ALGORITHMIC S-Z TRANSFORMATIONS FOR ANALOG-DIGITAL FILTER CONVERSION

Dalibor Biolek, Viera Biolková**)

Abstract: *This contribution links to publication [1] where a so-called generalized Pascal matrix has been introduced. Using this matrix, the coefficients of transfer functions of the continuous-time and discrete-time linear circuits can be recounted on the assumption that both circuits are connected by a general first-order s-z transformation. We explain the effective numerical procedure of computing all matrix elements for arbitrary first-order s-z transformation.*

1. Introduction

The so-called *s-z* transformations are often used for modeling interconnections between the continuous-time (CT) and discrete-time (DT) filters. The well-known transformations are:

bilinear (BL)

$$s = a\frac{z-1}{z+1}, \text{ or } z = \frac{a+s}{a-s}, \tag{1}$$

$$a = \frac{2}{T} = 2F. \tag{2}$$

backward-difference (BD)

$$sp = b\left(1 - z^{-1}\right) = b\frac{z-1}{z}, \text{ or } z = \frac{b}{b-s}, \tag{3}$$

$$b = \frac{1}{T} = F. \tag{4}$$

forward difference (FD)

$$s = b(z-1), \text{ or } z = 1 + \frac{s}{b}. \tag{5}$$

In equations (2) and (4), the symbols T / F represent sampling period / frequency of discrete-time filter.

In addition to the above *s-z* transformations, a set of so-called parametric transformations have been introduced. Their properties can be modified by an auxiliary parameter. To change it, we can pass from one to another known transformation. For example,

parametric BD-BL transformation [2]

$$s = \frac{1+r}{T}\frac{z-1}{z+r}, \text{ or } z = -\frac{1+r+srT}{sT-1-r} \tag{6}$$

is controlled by a parameter *r*. Changing it from 0 to 1, we pass from the BD to the BL transformation.

All known *s-z* transformations can be divided into first-order and higher-order ones [2]. The first-order transformations ensure that the order of the transformed filter will be the same as the order of the original filter. All the transformations mentioned above belong to the class of first-order transformations. It is shown in [1] the general form of first-order s-z transformation:

$$s = u\frac{1-z}{-v+wz}, \text{ or } z = \frac{u+vs}{u+ws}, \tag{7}$$

where u, $v \neq w$ are real constants. Choosing a proper set of constants leads to all the known transformations such as BL, BD, FD, and all parametric first-order ones.

2. Generalized Pascal matrix

Consider a DT linear filter with the transfer function

$$K_{DT}(z) = \frac{\sum_{k=0}^{N} a_k z^{-k}}{\sum_{k=0}^{N} b_k z^{-k}} = \frac{\sum_{k=0}^{N} a_k z^{N-k}}{\sum_{k=0}^{N} b_k z^{N-k}}, \tag{8}$$

where N is the filter order. The sampling frequency will be F and the sampling period T (see equation 2).

Consider the CT linear filter with the transfer function

$$K_{CT}(s) = \frac{\sum_{k=0}^{N} c_k s^{k}}{\sum_{k=0}^{N} d_k s^{k}}. \tag{9}$$

Let us assume that the DT filter has been designed from the CT prototype using the *s*-*z* transformation (7). Then an unambiguous relation between the *s*- and *z*- domain coefficients has to exist, independently for the numerator (a_k, c_k) and the denominator (b_k, d_k) [1]. For the numerator, this relation is derived in [1] using a matrix equation

$$\widetilde{\mathbf{C}} = \mathbf{MA}. \tag{10}$$

where the coefficients a_k of the DT circuit and the coefficients c_k of CT circuit are arranged to the vectors

$$\mathbf{A} = [a_0\ a_1\ a_2 \cdots a_N]^T, \quad \mathbf{C} = [c_0\ c_1\ c_2 \cdots c_N]^T, \tag{11}$$

and auxiliary vector $\widetilde{\mathbf{C}}$ is defined as follows:

$$\widetilde{\mathbf{C}} = \left[c_0 u^{-N}\ c_1 u^{1-N}\ c_2 u^{2-N} \cdots c_N u^{N-N}\right]^T. \tag{12}$$

In (10), $\mathbf{M}$ is a so-called $(N+1)$ x $(N+1)$ *generalized Pascal matrix* as a generalization of Pascal matrix [3,4] for BL transformation. Its components are defined by equation (13):

$$M_{ki} = \sum_{n=0}^{k} \binom{N-i}{n}\binom{i}{k-n} v^n w^{k-n} = \binom{i}{k} w^k + \sum_{n=1}^{k-1} \binom{N-i}{n}\binom{i}{k-n} v^n w^{k-n} + \binom{N-i}{k} v^k, \quad k,i = 0,1,\ldots N. \tag{13}$$

The boundary matrix elements are in a simple form given by equation (14). The other elements of the generalized Pascal matrix are determined by more complicated relations, see (13).

$$\begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ \binom{N}{1} v^1 & \cdots & \cdots & \cdots & \cdots & \binom{N}{1} w^1 \\ \binom{N}{2} v^2 & \cdots & \cdots & \cdots & \cdots & \binom{N}{2} w^2 \\ \binom{N}{3} v^3 & \cdots & \cdots & \cdots & \cdots & \binom{N}{3} w^3 \\ \vdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ \binom{N}{N} v^N & \cdots & \cdots & \cdots & \cdots & \binom{N}{N} w^N \end{bmatrix} \tag{14}$$

It is shown in [1] that the Pascal matrix can be easily compiled for particular *s*-*z* transformations (1), (3), (5) and (6). The examples for $N = 5$ are summarized in Table 1. In the trivial cases of BL, BD and FD transformations, we can derive algorithm how to compute all matrix elements from the boundary ones which are defined by equation (14). However, for a number of parametric and general first-order *s*-*z* transformations, this algorithm is not known at all. Then the complicated equation (13) represents a sole way of matrix element-to element compilation. This can be time-consuming especially for higher orders of transformed filters. In the following article we propose an effective numerical solution.

BL	BD	FD	parametric BL-BD
$u=a, v=1, w=-1$	$u=b, v=0, w=-1$	$u=b, v=1, w=0$	$u=\frac{1+r}{T}, v=r, w=-1$
$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 5 & 3 & 1 & -1 & -3 & -5 \\ 10 & 2 & -2 & -2 & 2 & 10 \\ 10 & -2 & -2 & 2 & 2 & -10 \\ 5 & -3 & 1 & 1 & -3 & 5 \\ 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & -1 & -2 & -3 & -4 & -5 \\ 0 & 0 & 1 & 3 & 6 & 10 \\ 0 & 0 & 0 & -1 & -4 & -10 \\ 0 & 0 & 0 & 0 & 1 & 5 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 5 & 4 & 3 & 2 & 1 & 0 \\ 10 & 6 & 3 & 1 & 0 & 0 \\ 10 & 4 & 1 & 0 & 0 & 0 \\ 5 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 5r & -1+4r & -2+3r & -3+2r & -4+r & -5 \\ 10r^2 & -4r+6r^2 & 1-6r+3r^2 & 3-6r+r^2 & 6-4r & 10 \\ 10r^3 & -6r^2+4r^3 & 3r-6r^2+r^3 & -1+6r-3r^2 & -4+6r & -10 \\ 5r^4 & -4r^3+r^4 & 3r^2-2r^3 & -2r+3r^2 & 1-4r & 5 \\ r^5 & -r^4 & r^3 & -r^2 & r & -1 \end{bmatrix}$

Tab.1. *An example of generalized Pascal matrices for N = 5 [1].*

3. Algorithm

As results from equation (13), the *k*-th column of the generalized Pascal matrix can be described by a convolution in regard of the subscript *k*:

$$M_{ki} = \sum_{n=0}^{k} \binom{N-i}{n}\binom{i}{k-n} v^n w^{k-n} = \left\{\binom{N-i}{k} v^k\right\} * \left\{\binom{i}{k} w^k\right\} \quad k,i = 0,1,\ldots N\,. \tag{15}$$

Applying Discrete Fourier transform (*DFT*) yields

$$M_i = DFT\{M_{ki}\} = DFT\left\{\binom{N-i}{k} v^k\right\} DFT\left\{\binom{i}{k} w^k\right\} \quad i = 0,1,\ldots N \tag{16}$$

where M_i is a vector of *DFT* coefficients of *i*-th column of the generalized Pascal matrix.

Equation (16) gives a rule how to compile (*N+1*) x (*N+1*) generalized Pascal matrix of general first-order *s-z* transformation with parameters *v* and *w*:

1. We set $i = 0$.
2. We compile vectors a_i and b_i of a size *1* x (*N+1*) according to the rule
$$a_i = \left[\binom{N-i}{0} v^0, \binom{N-i}{1} v^1, \binom{N-i}{2} v^2, \ldots, \binom{N-i}{N} v^N\right] \quad b_i = \left[\binom{i}{0} w^0, \binom{i}{1} w^1, \binom{i}{2} w^2, \ldots, \binom{i}{N} w^N\right].$$
3. We complete both vectors by as much zeros as needed to reach their length to integer power of two. New vectors a_i' and b_i' are obtained.
4. We perform Fast Fourier Transform (FFT) of vectors a_i' and b_i':
$$A_i' = FFT\{a_i'\} \quad B_i' = FFT\{b_i'\}.$$
5. We multiply vectors A_i' and B_i' element per element and perform inverse FFT:
$$M_{ki}' = IFFT\{A_i' B_i'\}$$
6. First $N+1$ elements of M_{ki}' give the *i*-th column of the generalized Pascal matrix.
7. $i = i+1$; if $i <= N$ then we go to the step 2 else end.

4. Example

Let us verify proposed algorithm for parametric BL-BD transformation with the parameter $r = 0.5$ and $N = 5$. According to the Table 1, $v = 0.5$, $w = -1$.

Following computation has been performed using MATLAB.

1. i = 0;
2. $a_i = [1\ \ 2.5\ \ 2.5\ \ 1.25\ \ 0.3125\ \ 0.03125]$, $b_i = [1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]$
3. $a_i^{'} = [1\ \ 2.5\ \ 2.5\ \ 1.25\ \ 0.3125\ \ 0.03125\ \ 0\ \ 0]$, $b_i^{'} = [1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]$
4. $A_i^{'} = [7.59375\ \ 1.54929\text{-}5.12955i\ \ -1.1875\text{-}1.28125i\ \ -0.17429\text{-}0.12955i\ \ 0.03125$ $-0.17429+0.12955i\ \ -1.1875+1.28125i\ \ 1.54929+5.12955i]$

   $B_i^{'} = [1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1\ \ 1]$
5. $A_i^{'}.B_i^{'} = [7.59375\ \ 1.54929\text{-}5.12955i\ \ -1.1875\text{-}1.28125i\ \ -0.17429\text{-}0.12955i$ 0.03125 $-0.17429+0.12955i\ \ -1.1875+1.28125i\ \ 1.54929+5.12955i]$

   $M_{ki}^{'} = IFFT\{A_i^{'}B_i^{'}\} = [1\ \ 2.5\ \ 2.5\ \ 1.25\ \ 0.3125\ \ 0.03125\ \ 0\ \ 0]$
6. $M_{ki} = [1\ \ 2.5\ \ 2.5\ \ 1.25\ \ 0.3125\ \ 0.03125]$, $i = 0$.

Setting $r = 0.5$ to the first column of Pascal matrix in the Table 1 leads to the verification of our result.

Alternatively, we can repeat our algorithm for $i = 1,2,3,4$, and 5 and compute the reminder matrix columns.

5. Conclusion

Proposed algorithm generates all elements of generalized Pascal matrix of arbitrary first-order *s*-*z* transformation (7). Described approach is based on the well-known FFT algorithm. This algorithm is applied N+1 times, where N is the order of continuous-or discrete-time filter whose coefficients are transformed. The structure of this procedure is not dependent on the type of *s*-*z* transformation. The required input data are N and the parameters v and w of a given *s*-*z* transformation.

Acknowledgement: *This publication has been supported by the Grant Agency of the Czech Republic, grant No. 102/98/P204.*

References:

[1] BIOLKOVÁ,V.-BIOLEK,D.: Generalized Pascal Matrix of First Order S-Z Transforms. Accepted for publication in ICECS´99 Pafos, Cyprus 1999.

[2] DOSTÁL,T.-POSPÍŠIL,J.: Switched circuits I. Rersearch Report No. FE-58, Dept. of Radioelectronics, TU Brno, 1985 (in Czech).

[3] KLEIN,W.: Finite Systemtheorie. Stuttgart, Treubner Studienbücher, 1976.

[4] PŠENIČKA,B.-PHAM KHAC DI: Transfer function computation using Pascal matrix. Electronic Horizon, 46, 1985, No. 7 (in Czech).

Prof. Dalibor Biolek, Inst. of Telecommunications, BUT Brno, Purkynova 118, 612 00 Brno, Czech Republic, phone (+420) 5 41149197, fax (+420) 5 41149192, email biolek@cs.vabo.cz

*) Ing. Viera Biolková, Inst. of Radioengineering, BUT Brno, Czech Republic, Purkynova 118, 612 00 Brno, Czech Republic, phone (+420) 5 41149148, fax (+420) 5 41149244, email biolkova@urel.fee.vutbr.cz