New Algorithms for the Derivation of the Transfer-Function Matrices of 2-D State-Space Discrete Systems

Hong Luo, *Member, IEEE*, Wu-Sheng Lu, *Senior Member, IEEE*, and Andreas Antoniou, *Fellow, IEEE*

***Abstract*— New algorithms for the derivation of the transfer-function matrices of two-dimensional (2-D) discrete systems from the Roesser and Fornasini–Marchesini state-space models are presented. Two key steps in developing the algorithms are as follows. First, the transfer-function matrix is reformulated in terms of the characteristic polynomials of the matrices involved. Second, an efficient algorithm for the determination of 1-D polynomial coefficients is developed and is, in turn, used to determine the coefficient matrices of the 2-D transfer-function matrix. The proposed algorithms are computationally efficient and reliable. The efficiency of the algorithms is illustrated by comparing the proposed method with two existing methods through examples.**

***Index Terms*— 2-D transfer-function matrix, 2-D discrete systems.**

I. Introduction

STATE-SPACE two-dimensional (2-D) discrete systems have been studied quite extensively during the past decade, and several useful methods for their analysis and design have been established [1]. These include methods for stability analysis [2]–[6], analysis of finite-wordlength effects [7], [8], design [9], [10], model reduction [11]–[13], and relevant computation issues [14], [15]. Since many of the available analysis and design methods are applicable only to the 2-D transfer-function matrix, it is often necessary to derive the transfer-function matrix from a state-space description of the system.

One of the commonly used state-space models for 2-D discrete systems is the Roesser model [16]. Several algorithms for the derivation of the 2-D transfer-function matrix from the Roesser state-space model have been proposed [17]–[22]. Those in [17]–[19] are basically extensions of the well-known Fadeeva algorithm [23] to the 2-D case while the algorithms in [20]–[22] are based on the discrete Fourier transform (DFT). Another popular state-space representation for 2-D discrete systems is the Fornasini–Marchesini model [24]. To date, no efficient algorithms for the derivation of the 2-D transfer-function matrix from the Fornasini–Marchesini state-space representation have been reported.

In Sections II and III of this paper, new algorithms for the derivation of the 2-D transfer-function matrix from the Roesser and Fornasini–Marchesini state-space models are presented. Two key steps in developing the new algorithms are as follows. First, the transfer-function matrix is reformulated in terms of the characteristic polynomials of several matrices that depend on one complex variable. Second, algorithms are proposed that identify the coefficients of a 1-D polynomial of order n when its values at $(n+1)$ points on the unit circle are known. Our algorithms entail solving a system of linear equations whose coefficient matrix is an unitary Vandermonde matrix. In Section IV, examples are given to illustrate the efficiency of the algorithms proposed and to compare them with the existing algorithms.

II. Derivation of the Transfer-Function Matrix from the Roesser State-Space Model

In this section, two algorithms for the derivation of the transfer-function matrix of a linear, shift-invariant, discrete, multivariable 2-D system from its Roesser state-space description are developed.

Consider the Roesser state-space model of a single-input, single-output (SISO) 2-D discrete system [16]

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

$$\begin{aligned} y(k,\,l) &= [\mathbf{c}_1 \quad \mathbf{c}_2]\begin{bmatrix}\mathbf{x}^h(k,\,l)\\ \mathbf{x}^v(k,\,l)\end{bmatrix} + du(k,\,l)\\ &= \mathbf{c}\mathbf{x} + du\end{aligned}\tag{1b}$$

where $\mathbf{x}^h \in \Re^m$, $\mathbf{x}^v \in \Re^n$ are the horizontal and vertical state vectors, respectively, and u and y are the input and output, respectively. If we define

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

where $\oplus$ denotes the direct sum, then the transfer-function matrix of the system is given by

$$\begin{aligned} H(z_1,\,z_2) &= [\mathbf{c}_1 \quad \mathbf{c}_2]\begin{bmatrix} z_1\mathbf{I}-\mathbf{A}_1 & -\mathbf{A}_2\\ -\mathbf{A}_3 & z_2\mathbf{I}-\mathbf{A}_4\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{bmatrix} + d\\ &= \mathbf{c}[\mathbf{I}(z_1,\,z_2)-\mathbf{A}]^{-1}\mathbf{b} + d\end{aligned}\tag{2}$$

Manuscript received October 11, 1994; revised May 10, 1996. This work was supported by Micronet, Networks of Centres of Excellence Program and to the Natural Sciences and Research Council of Canada. This paper was recommended by Associate Editor H. Reddy.

H. Luo is with the Voice Group, Micom Communications Corp., Simi Valley, CA 93063 USA.

W.-S. Lu and A. Antoniou are with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada V8W 3P6.

Publisher Item Identifier S 1057-7122(97)01500-6.

$H(z_1, z_2)$ in (2) can be written as

$$H(z_1, z_2) = \frac{\sum_{k=0}^{m} q_k(z_2) z_1^k}{\sum_{k=0}^{m} p_k(z_2) z_1^k} \tag{3}$$

where k is an integer, $p_k(z_2)$ and $q_k(z_2)$ are polynomials in z_2 of order not greater than n, and

$$\sum_{k=0}^{m} p_k(z_2) z_1^k = \det \begin{bmatrix} z_1\mathbf{I} - \mathbf{A}_1 & -\mathbf{A}_2 \\ -\mathbf{A}_3 & z_2\mathbf{I} - \mathbf{A}_4 \end{bmatrix}.$$

It follows that

$$p_m(z_2) = \det(z_2\mathbf{I} - \mathbf{A}_4) = P(z_2, \mathbf{A}_4) \tag{4}$$

where $P(z_2, \mathbf{A}_4)$ denotes the characteristic polynomial of $\mathbf{A}_4$ in variable z_2. Further, from (2) and the formula of matrix inversion [25], transfer function $H(z_1, z_2)$ can be expressed as

$$H(z_1, z_2) = l(z_2) + \mathbf{g}(z_2)[z_1\mathbf{I} - \mathbf{E}(z_2)]^{-1}\mathbf{f}(z_2) \tag{5}$$

where

$$\begin{aligned} \mathbf{E}(z_2) &= \mathbf{A}_1 + \mathbf{A}_2(z_2\mathbf{I} - \mathbf{A}_4)^{-1}\mathbf{A}_3 \\ \mathbf{f}(z_2) &= \mathbf{b}_1 + \mathbf{A}_2(z_2\mathbf{I} - \mathbf{A}_4)^{-1}\mathbf{b}_2 \\ \mathbf{g}(z_2) &= \mathbf{c}_1 + \mathbf{c}_2(z_2\mathbf{I} - \mathbf{A}_4)^{-1}\mathbf{A}_3 \\ l(z_2) &= d + \mathbf{c}_2(z_2\mathbf{I} - \mathbf{A}_4)^{-1}\mathbf{b}_2. \end{aligned}$$

Note that $(z_2\mathbf{I} - \mathbf{A}_4)^{-1}$ is a common term in $\mathbf{E}(z_2)$, $\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $l(z_2)$; hence, the above equations can be expressed as

$$\begin{bmatrix} \mathbf{E}(z_2) & \mathbf{f}(z_2) \\ \mathbf{g}(z_2) & l(z_2) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{b}_1 \\ \mathbf{c}_1 & d \end{bmatrix} + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{c}_2 \end{bmatrix} \cdot (z_2\mathbf{I} - \mathbf{A}_4)^{-1}[\mathbf{A}_3 \quad \mathbf{b}_2] \tag{6}$$

By using a well-known formula for the transfer function of a 1-D SISO state-space model (see [25, Appendix A.13]), (5) can be rewritten as

$$\begin{aligned} H(z_1, z_2) &= \frac{\det[z_1\mathbf{I} - \mathbf{E}(z_2) + \mathbf{f}(z_2)\mathbf{g}(z_2)]}{\det[z_1\mathbf{I} - \mathbf{E}(z_2)]} + l(z_2) - 1 \\ &= \frac{P[z_1, \mathbf{E}(z_2) - \mathbf{f}(z_2)\mathbf{g}(z_2)]}{P[z_1, \mathbf{E}(z_2)]} + l(z_2) - 1 \end{aligned} \tag{7}$$

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. Note that the denominator in (7) is a monic polynomial in z_1 but the denominator in (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) leads to

$$\begin{aligned} \sum_{k=0}^{m} q_k(z_2) z_1^k = P(z_2, \mathbf{A}_4)\{&P[z_1, \mathbf{E}(z_2) - \mathbf{f}(z_2)\mathbf{g}(z_2)] \\ &+ [l(z_2) - 1]P[z_1, \mathbf{E}(z_2)]\} \end{aligned} \tag{8a}$$

$$\sum_{k=0}^{m} p_k(z_2) z_1^k = P(z_2, \mathbf{A}_4)P[z_1, \mathbf{E}(z_2)] \tag{8b}$$

A. Algorithm for a SISO Roesser Model

The algorithm for a SISO Roesser model is derived using (8a) and (8b), and an efficient method for the determination of a 1-D polynomial as described below.

1) 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 $\alpha_n, \cdots, \alpha_1, \alpha_0$. Also let $\{z_2(l), 0 \le l \le n\}$ be $(n+1)$ points that are uniformly distributed on the unit circle of the complex z_2 plane, i.e.,

$$z_2(l) = e^{j2\pi l/(n+1)}, \qquad 0 \le l \le n \tag{9}$$

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

$$\boldsymbol{\alpha} = \mathbf{V}^{-1}(\mathbf{z}_2)\mathbf{q} \tag{10}$$

where $\boldsymbol{\alpha} = [\alpha_n \cdots \alpha_1\, \alpha_0]^T$, $\mathbf{q} = [p_0\, p_1 \cdots p_n]^T$, and $\mathbf{V}(\mathbf{z}_2)$ is the $(n+1) \times (n+1)$ Vandermonde matrix whose second to last column is

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

that is

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

Since $z_2(l)$ $(0 \le l \le n)$ are distinct, $\mathbf{V}(\mathbf{z}_2)$ is always nonsingular. More important, it follows from (9) that

$$\mathbf{V}^H(\mathbf{z}_2)\mathbf{V}(\mathbf{z}_2) = (n+1)\mathbf{I} \tag{11}$$

where $\mathbf{V}^H(\mathbf{z}_2)$ denotes the complex-conjugate transpose of $\mathbf{V}(\mathbf{z}_2)$. Therefore, $\mathbf{V}(\mathbf{z}_2)/\sqrt{n+1}$ is a unitary matrix and (10) can be written as

$$\boldsymbol{\alpha} = \frac{1}{n+1}\mathbf{V}^H(\mathbf{z}_2)\mathbf{q} \tag{12}$$

Equation (12) provides an efficient formula for the determination of 1-D polynomial $p(z_2)$.

2) Determination of the Coefficients of $p_k(z_2)$ and $q_k(z_2)$: Throughout this subsection it is assumed that matrix $\mathbf{A}_4$ has no eigenvalues on the unit circle, which is the case where the system is stable [4]. The case where $\mathbf{A}_4$ has eigenvalues on the unit circle will be considered in Section II-B.

Given a point z_2 on the unit circle, it follows from (6) that $\mathbf{E}(z_2)$, $\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $l(z_2)$ can be evaluated and used in (8a) and (8b) to obtain the values of $p_k(z_2)$ and $q_k(z_2)$ for $0 \le k \le m$ at the given point z_2. If this procedure is applied to each of the $n+1$ points defined by (9), then the values of every $q_k(z_2)$ and $p_k(z_2)$ on the set $\{z_2(l), 0 \le l \le n\}$ can be obtained. From these observations in conjunction with the analysis in Section II-A-1, we conclude that all polynomials $p_k(z_2)$ and $q_k(z_2)$ can be obtained using the following algorithm:

Algorithm 1:

Step 1: Use (6) to evaluate $\mathbf{E}(z_2)$, $\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $l(z_2)$ over the set of points defined in (9).

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

Step 3: Use (8a) and (8b) to obtain $p_k[z_2(l)]$ and $q_k[z_2(l)]$ for $0 \le l \le n$, $0 \le k \le m$.

Step 4: For each k $(0 \le k \le m)$, form vectors $\mathbf{q} = [p_0 \cdots p_n]^T$ and $\mathbf{q} = [q_0 \cdots q_n]^T$, and determine polynomials $p_k(z_2)$ and $q_k(z_2)$ by using (12)

B. The Unstable Case

If $\mathbf{A}_4$ has eigenvalues with unity modulus, the system is unstable. In such a case, the $n+1$ points defined by (9) need to be modified to

$$z_2(l) = re^{j2\pi l/(n+1)}, \qquad 0 \le l \le n \tag{13}$$

where $r > 0$ denotes the radius of a circle in the z_2 plane where $\mathbf{A}_4$ has no eigenvalues. With $\mathbf{q} = [p_0 \cdots p_n]^T$, (10) becomes

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

where

$$\begin{aligned}\mathbf{V}_r(\mathbf{z}_2) &= \begin{bmatrix} r^n z_2(0)^n & \cdots & rz_2(0) & 1 \\ \vdots & & \vdots & \vdots \\ r^n z_2(n)^n & \cdots & rz_2(n) & 1 \end{bmatrix} \\ &= \mathbf{V}(\mathbf{z}_2) \text{ diag } \{r^n, \cdots, r, 1\}\end{aligned}$$

and diag $\{r^n, \cdots, r, 1\}$ is the diagonal matrix with $r^n, \cdots, r, 1$ as the entries along its main diagonal. By (11),

$$\mathbf{V}_r(\mathbf{z}_2)^H \mathbf{V}_r(\mathbf{z}_2) = (n+1) \text{ diag } \{r^{2n}, \cdots, r^2, 1\}$$

which implies that

$$\mathbf{V}_r^{-1}(\mathbf{z}_2) = \frac{1}{n+1} \text{ diag } \{r^{-2n}, \cdots, r^{-2}, 1\}\mathbf{V}_r^H(\mathbf{z}_2).$$

Therefore, (12) is modified to

$$\begin{aligned}\boldsymbol{\alpha} &= \frac{1}{n+1} \text{ diag } \{r^{-2n}, \cdots, r^{-2}, 1\}\mathbf{V}_r^H(\mathbf{z}_2)\mathbf{q} \\ &= \frac{1}{n+1} \text{ diag } \{r^{-n}, \cdots, r^{-1}, 1\}\mathbf{V}^H(\mathbf{z}_2)\mathbf{q}.\end{aligned} \tag{14}$$

Note that (12) is a special case of (14) with $r = 1$, as may be expected.

C. Dual Algorithm

A dual algorithm to Algorithm 1 can be obtained when the roles of variables z_1 and z_2 are interchanged. By representing $H(z_1, z_2)$ in (2) as

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

where $\hat{p}_l(z_1)$ and $\hat{q}_l(z_1)$ are polynomials in z_1 of order not greater than m, it can be readily shown that

$$\begin{aligned}\sum_{l=0}^{n} \hat{q}_l(z_1) z_2^l = P(z_1, \mathbf{A}_1)\{&P[z_2, \hat{\mathbf{E}}(z_1) - \hat{\mathbf{f}}(z_1)\hat{\mathbf{g}}(z_1)] \\ &+ [\hat{l}(z_1) - 1]P[z_2, \hat{\mathbf{E}}(z_1)]\}\end{aligned} \tag{15a}$$

$$\sum_{l=0}^{n} \hat{p}_l(z_1) z_2^l = P(z_1, \mathbf{A}_1)P[z_2, \hat{\mathbf{E}}(z_1)] \tag{15b}$$

where $\hat{\mathbf{E}}(z_1)$, $\hat{\mathbf{f}}(z_1)$, $\hat{\mathbf{g}}(z_1)$, and $\hat{l}(z_1)$ can be obtained through the following matrix equation

$$\begin{bmatrix} \hat{\mathbf{E}}(z_1) & \hat{\mathbf{f}}(z_1) \\ \hat{\mathbf{g}}(z_1) & \hat{l}(z_1) \end{bmatrix} = \begin{bmatrix} \mathbf{A}_4 & \mathbf{b}_2 \\ \mathbf{c}_2 & d \end{bmatrix} + \begin{bmatrix} \mathbf{A}_3 \\ \mathbf{c}_1 \end{bmatrix} \cdot (z_1\mathbf{I} - \mathbf{A}_1)^{-1}[\mathbf{A}_2 \quad \mathbf{b}_1]. \tag{16}$$

Further, (12) needs to be modified as

$$\boldsymbol{\alpha} = \frac{1}{m+1}\mathbf{V}^H(\mathbf{z}_1)\mathbf{q} \tag{17}$$

where

$$\mathbf{z}_1 = [z_1(0) \quad z_1(1) \quad \cdots \quad z_1(m)]^T$$

with

$$z_1(k) = e^{j2\pi k/(m+1)}, \qquad 0 \le k \le m. \tag{18}$$

The above analysis leads to the following algorithm:

Algorithm 2:

Step 1: Use (16) to evaluate $\hat{\mathbf{E}}(z_1)$, $\hat{\mathbf{f}}(z_1)$, $\hat{\mathbf{g}}(z_1)$, and $\hat{l}(z_1)$ over the set of points defined by (18).

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

Step 3: Use (15a) and (15b) to obtain $\hat{q}_l[z_1(k)]$ and $\hat{p}_l[z_1(k)]$ for $0 \le l \le n$, $0 \le k \le m$.

Step 4: For each l $(0 \le l \le n)$, form vectors $\mathbf{q} = [\hat{p}_0 \cdots \hat{p}_m]^T$ and $\mathbf{q} = [\hat{q}_0 \cdots \hat{q}_m]^T$, and determine polynomials $\hat{p}_l(z_1)$ and $\hat{q}_l(z_1)$ by using (17).

Obviously, Algorithm 2 can be used to evaluate $H(z_1, z_2)$ only if matrix $\mathbf{A}_1$ has no eigenvalues on the unit circle. Modifications similar to those in (13), (14) should be made to deal with the case where $\mathbf{A}_1$ has eigenvalues on the unit circle.

D. The MIMO Case

Now consider the Roesser state-space model of a multi-input multi-output (MIMO) 2-D discrete system

$$\begin{aligned}\begin{bmatrix} \mathbf{x}^h(k+1, l) \\ \mathbf{x}^v(k, l+1) \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix}\begin{bmatrix} \mathbf{x}^h(k, l) \\ \mathbf{x}^v(k, l) \end{bmatrix} + \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix}\mathbf{u}(k, l) \\ &= \mathbf{Ax} + \mathbf{Bu}\end{aligned} \tag{19a}$$

$$\begin{aligned}\mathbf{y}(k, l) &= [\mathbf{C}_1 \quad \mathbf{C}_2]\begin{bmatrix} \mathbf{x}^h(k, l) \\ \mathbf{x}^v(k, l) \end{bmatrix} + \mathbf{Du}(k, l) \\ &= \mathbf{Cx} + \mathbf{Du}\end{aligned} \tag{19b}$$

where $\mathbf{u} \in \Re^t$ and $\mathbf{y} \in \Re^s$ are input and output vectors. The $s \times t$ transfer-function matrix of the system is given by

$$\begin{aligned}\mathbf{H}(z_1, z_2) &= [\mathbf{C}_1 \quad \mathbf{C}_2] \begin{bmatrix} z_1\mathbf{I} - \mathbf{A}_1 & -\mathbf{A}_2 \\ -\mathbf{A}_3 & z_2\mathbf{I} - \mathbf{A}_4 \end{bmatrix}^{-1} \\ &\quad \cdot \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{B}_2 \end{bmatrix} + \mathbf{D} \\ &= \mathbf{C}[\mathbf{I}(z_1, z_2) - \mathbf{A}]^{-1}\mathbf{B} + \mathbf{D}. \end{aligned} \tag{20}$$

Viewing the (k, l) entry of $\mathbf{H}(z_1, z_2)$ as a *scalar* 2-D rational function of order (m, n) given by

$$H_{kl}(z_1, z_2) = \mathbf{C}_k[\mathbf{I}(z_1, z_2) - \mathbf{A}]^{-1}\mathbf{B}_l + D_{kl}$$

where $\mathbf{C}_k$ and $\mathbf{B}_l$ are the kth row of $\mathbf{C}$ and the lth column of $\mathbf{B}$, respectively, and D_{kl} is the (k, l) entry of $\mathbf{D}$, $H_{kl}(z_1, z_2)$ can be considered to be $H(z_1, z_2)$ given by (2), which is the transfer function of the SISO Roesser state-space model given by (1a) and (1b) with $\mathbf{b} = \mathbf{B}_l$, $\mathbf{c} = \mathbf{C}_k$ and $d = D_{kl}$. Consequently, the transfer-function matrix $\mathbf{H}(z_1, z_2)$ in (20) can be evaluated entry by entry using Algorithm 1 or 2. This becomes apparent if we write $\mathbf{H}(z_1, z_2)$ in (20) as

$$\mathbf{H}(z_1, z_2) = \mathbf{L}(z_2) + \mathbf{G}(z_2)[z_1\mathbf{I} - \mathbf{E}(z_2)]^{-1}\mathbf{F}(z_2)$$

where

$$\begin{aligned}\begin{bmatrix} \mathbf{E}(z_2) & \mathbf{F}(z_2) \\ \mathbf{G}(z_2) & \mathbf{L}(z_2) \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_1 & \mathbf{B}_1 \\ \mathbf{C}_1 & \mathbf{D} \end{bmatrix} + \begin{bmatrix} \mathbf{A}_2 \\ \mathbf{C}_2 \end{bmatrix} \\ &\quad \cdot (z_2\mathbf{I} - \mathbf{A}_4)^{-1}[\mathbf{A}_3 \quad \mathbf{B}_2] \end{aligned} \tag{21}$$

$\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $l(z_2)$ in (5) are the lth column of $\mathbf{F}(z_2)$, the kth row of $\mathbf{G}(z_2)$, and the (k, l) entry of $\mathbf{L}(z_2)$ in (21), respectively. Consequently, (21) can be used to evaluate $\mathbf{E}(z_2)$, $\mathbf{f}(z_2)$, $\mathbf{g}(z_2)$, and $l(z_2)$ for each entry of $\mathbf{H}(z_1, z_2)$ when Algorithm 1 is applied.

Alternatively, (20) can be written as

$$\mathbf{H}(z_1, z_2) = \hat{\mathbf{L}}(z_1) + \hat{\mathbf{G}}(z_1)[z_2\mathbf{I} - \hat{\mathbf{E}}(z_1)]^{-1}\hat{\mathbf{F}}(z_1)$$

where

$$\begin{aligned}\begin{bmatrix} \hat{\mathbf{E}}(z_1) & \hat{\mathbf{F}}(z_1) \\ \hat{\mathbf{G}}(z_1) & \hat{\mathbf{L}}(z_1) \end{bmatrix} &= \begin{bmatrix} \mathbf{A}_4 & \mathbf{B}_2 \\ \mathbf{C}_2 & \mathbf{D} \end{bmatrix} + \begin{bmatrix} \mathbf{A}_3 \\ \mathbf{C}_1 \end{bmatrix} \\ &\quad \cdot (z_1\mathbf{I} - \mathbf{A}_1)^{-1}[\mathbf{A}_2 \quad \mathbf{B}_1] \end{aligned} \tag{22}$$

$\hat{\mathbf{f}}(z_1)$, $\hat{\mathbf{g}}(z_1)$, and $\hat{l}(z_1)$ in (16) are the lth column of $\hat{\mathbf{F}}(z_1)$, the kth row of $\hat{\mathbf{G}}(z_1)$, and the (k, l) entry of $\hat{\mathbf{L}}(z_1)$ in (22), respectively. Obviously, (22) is a key formula for the evaluation of $\hat{\mathbf{E}}(z_1)$, $\hat{\mathbf{f}}(z_1)$, $\hat{\mathbf{g}}(z_1)$, and $\hat{l}(z_1)$ for each entry of $\mathbf{H}(z_1, z_2)$ when Algorithm 2 is applied.

III. Derivation of the Transfer-Function Matrix from the Fornasini–Marchesini State-Space Model

In this section, two algorithms for the derivation of the 2-D transfer-function matrix of a linear, shift-invariant, discrete, multivariable 2-D system from the Fornasini–Marchesini state-space model are developed.

The Fornasini–Marchesini state-space model of a SISO 2-D discrete system is given by

$$\begin{aligned}\boldsymbol{x}(k+1, l+1) =& \boldsymbol{A}_1\boldsymbol{x}(k, l+1) + \boldsymbol{A}_2\boldsymbol{x}(k+1, l) \\ &+ \boldsymbol{b}_1u(k, l+1) + \boldsymbol{b}_2u(k+1, l)\end{aligned} \tag{23a}$$

$$y(k, l) = \boldsymbol{c}\boldsymbol{x}(k, l) + du(k, l) \tag{23b}$$

where $\boldsymbol{x} \in \Re^N$ is the state vector. The transfer function of the system can be expressed in terms of $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{b}_1$, $\boldsymbol{b}_2$, $\boldsymbol{c}$, and d, as

$$H(z_1, z_2) = \boldsymbol{c}(z_1z_2\mathbf{I} - z_2\boldsymbol{A}_1 - z_1\boldsymbol{A}_2)^{-1}(z_2\boldsymbol{b}_1 + z_1\boldsymbol{b}_2) + d \tag{24}$$

$$= \frac{\sum_{v=0}^{N} q_v(z_2)z_1^v}{\sum_{v=0}^{N} p_v(z_2)z_1^v} \tag{25}$$

where $p_v(z_2)$ and $q_v(z_2)$ are polynomials in z_2. As in (7), (24) can be written as

$$\begin{aligned}H(z_1, z_2) &= \frac{\det(z_1z_2\mathbf{I} - z_2\boldsymbol{A}_1 - z_1\boldsymbol{A}_2 + z_2\boldsymbol{b}_1\boldsymbol{c} + z_1\boldsymbol{b}_2\boldsymbol{c})}{\det(z_1z_2\mathbf{I} - z_2\boldsymbol{A}_1 - z_1\boldsymbol{A}_2)} \\ &\quad + d - 1 \\ &= \frac{\det(z_2\mathbf{I} - \boldsymbol{A}_2 + \boldsymbol{b}_2\boldsymbol{c})\,\det[z_1\mathbf{I} - \boldsymbol{F}(z_2)]}{\det(z_2\mathbf{I} - \boldsymbol{A}_2)\,\det[z_1\mathbf{I} - \boldsymbol{E}(z_2)]} + d - 1 \\ &= \frac{P(z_2, \boldsymbol{A}_2 - \boldsymbol{b}_2\boldsymbol{c})P[z_1, \boldsymbol{F}(z_2)]}{P(z_2, \boldsymbol{A}_2)P[z_1, \boldsymbol{E}(z_2)]} + d - 1 \end{aligned} \tag{26}$$

where

$$\boldsymbol{E}(z_2) = z_2\boldsymbol{A}_1(z_2\mathbf{I} - \boldsymbol{A}_2)^{-1} \tag{27a}$$

$$\boldsymbol{F}(z_2) = z_2(\boldsymbol{A}_1 - \boldsymbol{b}_1\boldsymbol{c})(z_2\mathbf{I} - \boldsymbol{A}_2 + \boldsymbol{b}_2\boldsymbol{c})^{-1}. \tag{27b}$$

In (26), $P(z_2, \boldsymbol{A}_2)$, $P(z_2, \boldsymbol{A}_2 - \boldsymbol{b}_2\ \boldsymbol{c})$, $P[z_1, \boldsymbol{E}(z_2)]$, $P[z_1, \boldsymbol{F}(z_2)]$ are the characteristic polynomials of $\boldsymbol{A}_2$, $\boldsymbol{A}_2 - \boldsymbol{b}_2\boldsymbol{c}$, $\boldsymbol{E}(z_2)$, and $\boldsymbol{F}(z_2)$, respectively. From (25) and (26), it follows that

$$\begin{aligned}\sum_{v=0}^{N} q_v(z_2)z_1^v =& P(z_2, \boldsymbol{A}_2 - \boldsymbol{b}_2\boldsymbol{c})P[z_1, \boldsymbol{F}(z_2)] \\ &+ (d-1)P(z_2, \boldsymbol{A}_2)P[z_1, \boldsymbol{E}(z_2)]\end{aligned} \tag{28a}$$

$$\sum_{v=0}^{N} p_v(z_2)z_1^v = P(z_2, \boldsymbol{A}_2)P[z_1, \boldsymbol{E}(z_2)]. \tag{28b}$$

A. Algorithm for a SISO Fornasini–Marchesini Model

The algorithm for the Fornasini–Marchesini model is based on (28a), (28b), and the assumption that matrices $\boldsymbol{A}_2$ and $\boldsymbol{A}_2 - \boldsymbol{b}_2\boldsymbol{c}$ have no eigenvalues on the unit circle. The method for the determination of a 1-D polynomial described in Section II-A-1 can be used here with some modifications. Specifically, (12) needs to be modified as

$$\boldsymbol{\alpha} = \frac{1}{N+1}\mathbf{V}^H(\boldsymbol{z}_2)\mathbf{q} \tag{29}$$

where

$$\boldsymbol{z}_2 = [z_2(0) \quad z_2(1) \quad \cdots \quad z_2(N)]^T$$

and

$$z_2(w) = e^{j2\pi w/(N+1)}, \qquad 0 \le w \le N. \tag{30}$$

The required algorithm can be constructed as follows:

Algorithm 3:

Step 1: Use (27a) and (27b) to evaluate $\boldsymbol{E}(z_2)$ and $\boldsymbol{F}(z_2)$ over the set of points defined in (30).

Step 2: Compute the determinants of $z_2\mathbf{I}-\boldsymbol{A}_2$ and $z_2\mathbf{I}-\boldsymbol{A}_2+\boldsymbol{b}_2\boldsymbol{c}$, and the characteristic equations of $\boldsymbol{E}(z_2)$ and $\boldsymbol{F}(z_2)$ for $z_2=z_2(w)$, $0\le w\le N$.

Step 3: Use (28a) and (28b) to obtain $p_v[z_2(w)]$ and $q_v[z_2(w)]$ for $0\le w\le N$, $0\le v\le N$.

Step 4: For each v $(0\le v\le N)$, form vectors $\mathbf{q}=[p_0\cdots p_N]^T$ and $\mathbf{q}=[q_0\cdots q_N]^T$, and determine polynomials $p_v(z_2)$ and $q_v(z_2)$ by using (29).

If $\boldsymbol{A}_2$ or $\boldsymbol{A}_2-\boldsymbol{b}_2\boldsymbol{c}$ has eigenvalues on the unit circle, then modifications similar to those in (13), (14) should be made.

B. Dual Algorithm

A dual algorithm to Algorithm 3 can be obtained when the roles of variables z_1 and z_2 are exchanged. By representing $H(z_1, z_2)$ in (24) as

$$H(z_1, z_2)=\frac{\sum_{w=0}^{N}\hat{q}_w(z_1)z_2^w}{\sum_{w=0}^{N}\hat{p}_w(z_1)z_2^w}$$

where $\hat{p}_w(z_1)$ and $\hat{q}_w(z_1)$ are polynomials in z_1, it can be readily shown that

$$\sum_{w=0}^{N}\hat{q}_w(z_1)z_2^w=P(z_1, \boldsymbol{A}_1-\boldsymbol{b}_1\boldsymbol{c})P[z_2, \hat{\boldsymbol{F}}(z_1)] + (d-1)P(z_1, \boldsymbol{A}_1)P[z_2, \hat{\boldsymbol{E}}(z_1)] \tag{31a}$$

$$\sum_{w=0}^{N}\hat{p}_w(z_1)z_2^w=P(z_1, \boldsymbol{A}_1)P[z_2, \hat{\boldsymbol{E}}(z_1)] \tag{31b}$$

where

$$\hat{\boldsymbol{E}}(z_1)=z_1\boldsymbol{A}_2(z_1\mathbf{I}-\boldsymbol{A}_1)^{-1} \tag{32a}$$

$$\hat{\boldsymbol{F}}(z_1)=z_1(\boldsymbol{A}_2-\boldsymbol{b}_2\boldsymbol{c})(z_1\mathbf{I}-\boldsymbol{A}_1+\boldsymbol{b}_1\boldsymbol{c})^{-1}. \tag{32b}$$

In (31a) and (31b), $P(z_1, \boldsymbol{A}_1)$, $P(z_1, \boldsymbol{A}_1-\boldsymbol{b}_1\boldsymbol{c})$, $P[z_2, \hat{\boldsymbol{E}}(z_1)]$, and $P[z_2, \hat{\boldsymbol{F}}(z_1)]$ are the characteristic polynomials of $\boldsymbol{A}_1$, $\boldsymbol{A}_1-\boldsymbol{b}_1\boldsymbol{c}$, $\hat{\boldsymbol{E}}(z_1)$, and $\hat{\boldsymbol{F}}(z_1)$, respectively. Further, (17) needs to be modified as

$$\boldsymbol{\alpha}=\frac{1}{N+1}\mathbf{V}^H(\boldsymbol{z}_1)\mathbf{q} \tag{33}$$

where

$$\boldsymbol{z}_1=[z_1(0) \quad z_1(1) \quad \cdots \quad z_1(N)]^T$$

with

$$z_1(v)=e^{j2\pi v/(N+1)}, \qquad 0\le v\le N. \tag{34}$$

The algorithm is as follows:

Algorithm 4:

Step 1: Use (32a) and (32b) to evaluate $\hat{\boldsymbol{E}}(z_1)$ and $\hat{\boldsymbol{F}}(z_1)$ over the set of points defined by (34).

Step 2: Compute the characteristic equations of $\boldsymbol{A}_1$, $\boldsymbol{A}_1-\boldsymbol{b}_1\boldsymbol{c}$, $\hat{\boldsymbol{E}}(z_1)$, and $\hat{\boldsymbol{F}}(z_1)$ for $z_1=z_1(v)$, $0\le v\le N$.

Step 3: Use (31a) and (31b) to obtain $\hat{q}_w[z_1(v)]$ and $\hat{p}_w[z_1(v)]$ for $0\le w\le N$, $0\le v\le N$.

Step 4: For each w $(0\le w\le N)$, form vectors $\mathbf{q}=[\hat{p}_0\cdots\hat{p}_N]^T$ and $\mathbf{q}=[\hat{q}_0\cdots\hat{q}_N]^T$, and determine polynomials $\hat{p}_w(z_1)$ and $\hat{q}_w(z_1)$ by using (33).

Obviously, Algorithm 4 can be used to evaluate $H(z_1, z_2)$ only if matrices $\boldsymbol{A}_1$ and $\boldsymbol{A}_1-\boldsymbol{b}_1\boldsymbol{c}$ have no eigenvalues on the unit circle. If matrix $\boldsymbol{A}_1$ or $\boldsymbol{A}_1-\boldsymbol{b}_1\boldsymbol{c}$ has eigenvalues on the unit circle, then modifications similar to (13), (14) should be made.

C. The MIMO Case

Consider now the Fornasini–Marchesini state-space model of a MIMO 2-D discrete system

$$\boldsymbol{x}(k+1, l+1)=\boldsymbol{A}_1\boldsymbol{x}(k, l+1)+\boldsymbol{A}_2\boldsymbol{x}(k+1, l) + \boldsymbol{B}_1\mathbf{u}(k, l+1)+\boldsymbol{B}_2\mathbf{u}(k+1, l) \tag{35a}$$

$$\mathbf{y}(k, l)=\boldsymbol{C}\boldsymbol{x}(k, l)+\mathbf{D}\mathbf{u}(k, l) \tag{35a}$$

where $\mathbf{u}\in\Re^t$, $\mathbf{y}\in\Re^s$ and $\mathbf{D}\in\Re^{s\times t}$. The $s\times t$ transfer-function matrix of the system can be expressed in terms of $\boldsymbol{A}_1$, $\boldsymbol{A}_2$, $\boldsymbol{B}_1$, $\boldsymbol{B}_2$, $\boldsymbol{C}$, and $\mathbf{D}$ as

$$\mathbf{H}(z_1, z_2)=\boldsymbol{C}(z_1z_2\mathbf{I}-z_2\boldsymbol{A}_1-z_1\boldsymbol{A}_2)^{-1} \cdot(z_2\boldsymbol{B}_1+z_1\boldsymbol{B}_2)+\mathbf{D} \tag{36}$$

whose entry (k, l) is a scalar rational function of order (N, N) given by

$$H_{kl}(z_1, z_2)=\boldsymbol{C}_k(z_1z_2\mathbf{I}-z_2\boldsymbol{A}_1-z_1\boldsymbol{A}_2)^{-1} \cdot(z_2\boldsymbol{B}_{1l}+z_1\boldsymbol{B}_{2l})+D_{kl} \tag{37}$$

where $\boldsymbol{C}_k$, $\boldsymbol{B}_{1l}$, and $\boldsymbol{B}_{2l}$ are the kth row of $\boldsymbol{C}$ and the lth column of $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, respectively. Therefore, the transfer-function matrix $\mathbf{H}(z_1, z_2)$ given by (36) can be evaluated entry by entry and each entry can be treated as a SISO transfer-function. Hence, (28a) associated with $H_{kl}(z_1, z_2)$ in (37) becomes

$$\sum_{v=0}^{N}q_v(z_2)z_1^v=P(z_2, \boldsymbol{A}_2-\boldsymbol{B}_{2l}\boldsymbol{C}_k)P[z_1, \tilde{\boldsymbol{F}}(z_2)] + (D_{kl}-1)P(z_2, \boldsymbol{A}_2)P[z_1, \boldsymbol{E}(z_2)] \tag{38}$$

where

$$\tilde{\boldsymbol{F}}(z_2)=z_2(\boldsymbol{A}_1-\boldsymbol{B}_{1l}\boldsymbol{C}_k)(z_2\mathbf{I}-\boldsymbol{A}_2+\boldsymbol{B}_{2l}\boldsymbol{C}_k)^{-1} \tag{39}$$

In (38), $P(z_2, \boldsymbol{A}_2-\boldsymbol{B}_{2l}\boldsymbol{C}_k)$ and $P[z_1, \tilde{\boldsymbol{F}}(z_2)]$ are the characteristic polynomials of $\boldsymbol{A}_2-\boldsymbol{B}_{2l}\boldsymbol{C}_k$ and $\tilde{\boldsymbol{F}}(z_2)$, respectively. Therefore, Algorithm 3 can be extended to deal with the MIMO case by substituting (38) and (39) into (28a) and (27b), respectively.

Similarly, (31a) becomes

$$\sum_{w=0}^{N}\hat{q}_w(z_1)z_2^w=P(z_1, \boldsymbol{A}_1-\boldsymbol{B}_{1l}\boldsymbol{C}_k)P[z_2, \breve{\mathbf{F}}(z_1)] + (D_{kl}-1)P(z_1, \boldsymbol{A}_1)P[z_2, \hat{\boldsymbol{E}}(z_2)] \tag{40}$$

where

$$\breve{\mathbf{F}}(z_1) = z_1(\boldsymbol{A}_2 - \boldsymbol{B}_{2l}\boldsymbol{C}_k)(z_1\mathbf{I} - \boldsymbol{A}_1 + \boldsymbol{B}_{1l}\boldsymbol{C}_k)^{-1} \tag{41}$$

$P(z_1, \boldsymbol{A}_1 - \boldsymbol{B}_{1l}\boldsymbol{C}_k)$ and $P[z_2, \breve{\mathbf{F}}(z_1)]$ are the characteristic polynomials of $\boldsymbol{A}_1 - \boldsymbol{B}_{1l}\boldsymbol{C}_k$, and $\breve{\mathbf{F}}(z_1)$, respectively. Therefore, Algorithm 4 can be extended to deal with the MIMO case by substituting (40) and (41) into (31a) and (32b), respectively.

IV. Examples

In Section IV-A, Algorithms 1 and 2 are applied to four 2-D discrete systems represented by the Roesser state-space model and the required amounts of computation are compared with those required by the existing algorithms [19], [22]. In Section IV-B, Algorithms 3 and 4 are applied to two systems represented by the Fornasini–Marchesini state-space model.

A. Examples for the Roesser Model

Example 1 is a 2-D discrete system of order (2, 6), which was used in [6] for stability analysis of 2-D systems. The system is represented by the Roesser state-space model with the matrices:

$$\mathbf{A}_1 = \begin{bmatrix} 0.500 & 0.007 \\ -0.007 & 0.500 \end{bmatrix}, \ \mathbf{A}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T$$

$$\mathbf{A}_2 = \begin{bmatrix} 0.012 & -0.008 & 0.028 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.012 & 0.008 & 0.012 \end{bmatrix}$$

$$\mathbf{A}_4 = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0.845 & -2.657 & 2.810 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -0.845 & -2.657 & -2.810 \end{bmatrix}$$

$$\begin{aligned} \mathbf{b} &= [\mathbf{b}_1^T \quad \mathbf{b}_2^T]^T \\ &= [0.134 \quad 1 \quad | \quad -0.657 \quad 0.036 \quad 0.269 \quad 0.805 \quad 1 \quad 2]^T \\ \mathbf{c} &= [\mathbf{c}_1 \quad \mathbf{c}_2] = [0.983 \quad 0.500 \quad | \quad -1 \quad 0 \quad 1 \quad 2 \quad 3 \quad 1] \\ d &= 0. \end{aligned}$$

Algorithms 1 and 2 proposed and the algorithms in [19] and [22] led to the transfer function

$$H(z_1, z_2) = \frac{[z_2^6 \quad \cdots \quad z_2 \quad 1]\mathbf{N}_t[z_1^2 \quad z_1 \quad 1]^T}{[z_2^6 \quad \cdots \quad z_2 \quad 1]\mathbf{D}_t[z_1^2 \quad z_1 \quad 1]^T}$$

where

$$\mathbf{D}_t = \begin{bmatrix} 1.0000 & -1.0000 & 0.2500 \\ 0.0000 & 0.0000 & 0.0001 \\ -2.5821 & 2.5821 & -0.6453 \\ 0.0000 & 0.0000 & 0.0002 \\ 2.3107 & -2.3107 & 0.5778 \\ 0.0000 & 0.0000 & 0.0000 \\ -0.7140 & 0.7140 & -0.1785 \end{bmatrix}$$

$$\mathbf{N}_t = \begin{bmatrix} 0.0000 & 0.6317 & -0.3094 \\ 7.5360 & -6.3818 & 1.3419 \\ -0.8882 & 2.0949 & -0.7442 \\ -23.9776 & 25.1974 & -6.4878 \\ 8.8500 & -9.2265 & 2.5091 \\ 16.5056 & -18.8216 & 5.3540 \\ -7.9463 & 6.2887 & -1.1409 \end{bmatrix}$$

TABLE I

Computational Complexity of the Algorithms for the Roesser Model

Algorithms	Flops			
	Example 1	Example 2	Example 3	Example 4
1	3.786×10^4	5.657×10^3	5.375×10^6	7.386×10^6
2	8.424×10^4	5.692×10^3	2.591×10^6	2.325×10^7
Fadeeva [19]	2.431×10^5	1.360×10^4	1.143×10^8	1.440×10^8
DFT [22]	1.815×10^5	2.060×10^4	2.645×10^7	3.178×10^7

The amounts of computation required by the various algorithms are listed in Table I.

Example 2 is a two-input two-output system represented by the Roesser model of order (2, 2), which was used to illustrate the algorithm in [19]. The model is given by (19a) and (19b) with

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \left[\begin{array}{cc|cc} 2 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ \hline 1 & 0 & -2 & 0 \\ 0 & 1 & 0 & -2 \end{array}\right]$$

$$\mathbf{B} = [\mathbf{B}_1^T \quad \mathbf{B}_2^T]^T = \left[\begin{array}{cc|cc} 1 & -1 & 2 & 1 \\ 1 & 0 & 1 & 0 \end{array}\right]^T$$

$$\mathbf{C} = [\mathbf{C}_1 \quad \mathbf{C}_2] = \left[\begin{array}{cc|cc} 1 & 1 & 0 & -1 \\ 0 & -1 & 1 & 1 \end{array}\right].$$

The transfer-function matrix obtained by using Algorithms 1 and 2, and the Algorithms in [19] and [22] is

$$\mathbf{H}(z_1, z_2) = \begin{bmatrix} H_1(z_1, z_2) & H_2(z_1, z_2) \\ H_3(z_1, z_2) & H_4(z_1, z_2) \end{bmatrix}$$

where the denominator is given by the matrix:

$$\mathbf{D}_t = \begin{bmatrix} 1 & -2 & -1 \\ 4 & -10 & -2 \\ 4 & -12 & 1 \end{bmatrix}$$

and the numerators are specified by $\mathbf{N}_{t1}$, $\mathbf{N}_{t2}$, $\mathbf{N}_{t3}$, and $\mathbf{N}_{t4}$ as follows: e14

$$\mathbf{N}_t = \begin{bmatrix} \mathbf{N}_{t1} & \mathbf{N}_{t2} \\ \mathbf{N}_{t3} & \mathbf{N}_{t4} \end{bmatrix} = \left[\begin{array}{ccc|ccc} 0 & 0 & 2 & 0 & 1 & 1 \\ -1 & 6 & 7 & 0 & 5 & 3 \\ -2 & 13 & 0 & 0 & 6 & 0 \\ \hline 0 & 1 & -3 & 0 & 0 & -1 \\ 3 & -3 & -14 & 1 & -1 & -5 \\ 6 & -13 & -8 & 2 & -3 & -4 \end{array}\right].$$

The amounts of computation required by the various algorithms are listed in Table I.

Example 3 is a 2-D SISO discrete system of order (16, 8) represented by the Roesser state-space model given in (1a) and (1b). Each element of $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, and d is a random number chosen from a normal distribution with zero mean and unit variance. The amounts of computation required by the algorithms are listed in Table I.

Example 4 is a four-input two-output 2-D discrete system of order (8, 16) represented by the Roesser state-space model in (19a) and (19b). Each element of $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ is a random number chosen from a normal distribution with zero mean and unit variance. The amounts of computation are listed in Table I.

From these examples, it is evident that both Algorithms 1 and 2 lead to a significant reduction in the amount of computation relative to the Fadeeva and DFT Algorithms [19], [22]. The DFT algorithm, which exploits the efficiency of the fast Fourier transform (FFT), is efficient for high-order systems; nevertheless, our algorithms are more efficient.

Algorithms 1 and 2 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 1 requires less computation than Algorithm 2 when $m < n$ (see Examples 1 and 4), and Algorithm 2 requires less computation when $m > n$ (see Example 3).

B. Examples for the Fornasini–Marchesini Model

Example 5 is a 2-D discrete system of order (1, 1), which was used in [26] to synthesize optimal Fornasini–Marchesini state-space model structures utilizing a 2-D similarity transformation matrix that is not block-diagonal. The system is represented by the Fornasini–Marchesini state-space model of (23a) and (23b) with

$$\begin{aligned} \boldsymbol{A}_1 &= \begin{bmatrix} 0 & 1 \\ 0 & -0.7243 \end{bmatrix} \qquad \boldsymbol{b}_1 = \begin{bmatrix} 0.1210 \\ -0.0456 \end{bmatrix} \\ \boldsymbol{A}_2 &= \begin{bmatrix} 0 & 0 \\ -0.5257 & -0.6815 \end{bmatrix} \qquad \boldsymbol{b}_2 = \begin{bmatrix} 0 \\ 0.0223 \end{bmatrix}. \\ \boldsymbol{c} &= [0 \quad 1] \qquad d = 0 \end{aligned}$$

It can be readily verified that the above system can be represented by the Roesser state-space model with

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} = \begin{bmatrix} -0.7243 & | & 1.0543 \\ --- & | & --- \\ -0.0304 & | & -0.6815 \end{bmatrix}, \\ \mathbf{b} &= \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} = \begin{bmatrix} 1 \\ - \\ 1 \end{bmatrix} \\ \mathbf{c} &= [\mathbf{c}_1 \quad \mathbf{c}_2] = [-0.0456 \quad | \quad 0.0223] \\ d &= 1. \end{aligned}$$

The transfer functions obtained by using Algorithms 1 to 4 are given by

$$H(z_1, z_2) = \frac{[z_2 \quad 1]\mathbf{N}_t[z_1 \quad 1]^T}{[z_2 \quad 1]\mathbf{D}_t[z_1 \quad 1]^T}$$

where

$$\begin{aligned} \mathbf{D}_t &= \begin{bmatrix} 1.0000 & 0.7243 \\ 0.6815 & 0.5257 \end{bmatrix} \\ \mathbf{N}_t &= \begin{bmatrix} 1.0000 & 0.6788 \\ 0.7038 & 0.4621 \end{bmatrix}. \end{aligned}$$

TABLE II
COMPUTATIONAL COMPLEXITY OF NEW ALGORITHMS

Algorithms	Flops	
	Example 5	Example 6
1	515	5657
2	514	5692
3	3515	61953
4	2648	60439

Example 6 is the 2-D discrete system in Example 2. It can be represented by the Fornasini–Marchesini model [24] with

$$\begin{aligned} \boldsymbol{A}_1 &= \begin{bmatrix} \mathbf{A}_1 & \mathbf{A}_2 \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \\ \boldsymbol{A}_2 &= \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{A}_3 & \mathbf{A}_4 \end{bmatrix} \\ \boldsymbol{B}_1 &= \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{0} \end{bmatrix} \\ \boldsymbol{B}_2 &= \begin{bmatrix} \mathbf{0} \\ \mathbf{B}_2 \end{bmatrix} \\ \boldsymbol{C} &= \mathbf{C}. \end{aligned} \tag{42}$$

The amounts of computation required by Examples 5 and 6 are listed in Table II.

As can be seen, Algorithms 1 and 2, i.e., the algorithms based on the Roesser model, are significantly more efficient than Algorithms 3 and 4, the algorithms based on the Fornasini–Marchesini model.

V. CONCLUSIONS

Two algorithms based on a 1-D polynomial determination technique for the derivation of the transfer-function matrix of a 2-D discrete system from the Roesser state-space model have been proposed. The computational efficiency of the algorithms has been examined and found to be superior relative to that of the algorithms described in [19], [22]. Then, two algorithms based on the Fornasini–Marchesini state-space model have been derived. A comparison of the algorithms based on the Roesser model (Algorithms 1 and 2) with the algorithms based on the Fornasini–Marchesini state-space model (Algorithms 3 and 4) has shown the former to be more efficient by a factor of about 10.

REFERENCES

[1] W.-S. Lu and A. Antoniou, *Two-Dimensional Digital Filters.* New York: Marcel Dekker, 1992.
[2] N. G. El-Agizi and M. M. Fahmy, "Two-dimensional digital filters with no overflow oscillations," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-27, pp. 465–469, June 1979.
[3] J. H. Lodge and M. M. Fahmy, "Stability and overflow oscillations in 2-D state space digital filters," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-29, pp. 1161–1171, Dec. 1981.
[4] W.-S. Lu and E. B. Lee, "Stability analysis for two-dimensional analysis," *IEEE Trans. Circuits Syst.,* vol. CAS-30, pp. 455–461, July 1983.
[5] ——, "Stability analysis for two-dimensional systems via a Lyapunov approach," *IEEE Trans. Circuits Syst,* vol. CAS-32, pp. 61–68, Jan. 1985.
[6] B. D. O. Anderson, P. Agathoklis, E. I. Jury, and M. Mansour, "Stability and the matrix Lyapunov equations for discrete two-dimensional systems," *IEEE Trans. Circuits Syst.,* vol. CAS-33, pp. 261–267, Mar. 1986.

[7] T. Lin, M. Kawamate, and T. Higuchi, "A unified study on the roundoff noise in 2-D state-space digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 724–730, July 1986.
[8] W.-S. Lu and A. Antoniou, "Synthesis of 2-D state-space fixed-point digital-filter structures with minimum roundoff noise," *IEEE Trans. Circuits Syst.*, vol. CAS-33, pp. 965–973, Oct. 1986.
[9] T. Lin, 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 Syst.*, vol. CAS-34, pp. 113–127, Feb. 1987.
[10] W.-S. Lu, 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, Oct. 1991.
[11] W.-S. Lu, E. B. Lee, and Q.-T. Zhang, "Model reduction of 2-D and delay-differential systems," *Int. J. Contr.*, vol. 46, pp. 2199–2218, Dec. 1987.
[12] K. Premaratne, E. I. Jury, and M. Mansour, "An algorithm for model reduction of 2-D discrete time systems," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 1116–1132, Sept. 1990.
[13] K. Zhou, J. L. Aravena, G. Gu, and D. Xiong, "Two-dimensional system model reduction by quasibalanced truncation and singular perturbation," Dep. Elect. Comput. Eng., Louisiana State University, Res. Rep., Sept. 1991.
[14] W.-S. Lu, H.-P. Wang, and A. Antoniou, "On the evaluation of the controllability and observability grammians of 2-D digital filters," in *Proc. Int. Symp. Circuits Syst.*, Singapore, June 1991, pp. 602–605.
[15] H. Luo, W.-S. Lu, and A. Antoniou, "Numerical solution of the two-dimensional Lyapunov equations and application in order reduction of recursive digital filters," in *Proc. IEEE Pacific Rim Conf. Commun., Comput. Signal Processing*, Victoria, BC, Canada, pp. 232–235, May 1993.
[16] R. P. Roesser, "A discrete state-space model for linear imaging processing," *IEEE Trans. Automatic Contr.*, vol. 20, pp. 1–10, Feb. 1975.
[17] C. S. Koo and C. T. Chen, "Fadeeva's algorithm for spatial dynamical equations," *Proc. IEEE*, vol. 65, pp. 975–976, June 1977.
[18] B. G. Mertzios and P. N. Paraskevopoulos, "Transfer function matrix of 2-D systems," *IEEE Trans. Automatic Contr.*, vol. 26, pp. 722–724, June 1981.
[19] B. G. Mertzios, "An algorithm for the computation of the transfer function matrix of two-dimensional systems," *J. Franklin Inst.*, vol. 321, pp. 73–80, Feb. 1986.
[20] K. S. Yeung and F. Kumbi, "Symbolic matrix inversion with application to electronic circuits," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 235–238, Feb. 1988.
[21] G. E. Antoniou, G. O. A. Glentis, D. A. Karras, and S. J. Varoufakis, "Transfer function determination of singular systems using the DTF," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 1140–1142, 1989.
[22] G. E. Antoniou, S. E. Mentzelopoulou, and G. O. A. Glentis, "Determination of transfer function of two dimensional generalized systems using the discrete Fourier transform," *Proc Inst. Elect. Eng. Pt. D*, vol. 138, pp. 327–330, July 1991.
[23] L. A. Zadeh and C. A. Desoer, *Linear System Theory*. New York: McGraw-Hill, 1963.
[24] E. Fornasini and G. Marchesini, "Doubly-indexed dynamical systems: State-space models and structural properties," *Math. Sys. Theory*, vol. 12, pp. 59–72, 1978.
[25] T. Kailath, *Linear Systems*. Englewood Clifts, NJ: Prentice Hall, 1981.
[26] T. Hinamoto, T. Hamanaka, and S. Maekawa, "A generalized study on the synthesis of 2-D state-space digital filters with minimum round-off noise," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1037–1042, Aug. 1988.
[27] W.-S. Lu and A. Antoniou, "On the synthesis of 2-D state-space digital filters with minimum roundoff noise," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 1424–1425, Nov. 1990.

Hong Luo (S'90–M'96) received the B.Sc. and M.Sc. degrees in mechanical engineering from Shanghai Jiao-Tong University, China, in 1983 and 1986, respectively, and the Ph.D. degree in electrical engineering from the University of Victoria, BC, Canada, in 1995.

She is currently a DSP design engineer with Micom Communications Corporation, Simi Valley, CA. Her research interests are in the areas of digital filters, digital signal and image processing, and telecommunication systems.

Wu-Sheng Lu (S'81–M'85–SM'90) received the B.S. degree in mathematics from Fudan University, China, in 1964, the M.S. degree in electrical engineering, and the Ph.D. degree in control science from the University of Minnesota, Minneapolis, in 1983 and 1984, respectively.

He was a post-Doctoral Fellow at the University of Victoria, Victoria, BC, Canada, in 1985 and held a visiting Assistant Professorship with the University of Minnesota in 1986. Since 1987, he has been with the University of Victoria where he is currently Professor. His teaching and research interests are in the areas of digital signal processing and robotics. He is the coauthor with A. Antoniou of *Two-Dimensional Digital Filters* (Marcel Dekker, 1992).

Dr. Lu served as Associate Editor of the *Canadian Journal of Electrical and Computer Engineering* in 1989, and was Editor of the same journal from 1990 to 1992. He was an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—PART II from 1993 to 1995.

Andreas Antoniou (M'69–SM'79–F'82) received the B.Sc. (Eng.) and Ph.D. degrees in electrical engineering from London University in 1963 and 1966, respectively.

From 1966 to 1969, he was Senior Scientific Officer at the Post Office Research Department, London, and from 1969 to 1970 he was a member of the Scientific Staff at the R&D Laboratories of Northern Electric Company Ltd., Ottawa, Ont., Canada. From 1970 to 1983, he was with the Department of Electrical and Computer Engineering, Concordia University, Montreal, Quebec, Canada, as Professor from June 1973 and as Chairman from December 1977. He was founding Chairman of the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada, from July 1, 1983 to June 30, 1990, and is now Professor in the same department. His teaching and research interests are in the areas of electronics, network synthesis digital system design, active and digital filters, and digital signal processing. He published extensively in these areas. One of his papers on gyrator circuits was awarded the Ambrose Fleming Premium by the Institution of Electrical Engineers, U.K. He is the author of *Digital Filters: Analysis, Design, and Applications* (McGraw-Hill) and the coauthor with W.-S. Lu of *Two-Dimensional Digital Filters* (Marcel Dekker, 1992).

Dr. Antoniou is a Member of the Association of Professional Engineers and Geoscientists of British Columbia, Canada, and a Fellow of the Institution of Electrical Engineers. He was elected Fellow of the IEEE for contributions to active and digital filters, and to electrical engineering education. He served as Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS from June 1983 to May 1985, and as Editor from June 1985 to May 1987, and is currently serving as a member of the Board of Governors of the Circuits and Systems Society.