Inversion of Shortened LDI Matrix for Designing Discrete-time Filters

Dalibor Biolek*
Department of Electrical Engineering, Faculty of Military Technology, UD Brno, Czech Republic

Jaromir Kuben, Jr.
Department of Mathematics and Statistics, Faculty of Science, Masaryk University Brno, Czech Republic

Jaromir Kuben
Department of Mathematics and Physics, Faculty of Military Technology, UD Brno, Czech Republic

* Corresponding author. E-mail: dalibor.biolek@unob.cz
Received: 27 January 2014; Accepted: 28 February 2014; Published online: 6 March 2014
DOI: 10.14416/j.ijast.2014.01.009

Abstract
The paper deals with the problem of the inversion of the so-called shortened LDI matrix for conversion between the coefficients of continuous-time and discrete-time linear filters whose s- and z-domain transfer functions are coupled via LDI transform. The elements of the inverse matrix are expressed by analytic formulae for general order of the matrix. The corresponding mathematical proofs are given.

Keywords*: Lossless Discrete Integration, LDI matrix, Matrix Inversion*

1 Introduction

In engineering practice, the relationship between the models of continuous-time system and system originating from it by discretizing the time axis is analyzed in two different aspects.

The first one is a numerical solution of differential equations in simulation programs where the solution is searched via approximating difference equation. The corresponding methods, such as Backward Euler, Trapezoidal, Gear, and others, are then universally usable for solving linear and non-linear problems [1,2]. The concrete integration algorithm introduces an error to the solution of the differential equation due to approximation of the derivative by difference relations. The nature and the gravity of this error generally depend on the type of the approximation, on the system model, on the type of its excitation, and on the selection of the integration step and the method of its control. It turns out that some simulation models exhibit large sensitivity of the solution to the way of the time axis discretization, and that then the numerical solution can be far from the reality [3,4]. Such anomalous manifestations of the numerical methods increase the level of uncertainty of the designer about the correctness of the outputs generated by commercial simulation programs. Therefore they are currently analyzed at large, among other with the use of the theory of chaos [5].

The second aspect, falling into the operator domain, deals with the relationship between the transfer functions of linear continuous-time (CT) and discrete-time (DT) systems [6,7]. For linear systems, one-to-one relations exist between both areas which result from the well-known connections between the system behavior in time, operator, and spectral domains.

The paper is concerned with the problem from the second aspect. The bilinear (BL) *s*-*z* transform belongs to frequently utilized methods of designing linear DT system from its CT prototype [8,9]. Probably the first attempts at algorithmic solution of the task of the conversion of the coefficients of *s*-domain transfer function of CT system into coefficients of *z*-domain transfer function of DT system for BL transform and several other transformations are described in [10] and [11]. It is shown that the vectors of the coefficients are matched via the so-called Pascal matrix. Its elements as well as the elements of its inverse matrix can be easily generated for an arbitrary order of the system, thus for an arbitrary matrix size [12]. The Pascal matrix is used for digital filter design [13-16], in numerical mathematics [17-22], and for signal processing [23-25]. Some papers indicate that the Pascal matrix also attracts the attention of pure mathematicians [26].

However, the BL transform is only one of more utilized first-order *s*-*z* transformations for converting *s*-domain transfer function to *z*-domain transfer function of the same order. Among other transforms, let us mention BD (Backward-Difference), FD (Forward-Difference) [8], and parametric transforms, the latter representing combinations of the basic transformations [27,28]. For these reasons, the so-called generalized Pascal matrix is introduced in [29], which includes the classical Pascal matrix as a special case, concurrently enabling conversions of the coefficients of CT and DT systems for all existing first-order *s*-*z* transforms. The rules for compiling the generalized Pascal matrix and also the inverse matrix are described in [30]. The rigorous mathematical analysis of the generalized Pascal matrix for various *s*-*z* transforms is given in [31] and [32].

Another useful s-z transform, the LDI (Lossless Discrete Integration) [33], plays an important role since the eighties of the last century, either directly or in combination with the BL transform, for designing low-sensitive digital filters and switched-capacitor filters based on analog ladder structures [34,35]. At later stage, it was utilized for designing switched-current filters [36,37] and switched-capacitor sigma-delta AD converters [38]. For example, the LDI transform is used for designing digital filters with low complexity coefficients in [39], Lossless Discrete Differentiator (LDD) in [40], and digital allpass filters in [41].

In contrast with the BL, BD, FD transforms and the parametric transforms derived from them, the LDI is not the first-order *s*-*z* transformation, such that the generalized Pascal matrix cannot be used for this case. The corresponding computations are provided by the so-called LDI matrix, introduced in [42]. However, unlike Pascal matrix, the LDI matrix is not square because the DT system has more coefficients than the CT system. Owing to a certain symmetry of the vector of the coefficients of the DT system, all the information about the elements of the LDI matrix is contained in its specific part, forming a square submatrix, denoted in [43] as *shortened LDI matrix*. General properties of the original and shortened LDI matrices are analyzed in [43], and effective procedure for their composition for general order is found therein. However, the analytical formulae of the elements of the matrix, coming from the inversion of shortened LDI matrix for backward DT-CT transformation, have not been known till now.

This paper presents the inverse matrix of the shortened LDI matrix in the analytical form. That completes the missing part of the matrix algorithm for bi-directional transformation between *s* and *z* domains, which was formerly built for the first-order *s*-*z* transform via generalized Pascal matrix.

The paper has the following structure: Section 2, following this Introduction, summarizes the definition of LDI transform and the resultant definitions of complete and shortened LDI matrices. The inverse matrix of the shortened LDI matrix in the analytic form is presented in Section 3. Section 4 presents the respective mathematical proofs. Section 5 describes software experiments, demonstrating the correctness and usefulness of the proposed method of the matrix inversion.

2 LDI Transform and Its Matrix Description

The LDI transform is defined by the well-known equation [33]

$$s = \frac{1}{T}(z^{1/2} - z^{-1/2}) \tag{1}$$

where s and z are operators of the Laplace and z transforms, and T is the sampling period, utilized for time axis discretization during CT to DT transformation.

Applying the substitution

$$x = z^{1/2}, \; f_S = 1/T, \tag{2}$$

where f_S is the sampling frequency, Eq. (1) is modified to the form

$$s = f_S\left(x - \frac{1}{x}\right) = f_S\frac{x^2 - 1}{x}, \, x \neq 0. \tag{3}$$

Consider N-th order CT linear system with the transfer function

$$K_{CT}(s) = \frac{\sum_{k=0}^{N} a_k s^k}{\sum_{k=0}^{N} b_k s^k}. \tag{4}$$

Applying the LDI transform (3) to (4) and arrangement yield the transfer function of DT system [43]

$$K_{DT}(x) = \frac{\sum_{k=0}^{2N} c_k x^k}{\sum_{k=0}^{2N} d_k x^k}. \tag{5}$$

The c_k coefficients in the numerator can be obtained from the ak coefficients in the numerator of the transfer function (4) of CT system. Similarly, the d_k coefficients in the denominator can be derived from the b_k coefficients in the denominator of (4). The procedure is given by the following uniform algorithm [43]:

$$\mathbf{C} = \mathbf{L}\tilde{\mathbf{A}},\ \mathbf{D} = \mathbf{L}\tilde{\mathbf{B}}. \tag{6}$$

Here **C** and **D** are vectors of the coefficients of DT system

$$\mathbf{C} = \left[c_0\ c_1\ c_2 \ldots c_{2N}\right]^T,\ \mathbf{D} = \left[d_0\ d_1\ d_2 \ldots d_{2N}\right]^T, \tag{7}$$

and $\tilde{\mathbf{A}}$, $\tilde{\mathbf{B}}$ are vectors of modified coefficients of the CT system

$$\tilde{\mathbf{A}} = \left[\tilde{a}_0\ \tilde{a}_1\ \tilde{a}_2 \ldots \tilde{a}_N\right]^T,\ \tilde{\mathbf{B}} = \left[\tilde{b}_0\ \tilde{b}_1\ \tilde{b}_2 \ldots \tilde{b}_N\right]^T, \tag{8}$$

where

$$\tilde{a}_i = f_S^i\, a_i,\ \tilde{b}_i = f_S^i\, b_i,\ i = 0, 1, ..., N. \tag{9}$$

The transformation matrices **L** appearing in both formulae (6) are identical. In [42] and [43], **L** is denoted LDI matrix. Its structure is obvious from the following notation of the first equation in (6) [43]:

$$c_{N-k} = \sum_{i=0}^{\left[\frac{N-k}{2}\right]} (-1)^{k+i} \binom{k+2i}{i} \tilde{a}_{k+2i},\ 0 \le k \le N, \tag{10}$$

$$c_{N+k} = \sum_{i=0}^{\left[\frac{N-k}{2}\right]} (-1)^{i} \binom{k+2i}{i} \tilde{a}_{k+2i},\ 0 \le k \le N, \tag{11}$$

where the notation [ ] denotes the integer part of the argument. The proof is given in [43].

Example of the matrix formula (10), (11) for $N = 5$ is given below as Eq. (12). It demonstrates specific symmetry of the LDI matrix, which can be described as follows:

$$c_{N+k} = (-1)^k\, c_{N-k}, \quad 0 \le k \le N. \tag{13}$$

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ \hline c_6 \\ c_7 \\ c_8 \\ c_9 \\ c_{10} \end{bmatrix} = \left[\begin{array}{cccccc} 0 & 0 & 0 & 0 & 0 & -\binom{5}{0} \\ 0 & 0 & 0 & 0 & \binom{4}{0} & 0 \\ 0 & 0 & 0 & -\binom{3}{0} & 0 & \binom{5}{1} \\ 0 & 0 & \binom{2}{0} & 0 & -\binom{4}{1} & 0 \\ 0 & -\binom{1}{0} & 0 & \binom{3}{1} & 0 & -\binom{5}{2} \\ \binom{0}{0} & 0 & -\binom{2}{1} & 0 & \binom{4}{2} & 0 \\ \hline 0 & \binom{1}{0} & 0 & -\binom{3}{1} & 0 & \binom{5}{2} \\ 0 & 0 & \binom{2}{0} & 0 & -\binom{4}{1} & 0 \\ 0 & 0 & 0 & \binom{3}{0} & 0 & -\binom{5}{1} \\ 0 & 0 & 0 & 0 & \binom{4}{0} & 0 \\ 0 & 0 & 0 & 0 & 0 & \binom{5}{0} \end{array}\right] \begin{bmatrix} \tilde{a}_0 \\ \tilde{a}_1 \\ \tilde{a}_2 \\ \tilde{a}_3 \\ \tilde{a}_4 \\ \tilde{a}_5 \end{bmatrix} = \left[\begin{array}{cccccc} 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 & 0 & 5 \\ 0 & 0 & 1 & 0 & -4 & 0 \\ 0 & -1 & 0 & 3 & 0 & -10 \\ 1 & 0 & -2 & 0 & 6 & 0 \\ \hline 0 & 1 & 0 & -3 & 0 & 10 \\ 0 & 0 & 1 & 0 & -4 & 0 \\ 0 & 0 & 0 & 1 & 0 & -5 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \begin{bmatrix} \tilde{a}_0 \\ \tilde{a}_1 \\ \tilde{a}_2 \\ \tilde{a}_3 \\ \tilde{a}_4 \\ \tilde{a}_5 \end{bmatrix} \tag{12}$$

It is obvious that the complete LDI matrix can be unambiguously restored only from its upper square submatrix, which is expressed by Eq. (10). In [43], this matrix $\mathbf{L}_S$ is called shortened LDI matrix. Its analytical notation results from Eq. (10):

$$\begin{bmatrix} 0 & \cdots & 0 & 0 & 0 & (-1)^N\binom{N}{0} \\ 0 & \cdots & 0 & 0 & (-1)^{N-1}\binom{N-1}{0} & 0 \\ 0 & \cdots & 0 & (-1)^{N-2}\binom{N-2}{0} & 0 & (-1)^{N-1}\binom{N}{1} \\ 0 & \cdots & (-1)^{N-3}\binom{N-3}{0} & 0 & (-1)^{N-2}\binom{N-1}{1} & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \end{bmatrix} \quad (14)$$

Note that the shortened $(N+1)\times(N+1)$ LDI matrix can be used for transforming the coefficients of Nth-order CT system into a subset of the coefficients of DT system according to the rule

$$\mathbf{C}_S = \mathbf{L}_S\tilde{\mathbf{A}},\ \mathbf{D}_S = \mathbf{L}_S\tilde{\mathbf{B}}, \quad (15)$$

where $\mathbf{C}_S$ and $\mathbf{D}_S$ are shortened vectors of the coefficients of DT system

$$\mathbf{C}_S = \left[c_0\ c_1\ c_2 \ldots c_N\right]^T,\ \mathbf{D}_S = \left[d_0\ d_1\ d_2 \ldots d_N\right]^T. \quad (16)$$

The remaining coefficients, not included in the shortened vectors, can be easily determined according to the rule of symmetry (13).

Detailed analysis of the properties of shortened LDI matrix is given in [43]. Similar analysis is provided below for the inverse matrix to the matrix $\mathbf{L}_S$.

3 Inversion of Shortened LDI Matrix

Beside the shortened LDI matrix $\mathbf{L}_S$, we will also work with a lower square submatrix of $\mathbf{L}$ of size $(N+1)\times(N+1)$, which we denote $\mathbf{L}_S^*$. This matrix is upper triangular with ones on the main diagonal, hence it is invertible. The main reason why we prefer this matrix is that triangular matrices are easier to handle (unlike $\mathbf{L}_S$ which has zero elements above the antidiagonal).

In the sequel, for two integers p, q the symbol $p|q$ means that p is a divisor of q and $p\nmid q$ means that p is not a divisor of q.

Let us denote $\mathbf{L}_S^* = \left\{a_{r,s}\right\}_{r,s=0}^N$ and $(\mathbf{L}_S^*)^{-1} = \left\{b_{r,s}\right\}_{r,s=0}^N$. Then according to (11)

$$a_{r,s} = \begin{cases} (-1)^{\frac{s-r}{2}}\dbinom{s}{\frac{s-r}{2}} & \text{for } r \le s \text{ and } 2\left|(r+s)\right., \\ 0 & \text{otherwise.} \end{cases} \quad (17)$$

In the following section it will be proven that

$$b_{r,s} = \begin{cases} \dbinom{\frac{r+s}{2}}{r} + \dbinom{\frac{r+s}{2}-1}{r} & \text{for } r < s \text{ and } 2\left|(r+s)\right., \\ 1 & \text{for } r = s, \\ 0 & \text{otherwise.} \end{cases} \quad (18)$$

The proof is quite crucial. It is well-known that if an augmented matrix $(\mathbf{L}_S^*|\mathbf{E})$, where E is the identity matrix, is transformed using elementary row operations to a matrix $(\mathbf{E}|\mathbf{P})$, then $\mathbf{P} = (\mathbf{L}_S^*)^{-1}$. Proceeding this way, rather complicated recurrence formulae are obtained from which the closed form (18) can be guessed, although it is not trivial (two partially overlapping Pascal triangles occur there). Nevertheless, once the formula for the inverse matrix is predicted, it is possible to verify its correctness—the product with $\mathbf{L}_S^*$ must be equal to the identity matrix.

The inverse matrix $(\mathbf{L}_S)^{-1}$ can then easily be obtained as follows. Consider a matrix

$$\mathbf{M} = \begin{bmatrix} 0 & 0 & \ldots & 0 & (-1)^N \\ 0 & 0 & \ldots & (-1)^{N-1} & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & -1 & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & 0 \end{bmatrix}$$

having zeros outside the antidiagonal. It is orthogonal, i.e., $\mathbf{M}^{-1} = \mathbf{M}^T$. From (10) and (11) it is evident that $\mathbf{L}_S = \mathbf{M}\mathbf{L}_S^*$, hence it is sufficient to rearrange the rows of $\mathbf{L}_S^*$ in the reverse order and then multiply them through consequently by $(-1)^N$, $(-1)^{N-1}$, …, –1, 1. If we denote $\mathbf{L}_S = \left\{c_{r,s}\right\}_{r,s=0}^N$, we obtain

$$c_{r,s} = \begin{cases} (-1)^{\frac{s-r+N}{2}}\dbinom{s}{\frac{r+s-N}{2}} & \text{for } N \le r+s \text{ and } 2\left|(r+s-N)\right., \\ 0 & \text{otherwise.} \end{cases} \quad (19)$$

Further, $(\mathbf{L}_S)^{-1} = (\mathbf{L}_S^*)^{-1}\mathbf{M}^{-1} = (\mathbf{L}_S^*)^{-1}\mathbf{M}^{\mathrm{T}}$ Thus it is sufficient to rearrange the columns of $(\mathbf{L}_S^*)^{-1}$ in the reverse order and then multiply them through in consequence by $(-1)^N$, $(-1)^{N-1}$, …, –1, 1. If we denote $(\mathbf{L}_S)^{-1} = \{d_{r,s}\}_{r,s=0}^{N}$, we obtain

$$d_{r,s} = \begin{cases} (-1)^{N-s}\left[\binom{\frac{r+N-s}{2}}{r} + \binom{\frac{r+N-s}{2}-1}{r}\right] & \text{for } r+s<N \text{ and } 2\,|\,(N-r-s), \\ (-1)^r & \text{for } r+s=N, \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

4 Proof of the Formula for the Inverse of $\mathbf{L}_S^*$

Let us denote N_0 the set of all nonnegative integers. To prove the main result, we will need the following auxiliary statement.

Lemma For every a, $b \in \mathbb{N}_0$, there is

$$\sum_{t=0}^{b}(-1)^t\binom{a+2t}{t}\binom{a+b+t}{a+2t} = (-1)^b. \tag{21}$$

Proof. Let L denotes the left-hand side of (21). For every $t \in \{0,\dots, b\}$, the equality

$$\binom{a+2t}{t}\binom{a+b+t}{a+2t} = \frac{(a+2t)!(a+b+t)!}{t!(a+t)!(a+2t)!(b-t)!} =$$

$$= \frac{(a+b+t)!}{t!(a+t)!(b-t)!} = \frac{b!(a+b+t)!}{t!(b-t)!(a+t)!b!} =$$

$$= \binom{b}{t}\binom{a+b+t}{a+t}$$

holds. Using this formula and the substitution $t \to b-t$ we get

$L =$

$$\sum_{t=0}^{b}(-1)^t\binom{b}{t}\binom{a+b+t}{a+t} = \sum_{t=0}^{b}(-1)^{b-t}\binom{b}{b-t}\binom{a+2b-t}{a+b-t} =$$

$$= \sum_{t=0}^{b}(-1)^{b-t}\binom{b}{t}\binom{a+2b-t}{a+b-t} = \tag{22}$$

$$= (-1)^b\sum_{t=0}^{b}(-1)^t\binom{b}{t}\binom{a+2b-t}{a+b-t}.$$

Consider a polynomial $P(x, y) = (1-x+y)^b(1+y)^{a+b}$. Applying twice binomial formula we get

$$P(x,y) = \sum_{t=0}^{b}(-1)^t\binom{b}{t}x^t(1+y)^{a+2b-t} =$$

$$= \sum_{t=0}^{b}(-1)^t\binom{b}{t}x^t\left[\sum_{u=0}^{a+2b-t}\binom{a+2b-t}{u}y^u\right].$$

The coefficient at the term $x^t y^{a+b-t}$ is thus

$$(-1)^t\binom{b}{t}\binom{a+2b-t}{a+b-t}.$$

Therefore, if we put $y = x$, the coefficient at x^{a+b} will be

$$\sum_{t=0}^{b}(-1)^t\binom{b}{t}\binom{a+2b-t}{a+b-t}.$$

On the other hand, since $P(x,x)=(1+x)^{a+b}$, this coefficient is obviously equal to 1. According to (22) this completes the proof of lemma.

Now we are ready to prove the main result.

Theorem Let $\mathbf{A} = \{a_{r,s}\}_{r,s=0}^{N}$ and $\mathbf{B} = \{b_{r,s}\}_{r,s=0}^{N}$ be $(N+1) \times (N+1)$ matrices whose entries are given by (17) and (18), respectively. Then $\mathbf{B} = \mathbf{A}^{-1}$.

Proof. We will be done if we prove that $\mathbf{AB} = \mathbf{E}$. Let $\mathbf{F} = \mathbf{AB} = \{f_{r,s}\}_{r,s=0}^{N}$. Both $\mathbf{A}$ and $\mathbf{B}$ are upper triangular matrices with ones in the principal diagonal, which implies that matrix $\mathbf{F}$ has this property as well.

So we will be done if we prove that all the entries of $\mathbf{F}$ above the principal diagonal are zeros. For $r < s$, $2 \nmid (r+s)$ obviously $f_{r,s} = 0$. For $r < s$, $2|(r+s)$ we have

$$f_{r,s} = \sum_{t=0}^{\frac{s-r}{2}} a_{r,r+2t}b_{r+2t,s} = (-1)^{\frac{s-r}{2}}\binom{s}{\frac{s-r}{2}} +$$

$$+\sum_{t=0}^{\frac{s-r}{2}-1}(-1)^{\frac{r+2t-r}{2}}\binom{r+2t}{\frac{r+2t-r}{2}}\times$$

$$\times\left[\binom{\frac{r+2t+s}{2}}{r+2t} + \binom{\frac{r+2t+s}{2}-1}{r+2t}\right] =$$

$$= \sum_{t=0}^{\frac{s-r}{2}} (-1)^t \binom{r+2t}{t} \binom{r+\frac{s-r}{2}+t}{r+2t} +$$

$$+ \sum_{t=0}^{\frac{s-r}{2}-1} (-1)^t \binom{r+2t}{t} \binom{r+\frac{s-r}{2}-1+t}{r+2t} =$$

$$= (-1)^{\frac{s-r}{2}} + (-1)^{\frac{s-r}{2}-1} = 0$$

according to Lemma. This completes the proof.

5 Software-based Experiments

To verify the correctness and usefulness of the above analytical formulae of the inverse of the LDI matrix, the following computations in Maple 16 have been performed on PC with Intel i5-3337U CPU 1.8 GHz with 8 GB RAM.

Note that the computation with the LDI matrix and also with its inverse has two practical aspects, especially for higher matrix sizes, which are typical for designing digital filters of orders exceeding values of hundreds: the computation time and particularly the computation accuracy. The latter aspect is crucial if filter zeros and poles are to be evaluated from the filter coefficients. Since such computations are extremely sensitive to the truncation errors of the coefficients, these coefficients should be represented by higher number of digits than the commonly used format on PCs, double-precision binary floating-point, can provide.

Until explicit formula (20) was known, the only way to find the inverse of $\mathbf{L}_S^*$ was the use of some standard numerical method for the evaluation of the inverse matrix. Since entries of the inverse of $\mathbf{L}_S^*$ are integers and the inverse LDI transform is sensitive to rounding errors, symbolic computations should be used. As the following Table 1 shows, this way (see the rows "symbol.") is very slow for large *N*. Alternatively, software floating-point numbers with a lot of digits have to be used to obtain accurate results (see the rows "numer."). The entry "Digits" represents a number of digits which are necessary for providing numerically correct results. The conventional numeric calculation utilizing hardware 64-bit IEEE binary floating-point numbers is faster, but it gives incorrect results for *N*=100 and 200 and fails for *N*=1000. Therefore, formula (20) represents the true progress in the study of the inverse LDI transform.

Table 1: Computational times for various *N* (first column). Computing $\mathbf{L}_S^*$ from Eq. (14) (second column), computing $\mathbf{L}_S^*$ inverse from Eq. (20) (third column) and by numerical inversion (fourth column).

	***N*/Digits**	**Computational Time [sec]**		
		$\mathbf{L}_S^*$ **(14)**	$\mathbf{L}_S^{*-1}$ **(20)**	$\mathbf{L}_S^{*-1}$ **inversion**
symbol.	100/-	0.125	0.109	1.578
	200/-	0.828	0.703	11.125
	1000/-	88.875	63.578	1520.453
numer.	100/33	0.141	0.140	0.218
	200/68	0.860	0.703	1.969
	1000/347	101.844	65.922	244.360

6 Conclusions

Analytical formulae for the elements of the matrix, which is the inverse matrix of the shortened LDI matrix, are derived in the paper. That completes the last and hitherto missing part of the matrix algorithm for the conversion between the coefficients of CT and DT systems, which are interconnected via the LDI transformation. The experiments from Section 5 demonstrate the fundamental limits of conventional numerical inversion of large LDI matrices. Then the analytic formula for the inverse matrix, proposed in the paper, can overcome the numerical problems associated with the conventional inversion.

Acknowledgments

This work has been supported by the Project for the development of K215 and K217 Departments, UD Brno, Czech Republic.

References

[1] K. S. Kundert, *The Designer's Guide to SPICE and SPECTRE*, Kluwer Academic Publishers, 1995.

[2] R. M. Kielkowski, *Inside Spice*, New York: McGraw-Hill New York, 1998.

[3] J.H.E. Cartwright and O. Piro, "The Dynamics of Runge-Kutta Methods, *Int. J. Bifurcation and Chaos*, vol. 2, pp. 427-449, 1992,

[4] R. Tetzlaff et al., *Memristors and Memristive Systems*, New York: Springer, ISBN:

978-1-4614-9067-8, 2014.

[5] M.R. de Lima, F. Claro, W. Ribeiro, S. Xavier, and A.L-Castillo, "The Numerical Connection between Map and its Differential Equation: Logistic and Other Systems," *Int. J. Nonlinear Sci. Numer. Simul.*, vol. 14(1), pp. 77-85, 2013.

[6] H. Baher, *Signal Processing and Integrated Circuits*, John Wiley & Sons, Ltd., 2012.

[7] A. Oppenheim and R.W. Schafer, *Discrete Time Signal Processing*, 3rd ed., Prentice Hall Signal Processing, 2009.

[8] R. Unbehauen and A. Cichocki, *MOS Switched-Capacitor and Continuous-Time Integrated Circuits and Systems*, New York: Springer Verlag, 1989.

[9] S. K. Mitra, *Digital Signal Processing: A Computer-Based Approach*, 3rd ed., New York: McGraw-Hill, 2006.

[10] A.M. Bush and D.C. Fielder, "Simplified Algebra For Bilinear and Related Transformations," *IEEE Trans. On Audio and Electroacoustics*, vol. AU21(2), pp. 127-128, 1973.

[11] E. I. Jury and O.W.C. Chan, "Combinatorial Rules for Some Useful Transformations," *IEEE Trans. On CAS*, vol. CT-20(5), pp. 476-480, Sep. 1973.

[12] W. Klein, *Finite Systemtheorie. Stuttgart: Teubner Studienbücher*, 1976.

[13] B. Psenicka, F. Garcia-Ugalde, and A. Herrera-Cammacho, "The Bilinear z-Transform by Pascal Matrix and its Application in the Design of Digital Filters," *IEEE Signal Processing Lett.*, vol. 9(11), pp. 368-370, Nov. 2002.

[14] B. Psenicka and F. Garcia-Ugalde, "Z Transform from Lowpass to Bandpass by Pascal Matrix," *IEEE Signal Processing Lett.*, vol. 11(2), pp. 282-284, Feb. 2004,

[15] J. Konopacki, "The Frequency Transformation by Matrix Operation and Its Application in IIR Filter Design," *IEEE Signal Processing Lett.*, vol. 12(1), pp. 5-8, Jan. 2005,

[16] F.J. Garcia-Ugalde, B. Psenicka, and M.O. Jimenez-Salinas, "Z Transformation by Pascal Matrix and its Applications in the Design of IIR Filters," *Journal of Applied Research and Technology*, vol. 9(3), pp. 355-366, 2011.

[17] X. Wang and L. Z. Lu, "A Fast Algorithm for Solving Linear Systems of the Pascal Type," *Applied Mathematics and Computation*, vol. 175(1), pp. 441-451, 2006.

[18] Z. S. Kang, "An Analytical Pascal Matrix Transform for s-to-z Domain Transfer Functions," *IEEE Signal Processing Lett.*, vol. 13(10), pp. 597-600, Feb. 2006,

[19] X. Wang and J. T. Zhout, "A Fast Eigenvalue Algorithm for Pascal Matrices," *Applied Mathematics and Computation*, vol. 183(2), pp. 711-716, 2006.

[20] X. Wang, "A Stable Fast Algorithm for Solving Linear Systems of the Pascal Type," *Journal of Computational Analysis and Applications*, vol. 9(4), pp. 411-419, 2007.

[21] L. Xiao-Guang, H. Ting-Zhu, R. Zhi-Gang, "A new algorithm for linear systems of the Pascal typed," *Journal of Computational and Applied Mathematics*, vol. 225(1), pp. 309-315, 2009.

[22] P. Alonso, J. Delgado, R. Gallego et al., "Conditioning and accurate computations with Pascal matrices," *Journal of Computational and Applied Mathematics*, vol. 252, pp. 21-26, 2013.

[23] M. F. Aburdene and T. J. Goodman, "The discrete Pascal transform and its applications," *IEEE Signal Processing Lett.*, vol. 12(7), pp. 493-495, Jul. 2005,

[24] M. Aburdene and T. Goodman, "Discrete Polynomials and Filter Design," in *Proc. 2005 Conf. on Information Sciences and Systems*, The John Hopkins University, Mar. 16-18, 2005.

[25] A. N. Skodras, "Fast Discrete Pascal Transform," *Electronics Lett.*, vol. 42(23), pp. 1367-1368, Nov. 2006,

[26] B. Lewis, Revisiting the Pascal Matrix, *The American Mathematical Monthly*, vol. 117(1), pp. 50-66, 2010.

[27] T. Dostal and J. Pospisil, "*Switched Circuits I*," Research Report No. FE-58, Dept. of Radioelectronics, Czech Republic: TU Brno, 1985,

[28] Y.Y. Fukui, N. Yabuki, and A. Kosaka, "A New s-z transformation and its switched capacitor realization," in *Proc. ISCAS'88*, 1988, pp. 1999-2002.

[29] V. Biolkova and D. Biolek, "Generalized Pascal Matrix of First Order S-Z Transforms," in *Proc. ICECS'99*, Pafos, Cyprus, 1999, pp. 929-931.

[30] D. Biolek and V. Biolkova, "Algorithmic S-Z

transformations for continuous-time to discrete-time filter conversion," in *Proc. IEEE Int. Symp. Circuits Systems*, vol. 1, 2001, pp. 588-590.

[31] T.-B. Deng, S. Chivapreecha, and K. Dejhan, "General-ized Pascal Matrices, Inverses, Computations and Properties Using One-to-One Rational Polynomial s-z Transformations," *IEEE Trans. on CAS-I*, vol. 55(9), pp. 2650-2663, Oct. 2008.

[32] T.-B. Deng, S. Chivapreecha, and K. Dejhan, "Unified Pascal Matrix for First-Order s-z Domain Transformations," *IEEE Trans. on Signal Processing*, vol. 57(6), pp. 2130-2139, 2009.

[33] G. M. Jacobs, D. J. Allstot, R. W. Brodersen, and P. R. Gray, "Design Techniques for MOS Switched Capacitor Ladder Filters," *IEEE Trans. on CAS*, vol. CAS-25, pp. 1014-1021, Dec. 1978.

[34] M. S. Lee and C. Chang, "Low-Sensitivity Switched-Capacitor Ladder Filters," *IEEE Trans. on CAS*, vol. CAS-27(6), pp. 475-480, Jun. 1980,

[35] M. S. Lee and C. Chang, "Switched-Capacitor Filters Using the LDI and Bilinear Transformations," *IEEE Trans. on CAS*, vol. CAS-28(4), pp. 265-270, Apr. 1981.

[36] A.C. M. de Queiroz, "Component Simulation Switched-Current Filters Using Backward Euler Transformation," in *Proc. MWSCAS'06*, vol. 2, Aug. 2006, pp. 245-248.

[37] C. Toumazou, J. B. Hughes, and N.C. Battersby, "Switched/Currents and Analogue Technique for Digital Technology," *IEE Peter Peregrinus Ltd*., 1993.

[38] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*, Wiley-Interscience: IEEE Press, 2005.

[39] T.W. Fox and L.E. Turner, "The Design of a Lossless Discrete Integrator Digital Filter With Low Complexity Coefficients," in *Proc. IEEE Pacific RIM Conference on Communications, Computers and Signal Processing*, vol. 2, 2001, pp. 567-570.

[40] Y. Horio and S. Mori, "Switched-Capacitor Lossless Discrete Differentiator With Modified Sample-And-Hold Sequence," *Electronics Letters*, vol. 21(22), pp. 1036-1037, 1985,

[41] K. Landernäs, "Implementation of Digit-Serial LDI/LDD Allpass Filters," Ph.D. dissertation, Sweden: Mälardalen University Press Dissertation No. 23, 2006.

[42] V. Biolkova and D. Biolek, "LDI matrix and its utilization for the circuit analysis and design," in Proc. *Radioelektronika'99*, Brno, Czech Republic, 1999, pp. 38-41.

[43] V. Biolkova, Z. Kolka, and D. Biolek, "Algorithmic utilization of LDI transform for discrete-time filter design," in P*roc. Mosharaka International Conference on Communications, Signals and Coding*, MIC-CSC'08, Amman, Jordan, 2008, pp. 16-21.