}_4^{pr})\, P(z_1,\, \mathbf{A}_1^{pr}) \quad (5.9a)$$

$$\sum_{k=0}^{m} q_k(z_2)\, z_1^k = P(z_2,\, \mathbf{A}_4^{pr}) \Big\{ P\left[z_1,\, \mathbf{A}_1^{pr} - \mathbf{f}(z_2)\, \mathbf{c}_1^{pr} \right] + \left[h(z_2) - 1 \right] P(z_1,\, \mathbf{A}_1^{pr}) \Big\} \quad (5.9b)$$

respectively.

Step 2: Determine the coefficient matrices of the transfer function of the reduced-order filter. The simplified Algorithms 4.1 or 4.2 can be used to determine the coefficient matrix $\mathbf{N}_{\mathrm{R}}$ of the numerator polynomial and the coefficient vectors $\mathbf{d}_{\mathrm{R}1}$ and $\mathbf{d}_{\mathrm{R}2}$ of the separable denominator polynomials. The transfer function of the reduced-order filter is given by

$$H^r(z_1, z_2) = \frac{\left[1 \;\; z_2^{-1} \;\; \cdots \;\; z_2^{-r_2} \right] \mathbf{N}_{\mathrm{R}} \left[1 \;\; z_1^{-1} \;\; \cdots \;\; z_1^{-r_1} \right]^T}{\left[1 \;\; z_2^{-1} \;\; \cdots \;\; z_2^{-r_2} \right] \mathbf{d}_{\mathrm{R}2}\, \mathbf{d}_{\mathrm{R}1}^T \left[1 \;\; z_1^{-1} \;\; \cdots \;\; z_1^{-r_1} \right]^T}$$

$$= \frac{\check{\mathbf{z}}_2^T\, \mathbf{N}_{\mathrm{R}}\, \check{\mathbf{z}}_1}{\check{\mathbf{z}}_2^T\, \mathbf{d}_{\mathrm{R}2}\, \mathbf{d}_{\mathrm{R}1}^T\, \check{\mathbf{z}}_1} \quad (5.10)$$

where

$$\check{\mathbf{z}}_1 = [1,\, z_1^{-1},\, \ldots,\, z_1^{-r_1}]^T, \qquad \check{\mathbf{z}}_2 = [1,\, z_2^{-1},\, \ldots,\, z_2^{-r_2}]^T$$

SNLUD: Determination of Section Number and LU Decomposition

The number of sections for the parallel structure of a nonrecursive or reduced-order recursive filter is determined by a trial-and-error approach. The implementation of this routine is as follows:

Step 1: Decompose the coefficient matrix of a nonrecursive or reduced-order recursive filter using the SVD method. That is, decompose the coefficient matrix $\mathbf{C}_{\text{H}}$ of the transfer function of a nonrecursive filter as

$$\mathbf{C}_{\text{H}} = \sum_{k=1}^{r_{\text{C}}} \hat{s}_k\, \hat{\mathbf{u}}_k\, \hat{\mathbf{v}}_k^T = \sum_{k=1}^{r_{\text{C}}} \check{\mathbf{u}}_k\, \check{\mathbf{v}}_k^T \tag{5.11}$$

or the coefficient matrix $\mathbf{N}_{\text{R}}$ of the numerator polynomial of the transfer function of a reduced-order recursive filter as

$$\mathbf{N}_{\text{R}} = \sum_{k=1}^{r_{\text{N}}} \check{s}_k\, \boldsymbol{u}_k\, \boldsymbol{v}_k^T = \sum_{k=1}^{r_{\text{N}}} \hat{\boldsymbol{u}}_k\, \hat{\boldsymbol{v}}_k^T \tag{5.12}$$

where

$$\hat{s}_1 \geq \hat{s}_2 \geq \cdots \geq \hat{s}_{r_{\text{C}}} > 0$$

$$\check{s}_1 \geq \check{s}_2 \geq \cdots \geq \check{s}_{r_{\text{N}}} > 0$$

are the nonzero singular values of $\mathbf{C}_{\text{H}}$ and $\mathbf{N}_{\text{R}}$, respectively; $\hat{\mathbf{u}}_k$ and $\hat{\mathbf{v}}_k$ are the kth pair of the left and right singular vectors of $\mathbf{C}_{\text{H}}$, while $\boldsymbol{u}_k$ and $\boldsymbol{v}_k$ are

the kth pair of the left and right singular vectors of $\mathbf{N}_R$; and

$$\check{\mathbf{u}}_k = \hat{s}_k^{1/2}\,\hat{\mathbf{u}}_k, \qquad \check{\mathbf{v}}_k = \hat{s}_k^{1/2}\,\hat{\mathbf{v}}_k$$

$$\hat{\boldsymbol{u}}_k = \check{s}_k^{1/2}\,\boldsymbol{u}_k, \qquad \hat{\boldsymbol{v}}_k = \check{s}_k^{1/2}\,\boldsymbol{v}_k$$

Step 2: Determine the number of sections for the parallel structure of a nonrecursive or reduced-order recursive filter. The number of sections r_s is the number of nonzero singular values used. The r_s is determined by a trial-and-error approach to assure that design specifications are satisfied.

Step 3: Perform an LU decomposition for a nonrecursive or reduced-order recursive filter. The LU decomposition of the coefficient matrix of a nonrecursive filter can be expressed as

$$\mathbf{C}_H = \mathbf{C}_L\,\mathbf{C}_U \tag{5.13}$$

while the LU decomposition of the coefficient matrix of the numerator polynomial of a recursive filter can be expressed as

$$\mathbf{N}_R = \mathbf{N}_L\,\mathbf{N}_U \tag{5.14}$$

Since $\mathbf{C}_L$, $\mathbf{C}_U$, $\mathbf{N}_L$, and $\mathbf{N}_U$ contain a large number of zero entries [33, 38], the so-called LUD realization is more efficient. For the design of a nonrecursive filter, the transfer function for the LUD realization is expressed as

$$\hat{H}(z_1, z_2) = \hat{\mathbf{z}}_2\,\mathbf{C}_L\,\mathbf{C}_U\,\hat{\mathbf{z}}_1^T = \sum_{k=1}^{r_s} \mathbf{c}_{Lk}(z_2)\,\mathbf{c}_{Uk}(z_1) \tag{5.15}$$

where $\mathbf{c}_{Uk}(z_1)$ and $\mathbf{c}_{Lk}(z_2)$ are the kth pair of 1-D transfer functions of orders m and n, respectively. Alternatively, for the design of a recursive filter, the transfer function for the LUD realization is expressed as

$$\hat{H}^r(z_1, z_2) = \frac{\check{\mathbf{z}}_2 \mathbf{N}_L \mathbf{N}_U \check{\mathbf{z}}_1^T}{\check{\mathbf{z}}_2 \mathbf{d}_{R2} \mathbf{d}_{R1}^T \check{\mathbf{z}}_1^T} = \frac{\sum_{k=1}^{r_s} \mathbf{n}_{Lk}(z_2)\, \mathbf{n}_{Uk}(z_1)}{\mathbf{d}_2(z_2)\, \mathbf{d}_1(z_1)} \tag{5.16}$$

where $\mathbf{n}_{Uk}(z_1)$ and $\mathbf{n}_{Lk}(z_2)$ are the kth pair of 1-D transfer functions of orders r_1 and r_2, respectively; and $\mathbf{d}_1(z_1)$, $\mathbf{d}_2(z_2)$ are two 1-D transfer functions of orders r_1 and r_2, respectively.

5.4 Design Groups

The design for nonrecursive and recursive filters can be divided into four groups:

- Group A: nonrecursive filters with specified filter order
- Group B: nonrecursive filters with specified passband and stopband ripples
- Group C: recursive filters with specified filter order
- Group D: recursive filters with specified passband and stopband ripples

The flowcharts shown in Figures 5.1 to 5.4 explain how the individual software routines can be used to perform the design task for each group of design problems.

Figure 5.1: Software routines used to perform a design task from group A.

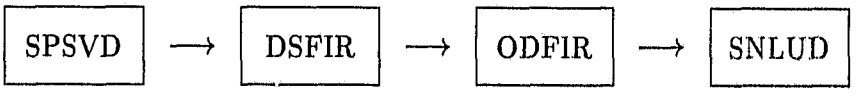

Figure 5.2: Software routines used to perform a design task from group B.

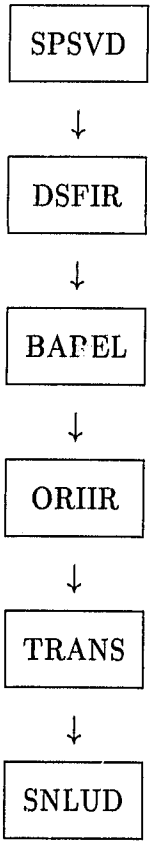

Figure 5.3: Software routines used to perform a design task from group C.

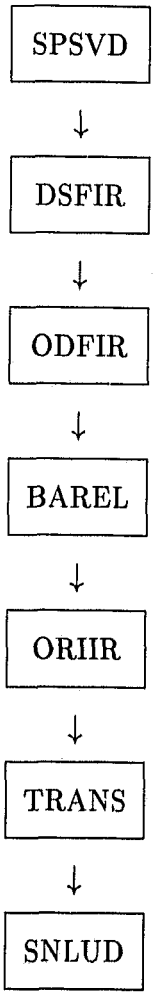

Figure 5.4: Software routines used to perform a design task from group D.

5.5 Design Examples

Four design examples are given to demonstrate the use of the design environment developed.

Example 5.1

The first example is to design a recursive, circularly symmetric, 2-D highpass digital filter with specified normalized passband and stopband edges $\omega_p = 0.6$ and $\omega_s = 0.4$, respectively. The maximum passband and stopband ripples ξ_P and ξ_S are required to be less than 0.05. This is a design that belongs to Group D. The ideal amplitude response is given by

$$\left| H^d(\omega_1, \omega_2) \right| = \begin{cases} 0 & \text{for } \sqrt{\omega_1^2 + \omega_2^2} \leq \omega_c \\ 1 & \text{otherwise} \end{cases}$$

where the normalized cutoff frequency is

$$\omega_c = \frac{\omega_p + \omega_s}{2} = 0.5$$

The design was obtained using function IIR_CS_HP_RS. The resulting digital filter is a stable, recursive, circularly symmetric, highpass filter of order (22, 25) that meets the above design requirements with maximum passband and stopband errors 0.0421 and 0.0444, respectively, and 8 parallel sections. The amplitude response of the filter designed is shown in Figure 5.5.

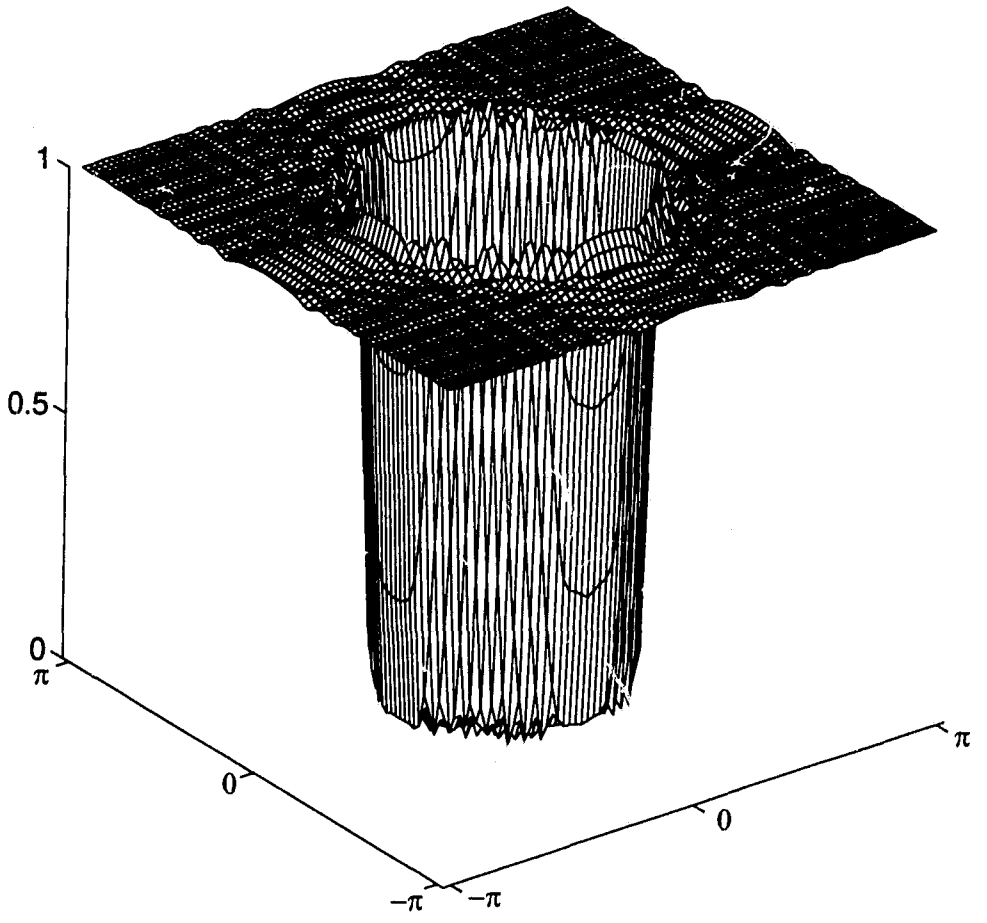

Figure 5.5: Amplitude response of a recursive circularly symmetric highpass filter designed using function IIR_CS_HP_RS with $\omega_p = 0.6$, $\omega_s = 0.4$, $\xi_P \leq 0.05$, and $\xi_S \leq 0.05$ (Example 5.1).

Example 5.2

The second example is to design a recursive, bandstop, 2-D fan filter with specified filter order of (30, 30). The design belongs to Group C. The ideal amplitude response is given by

$$\left| H^d(\omega_1, \omega_2) \right| = \begin{cases} 0 & \text{for } \omega_2 \leq 0.6\,\omega_1 - 0.02\,\pi \\ 1 & \text{for } \omega_2 \geq 0.6\,\omega_1 + 0.11\,\pi \end{cases}$$

A stable, recursive, bandstop, fan filter of order (30, 30) that meets the above design requirements with 20 parallel sections was obtained using function IIR_FN_BS_OS. The amplitude response of the filter designed is shown in Figure 5.6.

Example 5.3

The third example is to design a filter for a user in [24]. That is, the design of a recursive 2-D digital filter with specified filter order of (20, 20), whose amplitude response approximates the amplitude response shown in Figure 5.7. As can be seen from Figure 5.7, the amplitude response is non-standard. It is a *regularization* filter which can be used for the restoration of degraded images [55]. A stable, recursive, filter of order (20, 20) consisting of 2 parallel sections was obtained using function IIR_USER_OS. The amplitude response of the filter designed is shown in Figure 5.8.

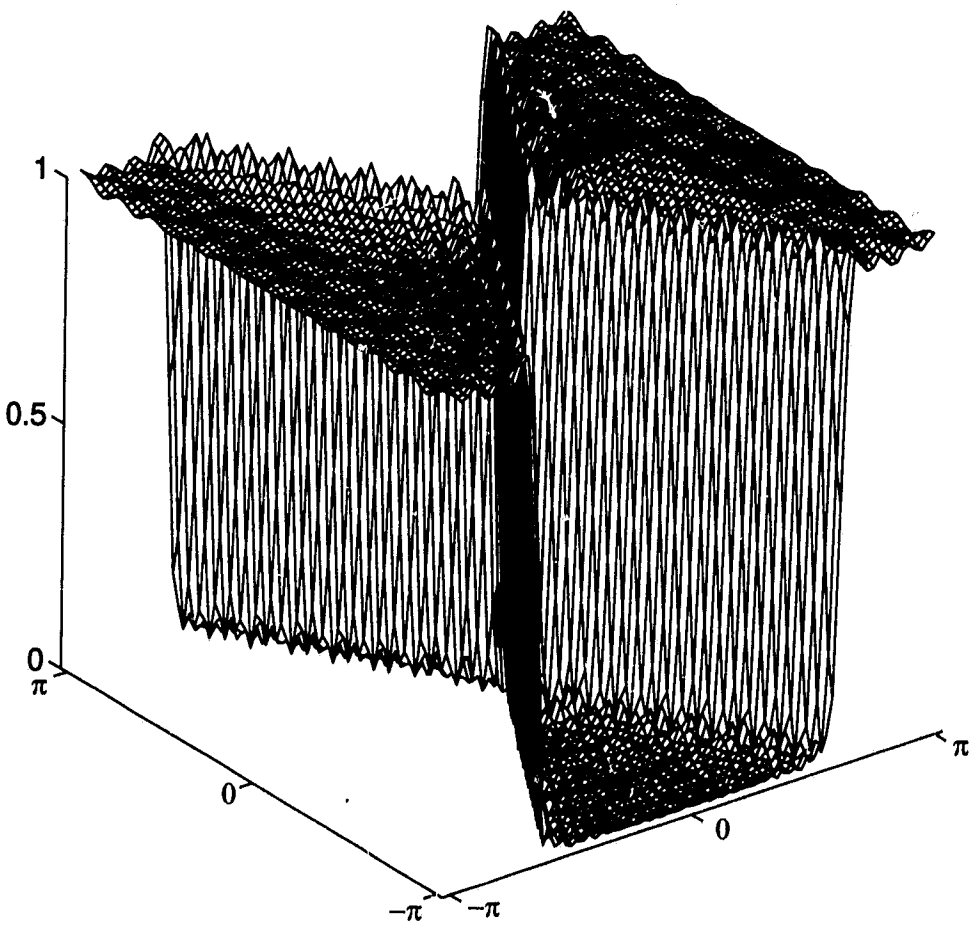

Figure 5.6: Amplitude response of a recursive bandstop fan filter designed using function IIR_FN_BS_OS with filter order of (30, 30) (Example 5.2).

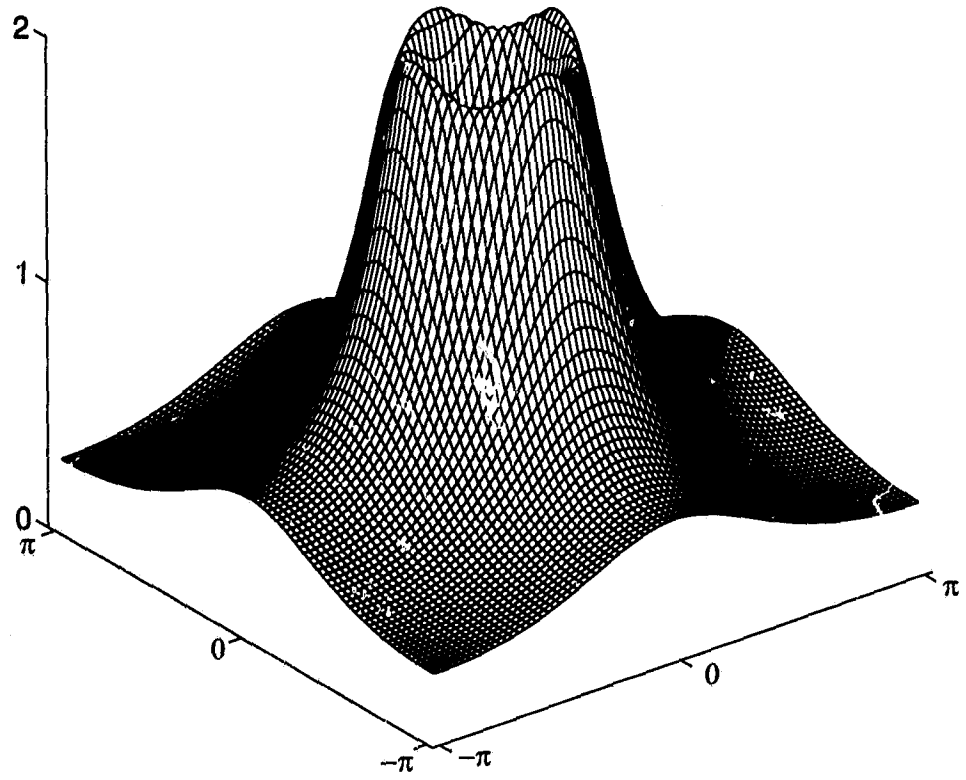

Figure 5.7: Amplitude response of the desired regularization filter (Example 5.3).

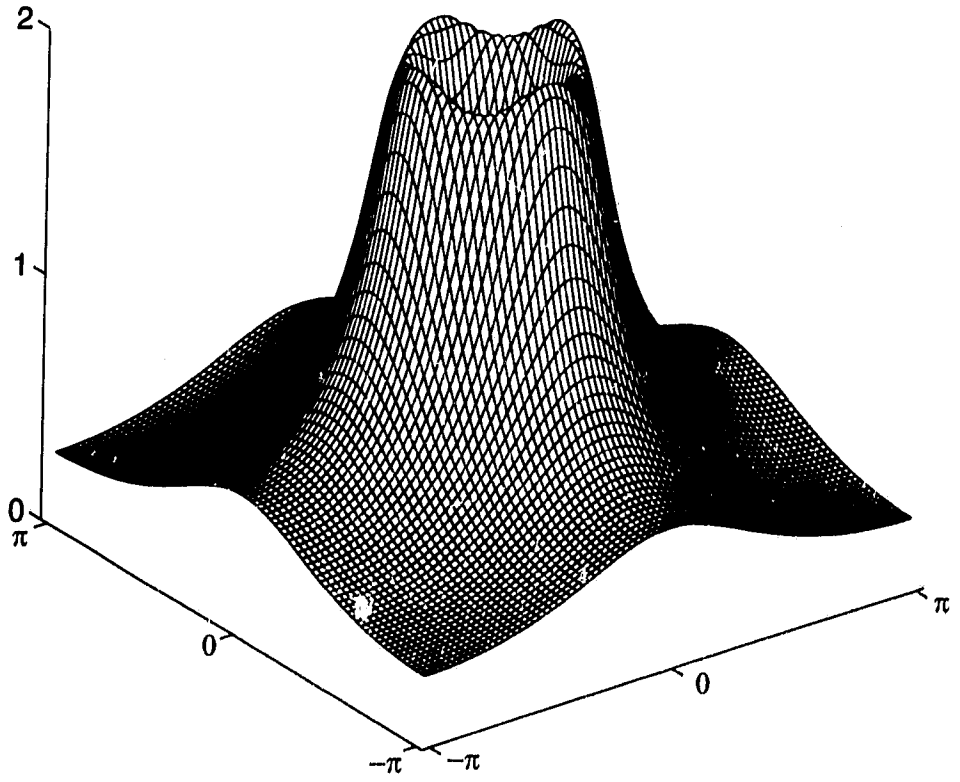

Figure 5.8: Amplitude response of a recursive regularization filter designed using function IIR_USER_OS with filter order of (20, 20) (Example 5.3).

Example 5.4

The last example is to design a nonrecursive circularly symmetric lowpass filter with specified normalized passband and stopband edges $\omega_p = 0.4$ and $\omega_s = 0.6$, respectively. The maximum passband and stopband ripples ξ_P and ξ_S are required to be less than 0.05. The design belongs to Group B. The amplitude response of the ideal circularly symmetric lowpass filter is given by

$$\left| H^d(\omega_1, \omega_2) \right| = \begin{cases} 1 & \text{for } \sqrt{\omega_1^2 + \omega_2^2} \leq \omega_c \\ 0 & \text{otherwise} \end{cases}$$

where the normalized cutoff frequency is

$$\omega_c = \frac{\omega_p + \omega_s}{2} = 0.5$$

A stable, nonrecursive, circularly symmetric, lowpass digital filter of order (35, 30) which meets the design requirements was designed using the user-interface software. The interface between the software and the user are illustrated in Table 5.4, where questions or comments provided by the software are denoted by Q's and the answers given by the user are denoted by A's. The amplitude response of the filter designed is shown in Figure 5.9.

Table 5.4: Questions and answers using user-interface design software

Questions	Answers
SVD Method for the Design of 2-D Digital Filters	
Q: Are you ready ? (Y/N)	A: Y
Types of 2-D Digital Filters: 1. Circularly Symmetric Lowpass Filter 2. Circularly Symmetric Highpass Filter 3. Circularly Symmetric Bandpass Filter 4. Circularly Symmetric Bandstop Filter 5. Bandpass Fan Filter 6. Bandstop Fan Filter 7. User-defined Filter 8. Quit	
Q: Choose the type of your filter:	A: 1
Types of Design Criteria: 1. Prescribed Maximum Passband and Stopband Ripples 2. Prescribed Filter Order	
Q: Choose one type of design criteria:	A: 1
Q: Enter normalized passband edge $(0.0 - 1.0)$:	A: 0.4
Q: Enter normalized stopband edge $(0.0 - 1.0)$:	A: 0.6
Q: Enter maximum passband ripple $(0.01 - 0.10)$:	A: 0.05
Q: Enter maximum stopband ripple $(0.01 - 0.10)$:	A: 0.05
Q: Enter sample points of amplitude response $(21 - 91)$:	A: 41
Q: Do you want to see the desired amplitude response $\mathbf{H}^d$?	A: Y
Q: Do you want to see the plot of singular values of $\mathbf{H}^d$?	A: Y

Table 5.4: Continuation

Questions	Answers
The Design Methods for 1-D Sub-Filters: 1. Kaiser Window Method 2. Hamming Window Method	
Q: Choose a method for 1-D filter designs:	*A*: 1
Q: Choose the order of 1-D filters in z_1 domain:	*A*: 20
Q: Choose the order of 1-D filters in z_2 domain:	*A*: 20
Q: Do you want to see amplitude response of 1-D filters ?	*A*: Y
Q: Do you want to see impulse response of 2-D filter ($\mathbf{C}_H$) ?	*A*: Y
Q: Do you want to see amplitude response of 2-D filter ?	*A*: Y
The maximum errors in passband and stopband are: 0.0671 0.0569	
Q: Are you satisfied with the design errors ?	*A*: N
Q: Re-enter the order of 1-D filters in z_1 domain:	*A*: 35
Q: Re-enter the order of 1-D filters in z_2 domain:	*A*: 30
Q: Do you want to see amplitude response of 2-D filter ?	*A*: Y
The maximum errors in passband and stopband are: 0.0480 0.0412	
Q: Are you satisfied with the design errors ?	*A*: Y
Types of Digital Filters to Be Realized: 1. Nonrecursive Filter 2. Recursive Filter	
Q: Select the realization type of your filter:	*A*: 1

Table 5.4: Continuation

Questions		Answers
Q:	Do you want to see the plot of singular values (sv) of $\mathbf{C}_{\text{II}}$?	*A*: Y
Q:	Select min(sv)/max(sv) ratio of $\mathbf{C}_{\text{II}}$ $(0.0001 - 0.05)$:	*A*: 0.001
	Impulse responses after section reduction: $\mathbf{C}_{\text{II}}^{s}$ Impulse responses before section reduction: $\mathbf{C}_{\text{II}}$	
Q:	Do you want to see the comparison of $\mathbf{C}_{\text{II}}^{s}$ and $\mathbf{C}_{\text{II}}$?	*A*: Y
	Amplitude responses after section reduction: $\mathbf{H}^{s}$ Amplitude responses before section reduction: $\mathbf{H}$	
Q:	Do you want to see the comparison of $\mathbf{H}^{s}$ and $\mathbf{H}$?	*A*: Y
Q:	Do you want to see the corresponding contours ?	*A*: Y
	The maximum errors in passband and stopband are: 0.0481 0.0412	
Q:	Are you satisfied with the design errors ?	*A*: Y
	Summary of the Design Results The order of the 2-D filter designed is: 35 30 The maximum errors in passband and stopband are: 0.0481 0.0412 The number of sections in parallel structure is: 8 The coefficients of 1-D sub-filters are contained in $\mathbf{C}_{\text{L}}$ and $\mathbf{C}_{\text{U}}$ The LUD realization is formed by $\mathbf{C}_{\text{L}}$ and $\mathbf{C}_{\text{U}}$	
Q:	Do you want to see $\mathbf{C}_{\text{L}}$ and $\mathbf{C}_{\text{U}}$? Y/N	*A*: N
	Thank You for Using the SVD Design Environment.	

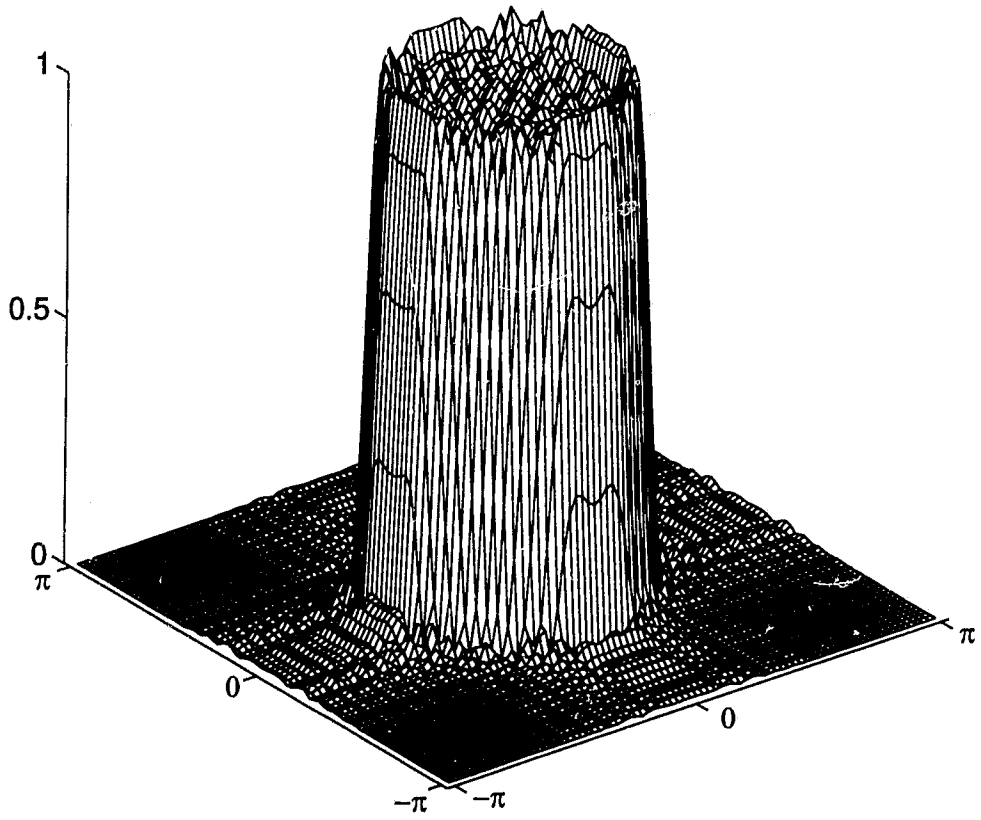

Figure 5.9: Amplitude response of a nonrecursive circularly symmetric low-pass filter designed using the user-interface software with $\omega_p = 0.4$, $\omega_s = 0.6$, $\xi_P \leq 0.05$, and $\xi_S \leq 0.05$ (Example 5.4).

5.6 Conclusion

A design environment for various recursive and nonrecursive 2-D digital filters has been developed. Using this environment, recursive and nonrecursive 2-D filters satisfying prescribed specifications such as maximum passband and stopband ripples or filter order, etc., can be easily designed. The indivi -dual routines of the design environment are flexible, efficient, and robust, and can be used as building blocks by designers to develop user-friendly software for other digital signal processing applications. The design environment has been used to design filters for image restoration [24], and is expected to be useful to engineers and researchers who are involved with other applications for the processing of 2-D data.

Chapter 6

Conclusions and Future Research

This chapter summarizes the conclusions of the dissertation and describes research that could be undertaken in the future.

6.1 Conclusions

In this dissertation, a sufficient condition for the existence of controllability and observability quasi-gramians has been presented. A new efficient algorithm for the computation of the quasi-gramians (Algorithm 2.2) has also been proposed. The algorithm involves solving two 1-D Lyapunov equations at each iteration. The resulting quasi-balanced model-reduction method is the most computationally efficient model-reduction method for 2-D discrete systems.

Two new reliable algorithms for the computation of structured gramians (Algorithms 2.3 and 2.4) have been developed. Both algorithms amount to solving unconstrained optimization problems. Algorithm 2.4 takes into account both system stability and approximation error. To the best knowledge

of the author, the structurally balanced model-reduction method obtained using Algorithm 2.4 is the only balanced model-reduction method that leads to a stable reduced-order system and a small approximation error.

The structurally balanced realization and model-reduction methods have been extended to the case of 2-D discrete systems with input and output weights, and resulted in new weighted structurally balanced realization and model-reduction methods for 2-D discrete systems. The development of the weighted structurally balanced realization and model-reduction methods can be divided into three crucial stages. The first stage is the introduction of two new auxiliary transfer-function matrices called weighted-input-to-state and state-to-weighted-output transfer-function matrices. The second stage is the definition of the 2-D weighted (structured) controllability and observability gramians. The third stage is the formulation of an unconstrained minimization problem to obtain the weighted structurally balanced model-reduction method that takes into account both system stability and approximation error.

It has been shown that if an original 2-D discrete system, and the input and output weights are all Q-stable, the resulting weighted reduced-order system obtained using the weighted balanced model reduction is also Q-stable regardless of the order of the reduced-order system. The design of a lowpass filter of order (4, 8) has been used to demonstrate that if appropriate input and output weights are used, the approximation accuracy of the weighted reduced-order filter in specified frequency ranges can be improved

when compared to the corresponding unweighted reduced-order filter.

As an integral part of a filter design process, new algorithms for the determination of the transfer-function matrix of a 2-D discrete system from the Roesser state-space model have been proposed. The development of the algorithms involves three distinct steps. First, the determination of the transfer-function matrix of a MIMO 2-D discrete system is reformulated as the evaluation of the transfer functions of a series of SISO 2-D systems. Second, the transfer function of a SISO 2-D system is reformulated in terms of the 1-D characteristic polynomials of several matrices involved that depend on one complex variable. Third, given the values of the complex variable over a set of distinct points on the unit circle, the coefficients of the 1-D polynomials are determined by solving a system of linear equations whose coefficient matrices are unitary Vandermonde matrices. The algorithms are found to be computationally more efficient when compared to that of the existing algorithms described in [7, 50]. The new algorithms have been extended to the Fornasini-Marchesini state-space model.

Finally, a general purpose design environment for various recursive and nonrecursive 2-D digital filters has been developed. The state-of-the-art design environment, which is developed in the MATLAB environment [45], consists of two modules. The first module is a user-interface software module which assists a novice to design a 2-D digital filter. The second module consists of a library of functions to assist an expert to design standard or nonstandard 2-D digital filters with specific design requirements. The soft-

ware functions involved in the design of recursive filters integrate the SVD design method with the balanced model reduction algorithms. As an integral part of the design environment, software routines for the determination of 2-D transfer-function matrix from state-space models are included.

Using the design environment, recursive and nonrecursive 2-D filters satisfying prescribed specifications such as maximum passband and stopband ripples, or filter order, or linear phase response, can be easily designed. The individual routines of the design environment are flexible, efficient, and robust, and can be used as building blocks by designers to develop user-friendly software for other digital signal processing applications. The design environment has been used to design filters for image restoration [24], and is expected to prove useful to engineers and researchers who are involved with other applications for the processing of 2-D data.

6.2 Suggested Future Research

- In this dissertation, weighted structurally balanced model-reduction methods for 2-D discrete systems with input and output weights are based on solving unconstrained optimization problems. The objective functions of the unconstrained optimization problems stated in (2.32) and (3.25) are not convex. It is known that linear matrix inequality techniques can be used to effectively solve *convex* problems that involve linear matrix inequalities. If the objective functions in (2.32) and

(3.25) can be reformulated as convex problems such that linear matrix inequality techniques [8] can be used, the computational efficiency of the weighted and unweighted structurally balanced model-reduction methods proposed are expected to be improved. A great deal of research will be needed for the reformulation.

- In the weighted structurally balanced model-reduction methods for 1-D and 2-D discrete systems with input and output weights, the relation between the discarded weighted structured Hankel singular values and the approximation error of a weighted reduced-order system needs to be investigated further.

- In Chapter 4, a comparison of the Algorithm 4.1 (Algorithm 4.2) with Algorithms 4.3 (Algorithm 4.4) has shown the former to be more efficient by a factor of about ten. How to improve the computational efficiency of the algorithms for the determination of transfer function matrices from the Fornasini-Marchesini state-space model is still an open question.

- The design environment developed in this research is based on the SVD design method, the algorithms for PBMR, and the algorithms for the determination of transfer-function matrix from the Roesser state-space model. The environment may be extended by including conventional design methods, such as the window, the McClellan transformation, and optimization design methods. The environment can also be ex-

tended to the case where the requirements for the maximum ripples are specified only in the passband and stopband (not in transition band) by using the proposed WSBMR.

- The design environment provided in this dissertation has been used in image restoration. The application of the design environment to other image processing or DSP applications needs to be explored further.

References

[1] Al-Saggaf, U. M., and G. F. Franklin, "Model reduction via balanced realizations: an extension and frequency weighted techniques", *IEEE Trans. Automat. Contr.*, vol. 33, pp. 687-692, July 1988.

[2] Anderson, B. D. O., P. Agathoklis, E. I. Jury, and M. Mansour, "Stability and the matrix Lyapunov equations for discrete two-dimensional systems", *IEEE Trans. Circuits and Systems*, vol. 33, pp. 261-267, March 1986.

[3] Antoniou, A., M. Ahmadi, and C. Charalambous, "Design for factorable lowpass 2-D filters satisfying prescribed specifications", *IEE Proc.*, vol. 128, pp. 53-60, April 1981.

[4] Antoniou, A., and W.-S. Lu, "Design of two-dimensional digital filters by using the singular-value decomposition", *IEEE Trans. Circuits and Systems*, vol. 34, pp. 1191-1198, October 1987.

[5] Antoniou, A., *Digital Filters: Analysis, Design, and Applications*, 2nd ed., New York: McGraw-Hill, 1993.

[6] Antoniou, G. E., G. O. A. Glentis, D. A. Karras, and S. J. Varoufakis, "Transfer function determination of singular systems using the DTF", *IEEE Trans. Circuits and Systems*, vol. 36, pp. 1140-1142, 1989.

[7] Antoniou, G. E., S. E. Mentzelopoulou, and G. O. A. Glentis, "Determination of transfer function of two-dimensional generalised systems using the discrete Fourier transform", *IEE Proc.*, vol. 138, pp. 327-330, July 1991.

[8] Boyd, S., L. Ghaoui, E. Feron, and V. B. Rishnan, *Linear Matrix Inequalities in System and Control Theory*, Philadelphia: SIAM (Society for Industrial and Applied Mathematics), 1994.

[9] Bose, N. K., *Multidimensional Systems: Theory and Applications*, New York: IEEE Press, 1979.

[10] Bose, N. K., *Applied Multidimensional Systems Theory*, New York: Van Nostrand Reinhold, 1982.

[11] Dudgeon, D. E., and Mersereau, R. M., *Multidimensional Digital Signal Processing*, New Jersey: Prentice Hall Inc., 1984.

[12] El-Agizi, N. G., and M. M. Fahmy, "Two-dimensional digital filters with no overflow oscillations", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 27, pp. 465-469, June 1979.

[13] Enns, D. F., "Model reduction with balanced realizations: an error bound and a frequency weighted generalization", *Proc. of the 23rd IEEE Conf. on Decision and Control*, Las Vegas, USA, pp. 127-132, December 1984.

[14] Fletcher, R., *Practical Methods of Optimization*, 2*nd* ed., New York: John Wiley, 1987.

[15] Fornasini, E., and G. Marchesini, "Doubly-indexed dynamical systems: state-space models and structural properties", *Mathematical System Theory*, vol. 12, pp. 59-72, 1978.

[16] Friedland, B., *Control System Design — An Introduction to State-Space Methods*, New York: McGraw Hill, 1986.

[17] Golub, G. H., and C. F. Van Loan, *Matrix Computation*, 2*nd* ed., Baltimore: The Johns Hopkins University Press, 1989.

[18] Hinamoto, T., T. Hamanaka, and S. Maekawa, "A generalized study on the synthesis of 2-D state-space digital filters with minimum roundoff noise", *IEEE Trans. Circuits and Systems*, vol. 35, pp. 1037-1042, August 1988.

[19] Huang, T. S., "Two-dimensional windows", *IEEE Trans. Audio Electroacoust.*, vol. 20, pp. 88-89, March 1972.

[20] Kailath, T., *Linear Systems*, New Jersey: Prentice Hall Inc., 1980.

[21] Kawamata M., and T. Higuchi, "Synthesis of 2-D separable denominator digital filters with minimum roundoff noise and no overflow oscillations", *IEEE Trans. Circuits and Systems*, vol. 33, pp. 365-372, April 1986.

[22] Koo, C. S., and C. T. Chen, "Fadeeva's algorithm for spatial dynamical equations", *IEE Proc.*, vol. 65, pp. 975-976, June 1977.

[23] Laub, A., J., "On computing balancing transformations", *Proc. of the Joint Automatic Control Conference*, FA 8-E, San Francisco, USA, 1980.

[24] Leung, C.-M., "Some Filtering techniques for digital image processing", Ph.D. Dissertation, Dept. of Electrical and Computer Engineering, University of Victoria, November 1994.

[25] Lim, J. S., *Two-Dimensional Signal and Image Processing*, New Jersey: Prentice Hall Inc., 1990.

[26] Lin, T., M. Kawamate, and T. Higuchi, "A unified study on the round-off noise in 2-D state-space digital filters", *IEEE Trans. Circuits and Systems*, vol. 33, pp. 724-730, July 1986.

[27] Lin, T., M. Kawamate, and T. Higuchi, "Design of 2-D digital filters with an arbitrary response and no overflow oscillations based on a new stability condition", *IEEE Trans. Circuits and Systems*, vol. 34, pp. 113-127, February 1987.

[28] Lodge, J. H., and M. M. Fahmy, "Stability and overflow oscillations in 2-D state-space digital filters", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 29, pp. 1161-1171, December 1981.

[29] Lu, W.-S., and E. B. Lee, "Stability analysis for two-dimensional analysis", *IEEE Trans. Circuits and Systems*, vol. 30, pp. 455-461, July 1983.

[30] Lu, W.-S., and E. B. Lee, "Stability analysis for two-dimensional systems via a Lyapunov approach", *IEEE Trans. Circuits and Systems*, vol. 32, pp. 61-68, January 1985.

[31] Lu, W.-S., and A. Antoniou, "Synthesis of 2-D state-space fixed-point digital-filter structures with minimum roundoff noise", *IEEE Trans. Circuits and Systems*, vol. 33, pp. 965-973, October 1986.

[32] Lu, W.-S., E. B. Lee, and Q.-T. Zhang, "Balanced approximation of two-dimensional and delay-differential systems", *Int. J. Contr.*, vol. 46, pp. 2199-2218, December 1987.

[33] Lu, W.-S., H.-P. Wang, and A. Antoniou, "Design of two-dimensional FIR digital filters by using the singular-value decomposition", *IEEE Trans. Circuits and Systems*, vol. 37, pp. 35-46, January 1990.

[34] Lu, W.-S., and A. Antoniou, "On the synthesis of 2-D state-space digital filters with minimum roundoff noise", *IEEE Trans. Circuits and Systems*, vol. 37, pp. 1424-1425, November 1990.

[35] Lu, W.-S., W.-P. Wang, and A. Antoniou, "Kaiser window MATLAB software for 1-D FIR filter design", *Research Report, Dept. of Electrical and Computer Engineering, University of Victoria*, 1990.

[36] Lu, W.-S., H.-P. Wang, and A. Antoniou, "On the evaluation of the controllability and observability grammians of 2-D digital filters", *Proc. of the IEEE International Symposium on Circuits and Systems*, Singapore, vol. 1, pp. 602-605, June 1991.

[37] Lu, W.-S., and A. Antoniou, "Design of 2-D digital filters with arbitrary amplitude and phase responses by using the singular-value decomposition", *Proc. of the IEEE International Symposium on Circuits and Systems*, Singapore, vol. 1, pp. 618-621, June 1991.

[38] Lu, W.-S., H.-P. Wang, and A. Antoniou, "Design of two-dimensional digital filters using the singular-value decomposition and balanced approximation method", *IEEE Trans. Signal Processing*, vol. 39, pp. 2253-2262, October 1991.

[39] Lu, W.-S., H.-P. Wang, and A. Antoniou, "An efficient method for the evaluation of the controllability and observability grammians of 2-D digital filters and systems", *IEEE Trans. Circuits and Systems*, vol. 39, pp. 695-704, October 1992.

[40] Lu, W.-S., and A. Antoniou, *Two-Dimensional Digital Filters*, New York: Marcel Dekker, 1992.

[41] Lu, W.-S., H. Luo, and A. Antoniou, "New algorithms for the derivation of the transfer function matrix of two-dimensional digital filters", *Proc.*

of the IEEE International Symposium on Circuits and Systems, Chicago, Illinois, vol. 1, pp. 591-594, May 1993.

[42] Luo, H., W.-S. Lu, and A. Antoniou, "An advanced software tool for the design of two-dimensional digital filters", *Proc. of the IEEE International Symposium on Circuits and Systems*, San Diego, California, vol. 4, pp. 1420-1423, May 1992.

[43] Luo, H., W.-S. Lu, and A. Antoniou, "Numerical solution of the two-dimensional Lyapunov equations and application in order reduction of recursive digital filters", *Proc. of the IEEE Pacific Rim Conf. on Communications, Computers and Signal Processing*, Victoria, Canada, vol. 1, pp. 232-235, May 1993.

[44] Maria, G. A., and M. M. Fahmy, "An L_p design technique for two-dimensional digital recursive filters", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 22, pp. 15-21, February 1974.

[45] *MATLAB: Reference Guide*, The Math Works Inc., 1992.

[46] McClellan, J. H., "The design of two-dimensional filters by transformations", *Proc. of the 7th Annual Princeton Conf. Information Sciences and Systems*, pp. 247-251, 1973.

[47] Mecklenbräuker, W. F. G., and R. M. Mersereau, "McClellan transformation for two-dimensional digital filtering: II — Implementation", *IEEE Trans. Circuits and Systems*, vol. 23, pp. 414-422, July 1976.

[48] Mersereau, R. M., W. F. G. Mecklenbräuker, and T. F. Jr. Quatieri, "McClellan transformations for two-dimensional digital filtering: I — Design", *IEEE Trans. Circuits and Systems*, vol. 23, pp. 405-413, July 1976.

[49] Mertzios, B. G., and P. N. Paraskevopoulos, "Transfer function matrix of 2-D systems", *IEEE Trans. Automat. Contr.*, vol. 26, pp. 722-724, June 1981.

[50] Mertzios, B. G., "An algorithm for the computation of the transfer function matrix of two-dimensional systems", *J. Franklin Inst.*, vol. 321, pp. 73-80, February 1986.

[51] Mitra, S. K., and M. P. Ekstrom, *Two-Dimensional Digital Signal Processing*, Dowden, Hutchinson and Ross, Stroudsburg, 1978.

[52] Moore, B. C., "Principal component analysis in linear systems: controllability, observability and model reduction", *IEEE Trans. Automat. Contr.*, vol. 26, pp. 17-32, February 1981.

[53] Premaratne, K., E. I. Jury, and M. Mansour, "An algorithm for model reduction of 2-D discrete time systems", *IEEE Trans. Circuits and Systems*, vol. 37, pp. 1116-1132, September 1990.

[54] Roesser, R. P., "A discrete state-space model for linear imaging processing", *IEEE Trans. Automat. Contr.*, vol. 20, pp. 1-10, February 1975.

[55] Unser, M., A. Aldroubi, and M. Eden, "Recursive regularization filters: design, properties, and applications", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, pp. 272-277, March 1991.

[56] Wang, W., K. Glover, J. C. Doyle, and C. Beck, "Model reduction of LFT systems", *Proc. of the 30th IEEE Conf. on Decision and Control*, Brighton, England, pp. 1233-1238, December 1991.

[57] Wang, H.-P., "Design of two-dimensional digital filters using singular-value decomposition", Ph.D. Dissertation, Dept. of Electrical and Computer Engineering, University of Victoria, November 1991.

[58] Woods, J. W., J. H. Lee, and J. Paul, "Two-dimensional IIR filter design with magnitude and phase error criteria", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 31, pp. 886-894, August 1983.

[59] Yeung, K. S., and F. Kumbi, "Symbolic matrix inversion with application to electronic circuits", *IEEE Trans. Circuits and Systems*, vol. 35, pp. 235-238, February 1988.

[60] Zadeh, L. A., and C. A. Desoer, *Linear System Theory*, New York: McGraw-Hill, 1963.

[61] Zhou, K., Y. Zheng, and T. Lu, "Stability and error bounds for discrete time frequency weighted balanced truncation", *Proc. of the 32nd IEEE Conf. on Decision and Control*, San Antonio, USA, December 1993.

[62] Zhou, K., J. L. Aravena, G. Gu, and D. Xiong, "Two-dimensional system model reduction by quasi-balanced truncation and singular perturbation", *IEEE Trans. Circuits and Systems*, vol. 41, pp. 593-602, September 1994.

Vita

Hong Luo

Personal Data

Place of Birth: Kunming, China

Educational Institutions Attended:

Shanghai Jiao Tong University, China	1979 to 1986
University of Victoria, Canada	1989 to 1995

Degrees Awarded:

M.Sc. (Eng.)	Shanghai Jiao Tong University	1986
B.Sc. (Eng.)	Shanghai Jiao Tong University	1983

Partial Copyright License

Title of Dissertation:

Analysis and Design of Two-Dimensional Digital Filters

Author:

Hong Luo

Aug. 24, 1995

Date