



# An efficient low order model for two-dimensional digital systems: Application to the 2D digital filters<sup>☆</sup>



**Lahcène Mitiche<sup>\*</sup>, Amel Baha Houda Adamou-Mitiche**

*Science and Technology Department, University of Djelfa, Algeria*

Received 7 September 2013; revised 7 January 2014; accepted 13 March 2014  
Available online 9 May 2014

## KEYWORDS

Two-dimensional digital systems;  
Stability;  
State space;  
Balanced realization;  
Model reduction

**Abstract** The work presented in this paper concerns with analysis and synthesis of the two-dimensional (2D) digital systems based on model order reduction, and application to the 2D-digital filters. The synthesis of the 2D-systems is performed with two methods, the Prony’s method (Prony modified) and Iterative method, in the spatial domain, and with the method of Semi-Definite iterative Programming (SDP), in the frequency domain. After synthesis, we make an order reduction of the synthesized model by the quasi-Gramians method. From several results and their interpretations, the 2D filter synthesized by the model reduction proposed method, presents advantages compared to the conventional direct synthesized filters.

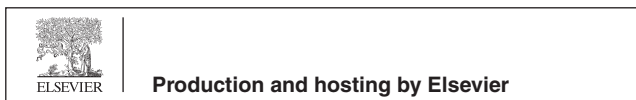
© 2014 King Saud University. Production and hosting by Elsevier B.V. All rights reserved.

## Contents

1. Introduction . . . . .	309
2. 2D IIR digital systems . . . . .	309
2.1. The design problem . . . . .	310
2.2. The stability problem . . . . .	310
2.3. Spatial domain synthesis . . . . .	310
(A) Prony’s method . . . . .	311
(B) Iterative algorithm . . . . .	311
2.4. Frequency domain design by the iterative Semi-Definite Programming . . . . .	311

<sup>\*</sup> Corresponding author. Tel.: +213 27 90 90 15.  
E-mail addresses: [l\\_mitiche@yahoo.fr](mailto:l_mitiche@yahoo.fr) (L. Mitiche), [amelmitiche@yahoo.fr](mailto:amelmitiche@yahoo.fr) (A.B.H. Adamou-Mitiche).

<sup>☆</sup> This work was partially performed at the LPTM, University of Cergy-Pontoise, France.  
Peer review under responsibility of King Saud University.



*List of symbols and abbreviations:*

1D	One-dimensional	$H(z_1, z_2)$	The 2D transfer function
2D	Two-dimensional	$(n, r)$	The full order of the original system
IIR	Infinite Impulse Response	SS	State Space
FIR	Finite Impulse Response	$X(i, j)$	The state vector
SDP	Semi-Definite Programming	$(A, B, C, D)$	The full order realization
$a(\dots)$ and $b(\dots)$	The model coefficients	$(A_r, B_r, C_r)$	The reduced order realization
$h(n_1, n_2)$	2D impulse response	SISO	Single Input Single Output
$\delta_p, \delta_s$	The pass-band and stop band tolerance (template)		

2.4.1.	Problem formulation	312
2.4.2.	Design steps	312
3.	2D digital system model reduction	312
3.1.	2D State-space models	312
3.2.	Model order reduction methods	313
3.2.1.	Balanced realization approximation	314
3.2.2.	Iterative Algorithm for computing the quasi-Gramians	315
4.	Illustrative simulations	316
4.1.	Part 1: 2D-filter design	316
4.2.	Part 2: Order reduction	316
4.3.	Interpretation	317
5.	Conclusion	317
	Acknowledgments	317
	References	317

## 1. Introduction

The fields of two-dimensional digital dynamical systems and signal processing have maintained tremendous vitality over the past four decades and there is a clear indication that this trend will continue. Advances in hardware technology provide capabilities in signal processing chips and microprocessors that were previously associated with mainframe computers. These advances allow sophisticated signal and image processing algorithms to be implemented in real time at a substantially reduced cost. New applications continue to be found and existing applications continue to expand in such diverse areas as control, communications, consumer electronics, medicine, defense, robotics, and geophysics (Salam, 2011; Ramamoorthy and Bruton, 1979).

At a conceptual level, there is a great deal of similarity between one-dimensional systems and two-dimensional systems (Sontag, 1978). Another problem is the absence of a fundamental theorem of algebra for two-dimensional polynomials. One-dimensional polynomials can be factored as a product of lower-order polynomials. An important structure for realizing one-dimensional systems is the cascade structure (Antoniu, 2001). In this case, the  $z$ -transform of the impulse response is factored as a product of lower-order polynomials, and the realizations of these lower-order factors are cascaded.

The  $z$ -transform of the impulse response of two-dimensional digital systems cannot, in general, be factored as a product of lower-order polynomials, and therefore, the cascade structure is not a general structure for achieving a two-dimensional digital system (Lim, 1990). Another

consequence of the nonfactorability of two-dimensional polynomials is the difficulty associated with issues related to system stability. In a one-dimensional system, the pole locations can be determined easily, and an unstable system can be stabilized without affecting the magnitude response by simple manipulation of pole locations. In a two-dimensional system, as the poles are surfaces rather than points, and there is no fundamental theorem of algebra, it is extremely difficult to determine the pole locations (Gonzalez and Woods, 2007).

The present paper is related particularly to the synthesis of the two-dimensional infinite impulse response (2D IIR) dynamical systems using model order reduction. In the first part of the paper, the synthesis is presented, both in the spatial domain with two methods (modified Prony's method and Iterative method) and in the frequency domain with iterative Semi-Definite Programming (SDP). In the second part of this paper, we describe an order reduction of the synthesized system using a quasi-Gramians approach.

## 2. 2D IIR digital systems

2D IIR systems with an arbitrary impulse response  $h(n_1, n_2)$  cannot be created as computing each output sample requires too many arithmetic operations (Lim, 1990; Gonzalez and Woods, 2007), which results in high estimate for the number of arithmetic operations needed. As a result, in addition to requiring  $h(n_1, n_2)$  to be real and stable, we require  $h(n_1, n_2)$  to have a rational  $z$ -transform corresponding to a recursively computable system.

### 2.1. The design problem

The problem of IIR system design is to determine a rational and stable function  $H(z_1, z_2)$  with a wedge support output mask that meets a given design specification. In other words, we wish to determine a stable computational procedure that is recursively computable and meets a design specification.

However, a given rational system function  $H(z_1, z_2)$  can lead to many different computational procedures (Lim, 1990). To make the relationship unique, we will adopt a convention when expressing  $H(z_1, z_2)$ .

Specifically, we will assume that  $a(0, 0)$  is always 1, so  $H(z_1, z_2)$  will then be in the form

$$H(z_1, z_2) = \frac{\sum_{(k_1, k_2) \in R_b} b(k_1, k_2) z_1^{-k_1} z_2^{-k_2}}{1 + \sum_{(k_1, k_2) \in R_a - (0,0)} a(k_1, k_2) z_1^{-k_1} z_2^{-k_2}}, \quad (1)$$

and  $a(0,0)$  results in put the transfer function  $H(z_1, z_2)$  being in the canonical form, and  $R_a - (0,0)$  represents the support region of  $a(k_1, k_2)$  without the origin,  $(0,0)$ .  $R_b$  represents the support region of  $b(k_1, k_2)$ .

The unique computational procedure corresponding to (1) is then given by

$$y(n_1, n_2) \leftarrow - \sum_{(k_1, k_2) \in R_a - (0,0)} \sum a(k_1, k_2) y(n_1 - k_1, n_2 - k_2) + \sum_{(k_1, k_2) \in R_b} \sum b(k_1, k_2) x(n_1 - k_1, n_2 - k_2), \quad (2)$$

where the sequences  $a(k_1, k_2)$  and  $b(k_1, k_2)$  are the model coefficients, and  $x(\dots)$  is the input signal.

The first step in the IIR model design is usually an initial determination of  $R_a$  and  $R_b$ , the support regions of  $a(k_1, k_2)$  and  $b(k_1, k_2)$ , respectively. If we are to determine the model coefficients by attempting to approximate some desired impulse response  $h_d(n_1, n_2)$  in the spatial domain, we will want to choose  $R_a$  and  $R_b$  such that  $h(n_1, n_2)$  will have at least approximately the same support region as  $h_d(n_1, n_2)$ .

Another consideration is related to the model specification parameters. In low pass filter design, for example, small  $\delta_p, \delta_s$  (filter templates) and transition regions will generally require a larger number of filter coefficients. It is often difficult to determine the number of model coefficients required to meet a given specification for a particular design algorithm, and an iterative procedure may become necessary (Lim, 1990).

One major difference between IIR and FIR (Finite Impulse Response) systems is related to stability. A FIR system is always stable as long as  $h(n_1, n_2)$  is bounded for all  $(n_1, n_2)$  (Antoniou, 2001; Lim, 1990), so the stability is never an issue. With IIR systems, however, ensuring stability is a major task. One approach to designing a stable IIR system is to impose a special structure on  $H(z_1, z_2)$  such that testing the stability and stabilizing an unstable system become relatively easy tasks. Such an approach, however, tends to impose a severe constraint on the design algorithm or to highly restrict the class of systems that can be designed (Lim, 1990). For example, if  $H(z_1, z_2)$  has a separable denominator polynomial of the form  $A_1(z_1)A_2(z_2)$ , testing the stability and stabilizing an unstable  $H(z_1, z_2)$  without affecting the magnitude response is a one-dimensional (1D) problem (Mandal et al., 2012). However, the class of systems that can be designed with a separable denominator polynomial without a significant increase in the

number of coefficients in the numerator polynomial of  $H(z_1, z_2)$  is restricted. An alternative approach is to design a system without considering the stability issue, and then test the stability of the resulting system and attempt to stabilize it if it proves unstable. However, testing stability and stabilizing an unstable system are not easy problems.

### 2.2. The stability problem

In the 1D case, testing the stability of a causal system whose system function is given by  $H(z) = \frac{1}{A(z)}$  is quite straightforward. As a 1D polynomial  $A(z)$  can always be factored straightforwardly as a product of first-order polynomials, we can easily determine the poles of  $H(z)$ . The stability of the causal system is equivalent to having all the poles inside the unit circle. The above approach cannot be used in testing the stability of a 2D first quadrant support system. That approach requires the specific location of all poles to be determined. Partly because a 2D polynomial  $A(z_1, z_2)$  cannot in general be factored as a product of lower-order polynomials, it is extremely difficult to determine all the pole surfaces of  $H(z_1, z_2) = \frac{1}{A(z_1, z_2)}$ , and the approach based on explicit determination of all pole surfaces has not led to successful practical procedures for the system stability testing (Lim, 1990; Gonzalez and Woods, 2007).

### 2.3. Spatial domain synthesis

The input often used in IIR system design is  $\delta(n_1, n_2)$ , and the desired impulse response, assumed given, is denoted by  $h_d(n_1, n_2)$ . Spatial domain design can be viewed as a system identification problem. Suppose we have an unknown system that we wish to model with a rational system function  $H(z_1, z_2)$ . One approach to estimating the system model parameters (model coefficients  $a(k_1, k_2)$  and  $b(k_1, k_2)$ ) is to require the impulse response of the designed system to be as close as possible in some sense to  $h_d(n_1, n_2)$ .

The error criterion used in the system design is

$$\text{Error} = \sum_{(n_1, n_2) \in R_e} \sum e^2(n_1, n_2), \quad (3a)$$

where

$$e(n_1, n_2) = h_d(n_1, n_2) - h(n_1, n_2), \quad (3b)$$

and  $R_e$  is the support region of the error sequence. Ideally,  $R_e$  coincides with all values of  $(n_1, n_2)$ .

Minimizing the error in (3) with respect to  $a(k_1, k_2)$  and  $b(k_1, k_2)$  is a nonlinear problem. An approach is to slightly modify the error in (3) such that the resulting algorithm leads to closed form solutions that require solving only sets of linear equations (Lim, 1990).

Consider the computational procedure given by (2). We will assume that there are  $p$  unknown values of  $a(k_1, k_2)$  and  $q + 1$  unknown values of  $b(k_1, k_2)$  and thus a total of  $N = p + q + 1$  model coefficients to be determined for a given pair  $(n_1, n_2)$ .

Replacing  $x(n_1, n_2)$  with  $\delta(n_1, n_2)$  and  $y(n_1, n_2)$  within (4) and noting that  $\sum_{(k_1, k_2) \in R_b} \sum b(k_1, k_2) \delta(n_1 - k_1, n_2 - k_2)$  is  $b(n_1, n_2)$ , we have

$$h_d(n_1, n_2) \leftarrow \sum_{(k_1, k_2) \in R_a - (0,0)} \sum a(k_1, k_2) h_d(n_1 - k_1, n_2 - k_2) + b(n_1, n_2). \quad (4)$$

As we wish to approximate  $h_d(n_1, n_2)$  as well as we can with  $h(n_1, n_2)$ , it is reasonable to define an error sequence  $e_M(n_1, n_2)$  as the difference between the left-hand and right-hand side expressions of (4)

$$e_M(n_1, n_2) = h_d(n_1, n_2) - \sum_{(k_1, k_2) \in \mathcal{R}_a - (0,0)} \sum a(k_1, k_2) h_d(n_1 - k_1, n_2 - k_2) - b(n_1, n_2). \quad (5)$$

It is clear that  $e_M(n_1, n_2)$  in (7) is not the same as  $e(n_1, n_2)$  in (3b). The subscript  $M$  in  $e_M(n_1, n_2)$  is used to emphasize that  $e_M(n_1, n_2)$  is a modification of  $e(n_1, n_2)$ . Minimizing  $e_M(n_1, n_2)$  with respect to the unknown coefficients  $a(k_1, k_2)$  and  $b(n_1, n_2)$  is a linear problem.

#### (A) Prony's method

In Prony's method, the error expression minimized:

$$\text{Error} = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} e_M^2(n_1, n_2), \quad (6)$$

where  $e_M(n_1, n_2)$  is given by (5). For practical computations, sums with a finite number of terms will be used.

The error in (6) is a quadratic form in the unknown parameters  $a(k_1, k_2)$  and  $b(n_1, n_2)$ . Careful observation of the error in (6) shows that it can be solved by first solving  $p$  linear equations for  $a(k_1, k_2)$  and then solving  $q + 1$  linear equations for  $b(n_1, n_2)$ . It is useful to rewrite (6) as

$$\text{Error} = E_1 + E_2, \quad (7a)$$

where

$$E_1 = \sum_{(n_1, n_2) \in \mathcal{R}_b} e_M^2(n_1, n_2), \quad (7b)$$

and

$$E_2 = \sum_{(n_1, n_2) \notin \mathcal{R}_b} e_M^2(n_1, n_2). \quad (7c)$$

The expression  $E_1$  in (7b) consists of  $q + 1$  terms, and  $E_2$  in (7c) consists of a large number of terms. Minimizing  $E_2$  in (7) with respect to  $a(k_1, k_2)$  results in  $p$  linear equations for  $p$  unknowns given by

$$\sum_{(k_1, k_2) \in \mathcal{R}_a - (0,0)} \sum a(k_1, k_2) r(k_1, k_2; l_1, l_2) = -r(0, 0; l_1, l_2), \quad (l_1, l_2) \in \mathcal{R}_a - (0, 0), \quad (8a)$$

where

$$r(k_1, k_2; l_1, l_2) = \sum_{(n_1, n_2) \notin \mathcal{R}_b} h_d(n_1 - k_1, n_2 - k_2) h_d(n_1 - l_1, n_2 - l_2). \quad (8b)$$

Once  $a(k_1, k_2)$  is determined, we can minimize the error in (7a) with respect to  $b(n_1, n_2)$ .

As Prony's method attempts to reduce the total square error, the resulting system is likely to be stable (Gonzalez and Woods, 2007).

#### (B) Iterative algorithm

The Iterative algorithm is an extension of a 1D system identification method developed by Steiglitz and McBride (Lim, 1990; Gonzalez and Woods, 2007).

From (6),  $e(n_1, n_2) = h_d(n_1, n_2) - h(n_1, n_2)$  is related to  $e_M(n_1, n_2)$  by

$$e_M(n_1, n_2) = a(n_1, n_2) * e(n_1, n_2). \quad (9)$$

Eq. (9) can be rewritten as

$$e(n_1, n_2) = v(n_1, n_2) * e_M(n_1, n_2). \quad (10)$$

The sequence  $v(n_1, n_2)$  is the inverse of  $a(n_1, n_2)$ .

From (5) and (10),

$$e(n_1, n_2) = v(n_1, n_2) * e_M(n_1, n_2) = v(n_1, n_2) * (a(n_1, n_2) * h_d(n_1, n_2) - b(n_1, n_2)). \quad (11)$$

From (11), if  $v(n_1, n_2)$  is somehow given, then  $e(n_1, n_2)$  is linear in both  $a(n_1, n_2)$  and  $b(n_1, n_2)$ , so minimization of  $\sum_{n_1} \sum_{n_2} e^2(n_1, n_2)$  with respect to  $a(n_1, n_2)$  and  $b(n_1, n_2)$  is a linear problem.

*Algorithm:*

- **Step 1:** We start with an initial estimate of  $a(n_1, n_2)$ , obtained using a method (e.g., Prony's).
- **Step 2:** We obtain  $v(n_1, n_2)$  from  $a(n_1, n_2)$ .
- **Step 3:** We minimize  $\sum_{n_1} \sum_{n_2} e^2(n_1, n_2)$  with respect to  $a(n_1, n_2)$  and  $b(n_1, n_2)$  by solving a set of linear equations.
- **Step 4:** We now have a new estimate of  $a(n_1, n_2)$ , and the process continues until we obtain the desired  $a(n_1, n_2)$  and  $b(n_1, n_2)$ .

#### 2.4. Frequency domain design by the iterative Semi-Definite Programming

Semi-Definite Programming (SDP) has recently attracted a great deal of research interest. Among other things, the optimization tool has been proven to be applicable to the design of various types of FIR digital systems. An attempt to extend the SDP approach to 2D IIR filters is made in Lu (2000). Throughout this section, the IIR systems are assumed to have separable denominators. This assumption simply imposes a constraint on the type of IIR systems being quadratically symmetric. Nevertheless, this class of systems is broad enough to cover practically all types of IIR systems that have been found useful in image/video processing (Lu, 2002).

Consider a quadratically symmetric 2D IIR digital system whose transfer function is given by

$$H(z_1, z_2) = \frac{B(z_1, z_2)}{A(z_1)A(z_2)}, \quad (12)$$

where  $B(z_1, z_2) = \sum_{k_1=0}^n \sum_{k_2=0}^n b(k_1, k_2) z_1^{-k_1} z_2^{-k_2}$  and  $A(z) = \sum_{k=0}^r a(k) z^{-k}$ ,  $a(0) = 1$ .

As the system is quadratically symmetric, we have  $b(k_1, k_2) = b(k_2, k_1)$ . As a result, there are only  $r + (n + 1)(n + 2)/2$  unknown variables in (12), which form a  $[r + (n + 1)(n + 2)/2]$ -dimension vector

$$x = [a_1 \cdots a_r b_{00} \cdots b_{nn} b_{10} b_{20} b_{21} \cdots b_{n0} \cdots b_{n, n-1}]^T, \quad (13)$$

where  $a(i) = a_i$ ,  $b(i, j) = b_{ij}$ . Denote the vector  $x$  in the  $k$ th iteration as  $x_k$  and the frequency response of the system for  $x = x_k$  as  $H(e^{j\omega_1}, e^{j\omega_2}, x_k)$ . In the neighborhood of  $x_k$ , the design variable can be expressed as  $x = x_k + \delta$ .

The transfer function can be approximated in terms of a linear function of  $\delta$  by

$$H(e^{j\omega_1}, e^{j\omega_2}, x) \approx H(e^{j\omega_1}, e^{j\omega_2}, x_k) + g_k^T \delta, \quad (14)$$

where  $g_k$  is the gradient of  $\text{de } H(e^{j\omega_1}, e^{j\omega_2}, x)$  for  $x = x_k$ .

#### 2.4.1. Problem formulation

The min-max design is obtained as a solution of the following optimization problem (Lu, 2000):

$$\text{Minimize } \hat{c}^T \hat{\delta}, \quad (15)$$

$$\text{Subject to } \begin{bmatrix} S_k & 0 \\ 0 & Y_k \end{bmatrix} \geq 0 \quad (16)$$

$$\text{with } \hat{c} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \hat{\delta} = \begin{bmatrix} \mu \\ \delta \end{bmatrix}, \quad (17)$$

where  $\mu$  is treated as an additional design variable, and

$$S_k = \text{diag}\{\Phi_k(\omega_1^{(1)}, \omega_2^{(1)}), \dots, \Phi_k(\omega_1^{(L)}, \omega_2^{(M)})\}, \quad (18)$$

$$\Phi_k(\omega_1, \omega_2) = \begin{bmatrix} I & Q_k^{\frac{1}{2}} \delta \\ \delta^T Q_k^{\frac{1}{2}} & \mu - 2\delta^T q_k - c_k \end{bmatrix} \geq 0, \quad (19)$$

$$Q_k = W(\omega_1, \omega_2) \Re e(\bar{g}_k g_k^T), \quad (20)$$

$W(\omega_1, \omega_2) \geq 0$  is a weighting function,  $\Re(\cdot)$  the real part of  $(\cdot)$ , and

$$q_k = W(\omega_1, \omega_2) \Re e\{[H(e^{j\omega_1}, e^{j\omega_2}, x_k) - H_d(\omega_1, \omega_2)] \bar{g}_k\}, \quad (21)$$

$$c_k = e(\omega_1, \omega_2, x_k). \quad (22)$$

$H_d(\omega_1, \omega_2)$  is the desired frequency response, for  $(\omega_1, \omega_2) \in \Omega$ , where  $\Omega = \{(\omega_1, \omega_2) : -\pi \leq \omega_1, \omega_2 \leq \pi\}$ , and

$$Y_k = \begin{bmatrix} P^{-1} - \tau I_r & D_k \\ D_k^T & P - \tau I_r \end{bmatrix} \geq 0, \quad (23)$$

where  $P$  is defined below, and  $\tau$  is a positive scalar that specifies the stability margin of the system, and

$$D_k = \begin{bmatrix} -(a_k + \delta_1)^T \\ \hat{I}_r \end{bmatrix}. \quad (24)$$

Denote the vectors formed from the first  $r$  components of  $x_k + \delta$  by  $a_k + \delta_1$ . As the denominator of  $H(z_1, z_2)$  is separable, it can be demonstrated that the IIR system with coefficient vector  $x_k + \delta$  is stable if and only if the magnitudes of the eigenvalues of matrices  $D_k$  are strictly less than one, where  $\hat{I}_r$  denotes a matrix of size  $(r-1) \times r$  obtained by augmenting the identity matrix with a zero column on the right. Applying the well-known Lyapunov theory (Kailath, 1981), one concludes that matrix  $D_k$  is stable if and only if there exists a positive definite matrix  $P$  such that

$$P - D_k^T P D_k \succ 0, \quad (25)$$

where  $M \succ 0$  denotes that matrix  $M$  is positive definite. The matrix  $P$  in (23) is not considered as a design variable. Rather, this positive definite matrix is fixed in each iteration and can be obtained by solving the Lyapunov equation

$$P - \hat{D}_k^T P \hat{D}_k = I, \quad (26)$$

where

$$\hat{D}_k = \begin{bmatrix} -a_k^T \\ \hat{I}_r \end{bmatrix}. \quad (27)$$

With  $P$  fixed in  $Y_k$ , the minimization problem in (15), (16) is an SDP problem of size  $1 + r + 0.5(n+1)(n+2)$ .

#### 2.4.2. Design steps

**Input:** The order of the IIR system  $(n, r)$ , the desired frequency response  $H_d(\omega_1, \omega_2)$ , and  $W(\omega_1, \omega_2)$ .

- **Step 1:** The proposed design method starts with an initial point  $x_0$  that corresponds to a stable system obtained using a conventional method.
- **Step 2:** With this  $x_0$ , a positive definite matrix  $P$  can be obtained by solving the Lyapunov Eq. (26), and the quantities  $Q_k$ ,  $q_k$ , and  $c_k$  can be evaluated by using (20)–(22).
- **Step 3:** Next, we solve the SDP problem in (15), (16).
- **Step 4:** The obtained solution  $\hat{x}^* = [\mu^* \quad \delta^{*T}]^T$  can be used to update  $x_0$  to  $x_1 = x_0 + \delta^*$ . The iteration continues until  $\|\delta^*\|$  is less than a prescribed tolerance  $\epsilon$ .

### 3. 2D digital system model reduction

It is often desirable to represent a high order system with a lower order system. A suitable model reduction procedure should provide a model that approximates the original well. It should produce stable models from a stable original, and it should be able to be implemented on a computer with high computational efficiency and reduced memory requirements.

The reduction of models in the state space (SS) realization environment has definite advantages. It is possible to apply the vast knowledge of matrix theory in the analysis, whereas the non-uniqueness of SS realization allows us to choose one that is better suited for the purpose at hand (Premaratne et al., 1990).

#### 3.1. 2D State-space models

Assuming that  $H(z, w) = \sum_{i=0}^{+\infty} \sum_{j=0}^{+\infty} h(i, j) z^{-i} w^{-j}$  is the transfer-function of a discrete 2D IIR filter of order  $(m, n)$ , where  $z^{-1}$  and  $w^{-1}$  are unit backward operators,  $H(z, w)$  can be written in the form

$$H(z, w) = \frac{\sum_{i=0}^m \sum_{j=0}^n h(i, j) z^{m-i} w^{n-j}}{\prod_{i=1}^m (z - z_i) \prod_{j=1}^n (w - w_j)}$$

This indicates that the 2D IIR filters belong to the class of filters with separable denominator, i.e., the denominator polynomial with two independent variables of the transfer-function of these filters can be written as a product of two polynomials, each dependent on a single variable only. The transfer-function of these filters is expressed as follows  $H(z, w) = \frac{N(z, w)}{D_1(z) D_2(w)}$ .

Any causal 2D system having a transfer-function with a separable denominator can be modeled in the local state-space Roesser's characterization in the form (Premaratne et al., 1990; Adamou-Mitiche et al., 2013; Adamou-Mitiche and Mitiche, 2013)

$$x(i, j) = \begin{bmatrix} x^h(i, j) \\ x^v(i, j) \end{bmatrix}, \quad (28a)$$

where  $x$  is the local state;  $x^h$ , an  $n$ -vector, is the horizontal state;  $x^v$ , an  $m$ -vector, is the vertical state; and

$$\begin{bmatrix} x^h(i+1,j) \\ x^v(i,j+1) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(i,j), \quad (28b)$$

$$y(i,j) = [C_1 \ C_2] \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + du(i,j), \quad (i,j) \geq (0,0), \quad (28c)$$

where  $u$ , the input, is an  $l$ -vector, and  $y$ , the output, is a  $p$ -vector. Clearly,  $x^h$ , the horizontal state, is propagated horizontally, and  $x^v$ , the vertical state, is propagated vertically by first-order difference equations.

The 2D transfer function can be written as

$$H(z_1, z_2) = C \left[ \begin{pmatrix} z_1 I & 0 \\ 0 & z_2 I \end{pmatrix} - A \right]^{-1} B + d, \quad (29)$$

where  $A$ ,  $B$ , and  $C$  are the block matrices in (28b) and (28c).

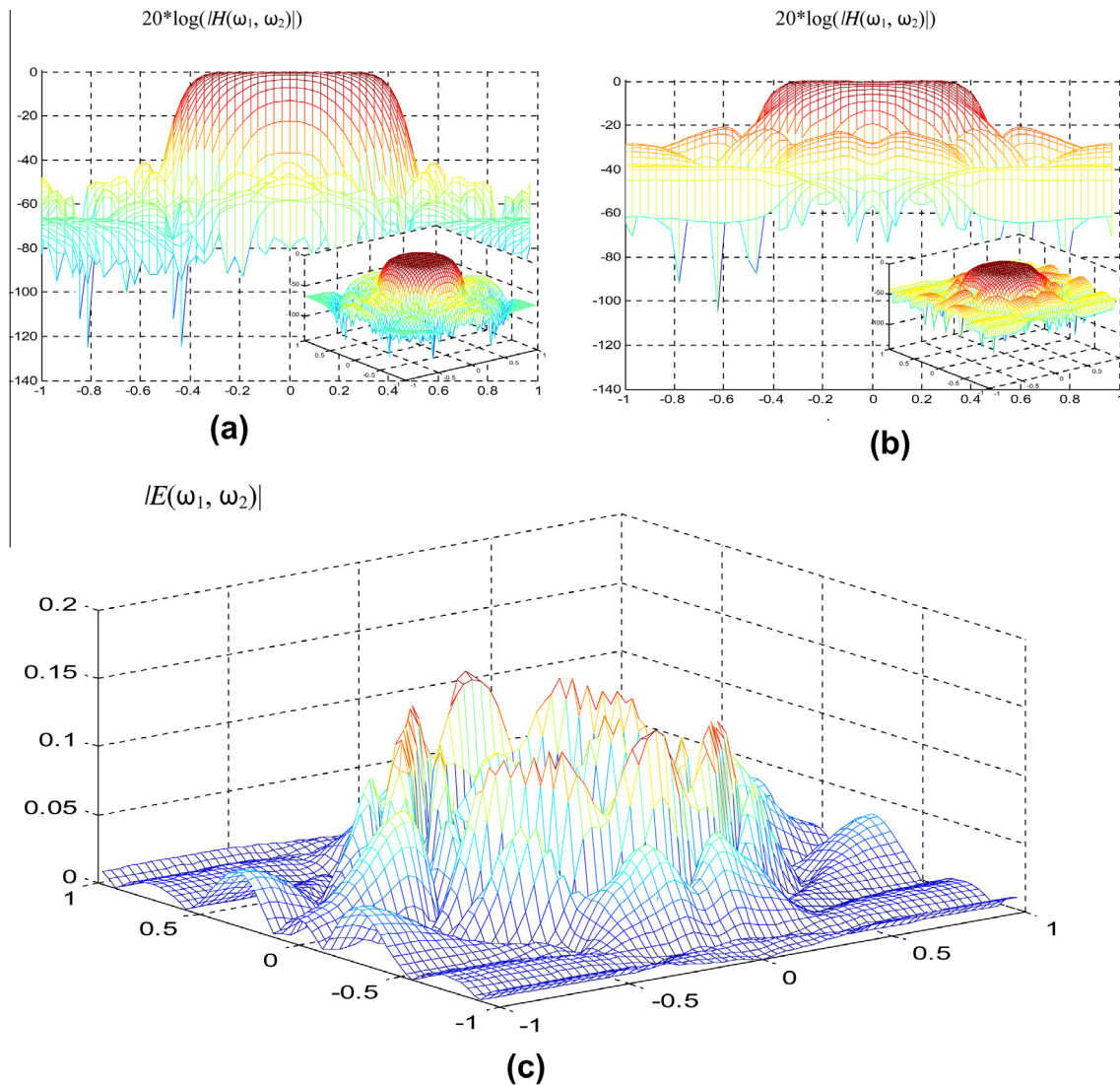
It is clear that there is a one-to-one correspondence between Roesser's model and circuit implementations with delay elements  $z^{-1}$  and  $z_2^{-1}$ .

Notice that the minimal state-space realizations are not possible for all 2D transfer functions (Kung et al., 1977; Attasi, 1975). However, minimal state-space realizations have been determined for a system with separable denominator (Givone and Roesser, 1973; Antoniou et al., 1988).

The state-space model sought is of the Givone–Roesser type (Shanks et al., 1972), described with  $l = p = 1$ , and  $A_1, A_2, A_3, A_4, B_1, B_2, C_1$  and  $C_2$  of dimension  $(n \times n), (n \times m), (m \times n), (m \times m), (n \times 1), (m \times 1), (1 \times n)$  and  $(1 \times m)$ , respectively.

### 3.2. Model order reduction methods

The most popular 1D model reduction techniques are based on the concept of balanced realization, which was originally proposed by Moore (Moore, 1981). Given a discrete system, its balanced realization describes the system in a state space



**Figure 1** Low pass IIR filter. (a) Magnitude of the original  $5 \times 5$  filter via Iterative method. (b) Magnitude of the reduced-order  $4 \times 5$  filter. (c) Spectral error between original and reduced filter.

representation in which the importance of the  $i$ th state variable can be measured by the  $i$ th Hankel singular value of the system. This suggests that one way of obtaining a low-order approximation of a state-space model is to form a balanced realization (Laub, 1980) and then to retain those states corresponding to the  $r$  largest Hankel singular values, where  $r$  is the order of the reduced-order system.

One of the problems in the study of 2D model reduction is the extension of the 1D model reduction algorithms to 2D models (Xiao et al., 1998, 2001). In our case, the balanced realization concept is extended to the 2D case. As a balanced realization is essentially determined by the controllability and observability Gramians of the system, and as there are several types of Gramians of the system that can be properly defined for a given 2D system, there are different types of balanced realizations for a 2D discrete system, leading to different balanced approximations (Lu et al., 1987, 1996).

Consider the Givone–Roesser state space model of a Single Input Single Output (SISO) system described in (28), the controllability and observability quasi-Gramians (Wang et al., 1991; Zhou et al., 1994) are defined by the positive definite block-diagonal matrices  $P^q = \text{diag}(P_1, P_2)$  and  $Q^q = \text{diag}(Q_1, Q_2)$ , where  $P_i$  and  $Q_i$  ( $i = 1, 2$ ) satisfy the Lyapunov equations

$$A_1 P_1 A_1^T - P_1 + B_1 B_1^T + A_2 P_2 A_2^T = 0,$$

$$A_4 P_2 A_4^T - P_2 + B_2 B_2^T + A_3 P_1 A_3^T = 0,$$

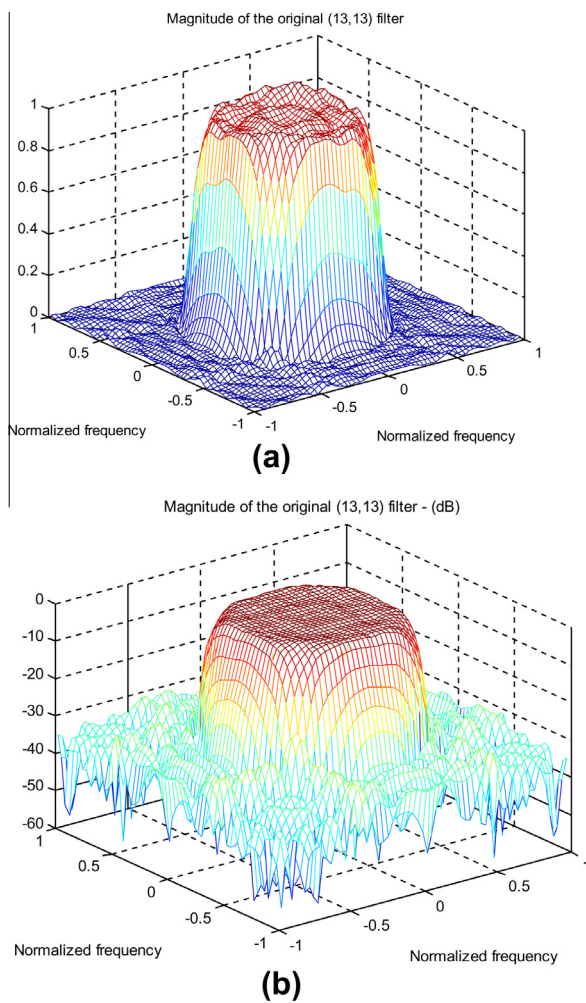
$$A_1^T Q_1 A_1 - Q_1 + C_1^T C_1 + A_3^T Q_2 A_3 = 0,$$

$$A_4^T Q_2 A_4 - Q_2 + C_2^T C_2 + A_2^T Q_1 A_2 = 0.$$

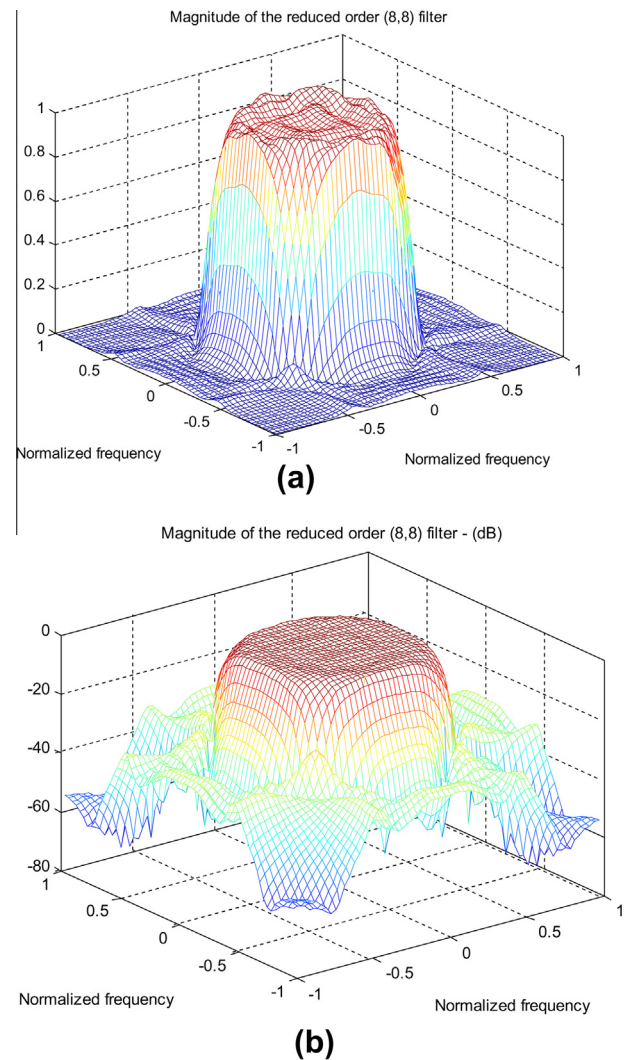
### 3.2.1. Balanced realization approximation

The upper left and lower right diagonal blocks of the observability and controllability quasi-Gramians are used to compute the transformation matrix  $T = T_1 \oplus T_2$  by using, for example, Laub’s algorithm (Zhou et al., 1994), such that the realization characterized by  $(T^{-1}AT, T^{-1}B, CT, d)$  is balanced. A reduced-order system of order  $(r_1, r_2)$ , denoted by  $(A_r, B_r, C_r, d)$ , can be obtained by truncating the matrices  $A, B$ , and  $C$  as

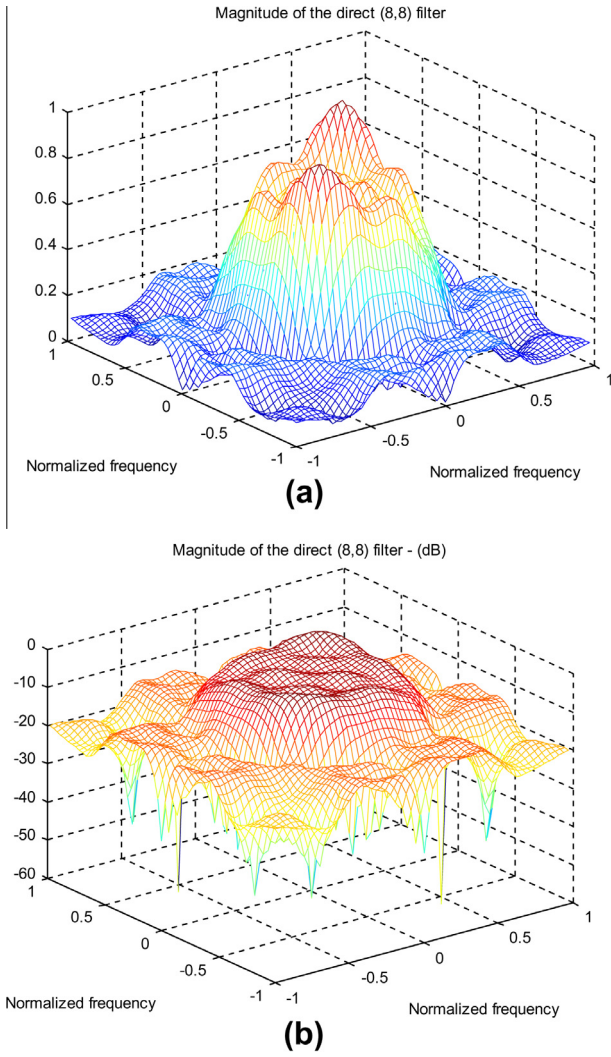
$$A_r = \begin{bmatrix} A_{1r} & A_{2r} \\ A_{3r} & A_{4r} \end{bmatrix}, B_r = \begin{bmatrix} B_{1r} \\ B_{2r} \end{bmatrix}, C_r = [C_{1r} \quad C_{2r}],$$



**Figure 2** Low pass IIR filter. (a) Magnitude of the original  $13 \times 13$  filter via SDP method. (b) Magnitude-dB of the original  $13 \times 13$  filter via SDP method.



**Figure 3** Low pass IIR filter. (a) Magnitude of the reduced order  $8 \times 8$  filter. (b) Magnitude-dB of the reduced order  $8 \times 8$  filter.



**Figure 4** Low pass IIR filter. (a) Magnitude of the direct SDP  $8 \times 8$  filter. (b) Magnitude-dB of the direct SDP  $8 \times 8$  filter.

where

$$A_{1r} = A_1(1:r_1, 1:r_1), A_{2r} = A_2(1:r_1, 1:r_2), A_{3r} = A_3(1:r_2, 1:r_1), \\ A_{4r} = A_4(1:r_2, 1:r_2), B_{1r} = B_1(1:r_1), B_{2r} = B_2(1:r_2), C_{1r} = C_1(1:r_1), \\ \text{and } C_{2r} = C_2(1:r_2) \text{ (Lu et al., 1987).}$$

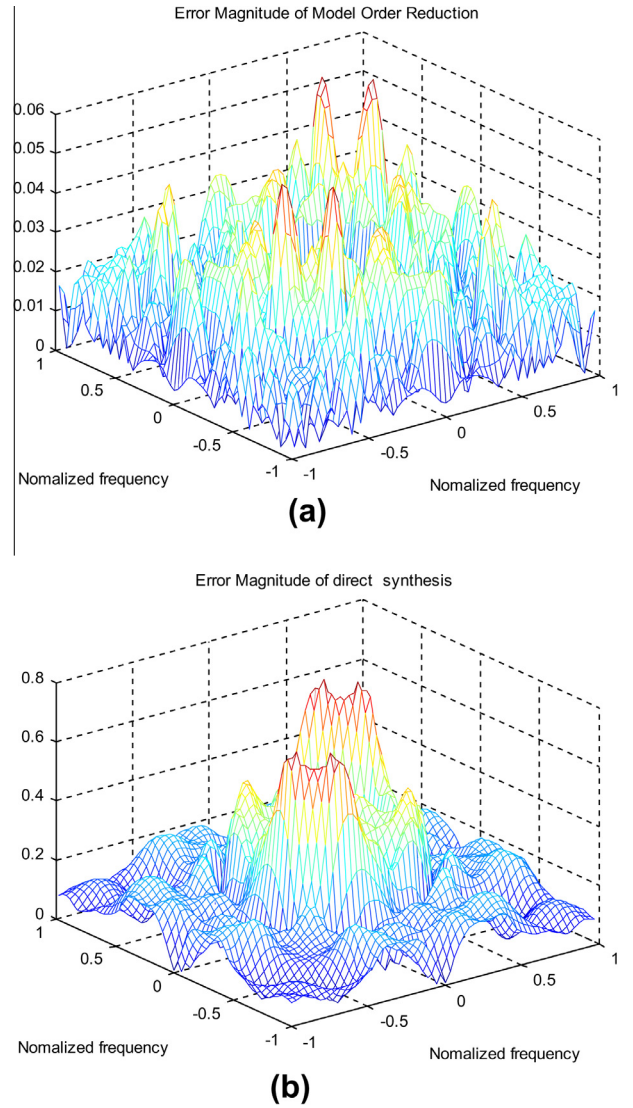
### 3.2.2. Iterative Algorithm for computing the quasi-Gramians

An Iterative method for the computation of quasi-Gramians is described, where each iteration involves solving two 1D Lyapunov equations. For a 2D stable system, the algorithm converges very quickly to the 2D quasi-Gramians (Luo et al., 1993).

- **Step 1:** Set  $P_2^{(0)} = Q_2^{(0)} = 0$ , and  $k = 1$ .
- **Step 2:** Solve the following 1D Lyapunov equations for  $P_1^{(k)}$  and  $Q_1^{(k)}$

$$A_1 P_1^{(k)} A_1^T - P_1^{(k)} + F_1 = 0, \quad (30a)$$

$$A_1^T Q_1^{(k)} A_1 - Q_1^{(k)} + G_1 = 0, \quad (30c)$$



**Figure 5** (a) Error Magnitude of model order reduction. (b) Error Magnitude of direct synthesis.

where  $F_1 = B_1 B_1^T + A_2 P_2^{(k-1)} A_2^T$ ,

$$G_1 = C_1^T C_1 + A_3^T Q_2^{(k-1)} A_3.$$

- **Step 3:** Solve the following 1D Lyapunov equations for  $P_2^{(k)}$  and  $Q_2^{(k)}$ :

$$A_4 P_2^{(k)} A_4^T - P_2^{(k)} + F_2 = 0, \quad (30b)$$

$$A_4^T Q_2^{(k)} A_4 - Q_2^{(k)} + G_2 = 0, \quad (30d)$$

where  $F_2 = B_2 B_2^T + A_3 P_1^{(k)} A_3^T$ ,

$$G_2 = C_2^T C_2 + A_2^T Q_1^{(k)} A_2.$$

- **Step 4:** Set  $k = k + 1$ , and repeat Step 2 and Step 3 until

$$\|P_i^{(k)} - P_i^{(k-1)}\| < \varepsilon, \quad (i = 1, 2),$$



$$\|Q_i^{(k)} - Q_i^{(k-1)}\| < \varepsilon, \quad (i = 1, 2),$$

where  $\varepsilon$  is a prescribed tolerance (Luo et al., 1993).

The overall algorithm can be summarized as follows:

- (1) Use a design method (Prony's, Iterative, SDP) to design a system satisfying the design specifications.
- (2) Apply a model order reduction procedure to obtain a low-complexity system.

#### 4. Illustrative simulations

We divided the simulation into two parts, 2D-filter design and order reduction.

The interpretation of the results is given at the end of this section.

##### 4.1. Part 1: 2D-filter design

The design was performed in two domains.

- For the spatial domain, two methods were used: Prony's method and the Iterative method.

The numerator  $b(n_1, n_2)$ , and the denominator  $a(n_1, n_2)$  matrices were generated, then, we used the function `Impulse_2D.m(E)` to produce the impulse response and frequency response.

<sup>(E)</sup>The function  $h = \text{impulse\_2D}(N, M, a, b)$  computes the 2D impulse response of the 2D IIR filters. The input variables  $N, M, a$ , and  $b$ , are, respectively, the number of rows, number of columns of  $h(\cdot)$ , the numerator polynomial  $A(\cdot)$ , and the denominator polynomial  $B(\cdot)$ . The function output variable  $h(\cdot)$  is the impulse response of the 2D filter.

The order of the original IIR low pass filter is  $(n, m) = (5, 5)$ , and the corresponding pass band and stop band are given by  $R_p = [0 \ 0.4]$ , and  $R_s = [0.5 \ 1]$ .

- For the frequency domain, we used Semi-Definite Programming (SDP) to do the same.

The order of the original IIR low pass filter is  $(n, m) = (13, 13)$ , and the corresponding pass band and stop band are given by  $R_p = [0 \ 0.4]$ , and  $R_s = [0.5 \ 1]$ , and number of iterations is  $it = 2$ .

##### 4.2. Part 2: Order reduction

We applied the method of quasi-Gramians to the low pass filters designed in the first part. First, we used our function `tf2ss_2D.m` to transform the transfer function matrices  $a$  and  $b$  to the state-space model  $(A, B, C, d)$ , and then we applied the quasi-Gramian method to produce a reduced-order model  $(A_r, B_r, C_r, d)$  (see Figures 1–7).

For the Iterative method, the order of original filter is  $n = 5$  and  $m = 5$ , and the total number of coefficients is  $(5 \times 5_{\text{(matrix } b)} + 5 \times 5_{\text{(matrix } a)}) \times 4 = 200$ .

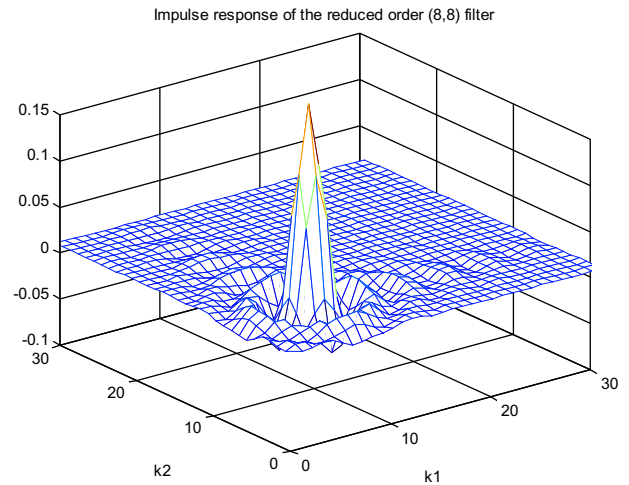


Figure 6 Impulse response of the reduced order  $8 \times 8$  filter.

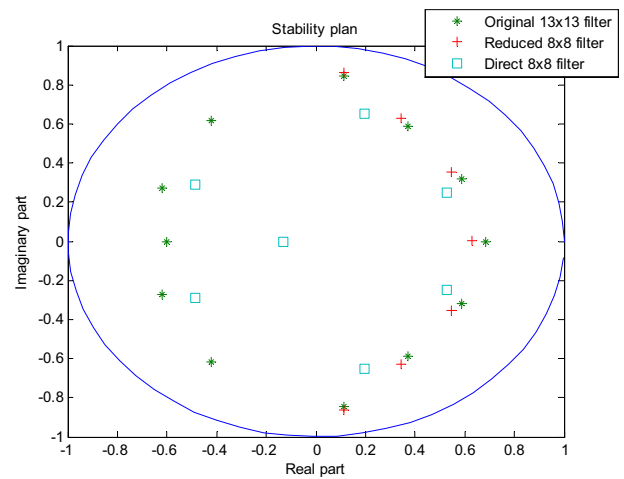


Figure 7 Stability plan.

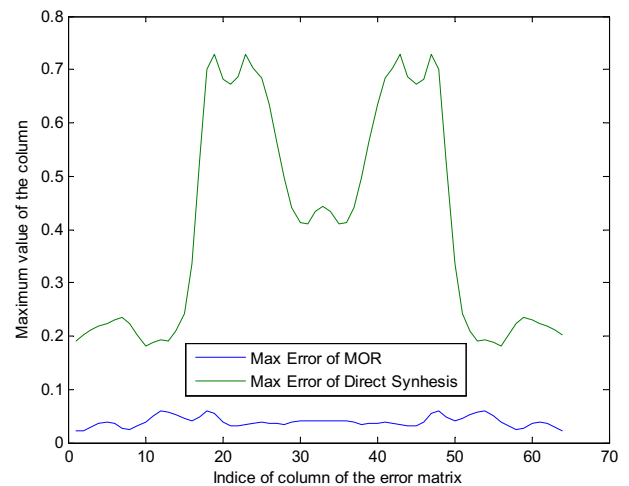


Figure 8 Maximum error distribution of the direct synthesis and the proposed approach.

**Table 1** Pole magnitude of the direct and the reduced order filters.

	$ p_1 $	$ p_2 $	$ p_3 $	$ p_4 $	$ p_5 $	$ p_6 $	$ p_7 $
Direct synthesis	0.6783	0.6783	0.5846	0.5846	0.5661	0.5661	0.1291
Proposed approach	0.8674	0.8674	0.7138	0.6507	0.6299	0.7138	0.6507

The dimensions of the matrices  $A$ ,  $B$ , and  $C$  are  $(n + 2 \times m) \times (n + 2 \times m)$ ,  $(n + 2 \times m) \times 1$ , and  $1 \times (n + 2 \times m)$ , respectively, and the number of iterations is  $it = 10$ .

The order of reduced filter is  $r_1 = 4$  and  $r_2 = 5$ , and the total number of coefficients is  $(4 \times 5$  (matrix  $b$ ) +  $4 \times 5$  (matrix  $a$ ))  $\times 4 = 160$ .

For the SDP method, the order of original filter is  $(n, m) = (13, 13)$ , and the total number of coefficients is  $13 \times 13$  (matrix  $b$ ) +  $13 \times 13$  (matrix  $a$ ) = 338.

The dimensions of the matrices  $A$ ,  $B$ , and  $C$  are  $(n + m) \times (n + m)$ ,  $(n + m) \times 1$ , and  $1 \times (n + m)$ , respectively, and the number of iterations is  $it = 2$ . The order of the reduced filter is  $r_1 = 8$  and  $r_2 = 8$ , and the total number of coefficients is:  $8 \times 8$  (matrix  $b$ ) +  $8 \times 8$  (matrix  $a$ ) = 128.

To thoroughly evaluate the performance of our synthesized filter by order reduction, we synthesized a direct filter of order (8, 8), following the same frequency specifications (template) using the SDP method, and we compared the two filters.

#### 4.3. Interpretation

For the design step, there was not much difference between the Prony's and the Iterative methods. However, we did observe that there is a small improvement for the stop band (attenuation) when the Iterative method is applied. For the Prony's approach,  $\min(H(\omega_1, \omega_2)) = -49.09$  dB, and  $\min(H(\omega_1, \omega_2)) = -55.77$  dB for the Iterative method.

The results obtained with the SDP method are comparable to the other two methods, but we found a performance decrease in both bands (pass band and stop band). It is possible to improve the results by increasing the number of coefficients ( $n$ ,  $r$ ) or the number of iterations. A good aspect of this method is the stability, which can be verified by the stability criterion, i.e.,  $\max(\text{abs}(\text{roots}(a(:, 1)))) < 1$ , where ( $a$ ) is the denominator matrix.

In the proposed example, we found  $\max(\text{abs}(\text{roots}(a(:, 1)))) = 0.8926$ , where  $\max$ ,  $\text{abs}$ , and  $\text{roots}$  are MATLAB functions (**MATLAB**).

For the order reduction step, we notice that the filter designed by the Prony's and Iterative methods has a non-minimal realization, and the reduced filter can be unstable (see **Figure 1**), the pass band and stop band errors of the reduced order filter are higher than those of the original filter. Using the SDP method, the stability of the reduced order filter is always preserved. In the proposed example,  $\max(\text{abs}(\text{roots}(a(:, 1)))) = 0.8926$  for the original filter, and  $\max(\text{abs}(\text{roots}(a_r(:, 1)))) = 0.9119$  for the reduced order filter.

The results obtained indicate that the reduced low pass filter is acceptable. Note that the number of coefficients decreases from 338 to 128, and the max error between the reduced and original filter is  $\max(E) \leq 0.06$  (cf. **Figure 5a**). **Figure 8** shows the distributions of the maximum error via the direct synthesis, and the proposed approach.

For the (8, 8) reduced-order filter, obtained by the order reduction of the (13, 13) filter synthesized by SDP, we note that

it closely follows the frequency behavior of the original filter of complete order (see **Figures 3 and 8**). The reduced order filter is always stable (see **Figures 6 and 7**).

To highlight the reduced-order filter using our proposed approach, we synthesized another filter of the same order (8, 8) directly using the SDP method (see **Figure 4**). It is clear that the frequency response of the (8, 8) filter via model order reduction fit the desired specification better than the one designed directly (see **Figure 5**).

Another important result, demonstrating the performance of our filters resulting from the order reduction, is that the poles (see **Figure 7** and **Table 1**) are closer to the unit circle, which is a characterization of the filter selectivity.

## 5. Conclusion

In this paper, we present a new order reduction method for 2D digital systems synthesis.

In a first step, we designed full order 2D IIR systems using two methods (Iterative and SDP methods).

After various simulations, the SDP technique was retained because it always yields a stable filter.

In the second step, order reduction based on the quasi-Gradients of the original filter was achieved. The approximate reduced order filter presents some interesting key characteristics, such as the stability and the perfect frequency fitting of the original filter behavior. This filter is better than the filters (of the same order) synthesized directly by SDP method. This superiority was proven by several dynamical system simulations.

The superiority of our model obtained by model order reduction is justified by the fact that in the order reduction operations, the states of initial models with small contribution to the complete behavior of the filter are eliminated, and only the dominant states are kept.

## Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions for improving the quality of the paper. They are also grateful to Dr. Mansour Alsulaiman, Editor-in-Chief, for handling our paper.

## References

- Adamou-Mitiche, A.B.H., Mitiche, L., 2013. A novel algorithm for 2D-IIR filters synthesis via 2D-FIR filters model reduction. *WSEAS Trans. Signal Process.* 9 (4), E-ISSN: 2224-3488.
- Adamou-Mitiche, A.B.H., Mitiche, L., Sima, V., 2013. On the synthesis of digital two dimensional filters based on FIR filters approximation. In: *IEEE 5th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO)*, Tunisia, pp. 1-6, 28-30. doi: 10.1109/ICMSAO.2013.655266, Print ISBN:978-1-4673-5812-5.

- Antoniou, Andreas, 2001. *Digital Filters: Analysis, Design and Applications*. Tata McGraw-Hill Education.
- Antoniou, G.E., Paraskevopoulos, P.N., Varoufakis, S.J., 1988. Minimal state-space realization of factorable 2-D transfer function. *IEEE Trans. Circuits Syst.* 35, 1055–1058.
- Attasi, S. Modélisation et traitement des suites à deux indices, IRIA, Rapport Lab, Sept. 1975.
- Givone, D., Roesser, R.P., 1973. Minimization of multidimensional linear iterative circuits. *IEEE Trans. Comput. C-22*, 673–678.
- Gonzalez, R.C., Woods, R.E., 2007. *Digital Image Processing*, third ed. Prentice Hall.
- Kailath, T., 1981. *Linear Systems*. Prentice-Hall.
- Kung, S.Y., Levi, B.C., Morf, M., Kailath, T., 1977. New results in 2-D systems theory, Part II: 2-D state-space models-realization and the notions of controllability, observability, and minimality. *Proc. IEEE* 65, 945–961.
- A. J. Laub, On computing balancing transformations. In: *Proc. JACC*, FA 8-E, San Francisco, 1980.
- Lim, J.S., 1990. *Two-Dimensional Signal and Image Processing*. Prentice Hall.
- Lu, W.S., 2000. Design of stable 2-D IIR digital filters using iterative semi definite programming, In: *Proc. ISCAS'00*, Geneva, Switzerland.
- Lu, W.S., 2002. A unified approach for the design of 2-D digital filters via semi definite programming. *IEEE Trans. Circuits Syst.* 49, 814–826.
- Lu, W.-S., Lee, E.B., Zhang, Q.T., 1987. Balanced approximation of two-dimensional and delay differential systems. *Int. J. Control* 47, 2199–2218.
- Lu, W.S., Luo, H., Antonio, A., 1996. Recent results on model reduction methods for 2-D discrete systems. *IEEE Int. Symp. Circuits Syst.*, 348–351.
- Luo, H., Lu, W. S., Antoniou, A., 1993. Numerical solution of the two-dimensional Lyapunov equations and application in order reduction of recursive digital filters. In: *Proc. IEEE Pacific Rim Con & Commun., Process.*, Victoria, B.C., Canada, pp. 232–235.
- Mandal, Sangeeta, Ghoshal, Sakti Prasad, Kar, Rajib, Mandal, Durbadal, 2012. Design of optimal linear phase FIR high pass filter using craziness based particle swarm optimization technique. *J. King Saud Univ. Comput. Inf. Sci.* 24 (1), 83–92. <http://dx.doi.org/10.1016/j.jksuci.2011.10.007>, ISSN 1319–1578.
- MATLAB, The Matworks™, Ver. (R2009b).
- Moore, B.C., 1981. Principal component analysis in linear systems: Controllability, observability and model reduction. *IEEE Trans. Automat. Control* 26, 17–32.
- Premaratne, K., Jury, E.I., Mansour, M., 1990. An algorithm for model reduction of 2-D discrete time systems. *IEEE Trans. Circuits Syst.* 38, 1116–1132.
- Ramamoorthy, P.A., Bruton, L.T., 1979. Design of stable two-dimensional analogue and digital filters with applications in image processing. *Int. J. Circ. Theor. Appl.* 7, 229–245. <http://dx.doi.org/10.1002/cta.4490070210>.
- Salam, Muhammad Tariqus, 2011. *Two-dimensional Digital Filters for Image Processing Applications: Design, Implementation, and Validation*. LAP LAMBERT Academic Publishing, ISBN-13: 978-3845436180.
- Shanks, J.L., Treitel, S., Justice, J.H., 1972. Stability and synthesis of tow-dimensional recursive filters. *IEEE Trans. Audio Electroacoust. AU-20*, 115–128.
- Sontag, E.D., 1978. On first order equations for multidimensional filters. *IEEE Trans. Acoust. Speech Signal Process. ASSP-26*, 480–482.
- Wang, W., Glover, K., Doyle, J.C., Beck, C., 1991. Model reduction of LFT systems”, In: *Proc. 30th IEEE Conf. on Decision and Control*, Brighton, UK, 1991, pp. 1233–1238.
- Xiao, C., Sreeram, V., Liu, W.Q., Venetsanopoulos, A.N., 1998. Identification and model reduction of 2-D systems via the extended impulse response Gramians. *Automatica* 34 (1), 93–101.
- Xiao, C., Olivier, J.C., Agathoklis, P., 2001. Design of linear phase IIR filters via weighted least-squares approximation. *ICASSP 6*, 3817–3820.
- Zhou, K., Aravena, J.L., Gu, G., Xiong, D., 1994. Two-dimensional system model reduction by quasi-balanced truncation and singular perturbation. *IEEE Trans. Circuits Syst.* 41, 593–602.