

Olle KOTTA

## RECURSIVE REALIZATION OF 2-D LINEAR SYSTEMS

(Presented by N. Alumäe)

### 1. Introduction

In recent years, several authors have discussed the two-dimensional (2-D) realization problems of the 2-D input-output description by different 2-D state space models in a variety of situations. The input-output description have been expressed in terms of 2-D Markov parameters [1-3], or in terms of 2-D transfer function [4-10]. For a general 2-D system, at this moment there is no technique available for constructing minimal realizations [11]. For the important special case of 2-D systems with separable denominator transfer function, T. Hinamoto [1, 2] has presented the algorithms for complete and partial minimal realizations from the given set of 2-D Markov parameters; the Roesser's model is used to represent the system. T. Hinamoto and F. W. Fairman [3] have treated the canonical minimal partial realization problem in the same situation.

In this paper, we also consider the partial minimal realization problem for systems with separable denominator transfer function. Unlike [1-3] we do not look for a solution for some fixed  $N$  ( $N$  is the number of given 2-D Markov parameters), but look for a recursive solution where each extension of the sequence of 2-D Markov parameters can be met by calculating just a few new elements in the corresponding realization. Our approach here is an extension of the method proposed by J. Rissanen for the 1-D case [12].

### 2. Problem statement

Consider the Roesser's state space model of the 2-D system with separable denominator transfer function abbreviated by  $\Sigma(n, m) = (A_1, A_3, A_4, b_1, b_2, c_1, c_2)$  [1]:

$$\begin{aligned}x^h(i+1, j) &= A_1 x^h(i, j) + b_1 u(i, j), \\x^v(i, j+1) &= A_3 x^h(i, j) + A_4 x^v(i, j) + b_2 u(i, j), \\y(i, j) &= c_1 x^h(i, j) + c_2 x^v(i, j), \quad i, j \geq 0,\end{aligned}\tag{1}$$

where  $x^h(i, j)$  is an  $n$  dimensional horizontal state vector,  $x^v(i, j)$  is an  $m$  dimensional vertical state vector,  $u(i, j)$  is a scalar input,  $y(i, j)$  is a scalar output, and  $A_1, A_3, A_4, b_1, b_2, c_1, c_2$  are real matrices of appropriate dimensions. For the system (1), the set of boundary conditions consists of  $x^h(0, j)$ , for  $j \geq 0$  and  $x^v(i, 0)$  for  $i \geq 0$ .

If the boundary conditions are equal to zero and an impulse is applied as an input ( $u(0, 0) = 1$ ,  $u(i, j) = 0$  in other cases), then the impulse response of the system (1) is [1]

$$y(i, 0) = c_1 A_1^{i-1} b_1 \triangleq \omega_{i0}, \quad y(0, j) = c_2 A_4^{j-1} b_2 \triangleq \omega_{0j},$$

$$y(i, j) = c_2 A_4^{j-1} A_3 A_1^{i-1} b_1 \triangleq \omega_{ij}, \quad i > 0, \quad j > 0.$$

The real numbers  $\omega_{ij}$  are also called the 2-D Markov parameters of the system (1). Let us define an  $2k+1$  dimensional vector  $W_k$  as

$$W_k(\omega_{k0}, \omega_{k1}, \dots, \omega_{kk}, \omega_{k-1,k}, \dots, \omega_{1k}, \omega_{0k}), \quad k=1, 2, \dots$$

The partial realization problem can be defined as follows [1]. Given the finite sequence  $\{W_k; k=1, 2, \dots, N\}$ , system (1) is said to be a partial realization of that sequence if and only if the Markov parameters formed from the realized system match the given set of Markov parameters contained in the sequence. Furthermore, the realization is said to be a minimal partial realization if and only if  $n$  and  $m$  are minimal among all other partial realizations.

In this paper we consider the following more general variant of the minimal partial realization problem. Let  $\sum_{N'}$  denote a minimal partial realization by Roesser's state space model of the sequence  $\{W_k; k=1, \dots, N\}$ . Given the sequence  $\{W_k; k=1, \dots, N\}$ , find a sequence of minimal partial realization  $\{\sum_{N'}, N'=1, 2, \dots, N\}$  such that  $\sum_{N'} \subseteq \sum_{N''}$ , if  $N' < N''$ . Here the inclusion symbol means that the matrices in  $\sum_{N'}$  appear as submatrices of the corresponding ones in  $\sum_{N''}$ .

### 3. Generalized Hankel matrices and their factorization

In this section we describe two matrices introduced in [1], which play, in the case of 2-D systems, a role similar to the one of the Hankel matrix in the 1-D realization problem. These matrices are given as follows:

$$H_{L,n+1}^{(s)} = \|h_{i+j-1}^{(s)}\|, \quad i=1, \dots, L, \quad j=1, \dots, n+1,$$

$$V_{m+1,J}^{(t)} = \|v_{i+j-1}^{(t)}\|, \quad i=1, \dots, m+1, \quad j=1, \dots, J,$$

where

$$h_i^{(s)} = (\omega_{i0}, \omega_{i1}, \dots, \omega_{is})^T, \quad i=1, 2, \dots,$$

$$v_i^{(t)} = (\omega_{0i}, \omega_{1i}, \dots, \omega_{ti}), \quad i=1, 2, \dots$$

The former is called the horizontal Hankel matrix and the latter is called the vertical Hankel matrix.

The recursive realization algorithm, proposed in this paper, is based on the following types of factorizations of the horizontal and vertical Hankel matrices

$$H_{L,n+1}^{(s)} = S(L, n+1)R(n+1); \quad L \geq n+1, \quad \text{rank } H_{L,n+1}^{(s)} \geq n \quad (2)$$

and

$$V_{m+1,J}^{(t)} = P(m+1)Q(m+1, J), \quad J \geq m+1, \quad \text{rank } V_{m+1,J}^{(t)} \geq m. \quad (3)$$

Here  $R(n+1)$  is upper triangular with 1's on the main diagonal and  $P(m+1)$  is lower triangular with 1's on the main diagonal. These factors are not unique. By setting certain elements of  $S(L, n+1)$  and  $Q(m+1, J)$  equal to zero, we shall be able to calculate the elements of factors one

by one and, moreover, an addition of rows and columns to  $H_{L, n+1}^{(s)}$  and  $V_{m+1, J}^{(t)}$  will not change the elements already calculated.

**Algorithm 1.** Factorization of horizontal Hankel matrix.

*Step 1.* Set the first column in  $S(L, n+1)$  equal to that of  $H_{L, n+1}^{(s)}$ .

*Step 2.* Recursively, at the  $k+1$ th step all the  $s_{ij}$  (elements of  $S(L, n+1)$ ) and  $r_{ij}$  (elements of  $R(n+1)$ ) for  $j=1, \dots, k$  have been determined. Suppose, first, that the  $k$ th column of  $S(L, n+1)$  is not zero. Let  $q(k)$  be the least integer such that  $s_{q(k), k} \neq 0$ . Set  $s_{q(k)+i, p} = 0$  for  $p > k, i=0, \dots, s$ . Equation (2) leads to a set of  $k$  equations one for each row  $q(j), j=1, \dots, k$ . From these the unknowns  $r_{1, k+1}, \dots, r_{k, k+1}$  can be found one by one. These determine the  $k+1$ th column of  $S(L, n+1)$ .

Suppose, next, that the  $k$ th column of  $S(L, n+1)$  is zero. In this case set the nondiagonal elements in the  $k+1$ th column of  $R(n+1)$  to arbitrary values, say, to zero.

**Algorithm 2.** Factorization of vertical Hankel matrix.

*Step 1.* Set the first row in  $Q(m+1, J)$  equal to that of  $V_{m+1, J}^{(t)}$ .

*Step 2.* Recursively, at the  $k+1$ th step all the  $q_{ij}$  (elements of  $Q(m+1, J)$ ) and  $p_{ij}$  (elements of  $P(m+1)$ ) for  $i=1, \dots, k$  have been determined. Suppose, first, that the  $k$ th row of  $Q(m+1, J)$  is not zero. Let  $s(k)$  be the least integer such that  $q_{k, s(k)} \neq 0$ . Set  $q_{p, s(k)+i} = 0$  for  $p > k, i=0, 1, \dots, s$ . Equation (3) leads to a set of  $k$  equations one for each column  $s(j), j=1, \dots, k$ . From these the unknowns  $p_{k+1, 1}, \dots, p_{k+1, k}$  can be found one by one. These determine the  $k+1$ th row of  $Q(m+1, J)$ .

Suppose, next, that the  $k$ -th row is zero. In this case set the nondiagonal elements of the  $k+1$ th row of  $P(m+1)$  to arbitrary values, say, to zero.

#### 4. Preliminary results

Let us define

$$A_1(n) = R^*(n)R^{-1}(n), \quad (4)$$

where  $R^*(n)$  is formed from  $R(n+1)$  by omitting the first column and last row,

$$b_1^T(n) = [1 \ 0 \ \dots \ 0], \quad (5)$$

$$c_1(n) = [s_{11} \ \dots \ s_{1n}], \quad (6)$$

$$A_4(m) = P^{-1}(m)P^*(m), \quad (7)$$

where  $P^*(m)$  is formed from  $P(m+1)$  by omitting the first row and the last column,

$$b_2^T(m) = [q_{11} \ \dots \ q_{m1}], \quad (8)$$

$$c_2(m) = [1 \ 0 \ \dots \ 0], \quad (9)$$

$$A_3(m, n) = \Theta^{-1}(m)W_{nm}^T \Theta^{-1}(n), \quad (10)$$

where

$$\Theta^l(m) = [c_2^T(m), A_4^T(m) c_2^T(m), \dots, (A_4^{m-1}(m))^T c_2^T(m)]^T,$$

$$\Theta(n) = [b_1(n); A_1(n)b_1(n), \dots, A_1^{n-1}(n)b_1(n)],$$

$$W_{nm} = \|\omega_{kl}\|, \quad k=1, \dots, n, \quad l=1, \dots, m.$$

Now without proof being too long to present, we shall give a theorem on which the realization algorithm is based.

**Theorem.** Given  $W_k, k=1, 2, \dots$ , let the rank of the matrix  $H_{n,n}^{(2n-1)}$  be  $n$  and  $L$  any integer such that  $H_{L,n+1}^{(L+n)}$  has rank  $n$  ( $n+1 \leq L \leq L_1$ ); let the rank of the matrix  $V_{m,m}^{(2m-1)}$  be  $m$  and  $J$  any integer such that  $V_{m+1,J}^{(m+J)}$  has rank  $m$  ( $m+1 \leq J \leq J_1$ ). Then  $\sum (n, m) = [A_1(n), A_3(m, n), A_4(m), b_1(n), b_2(m), c_1(n), c_2(m)]$  is a minimal realization of the sequence  $\{W_1, \dots, W_T\}$ , where  $T \in [2n+1, L_1+n] \cap [2m+1, J_1+m]$ .

Observe that the matrices (4), (7) and (10) can be computed recursively due to their special form. The following holds

$$A_1(n+1) = \begin{bmatrix} A_1(n) & \vdots \\ \dots & \vdots \\ 0 \dots 0 & 1 \end{bmatrix} \delta_{n+1}, \quad (11)$$

where

$$\delta_{n+1} = [-r_{n+1}^T \ A_1^T(n) \ \vdots \ r_{n,n+1}] + r_{n+2}^T,$$

$$r_{n+i}^T = [r_{1,n+i} \ \dots \ r_{n+i-1,n+i}], \quad i=1, 2;$$

$$A_4(m+1) = \begin{bmatrix} 0 \\ \vdots \\ A_4(m) \\ \vdots \\ 0 \\ 1 \\ \vdots \\ \dots \\ \gamma_{m+1} \end{bmatrix}, \quad (12)$$

where

$$\gamma_{m+1} = [-\rho_{m+1} \ A_4(m) \ \vdots \ \rho_{m+1,m}] + \rho_{m+2},$$

$$\rho_{m+i} = [\rho_{m+i,1} \ \dots \ \rho_{m+i,m+i-1}], \quad i=1, 2;$$

$$A_3(m+1; n) = \begin{bmatrix} A_3(m, n) \\ (\alpha_{m+1} \ W_{mn} + \omega_{m+1}) C^{-1}(n) \end{bmatrix}, \quad (13)$$

where

$$\alpha_{m+1} = -c_2(m+1) A_4^m(m+1) I(m+1, m) O^{-1}(m),$$

$$\omega_{m+1} = \omega_{1,m+1} \ \dots \ \omega_{n,m+1},$$

and  $I(m+1, m)$  is the  $m+1$ -dimensional unit matrix where the last column is omitted;

$$A_3(m, n+1) = [A_3(m, n) \ \vdots \ O^{-1}(m) (W_{mn} \beta_{n+1} + \omega_{n+1})], \quad (14)$$

where

$$\beta_{n+1} = -C^{-1}(n) I(n, n+1) A_1^n(n+1) b_1(n+1),$$

$$\omega_{n+1} = [\omega_{n+1,1}, \ \dots, \ \omega_{n+1,m}]^T$$

and  $I(n, n+1)$  is the  $n+1$ -dimensional unit matrix where the last row is omitted.

## 5. Realization algorithm

*Step 1.* Pick the least  $K$  such that  $h_K^{(2K+1)} \neq 0$ . Set  $N=2K-1$ ,  $IH=JH=K$ . Pick the least  $L$  such that  $v_L^{(2L+1)} \neq 0$ . Set  $M=2L-1$ ,  $IV=JV=L$ ,  $IND=1$ .

*Step 2.*  $JH=JH+1$ ,  $N=N+1$ .

*Step 3.* If  $JH \leq IH$ , go to Step 4; otherwise  $IH=IH+1$ ,  $N=N+1$ , go to Step 4.

*Step 4.* Apply the factorization algorithm 1 on  $H_{IH, JH}^{(N)}$ .

*Step 5.* If the last column of  $S(IH, JH)$  is nonzero, go to Step 2; otherwise calculate  $A_1(JH-1)$ ,  $b_1(JH-1)$ ;  $c_1(JH-1)$ .

*Step 6.* If  $IND=1$ , go to Step 7; otherwise go to Step 11.

*Step 7.*  $IV=IV+1$ ,  $M=M+1$ .

*Step 8.* If  $IV \leq JV$ , go to Step 9; otherwise  $JV=JV+1$ ,  $M=M+1$ , go to Step 9.

*Step 9.* Apply the factorization algorithm 2 on  $V_{IV, JV}^{(M)}$ .

*Step 10.* If the last row of  $Q(IV, JV)$  is nonzero, go Step 7; otherwise calculate  $A_4(IV-1)$ ,  $b_2(IV-1)$ ,  $c_2(IV-1)$ .

*Step 11.* If  $N < M$ , then  $IH=IH+1$ ,  $N=N+1$ , apply the factorization algorithm 1 on  $H_{IH, JH}^{(N)}$ , go to Step 12; If  $N > M$ , then  $JV=JV+1$ ,  $M=M+1$ , apply the factorization algorithm 2 on  $V_{IV, JV}^{(M)}$ , go to Step 13; If  $N=M$ , then calculate  $A_3(IV-1, JH-1)$ , go to Step 14.

*Step 12.* If the last column of  $S(IH, JH)$  is zero, go to Step 11; otherwise  $IND=0$ , go to Step 2.

*Step 13.* If the last row of  $Q(IV, JV)$  is zero, go to Step 11; otherwise go to Step 7.

*Step 14.* Print  $N(=M)$ ,  $A_1, A_3, A_4, b_1, b_2, c_1, c_2$ . If you want to continue, then  $IH=IH+1$ ,  $N=N+1$ , apply the factorization algorithm 1 on  $H_{IH, JH}^{(N)}$ , go to Step 15; otherwise go to Step 17.

*Step 15.* If the last column of  $S(IH, JH)$  is nonzero, then  $IND=0$  go to Step 2; otherwise  $JV=JV+1$ ,  $M=M+1$ , apply the factorization algorithm 2 on  $V_{IV, JV}^{(M)}$ .

*Step 16.* If the last row of  $Q(IV, JV)$  is nonzero, go to Step 7; otherwise go to Step 14.

*Step 17.* Stop.

**Remark 1.** In this algorithm each time the factorization algorithm 1(2) is called, only one additional column (row) is calculated.

**Remark 2.** The matrices  $A_1, A_3, A_4$  are calculated from the formulas (4), (10), (7) or once calculated, are recursively updated by formulas (11)–(14).

## 6. Numerical example

Suppose the given sequence  $\{W_k; k=1, 2, \dots, 5\}$  is specified as

$$W_1 = (8 \ 4 \ 4);$$

$$W_2 = (4 \ 2 \ 1 \ 2 \ 2);$$

$$W_3 = (4 \ 2 \ 1 \ 0,5 \ 0,5 \ 1 \ 1);$$

$$W_4 = (3 \ 1,5 \ 0,75 \ 0,375 \ 0,1875 \ 0,25 \ 0,25 \ 0,5 \ 0,5);$$

$$W_5 = (2,5 \ 1,25 \ 0,625 \ 0,3125 \ 0,15625 \ 0,078125 \ 0,09375 \ 0,125 \ 0,125 \ 0,5 \ 0,5)$$

*Step 1.*  $K=1$ ,  $N=1$ ,  $IH=JH=1$ ,  $L=1$ ,  $M=1$ ,  $IV=JV=1$ ,  $IND=1$ .

*Step 2.*  $JH=2$ ,  $N=2$ .

Step 3. As  $JH > IH$ ,  $IH=2$ ,  $N=3$ .

Step 4. Applying the factorization algorithm 1 on  $H_{2,2}^{(3)}$  gives

$$\left[ \begin{array}{cc|cc} 8 & 4 & 8 & 0 \\ 4 & 2 & 4 & 0 \\ 2 & 1 & 2 & 0 \\ 1 & 0,5 & 1 & 0 \\ \hline 4 & 4 & 4 & 2 \\ 2 & 2 & 2 & 1 \\ 1 & 1 & 1 & 0,5 \\ 0,5 & 0,5 & 0,5 & 0,25 \end{array} \right] = \left[ \begin{array}{cc|cc} 8 & 0 & 8 & 0 \\ 4 & 0 & 4 & 0 \\ 2 & 0 & 2 & 0 \\ 1 & 0 & 1 & 0 \\ \hline 4 & 2 & 4 & 2 \\ 2 & 1 & 2 & 1 \\ 1 & 0,5 & 1 & 0,5 \\ 0,5 & 0,25 & 0,5 & 0,25 \end{array} \right] \left| \begin{array}{ccc} 1 & & \\ & 0,5 & \\ & & 1 \end{array} \right|$$

Step 5. As the last column of  $S(2, 2)$  is nonzero, we go to Step 2.

Step 2.  $JH=3$ ,  $N=4$ .

Step 3. As  $JH > IH$ ,  $IH=3$ ,  $N=5$ .

Step 4. Applying the factorization algorithm 1 on  $H_{3,3}^{(5)}$  gives

$$\left[ \begin{array}{ccc|ccc} 8 & 4 & 4 & 8 & 0 & 0 \\ 4 & 2 & 2 & 4 & 0 & 0 \\ 2 & 1 & 1 & 2 & 0 & 0 \\ 1 & 0,5 & 0,5 & 1 & 0 & 0 \\ 0,5 & 0,25 & 0,25 & 0,5 & 0 & 0 \\ 0,25 & 0,125 & 0,125 & 0,25 & 0 & 0 \\ \hline 4 & 4 & 3 & 4 & 2 & 0 \\ 2 & 2 & 1,5 & 2 & 1 & 0 \\ 1 & 1 & 0,75 & 1 & 0,5 & 0 \\ 0,5 & 0,5 & 0,375 & 0,5 & 0,25 & 0 \\ 0,25 & 0,25 & 0,1875 & 0,25 & 0,125 & 0 \\ 0,125 & 0,125 & 0,09375 & 0,125 & 0,0625 & 0 \\ \hline 4 & 3 & 2,5 & 4 & 1 & 0 \\ 2 & 1,5 & 1,25 & 2 & 0,5 & 0 \\ 1 & 0,75 & 0,625 & 1 & 0,25 & 0 \\ 0,5 & 0,375 & 0,3125 & 0,5 & 0,125 & 0 \\ 0,25 & 0,1875 & 0,15625 & 0,25 & 0,0625 & 0 \\ 0,125 & 0,09375 & 0,078125 & 0,125 & 0,03125 & 0 \end{array} \right] = \left[ \begin{array}{ccc|ccc} 8 & 0 & 0 & 8 & 0 & 0 \\ 4 & 0 & 0 & 4 & 0 & 0 \\ 2 & 0 & 0 & 2 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ \hline 4 & 2 & 0 & 4 & 2 & 0 \\ 2 & 1 & 0 & 2 & 1 & 0 \\ 1 & 0,5 & 0 & 1 & 0,5 & 0 \\ 0,5 & 0,25 & 0 & 0,5 & 0,25 & 0 \\ 0,25 & 0,125 & 0 & 0,25 & 0,125 & 0 \\ 0,125 & 0,0625 & 0 & 0,125 & 0,0625 & 0 \\ \hline 4 & 1 & 0 & 4 & 1 & 0 \\ 2 & 0,5 & 0 & 2 & 0,5 & 0 \\ 1 & 0,25 & 0 & 1 & 0,25 & 0 \\ 0,5 & 0,125 & 0 & 0,5 & 0,125 & 0 \\ 0,25 & 0,0625 & 0 & 0,25 & 0,0625 & 0 \\ 0,125 & 0,03125 & 0 & 0,125 & 0,03125 & 0 \end{array} \right] \left| \begin{array}{ccc} 1 & & \\ & 0,5 & \\ & & 0,5 \\ & & & 1 \\ & & & & 0,5 \\ & & & & & 0,5 \end{array} \right|$$

Step 5. As the last column of  $S(3, 3)$  is zero, we calculate

$$A_1(2) = \begin{bmatrix} 0,5 & 0,25 \\ 1 & 0 \end{bmatrix}, \quad b_1^T(2) = [1 \ 0], \quad c_1(2) = [8 \ 0].$$

Step 6. As  $IND=1$ , we go to Step 7.

Step 7.  $IV=2$ ,  $M=2$ .

Step 8. As  $IV > JV$ ,  $JV=2$ ,  $M=3$ .

Step 9. Applying the factorization algorithm 2 on  $V_{2,2}^{(3)}$  gives

$$\left[ \begin{array}{cccc|cccc} 4 & 4 & 2 & 2 & 2 & 2 & 1 & 1 \\ 2 & 2 & 1 & 1 & 1 & 1 & 0,5 & 0,5 \end{array} \right] = \begin{bmatrix} 1 \\ 0,5 & 1 \end{bmatrix} \left[ \begin{array}{cccc|cccc} 4 & 4 & 2 & 2 & 2 & 2 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array} \right].$$

Step 10. As the last row of  $Q(2, 2)$  is zero, we calculate  $A_4(1)=0,5$ ,  $b_2(1)=4$ ,  $c_2(1)=1$ .

Step 11. As  $N > M$ ,  $JV=3$ ,  $M=4$ . Applying the factorization algorithm 2 on  $V_{2,3}^{(4)}$  gives

$$\left[ \begin{array}{cccc|cccc|cccc} 4 & 4 & 2 & 2 & 1,5 & 2 & 2 & 1 & 1 & 0,75 & 1 & 1 & 0,5 & 0,5 & 0,375 \\ 2 & 2 & 1 & 1 & 0,75 & 1 & 1 & 0,5 & 0,5 & 0,375 & 0,5 & 0,5 & 0,25 & 0,25 & 0,1875 \end{array} \right] =$$



**РЕКУРСИВНЫЙ АЛГОРИТМ РЕАЛИЗАЦИИ ДВУМЕРНЫХ  
ЛИНЕЙНЫХ СИСТЕМ**

Рассматривается задача идентификации внутреннего описания двумерной (2-D) линейной динамической системы с отделимым знаменателем передаточной функции по ее внешнему поведению. Внешнее описание задано с помощью двумерных марковских параметров. Внутреннее описание ищется в виде уравнений состояния (модель Роестера). Предлагается и обосновывается алгоритм минимальной частичной реализации. Он представляет собой рекуррентный процесс, в котором реализация уточняется по мере удлинения последовательности двумерных марковских параметров, в то время как в существующих алгоритмах решение получается лишь для фиксированного конечного набора значений параметров. В основе алгоритма лежит специального вида разложение горизонтальной и вертикальной ганкелевых матриц. Предложенный алгоритм является обобщением алгоритма рекурсивной реализации одномерных (1-D) систем.