A Review on the Minimal State Space Realization for a Class of N-D Discrete Time Lossless Bounded Real Functions

George Antoniou *, Nikos Mastorakis †

Abstract: A review on the minimal state space realizations for a class of N-Dimensional (N-D) discrete - time lossless bounded real functions is presented. Initially the minimal representation is derived from the circuit representation which in fact has minimum number of delay elements and is a particular form of N-D lattice filters. The corresponding transfer functions are characterized by the all-pass filter property.

Keywords. Multidimensional systems, Circuit theory, Realization, State space, Lossless bounded real functions.

1 Introduction

The problem of representing a N-D system described by a transfer function, with a N-D state space model realization having minimal dimension has been the focus of considerable effort in recent years [1]–[4].

In going from one dimension (1-D) to higher dimensions, this problem increases greatly and abruptly in difficulty. This is due to the fact that, in contradistinction to 1-D systems, in the case of two-dimensions (2-D) or higher dimensions simultaneous holding of observability and controllability does not necessarily imply minimal state space realization. In fact, obtaining minimal realization is not always possible. Minimal realizations exist only for special systems [5]. Realizing a system with a multidimensional state space model having minimal dimensional is a significant and non-trivial problem. The need to provide minimal realization arises out of several requirements. To begin with, in general, non-minimal realizations often cause theoretical and/or computational difficulties. Also, it is important to keep in mind that, the amount of data handled by N-D systems is often very large. It is noted that some times non-minimal realizations are useful especially in circuit design due to more relaxed constraints that may offer.

In this paper a review on the minimal state space realizations, for a class of N-D discrete time lossless bounded real (DTLBR) function, is presented. Minimal realizations have been proposed for 2-D first-order DTLBR scalar functions [7],[8] and 2-D square DTLBR matrices [9]. For the case of 1-D systems a minimal state space representation was proposed by M. Mansour in [6]. Recently minimal state space realization for first order N-D DTLBR was obtained in conjunction with a sufficient condition for the stability for N-D DTLBR matrix [10]. It is noted that for the case of 1-D, all-pass functions are equivalent to DTLBR functions and for N-D systems, it has been shown that the transfer function of a N-D lattice filter becomes a N-D function if and only if the reflection coefficients $|\Delta_i| < 1$, $i = 1, 2, \cdots, k$ [6].

2 Problem Formulation

The first-order N-D lattice filter generated from 1-D lattice filters was suggested by Matsumoto et al., in [10]. In [12] and [13] the circuit implementations were extended to general 2 and 3-D filters. Using these circuit representations, which in fact are characterized by minimal number of delay elements, the corresponding state space realization, for N-D systems, are derived.

For N-D systems various state space models have been proposed for representing a given transfer function [5]. In this paper the N-D model of the Givone-Roesser setting is used. The structure of this model is as follows [11]:

$$
\begin{align*}
\dot{x}(i_1, i_2, \cdots, i_k) &= Ax(i_1, i_2, \cdots, i_k) &+ bu(i_1, i_2, \cdots, i_k) \\
y(i_1, i_2, \cdots, i_k) &= c'x(i_1, i_2, \cdots, i_k) &+ du(i_1, i_2, \cdots, i_k)
\end{align*}
$$

where
\[ \dot{x}(i_1, i_2, \ldots, i_k) = \begin{bmatrix} x^{d_1}(i_1 + 1, i_2, \ldots, i_k) \\ x^{d_2}(i_1, i_2 + 1, i_3, \ldots, i_k) \\ \vdots \\ x^{d_k}(i_1, i_2, \ldots, i_k + 1) \end{bmatrix} \]

\[ x(i_1, i_2, \ldots, i_k) = \begin{bmatrix} x^{d_1}(i_1, i_2, \ldots, i_k) \\ x^{d_2}(i_1, i_2, \ldots, i_k) \\ \vdots \\ x^{d_k}(i_1, i_2, \ldots, i_k) \end{bmatrix} \]

\[ A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1k} \\ A_{21} & A_{22} & \cdots & A_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ A_{k1} & A_{k2} & \cdots & A_{kk} \end{bmatrix} \]

\[ b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix}, \quad c' = [ c_1, c_2, \ldots, c_k ], \quad d \in \mathbb{R} \]

Applying the N–D \( Z \) transform, the equation that relates the state space model (1) and its transfer function representation is:

\[ H(z_1, z_2, \ldots, z_k) = b[\mathcal{I} - A]c' + d \quad (2) \]

where, \( \mathcal{I} = z_1 \mathbb{I} \oplus z_2 \mathbb{I} \oplus \cdots \oplus z_k \mathbb{I}, \) with \( \oplus \) denoting the direct sum.

### 3 2–D State Space Realization

In order to derive the state space matrices \( A, b, c' \) and the scalar \( d \) for the state space model (1), it is assumed that the outputs of the delay elements \( z_1^{-1}, z_2^{-1} \) correspond to the states of the model (1) [13]. Moreover writing one state equation for every delay element, after some algebraic manipulations, we conclude the matrices \( A, b, c' \) and the scalar \( d \), of the state space model (2) may be derived by inspection as having the structure (1), where,

\[ \dot{x}(i_1, i_2) = Ax(i_1, i_2) + bu(i_1, i_2) \quad (3) \]

\[ y(i_1, i_2) = c'x(i_1, i_2) + du(i_1, i_2) \]

where

\[ \dot{x}(i_1, i_2) = \begin{bmatrix} x^{d_1}(i_1 + 1, i_2) \\ x^{d_2}(i_1, i_2 + 1) \end{bmatrix} \]

\[ x(i_1, i_2) = \begin{bmatrix} x^{d_1}(i_1, i_2) \\ x^{d_2}(i_1, i_2) \end{bmatrix} \]

\[ A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \]

Submatrix \( A_{11} \)

\[ a_{i,j} = \begin{cases} -\Delta_2(k+1-i) & \text{if } i \geq j \\ 0 & \text{otherwise} \end{cases} \]

Submatrix \( A_{12} \)

\[ a_{i,j} = \begin{cases} -\Delta_2(k+1-i) & \text{if } i > j \\ 1 - \Delta_2^{2k} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \]

Submatrix \( A_{21} \)

\[ a_{i,j} = \begin{cases} -\Delta_2(k+1-i) & \text{if } i \geq j \\ 1 - \Delta_2^{2k} & \text{if } j = i + 1 \\ 0 & \text{otherwise} \end{cases} \]

Submatrix \( A_{22} \)

\[ a_{i,j} = \begin{cases} -\Delta_2(k+1-i) & \text{if } i \geq j \\ 0 & \text{otherwise} \end{cases} \]

Vector \( b \)

\[ b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \]

where

\[ b_1 = b_i = \Delta_2(k+1-i) \]

and

\[ b_2 = b_i = \Delta_2(k+1-i) - 2 \]

vector \( c' \)

\[ c' = [ 1 - \Delta_2^k, 0, \ldots, 0 ] \]

and the scalar

\[ d = \Delta_2^k \]

Note that, \( \Delta_0 = 1. \)

### 3.1 Example: First–order 2–D filter

For the First–order 2–D filter, \( k = 1, \) the 2–D Givone–Roesser model (3) becomes,

\[ \dot{x}(i_1, i_2) = Ax(i_1, i_2) + bu(i_1, i_2) \]

\[ y(i_1, i_2) = c'x(i_1, i_2) + du(i_1, i_2) \quad (4) \]
where,

\[
A = \begin{bmatrix}
-\Delta_1 \Delta_2 & 1 - \Delta_1^2 \\
-\Delta_2 & -\Delta_1
\end{bmatrix}, \quad b = \begin{bmatrix} \Delta_1 \\ 1 \end{bmatrix}
\]

\[
c' = \begin{bmatrix} 1 - \Delta_2^2 \\ 0 \end{bmatrix}, \quad d = \Delta_2
\]

Using (2), the transfer function of the above model (4) is

\[
H(z_1, z_2) = \frac{1 + \Delta_1 z_1 + \Delta_2 z_2 + \Delta_3 z_1 z_2}{\Delta_2 + \Delta_1 \Delta_2 z_1 + \Delta_1 \Delta_2 z_2 + \Delta_2 z_1 z_2}
\] (5)

It is evident that the above transfer function is characterized by the all-pass property as in [6].

4 3-D State Space Realization

In order to derive the state space matrices \(A, b, c'\) and the scalar \(d\) of 3-D state space model, we assume that the outputs of the delay elements \(z_1^{-1}, z_2^{-1}, z_3^{-1}\) correspond to the states of the 3-D model [12]. Following the same procedure as in the 2-D case, the state space sought is:

\[
\begin{align*}
\dot{x}(i_1, i_2, i_3) &= A x(i_1, i_2, i_3) + b u(i_1, i_2, i_3) \\
y(i_1, i_2, i_3) &= c' x(i_1, i_2, i_3) + d u(i_1, i_2, i_3)
\end{align*}
\] (6)

where

\[
x(i_1, i_2, i_3) = \begin{bmatrix} x^{d_1}(i_1 + 1, i_2, i_3) \\ x^{d_2}(i_1, i_2 + 1, i_3) \\ x^{d_3}(i_1, i_2, i_3 + 1) \end{bmatrix}
\]

\[
x(i_1, i_2, i_3) = \begin{bmatrix} x^{d_1}(i_1, i_2, i_3) \\ x^{d_2}(i_1, i_2, i_3) \\ x^{d_3}(i_1, i_2, i_3) \end{bmatrix}
\]

\[
A = \begin{bmatrix}
A_{11} & A_{12} & A_{13} \\
A_{21} & A_{22} & A_{23} \\
A_{31} & A_{32} & A_{33}
\end{bmatrix}
\]

Submatrix \(A_{11}\)

\[
a_{i,j} = \begin{cases} -\Delta_3(k+1-i-1)\Delta_3(k+1-j) & \text{if } i > j \\
0 & \text{otherwise}
\end{cases}
\]

Submatrix \(A_{12}\)

\[
a_{i,j} = \begin{cases} -\Delta_3(k+1-i-1)\Delta_3(k+1-j-1) & \text{if } i > j \\
1 - \Delta_2^2 & \text{if } i = j \\
0 & \text{otherwise}
\end{cases}
\]

Vector \(b\)

\[
b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}
\]

where

\[
b_1 = b_i = \Delta_2(k+1-i-1)
\]

\[
b_2 = b_i = \Delta_2(k+1-i-2)
\]

\[
b_3 = b_i = \Delta_2(k+1-i-3)
\]

vector \(c'\)

\[
c' = \begin{bmatrix} c_1 & c_2 & c_3 \end{bmatrix}
\]
where \[ c_1 = 1 - \Delta_3^2, c_2 = 0, c_3 = 0 \]
and the scalar
\[ d = \Delta_3 \]
Note that, \( \Delta_0 = 1 \).

4.1 Example: First-order 3-D filter

For the First-order 3-D filter, \( k = 1 \), the 3-D Givone-Roesser model (6) becomes,

\[
\begin{align*}
\dot{x}(i_1, i_2, i_3) &= Ax(i_1, i_2, i_3) + bu(i_1, i_2, i_3) \\
y(i_1, i_2, i_3) &= c'x(i_1, i_2, i_3) + du(i_1, i_2, i_3)
\end{align*}
\]  \tag{7}

where,
\[
A = \begin{bmatrix} -\Delta_2 \Delta_3 & 1 - \Delta_3^2 & 0 \\ -\Delta_1 \Delta_3 & -\Delta_1 \Delta_2 & 1 - \Delta_1^2 \\ -\Delta_3 & \Delta_2 & \Delta_1 \end{bmatrix}
\]
\[
b = \begin{bmatrix} \Delta_2 \\ \Delta_1 \\ 1 \end{bmatrix}
\]
\[
c' = \begin{bmatrix} 1 - \Delta_3^2 & 0 & 0 \end{bmatrix}
\]
\[
d = \Delta_3
\]

Using (2), the transfer function of the above model (7) is

\[
\begin{array}{c|c|c|c|c|c}
\text{num:} & 1 & z_1 & z_2 & z_2 z_3 & z_1 \\
\text{den:} & \Delta_3 & \Delta_1 \Delta_3 & \Delta_1 \Delta_2 \Delta_3 & \Delta_2 \Delta_3 & \Delta_3 \\
\hline
\text{num:} & \Delta_1 \Delta_3 & \Delta_1 z_3 & \Delta_2 z_3 & \Delta_3 \\
\text{den:} & \Delta_1 \Delta_2 & \Delta_1 & 1 \\
\end{array}
\]

It is evident that the above transfer function is characterized by the all-pass property.

5 N-D State Space Realization

Similarly as in the 3-D case, in order to derive the state space matrices \( A, b, c' \) and the scalar \( d \) of 2-D state space model, we assume that the outputs of the delay elements \( z_1^{-1}, z_2^{-1}, \ldots, z_k^{-1} \) correspond to the states of the model (1). Following the same procedure as for the 2 and 3-D case, the N-D state space model is:

\[
A = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1k} \\ A_{21} & A_{22} & \cdots & A_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ A_{k1} & A_{k2} & \cdots & A_{kk} \end{bmatrix}
\]

with,

\[
A_{11} = \begin{bmatrix} a_{11} & \cdots & a_{11}^k \\ \vdots & \ddots & \vdots \\ a_{11}^k & \cdots & a_{11}^k \end{bmatrix}
\]

where,
\[
a_{1,1}^{\lambda,\mu} = \begin{cases} -\Delta_N(k+1-\lambda)-1 \Delta_N(k+1-\mu)-1 & \text{if } \lambda \leq \mu \\ 0 & \text{otherwise} \end{cases}
\]

Submatrix \( A_{12} \)

\[
A_{12} = \begin{bmatrix} a_{12}^1 & \cdots & a_{12}^k \\ \vdots & \ddots & \vdots \\ a_{12}^k & \cdots & a_{12}^k \end{bmatrix}
\]

where,
\[
a_{1,2}^{\lambda,\mu} = \begin{cases} -\Delta_N(k+1-\lambda)-1 \Delta_N(k+1-\mu)-1 & \text{if } \lambda > \mu \\ 1 - \Delta_N(k+1) & \text{if } \lambda = \mu \\ 0 & \text{otherwise} \end{cases}
\]

Submatrix \( A_{13} \)

\[
A_{13} = \begin{bmatrix} a_{13}^1 & \cdots & a_{13}^k \\ \vdots & \ddots & \vdots \\ a_{13}^k & \cdots & a_{13}^k \end{bmatrix}
\]

where,
\[
a_{1,3}^{\lambda,\mu} = \begin{cases} -\Delta_N(k+1-\lambda)-1 \Delta_N(k+1-\mu)-2 & \text{if } \lambda > \mu \\ 0 & \text{otherwise} \end{cases}
\]

Submatrix \( A_{1N} \)

\[
A_{1N} = \begin{bmatrix} a_{1N}^1 & \cdots & a_{1N}^k \\ \vdots & \ddots & \vdots \\ a_{1N}^k & \cdots & a_{1N}^k \end{bmatrix}
\]

where,
\[
a_{1,N}^{\lambda,\mu} = \begin{cases} -\Delta_N(k+1-\lambda)-1 \Delta_N(k+1-\mu)-(N-1) & \text{if } \lambda > \mu \\ 0 & \text{otherwise} \end{cases}
\]

The overall algorithm, for determining the matrices \( A, b, c' \) and the scalar \( d \), for a N-D DTLBR function is:

FOR \( i \geq j \) THEN

\[
a_{i,j}^{\lambda,\mu} = \begin{cases} -\Delta_N(k+1-\lambda)-1 \Delta_N(k+1-\mu)-(j-1) & \text{if } \lambda \geq \mu \\ 0 & \text{otherwise} \end{cases}
\]
FOR $j = i + 1$ THEN

$$a_{\lambda,j}^{\mu} = \begin{cases} 
-\Delta_N^{(k+1-\lambda)-i} & \text{if } \lambda > \mu \\
\frac{1 - \Delta_N^{(k+1-\lambda)-i}}{\Delta_N^{(k+1-\lambda)-i}} & \text{if } \lambda = \mu \\
0 & \text{otherwise}
\end{cases}$$

FOR $i < j$ THEN

$$a_{i,j}^{\lambda,\mu} = \begin{cases} 
-\Delta_N^{(k+1-\lambda)-j} & \text{if } \lambda > \mu \\
0 & \text{otherwise}
\end{cases}$$

The column vector $b$ is

$$b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_k \end{bmatrix} = col[\delta_N^{(k+1-i)}]$$

The vector row vector $c'$ is

$$c' = [c_1 \ldots c_k] = [1 - \Delta_N^{k} 0 \ldots 0],$$

and the constant $d = \Delta_N^{k}$, where $i = 0, \ldots, N$, $j = 1, \ldots, N$, $\lambda = 1, \ldots, k$, $\mu = 1, \ldots, k$.

It is noted the presented results are easily verified by inspection rather than by algebraic manipulations.

6 Conclusion

In this paper an attempt has been made to review minimal state space realization for a class of N-D DTLBR functions. The circuit implementations have minimum number of delay elements and the state space matrices $A, b, c'$, and the constant $d$ of the Givone-Roesser model initially were derived by inspection. In this paper the state space matrices $A, b, c'$, and the constant $d$ of the Givone-Roesser model are given in closed form.

7 References


