Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Model Reference Adaptive Control of 2-D Discrete Systems with Unbounded Variables along Two Dimensions

Ali Madady (Islamic Azad University of Islamshahr)

Abstract—**Adaptive control is an effective method of controlling unknown dynamical systems. While many research results on one-dimensional (1-D) adaptive control are available, little has been accomplished in the area of 2-D system theory. The main reason is due primarily to the difficult algebra of 2-D systems and the complexity of the underlying theory. In particular, when both independent variables in the 2-D space are unbounded, the problem is very involved. In this paper, we propose a model reference adaptive control scheme for 2-D discrete systems which are described by Roesser state space model and their both independent variables are unbounded. The input of the underlying 2-D system is assigned according to a closed-loop control law incorporating the system state and the reference model state as well as input. In this closed-loop control law, certain feedback gains are fixed, but others are adjustable. Those adjustable feedback gains are updated two-dimensionally subsequently, utilizing the gradient approach and based on the error between the actual system and its corresponding reference model. The stability of the presented 2-D model reference adaptive control (2-DMRAC) system is analyzed.**

I. Introduction

In the past three decades, a growing interest has developed in two-dimensional (2-D) systems [1-3]. The evolution of 2-D systems theories from the concepts and results in 1-D systems is well known. Correspondingly, such topics as modeling, stability, stabilization by the state and output feed-back, controllability and observability, pole placement and model matching, optimal control problems, observer and state estimation, transfer function identification widely studied in 1-D systems are studied in 2-D systems as well [4,5]. Many applications in image processing, 2-D filter design, iterative learning and some industrial processes modeling and control are reported [5-8]. However, there are many unsolved problems in 2-D systems in contrast with their corresponding 1-D case.

In this paper, we consider the designing 2-D control systems. The various approaches as stabilization by state and output feedback [9,10], pole assignment [11-13], exact model matching and model following [14-16], optimal control methods [17,18] and robust control [19,20] are studied in this category.

Adaptive control is an effective method of controlling unknown dynamical systems. Similar to the 1-D case [21-23], we can use the adaptive techniques to control the 2-D unknown systems. In the 1-D case, adaptive control systems are accomplished either by self-tuning or model reference approaches. There, the self-tuning adaptive control systems are constructed by combination of an on-line procedures to estimate the system unknown parameters and a traditional control policy. While in the model reference approach the parameter estimation is not accomplished but the system input is assigned by an adaptive control law which includes adjustable gains, and these gains are updated so that the system follows a given model.

These rather classical 1-D approaches can be extended to the 2-D systems. Indeed, it is possible to propose both the self-tuning and model reference ideas for controlling the 2-D unknown dynamical systems. For instance, a 2-D self-tuning adaptive controller can be obtained by combining a 2-D on-line parameter estimation algorithm and any 2-D classical controller, as shown in [24]. Similarly, we may determine the system input by an adaptive control law so that the system output follows a given reference model, without estimating the system unknown parameters.

Several on-line parameter estimation algorithms are presented for 2-D systems [25,26], which in order to construct the corresponding 2-D self-tuning adaptive controller we can combine any kind of them with any kind of the 2-D traditional controllers.

In [24], algorithms are given for setting in the self-tuning mode by combining them with recursive parameter estimation, and forgetting factors are given for these algorithms. But no convergence and stability analyses are given to these algorithms and also one of the independent variables of the underlying 2-D systems is assumed to be bounded. In [27] and [28], the problem of model reference adaptive control for 2-D systems is investigated, but there exist two limitations in these approaches. First, the parameters of the under control 2-D system are assumed to be known and hence the formulated problem is not truly an adaptive one. Secondly, one of the independent variables of the underlying 2-D systems is bounded. In [29], a similar problem is formulated and solved for 2-D continuous-time systems in a special case, in which one of the 2-D space coordinates is bounded. The self-tuning adaptive control of a simple class of a 2-D shift-invariant system which is finite along one dimension and infinite along the other dimension are studied in [30].

However, since one of the independent variables of the underlying 2-D systems is assumed to be bounded in all above investigations, these are not applicable in the general case where the both independent variables are unbounded.

Author is with the Islamic Azad University of Islamshahr (as a part time faculty member), Islamshahr, IRAN (e-mail: drmadady@yahoo.com).

The aim of this paper is to extend the model reference adaptive control approach to the 2-D case when both of the independent variables of the underlying 2-D system are unbounded. The paper is organized as follows. Section 2 formulates our 2-D model reference adaptive control (2-DMRAC) problem. In section 3, we solve the 2-DMRAC problem. The stability is analyzed in section 4. Conclusion is deferred to section 5.

II. POBLEM STATEMENT

The most popular two-dimensional state space models are the discrete models which are proposed by Roesser [1], Fornasini-Marchesini [2] and Kurek [3]. In this paper we consider the Roesser model. In the Roesser model the local state of the system is decomposed into two components, namely the *horizontal* state x^h and the *vertical* state x^v. This model is as follows [1].

$$\begin{bmatrix} x^h(i+1,j) \\ x^v(i,j+1) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(i,j) \qquad i,j=0,1,... \tag{1}$$

where i and j are non-negative integer-valued horizontal and vertical coordinates; $x^h \in \mathbb{R}^{m^h}$ and $x^v \in \mathbb{R}^{m^v}$ are the state components which are propagated respectively horizontally and vertically by a set of first-order difference equations; $u \in \mathbb{R}$ is the control input. The coefficient matrices A_k (for $k=1,2,3,4$) and B_l (for $l=1,2$) are real with appropriate dimensions. The global boundary conditions for (1) are given by:

$$x^h(0,j) \ and \ x^v(i,0) \quad i,j=0,1,... \tag{2}$$

We assume that the following reference model is given:

$$\begin{bmatrix} x_0^h(i+1,j) \\ x_0^v(i,j+1) \end{bmatrix} = \begin{bmatrix} A_{01} & A_{02} \\ A_{03} & A_{04} \end{bmatrix} \begin{bmatrix} x_0^h(i,j) \\ x_0^v(i,j) \end{bmatrix} + \begin{bmatrix} B_{01} \\ B_{02} \end{bmatrix} r(i,j) \qquad i,j=0,1,... \tag{3}$$

where $x_0^h \in \mathbb{R}^{m^h}$, $x_0^v \in \mathbb{R}^{m^v}$ and $r \in \mathbb{R}$.

For simplicity, we choose the following notations:

$$m = m^h + m^v \ , \ q = m^h + m^v + 1$$

$$A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix}, B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, A_0 = \begin{bmatrix} A_{01} & A_{02} \\ A_{03} & A_{04} \end{bmatrix}, B_0 = \begin{bmatrix} B_{01} \\ B_{02} \end{bmatrix}$$

and:

$$x(i,j) = \begin{bmatrix} x^h(i,j) \\ x^v(i,j) \end{bmatrix}, x_0(i,j) = \begin{bmatrix} x_0^h(i,j) \\ x_0^v(i,j) \end{bmatrix}$$
$$x'(i,j) = \begin{bmatrix} x^h(i+1,j) \\ x^v(i,j+1) \end{bmatrix}, x_0'(i,j) = \begin{bmatrix} x_0^h(i+1,j) \\ x_0^v(i,j+1) \end{bmatrix} \tag{4}$$

Now, we define the 2-D model reference adaptive control (2-DMRAC) problem as follows.

Problem formulation:

Let the coefficient matrices A and B of the system (1) be unknown, and the local states x^h and x^v be measurable. Determine the control input of (1) so that the following tracking can be established:

$$\lim_{i \ and/or \ j \to \infty} C(x'(i,j) - x_0'(i,j)) = 0 \quad i,j=0,1,... \tag{5}$$

where $C \in \mathbb{R}^{1 \times m}$ is a known constant matrix.

The solution procedure of the above problem will be presented in the next section.

Here, it should be pointed out that in (5), C must be chosen based on the system controlling aim. For example, if it is desired that the sum of the first and last components of $x'(i,j)$ track the $x_0'(i,j)$ then C must be chosen as:

$$C = [1 \ 0 \ 0 \cdots 0 \ 0 \ 1]$$

However, we assume that the scalar CB is nonzero.

The coefficient matrices A_{0k} (for $k=1,2,3,4$), B_{0l} (for $l=1,2$), the input sequence $r(i,j)$ and the boundary conditions $x_0^h(0,j)$, $x_0^v(i,0)$ (for $i,j=0,1,...$) of the reference model (3) are properly selected, so that the resulted solution for $Cx_0'(i,j)$ indicates the desired trajectory for $Cx'(i,j)$ and also the scalar CB_0 be nonzero.

III. SOLUTION PROCEDURE OF THE 2-DMRAC PROBLEM

As solution method of any control problem, using the feedback is considered as the basic idea for solving our 2-DMRAC problem. Therefore, we choose the input of (1) as following:

$$u(i,j) = Q(i,j)\left\{\theta(i,j)x(i,j) + P^* x_0(i,j) + r(i,j)\right\} \tag{6}$$

where $\theta(i,j) \in \mathbb{R}^{1\times m}$ and $Q(i,j) \in \mathbb{R}$ are adjustable and the $P^* \in \mathbb{R}^{1\times m}$ is constant.

Substituting for $u(i,j)$ from (6) into (1) yields:

$$\begin{aligned} x'(i,j) &= \{A + BQ(i,j)\theta(i,j)\} x(i,j) \\ &+ BQ(i,j)P^* x_0(i,j) + BQ(i,j)r(i,j) \\ & i,j=0,1,... \end{aligned} \tag{7.1}$$

Multiplying the above relation from left side by C results:

$$\begin{aligned} Cx'(i,j) &= C\{A + BQ(i,j)\theta(i,j)\} x(i,j) \\ &+ CBQ(i,j)P^* x_0(i,j) + CBQ(i,j)r(i,j) \\ & i,j=0,1,... \end{aligned} \tag{7.2}$$

Also from (3) we have:

$$Cx_0'(i,j) = CA_0x_0(i,j) + CB_0r(i,j)\ i,j = 0,1,... \quad (8)$$

The comparison of (7.2) and (8) yields that in order to satisfy (5), the adjustment of the matrix $\theta(i,j)$ and the scalar $Q(i,j)$ must be such that these to come close respectively to a constant matrix and a constant scalar, namely $\theta^* \in \mathbb{R}^{1\times m}$ and $Q^* \in \mathbb{R}$, so that:

$$C(A + BQ^*\theta^*) = 0, CBQ^*P^* = CA_0, CBQ^* = CB_0$$

or:

$$\theta^* = \frac{-1}{CB_0}CA \ ,\ P^* = \frac{1}{CB_0}CA_0 \ ,\ Q^* = \frac{CB_0}{CB} \quad (9)$$

Considering that A and B are unknown, P^* will be known, but θ^* and Q^* are unknown. After obtaining P^* from (9), the closed-loop control law (6) can be applied for determining the system input.

Now, the main problem is constructing a 2-D adaptive law for updating the $\theta(i,j)$ and $Q(i,j)$ using all available information, so that (5) can be established. We assume that the scalar $Q(i,j)$ is assigned so that it is nonzero.

We define the tracking error between the system (1) and reference model (3) as follows:

$$e(i,j) = C\left(x'(i,j) - x_0'(i,j)\right) \quad i,j = 0,1,... \quad (10)$$

Using (6), (7.2), (8) and (9), the tracking error equation is obtained as follows:

$$e(i,j) = CB_0\psi(i,j)z(i,j) \quad i,j = 0,1,... \quad (11)$$

where

$$z(i,j) = \begin{bmatrix} x^h(i,j) \\ u(i,j) \\ x^v(i,j) \end{bmatrix},\ \psi(i,j) = \Lambda(i,j) - \Lambda^* \quad (12.1)$$

$$\begin{aligned} \Lambda(i,j) &= \begin{bmatrix} \theta_1(i,j) & -Q^{-1}(i,j) & \theta_2(i,j) \end{bmatrix} \\ \Lambda^* &= \begin{bmatrix} \theta_1^* & -Q^{*-1} & \theta_2^* \end{bmatrix} \end{aligned} \quad (12.2)$$

$\theta_1(i,j)$, θ_1^* are respectively the m^h first components and $\theta_2(i,j)$, θ_2^* are respectively the m^v last components of the $\theta(i,j)$ and θ^*, i.e.

$$\begin{aligned} &\theta(i,j) = [\theta_1(i,j)\ \ \theta_2(i,j)],\ \theta^* = \begin{bmatrix} \theta_1^* & \theta_2^* \end{bmatrix} \\ &\theta_1(i,j), \theta_1^* \in \mathbb{R}^{1\times m^h},\ \theta_2(i,j), \theta_2^* \in \mathbb{R}^{1\times m^v} \end{aligned} \quad (13)$$

The scalars $Q^{-1}(i,j)$ and Q^{*-1} are respectively the inverses of the $Q(i,j)$ and Q^*.

Considering the error equation (11), we try to adjust the $\Lambda(i,j)$ instead of $\theta(i,j)$ and $Q(i,j)$.

The matrix $\Lambda(i,j)$, we name the adjustable feedback gain matrix, has one row and q columns and q is defined in (4) which is at least three. For constructing a 2-D adaptive adjusting law, we decompose the columns of $\Lambda(i,j)$ into two blocks, namely the *horizontal* and *vertical* blocks. Let $\Lambda^h(i,j)$ and $\Lambda^v(i,j)$ denote the horizontal and vertical blocks respectively:

$$\begin{gathered} \Lambda(i,j) = \begin{bmatrix} \Lambda^h(i,j) & \Lambda^v(i,j) \end{bmatrix} \\ \Lambda^h(i,j) \in \mathbb{R}^{1\times q^h}, \Lambda^v(i,j) \in \mathbb{R}^{1\times q^v} \\ q^h + q^v = q \end{gathered} \quad (14.1)$$

We accomplish similar decomposition for Λ^* as follows:

$$\Lambda^* = \begin{bmatrix} \Lambda^{*h} & \Lambda^{*v} \end{bmatrix} \Lambda^{*h} \in \mathbb{R}^{1\times q^h}, \Lambda^{*v} \in \mathbb{R}^{1\times q^v} \quad (14.2)$$

We adjust the $\Lambda^h(i,j)$ and $\Lambda^v(i,j)$ respectively along the horizontal (i) and vertical (j) directions, as follows.

$$\begin{gathered} \begin{bmatrix} \Lambda^h(i+1,j) & \Lambda^v(i,j+1) \end{bmatrix} = \begin{bmatrix} \Lambda^h(i,j) & \Lambda^v(i,j) \end{bmatrix} + \Delta(i,j) \\ i,j = 0,1,... \end{gathered} \quad (15)$$

where $\Delta(i,j)$ is a modifier term, and it must be determined in a suitable manner.

The relation (15) is a 2-D algorithm, in which $\Lambda^h(i,j)$ and $\Lambda^v(i,j)$ are adjusted along horizontally and vertically directions respectively. For running this 2-D algorithm, the quantities $\Lambda^h(0,j)$ and $\Lambda^v(i,0)$ are needed, in addition to the modifier term $\Delta(i,j)$. We call the quantities $\left\{\Lambda^h(0,j), j = 0,1,...\right\}$ and $\left\{\Lambda^v(i,0), i = 0,1,...\right\}$ the boundary conditions of the (15), which must be adjusted according to a suitable manner.

The adjustment of the boundary conditions $\Lambda^h(0,j)$ and $\Lambda^v(i,0)$ is a 1-D adjusting problem, and we consider that as follows.

$$\Lambda^h(0,j+1) = \Lambda^h(0,j) + \Delta_0^h(j) \quad j = 0,1,... \quad (16.h)$$

$$\Lambda^v(i+1,0) = \Lambda^v(i,0) + \Delta_0^v(i) \quad i = 0,1,... \quad (16.v)$$

where $\Delta_0^h(j)$ and $\Delta_0^v(i)$ are appropriate modifier quantities.

Thus, the 2-D adjusting algorithm needs the two minor 1-D adjusting algorithms for its boundary conditions, while in the 1-D case there is no the boundary conditions problem.

This is a principal difference between the 1-D and 2-D adjustment algorithms.

For completing the underlying algorithm, the modifier terms $\Delta(i,j)$, $\Delta_0^h(j)$ and $\Delta_0^v(i)$ must be determined.

Determination of the $\Delta(i,j)$

The modifier term $\Delta(i,j)$ in (15) should be chosen such that with increasing i and/or j, the value of tracking error $e(i,j)$, which is defined in (10), to be decreased. We know that *Moving against the gradient direction causes the maximum decrease*. We use this fact in order to obtain an appropriate amount for $\Delta(i,j)$. For this purpose the following quadratic cost function is considered on $e(i,j)$:

$$g(i,j) = \frac{1}{2}e^2(i,j) \qquad i,j = 0,1,\ldots \tag{17}$$

Here, $\Delta(i,j)$ is chosen as following:

$$\Delta(i,j) = \mu(i,j)\left(-\frac{\nabla g(i,j)}{\nabla \Lambda(i,j)}\right) \quad i,j = 0,1,\ldots \tag{18}$$

where $\mu(i,j)$ is a positive scalar named the algorithm step size, $\frac{\nabla g}{\nabla \Lambda}$ demonstrates the gradient of g with respect to Λ.

Using (11) and (17) it is easy to show

$$\frac{\nabla g(i,j)}{\nabla \Lambda(i,j)} = B_0^T C^T e(i,j) z^T(i,j) \tag{19}$$

Finally, from (18) and (19) the 2-D algorithm (15) will become as follows:

$$\begin{aligned}\left[\Lambda^h(i+1,j) \quad \Lambda^v(i,j+1)\right] &= \left[\Lambda^h(i,j) \quad \Lambda^v(i,j)\right] \\ &\quad - \mu(i,j) B_0^T C^T e(i,j) z^T(i,j)\end{aligned} \tag{20}$$

Determination of the $\Delta_0^h(j)$ *and* $\Delta_0^v(i)$

Considering the form of the 2-D algorithm (20), its boundary conditions adjustment manners, that are the 1-D algorithms (16.h) and (16.v), are offered as follows:

$$\Lambda^h(0,j+1) = \Lambda^h(0,j) - \mu(0,j) B_0^T C^T e(0,j) z^{h^T}(0,j) \tag{21.h}$$

$$\Lambda^v(i+1,0) = \Lambda^v(i,0) - \mu(i,0) B_0^T C^T e(i,0) z^{v^T}(i,0) \tag{21.v}$$

where $z^h(i,j)$ and $z^v(i,j)$ respectively consist of the first q^h and the last q^v components of the $z(i,j)$, i.e.

$$z(i,j) = \begin{bmatrix} z^h(i,j) \\ z^v(i,j) \end{bmatrix} \; z^h(i,j) \in \mathbb{R}^{q^h}, z^v(i,j) \in \mathbb{R}^{q^v} \tag{22}$$

IV. Stability Analysis Of The Presented 2-DMRAC Procedure

In this section the stability of the presented 2-DMRAC procedure that consists of the control law (6), the 2-D adjusting algorithm (20) and the 1-D adjusting algorithms (21.h), (21.v) is analyzed.

The proof of Theorems is not given here for limitation in the paper pages number and we will give it in the journal version of paper. However, interested readers can find it in [31].

A. The Concept of the Stability and a Sufficient Condition for Stability

The Stability of the presented 2-DMRAC procedure is defined follows:

Definition 1- The presented 2-DMRAC procedure is stable, if the desired relation (5) to be satisfied, i.e.

$$\begin{matrix}\lim e(i,j) = 0 \\ i \text{ and/or } j \to \infty\end{matrix} \tag{23}$$

Theorem 1- The presented 2-DMRAC procedure is stable if the following conditions hold:

1- In the 2-D adjusting algorithm (20), the step size $\mu(i,j)$ is chosen in the following interval:

$$0 < \mu(i,j) < \frac{2}{(CB_0)^2 z^T(i,j) z(i,j)} \quad i,j = 0,1,\ldots \tag{24}$$

2-The adjusting algorithms of the boundary conditions $\Lambda^h(0,j)$ and $\Lambda^v(i,0)$, which are the 1-D algorithms (21.h) and (21.v), are such that the following relations will be satisfied:

$$\begin{matrix}\lim f(k) = 0 \\ k \to \infty\end{matrix} \tag{25}$$

where:

$$\begin{aligned} f(k) &= V^h(0,k) + V^v(k,0) \\ V^h(0,k) &= \psi^h(0,k)\psi^{h^T}(0,k) \quad k = 0,1,\ldots \\ V^v(k,0) &= \psi^v(k,0)\psi^{v^T}(k,0) \end{aligned} \tag{26}$$

and $\psi^h(i,j)$, $\psi^v(i,j)$ respectively denote the horizontal and vertical blocks of $\psi(i,j)$, which is defined in (12), i.e.

$$\begin{aligned}\psi(i,j) &= \left[\psi^h(i,j) \quad \psi^v(i,j)\right] \\ \psi^h(i,j) &= \Lambda^h(i,j) - \Lambda^{*h}, \psi^v(i,j) = \Lambda^v(i,j) - \Lambda^{*v}\end{aligned} \tag{27}$$

■

B. Determination of the Requirements Conditions for the Boundary Conditions Adjustment Algorithms

The condition (25) is used in the proof of the Theorem1. The sequence $f(k)$ depends on the 1-D adjusting algorithms (21.h) and (21.v). We must obtain some conditions for these algorithms, such that the conditions guaranty the requirement (25).

Theorem 2.h- The algorithm (21.h) results:

$$\lim_{k \to \infty} V^h(0,k) = 0 \tag{28}$$

if the three following conditions hold:

1- The step size $\mu(0,j)$ is chosen in the following interval:

$$0 < \mu(0,j) < \frac{2}{(CB_0)^2 \, z^T(0,j) z(0,j)} \quad j = 0,1,\ldots \tag{29}$$

2-The boundary condition $x^h(0,j)$ of the system (1), and the boundary condition $x_0^h(0,j)$ and the boundary input $r(0,j)$ of the reference model (3) are sufficiently general so that there exists a natural number as n^{*h} such that the matrix $R^h(k,n^{*h}) \in \mathbb{R}^{(m+1)(2m^h+1)\times(n^{*h}+1)}$, which is defined below, has full row rank for any beginning point k.

$$R^h(k,n^{*h}) = \left[\begin{array}{cccc} x^h(0,k) & x^h(0,k+1) & \cdots & x^h(0,k+n^{*h}) \\ x_0^h(0,k) & x_0^h(0,k+1) & \cdots & x_0^h(0,k+n^{*h}) \\ r(0,k) & r(0,k+1) & & r(0,k+n^{*h}) \\ \hdashline x^h(0,k+1) & x^h(0,k+2) & & x^h(0,k+n^{*h}+1) \\ x_0^h(0,k+1) & x_0^h(0,k+2) & & x_0^h(0,k+n^{*h}+1) \\ r(0,k+1) & r(0,k+2) & & r(0,k+n^{*h}+1) \\ \hdashline \vdots & \vdots & \ddots & \vdots \\ \hdashline x^h(0,k+m) & x^h(0,k+m+1) & & x^h(0,k+m+n^{*h}) \\ x_0^h(0,k+m) & x_0^h(0,k+m+1) & \cdots & x_0^h(0,k+m+n^{*h}) \\ r(0,k+m) & r(0,k+m+1) & \cdots & r(0,k+m+n^{*h}) \end{array}\right] \tag{30}$$

3- In the decomposition (14) we have:

$$q^h \le m^h + 1 \tag{31}$$

that is the $z^h(i,j)$ does not consist of any component of the $x^v(i,j)$.

■

Theorem 2.v- The algorithm (21.v) results:

$$\lim_{k \to \infty} V^v(k,0) = 0 \tag{32}$$

if the following conditions hold:

1- The step size $\mu(i,0)$ is chosen in the following interval:

$$0 < \mu(i,0) < \frac{2}{(CB_0)^2 \, z^T(i,0) z(i,0)} \quad i = 0,1,\ldots \tag{33}$$

2-The boundary condition $x^v(i,0)$ of the system (1), and the boundary condition $x_0^v(i,0)$ and the boundary input $r(i,0)$ of the reference model (3) are sufficiently general so that there exists a natural number as n^{*v} such that the matrix $R^v(k,n^{*v}) \in \mathbb{R}^{(m+1)(2m^v+1)\times(n^{*v}+1)}$, which is given below, has full row rank for any beginning point k.

$$R^v(k,n^{*v}) = \left[\begin{array}{cccc} x^v(k,0) & x^v(k+1,0) & \cdots & x^v(k+n^{*v},0) \\ x_0^v(k,0) & x_0^v(k+1,0) & \cdots & x_0^v(k+n^{*v},0) \\ r(k,0) & r(k+1,0) & & r(k+n^{*v},0) \\ \hdashline x^v(k+1,0) & x^v(k+2,0) & & x^v(k+n^{*v}+1,0) \\ x_0^v(k+1,0) & x_0^v(k+2,0) & & x_0^v(k+n^{*v}+1,0) \\ r(k+1,0) & r(k+2,0) & & r(k+n^{*v}+1,0) \\ \hdashline \vdots & \vdots & \ddots & \vdots \\ \hdashline x^v(k+m,0) & x^v(k+m+1,0) & & x^v(k+m+n^{*v},0) \\ x_0^v(k+m,0) & x_0^v(k+m+1,0) & \cdots & x_0^v(k+m+n^{*v},0) \\ r(k+m,0) & r(k+m+1,0) & \cdots & r(k+m+n^{*v},0) \end{array}\right] \tag{34}$$

3- In the decomposition (14) we have:

$$q^v \le m^v + 1 \tag{35}$$

that is the $z^v(i,j)$ does not consist of any component of the $x^h(i,j)$.

■

Comment 1- If the conditions of Theorems 2.h and 2.v are satisfied then we have:

$$\lim_{k \to \infty} f(k) = \lim_{k \to \infty} \left(V^h(0,k) + V^v(k,0) \right) = 0$$

Thus, the requirement (25) will be guarantied.

*Comment 2-*The conditions 1 of Theorems 2.h and 2.v are not new conditions but theses are a part of the condition 1 of Theorem 1.

Comment 3- Considering the conditions 3 of Theorems 2.h and 2.v we must choose the dimensions of the decomposition (14), i.e. q^h and q^v as follows:

$$q^h = m^h + 1, q^v = m^v \text{ or } q^h = m^h, q^v = m^v + 1 \tag{36}$$

*Comment 4-*The conditions 2 of Theorems 2.h and 2.v are the PE (persistence of excitation) conditions of the presented 2-DMRAC system, and considering these the following definition is given:

Definition 2- The reference model (3) is said to be tractable by the system (1) if there exist n^{*h} and n^{*v} so that $R^h(k,n^{*h})$ and $R^v(k,n^{*v})$, which are respectively given in (30) and (34), have full row rank for any beginning point k.

V. Conclusion

In this paper the 2-DMRAC problem is formulated in a general case, when both of the 2-D space coordinates are possibly unbounded, and the corresponding solution is presented. In this solution procedure, the system input is assigned as a closed-loop control law in terms of the system state and reference model state and input vectors. There exist some adjustable feedback gains in this control law, which are augmented into a matrix form, and is called the adjustable feedback gain matrix.

For adjusting the feedback gain matrix, in two dimensions, its columns are decomposed into the horizontal and vertical blocks. Based on this decomposition a 2-D recursive algorithm is proposed to adjust the feedback gain matrix in the manner that the horizontal and vertical blocks are adjusted along the horizontal and vertical directions, respectively. This approach to adjust in two dimensions can be interpreted as a generic 2-D adjustment law. It is observed that this 2-D adjusting algorithm needs the two minor 1-D adjusting algorithms for its boundary conditions; while in the 1-D case this problem does not exist. This is a principal difference between the 1-D and 2-D adjustment algorithms. The proper values for the modifying these terms; are obtained by utilizing the gradient approach based on the error between the system and reference model.

The stability of the presented 2-DMRAC procedure is analyzed and stability conditions are obtained in terms of algorithm step size range; and the properties of the system boundary conditions as well as its reference model boundary conditions and boundary inputs.

References

[1] R. P. Roesser, "A discrete state space model for linear image processing," *IEEE Transactions on Automatic Control*, vol. 20, pp. 1–10, 1975.

[2] E. Fornasini, and G. Marchesini, "Doubly indexed dynamical systems: state-space models and structural Properties," *Math. Systems Theory*, vol. 12, pp. 59–72, 1978.

[3] J. Kurek, "The general state-space model for two-dimensional linear digital systems," *IEEE Transactions on Automatic Control*, vol. 30, pp. 600–602, 1985.

[4] N. K. Bose, *Applied Multidimensional Systems Theory*. New York: Van Nostrand Reinhold, 1982.

[5] T. Kaczorek, *Two Dimensional Linear Systems*. Berlin: Springer, 1985.

[6] W. Lu, and A. Antoniou, *Two-Dimensional Digital Filters*. New York: Marcel Dekker, 1992.

[7] S. K. Mitra, and P. E. Michael, *Two-Dimensional Digital Signal Processing*. New York: Marcel Dekker, 1992.

[8] D. H. Owens, N. Amann, E. Rogers , and M. French, "Analysis of linear iterative learning control schemes: a 2-d repetitive systems approach," *Multidimensional Systems and Signal Processing,* vol. 11, pp.125–177, 2000.

[9] E. Walach, and E. Zeheb, "Generalized zero sets of multi-parameter polynomials and feedback stabilization," *IEEE Transactions on Circuits and System,* vol. 29, pp.13–23, 1982.

[10] M. Bisiacco, E. Fornasini, and G. Marchesini, "Feedback stabilization of 2-d systems," in *Proceedings of the 24th Conference on Decision and Control*, December 1985, pp.1579–1583.

[11] M. Sebek, "On 2-d pole placement," *IEEE Transactions on Automatic Control,* vol. 30, pp.819–822, 1985.

[12] P. Agathokiis,and S. Foda, "Characteristic polynomial assignment in 2-dimensional discrete systems," *IEE Proceedings*, vol. 135(D), pp.253–256,1988.

[13] J. Wood, U. Oberst, E. Rogers, and D. H. Owens, "A behavioral approach to the pole structure of one-dimensional and multi-dimensional linear systems," *SIAM Journal of Control and Optimization*, vol. 38, pp.627–661, 2000

[14] P. N. Paraskevopoulos, "Exact model matching of 2-d systems via state feedback," *Journal of the Franklin Institute*, vol. 308, no. 5, pp.475–486, printed in Northern Ireland November 1979.

[15] Y. Yasuda, "On the synthesis of model-following two-dimensional digital systems," *International Journal of Control*, vol. 34, pp.201–210, 1981.

[16] M. Sebek, "2-D exact model matching," *IEEE Transactions on Automatic Control*, vol. 28, pp.215–217, 1983.

[17] T. Kaczorek, and J. Klamka, "Minimum energy control of 2-d linear systems with variable coefficients," *International Journal of Control*, vol. 44, pp.645–650, 1989.

[18] C. Li, and M. S. Fadali, "Optimal control of 2-d systems," *IEEE Transactions on Automatic Control*, vol. 36, pp.223–228, 1991.

[19] C. Du, L. Xie, and C. Zhang, "H_∞ control and robust stabilization of two-dimensional systems in Rosser model," *Automatica*, vol. 37, no. 2, pp.205–211, 2001.

[20] L. Xie, C. Du, Y. C. Soh, and C. Zhang, "H_∞ and robust control of 2-d systems in FM second model," *Multidimensional Systems and Signal Processing*, vol. 13, pp.265–287, 2002.

[21] K. S. Narendra, and A. M. Annaswamy, *Stable Adaptive Systems*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

[22] K. J. Astrom, and B. Wittenmark, *Adaptive Control*. Reading, MA: Addison-Wesley, 1995.

[23] J. Karl, G. C. Goodwin, and P. Kumar, *Adaptive Control, Filtering, and Signal Processing*. New York: Springer, 1995.

[24] W. P. Heath, *Self-Tuning Control for Two-Dimensional Processes*. New York: Marcel Dekker, 1992.

[25] W. P. Heath, and P. E. Wellsted, "Self-tuning prediction and control for two-dimensional processes –part 2: parameter estimation, set-point tracking and offset handling," *International Journal of Control*, vol. 62, no. 2, pp.239–269, 1995.

[26] A. Madady, and M. Shafiee, "Two-dimensional adaptive parameters estimation," in *Proceedings (CD-ROM) of the European Control Conference 2003 (ECC-2003)*, University of Cambridge, UK, September 2003.

[27] M. Yamada, L. Xu, and O. Saito, "2D model-following servo system," *Multidimensional Systems and Signal Processing*, vol. 10, pp.71–91, 1999.

[28] M. Yamada, L. Xu, and O. Saito, "2D model-following servo system and its application to linear multi-pass processes," in *Proceedings of the 14th Triennial IFAC World Congress,* Beijing, China, July 1999; pp.241–246.

[29] M. Bohm, M. A. Demetrious, S. Reich, and I. G. Rosen, "Model reference adaptive control of distributed parameter systems," *SIAM Journal of Control and Optimization*, vol. 36, no. 1, pp.33–83, 1998.

[30] H. Fan, and C. Wen, "Adaptive control of a class of 2-d discrete systems," *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, vol. 50, no. 1, pp.166–172, 2003.

[31] A. Madady, "Synthesis stable adaptive procedures for 2-D systems," Ph.D. dissertation, Dept. Elect. Eng., Amir Kabir University of Technology, Tehran, Iran, 2004.