Infinite Impulse Response Digital Filters Design

Phuoc Si Nguyen
Faculty of Electrical and Electronic Engineering, Department of School of Communications and Informatics, Victoria University of Technology, Melbourne, Australia
Email: nguyensiphuoc@frequencytransform.com

***Abstract*—Infinite Impulse Response (IIR) filters can be designed from an analogue low pass prototype by using frequency transformation in the s-domain and bilinear z-transformation with pre-warping frequency; this method is known as frequency transformation from the s-domain to the z-domain. This paper will introduce a new method to transform an IIR digital filter to another type of IIR digital filter (low pass, high pass, band pass, band stop or narrow band) using a technique based on inverse bilinear z-transformation and inverse matrices. First, a matrix equation is derived from inverse bilinear z-transformation and Pascal's triangle. This Low Pass Digital to Digital Filter Pascal Matrix Equation is used to transform a low pass digital filter to other digital filter types. From this equation and the inverse matrix, a Digital to Digital Filter Pascal Matrix Equation can be derived that is able to transform any IIR digital filter. This paper will also introduce some specific matrices to replace the inverse matrix, which is difficult to determine due to the larger size of the matrix in the current method. This will make computing and hand calculation easier when transforming from one IIR digital filter to another in the digital domain.**

***Index Terms*—bilinear z-transformation, frequency transformation, inverse bilinear z-transformation, IIR digital filters, warping frequency, pre_warping frequency**

I. Introduction

Currently, the most common method to design an Infinite Impulse Response (IIR) digital filter uses a reference analogue low pass prototype with a desirable class (such as Butterworth, Chebyshev or elliptic), then transforms it to another type of filter (low pass, high pass, band pass, band stop or narrow band filter) using frequency transformation in the s-domain and then converting the resulting analogue filter into an equivalent digital filter using bilinear z-transformation with pre-warping frequency [1], [2]. This method is known as frequency transformation in the s-domain to z-domain [3], [4] and is described mathematically by way of a matrix equation, called the Analog Low Pass Prototype to Digital Filter Pascal Matrix Equation. This paper will introduce a new method to design IIR digital filters from a low pass IIR digital filter using inverse bilinear z-transformation; with the support of the Pascal's triangle, a matrix equation is derived, called the Low Pass Digital to Digital Filter Pascal Matrix Equation. From this matrix equation and inverse matrix, an IIR digital filter can be designed from any type of IIR digital filter. This process is described in a general mathematical way as a Pascal matrix equation called the Digital to Digital Filter Pascal Matrix Equation. The main purpose of this equation is to simplify operations involving the matrix multiplication, which will make it more effective for programming and calculating transforming one IIR digital to another.

II. Analog Low Pass Prototype To Digital Filter Pascal Matrix Equation

Frequency transformation from the s-domain to the z-domain is a method to transform an analogue low pass prototype into a digital filter (low pass with different cut-off frequency, high pass, band pass, band stop or narrow band) as shown as a block diagram in Fig. 1.

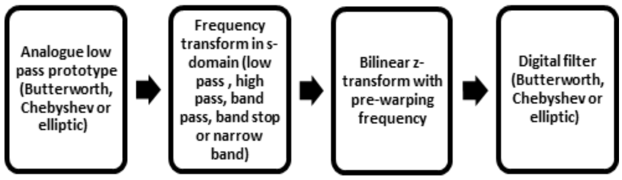

Figure 1. Frequency transformation from s-domain to z-domain

The technique uses one-to-one mapping poles and zeros on a stable region in the s-domain inside a unit circle in the z-domain [5]. The main advantage of this method is in transforming a stable designed analogue low pass prototype filter to a stable digital filter for which the frequency response has the same characteristics [6], [7] as those of the analogue low pass filter.

TABLE I. Transforming an Analogue Low Pass Prototype to a Digital Filter

Transforming	s=f(z)
Low pass to low pass	$c\frac{1-z^{-1}}{1+z^{-1}}$
Low pass to high pass	$t\frac{1+z^{-1}}{1-z^{-1}}$
Low pass to band pass	$U\frac{1-z^{-1}}{1+z^{-1}}+L\frac{1+z^{-1}}{1-z^{-1}}$
Low pass to band stop	$\frac{1}{U\frac{1-z^{-1}}{1+z^{-1}}+L\frac{1+z^{-1}}{1-z^{-1}}}$
Low pass to narrow band	$U_Q\frac{1-z^{-1}}{1+z^{-1}}+L_Q\frac{1+z^{-1}}{1-z^{-1}}$
Low pass to Notch	$\frac{1}{U_Q\frac{1-z^{-1}}{1+z^{-1}}+L_Q\frac{1+z^{-1}}{1-z^{-1}}}$

Manuscript received August 22, 2016; revised April 2, 2017.

Table I further details the block diagram shown in Fig. 1. It demonstrates how to transform a given analogue low pass prototype to a desired digital filter.

Let f_s be the sampling frequency; f_c, the cut-off frequency for the low pass and high pass digital filter; and f_U and f_L, the upper and lower frequency of the band pass and band stop. In the case of narrow band pass and notch digital filters, f_0 is the centre frequency and Q is a quality factor. The parameters c, t, U, L, U_Q and L_Q can be calculated as:

$$c = \cot\left(\pi \frac{f_c}{f_s}\right) \qquad t = \tan\left(\pi \frac{f_c}{f_s}\right)$$

$$\begin{cases} c_U = \cot\left(\pi \dfrac{f_U}{f_s}\right) \\ t_L = \tan\left(\pi \dfrac{f_L}{f_s}\right) \end{cases} \Rightarrow \begin{cases} U = \dfrac{c_U}{1 - c_U t_L} \\ L = \dfrac{t_L}{1 - c_U t_L} \end{cases}$$

$$\begin{cases} c_0 = \cot\left(\pi \dfrac{f_0}{f_s}\right) \\ t_0 = \tan\left(\pi \dfrac{f_0}{f}\right) \end{cases} \Rightarrow \begin{cases} U_Q = Qc_o \\ L_Q = Qt_0 \end{cases}$$

Let $H(s)$ be the transfer function of the analogue low pass prototype and $H(z)$, the transfer function of the digital filter. These can be written as below, where A_i, B_i, a_i and b_i are real coefficients, n is the nth order of the analogue low pass prototype and N is the Nth order of the digital filter. $N = n$ for low pass and high pass, and $N = 2n$ for band pass, band stop and narrow band filters:

$$H(s) = \frac{\sum_{i=0}^{n} A_i s^i}{\sum_{i=0}^{n} B_i s^i} = \frac{A_0 + A_1 s + \ldots + A_n s^n}{B_0 + B_1 s + \ldots + B_n s^n}$$

$$H(z) = \frac{\sum_{i=0}^{N} a_i z^{-i}}{\sum_{i=0}^{N} b_i z^{-i}} = \frac{a_0 + a_1 z^{-1} + \ldots + a_n z^{-N}}{b_0 + b_1 z^{1} + \ldots + b_n z^{-N}}$$

Following Table I, the relationship between coefficients A_i and B_i of the given analogue low pass prototype filter, and coefficients a_i and b_i of the desired digital filter, can be formulated as a matrix equation called the Analog Low Pass Prototype to Digital Filters Pascal Matrix Equation:

$$\begin{cases} \left[a_{i=0\to N}\right]_{(N+1;1)} = \left[P\right]_{(N+1;N+1)} \left(\left[A_{i=0\to n}\right]_{(1;n+1)} \left[T\right]_{(n+1;N+1)}\right)_{(N+1;1)} \\ \left[b_{i=0\to N}\right]_{(N+1;1)} = \left[P\right]_{(N+1;N+1)} \left(\left[B_{i=0\to n}\right]_{(1;n+1)} \left[T\right]_{(n+1;N+1)}\right)_{(N+1;1)} \end{cases} \tag{1}$$

A. Matrix [P]

Matrix [P] contains the positive and negative binomial coefficients of the Pascal's triangle. There are two types of matrix [P]: [P_{LP}] for a digital low pass filter and [P_{HBS}] for digital high pass, band pass, band stop and narrow band filters. These can defined as:

$$P_{LP} = \left[\begin{cases} \left(P_{LP}\right)_{i=1;\, j=1\to n+1} = 1 \\ \left(P_{LP}\right)_{i=n+1;\, j=1\to n+1} = (-1)^{j-1} \\ \left(P_{LP}\right)_{i=1\to n+1;\, j=1} = \binom{n}{i-1} \\ \left(P_{LP}\right)_{i=1\to n+1;\, j=n+1} = (-1)^{i-1}\binom{n}{i-1} \\ \left(P_{LP}\right)_{i;j} = \left(P_{LP}\right)_{i;j-1} - \left(P_{LP}\right)_{i-1;j-1} - \left(P_{LP}\right)_{i-1;j} \end{cases}\right]$$

Let [I] be an anti-diagonal unit matrix:

$$I = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & \iddots & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}_{(N+1;N+1)}$$

The matrix [P_{HBS}] can be calculated as [P_{HBS}] = [IP_{LP}].

B. Matrix [T]

Depending on the conversion, matrix [T] is either [T_x] or [T_{UL}]. For low pass to low pass filters x is replaced by c; for low pass to high pass x is replaced by t in matrix [T_x]. Matrix [T_{UL}] can be derived from the Pascal's triangle expansion of $(U+L)^n$ by inserting zeros. This matrix is used when transforming low pass to band pass, band stop and narrow band filters:

$$T_x = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & x & 0 & 0 & 0 \\ 0 & 0 & x^2 & 0 & 0 \\ 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & 0 & x^n \end{bmatrix}$$

$$T_{UL} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & U & 0 & L & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & U^2 & 0 & 2UL & 0 & L^2 & 0 & 0 & 0 \\ 0 & 0 & U^3 & 0 & 3U^2L & 0 & 3UL^2 & 0 & L^3 & 0 & 0 \\ 0 & \iddots & 0 & \ldots & 0 & \ldots & 0 & \ldots & 0 & \ddots & 0 \\ U^n & 0 & \ldots & 0 & \ldots & 0 & \ldots & 0 & \ldots & 0 & L^n \end{bmatrix}$$

The procedure of converting an analog low pass filter to a digital filter were studied, the next section will introduce how to transform a digital low pass filter to another type of digital filter.

III. Digital Low Pass To Digital Filter Pascal Matrix Equation

Following Table I, transforming from an analogue low pass prototype to a low pass digital filter with the cut-off frequency f_c can be done using $s = c[(1 - z^{-1)}/(1 + z^{-1})]$. Applying inverse bilinear z-transformation [1], a low pass digital filter can be converted back to an analogue low pass filter as:

$$z^{-1} = \frac{c - s}{c + s} \tag{2}$$

Let f_{cn} be a new cut-off frequency of the low pass digital filter and $c_n = \cot[x(f_{cn}/f_s)]$. From Table I and (2), a digital filter can be designed from a given low pass digital filter as shown in Table II.

TABLE II. Transforming an Analogue Low Pass Prototype to a Digital Filter

Transforming	z=Z(z^{-1})
Low pass to low pass	$\frac{c - c_N + (c + c_N)z^{-1}}{c + c_N + (c - c_N)z^{-1}}$
Low pass to high pass	$\frac{c - t - (c + t)z^{-1}}{c + t - (c - t)z^{-1}}$
Low pass to band pass	$\frac{c - U - L + 2(U - L)z^{-1} - (U + L + c)z^{-2}}{c + U + L - 2(U - L)z^{-1} + (U + L - c)z^{-2}}$
Low pass to band stop	$\frac{cU + cL - 1 - 2c(U - L)z^{-1} + (cU + cL + 1)z^{-2}}{cU + cL + 1 - 2c(U - L)z^{-1} + cU + cL - 1)z^{-2}}$

In the case of a narrow band, U and L are replaced by U_Q and L_Q. Let $H_g(z)$ be a transfer function for a given low pass digital filter and $H_d(z)$, a transfer function for a desired digital filter, as shown in (3) and (4):

$$H_{LP}(z)=\frac{\sum_{i=0}^{n} a_{LP(i)} z^{-i}}{\sum_{i=0}^{n} b_{LP(i)} z^{-i}} \qquad H_d(z)=\frac{\sum_{i=0}^{n} a_{d(i)} z^{-i}}{\sum_{i=0}^{n} b_{d(i)} z^{-i}}$$

The relationship between a_{LP}, b_{LP}, a_d and b_d can be described as a matrix equation, the Low Pass Digital to Digital Filter Pascal Matrix Equation:

$$\begin{cases} a_d = a_{LP} P_{LP}^{tr} T_{LP} T P^{tr} \\ b_d = b_{LP} P_{LP}^{tr} T_{LP} T P^{tr} \end{cases} \tag{3}$$

where the matrices $[P_{LP}^{tr}]$ and $[P^{tr}]$ are the transpose of the matrix $[P]$. Matrix $[T_{LP}]$ can be found from a matrix $[T_y]$ by replacing c with y as follows:

$$T_y=\begin{bmatrix} y^n & 0 & 0 & 0 & 0 \\ 0 & y^{n-1} & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & y & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The next section will introduce an alternative method to transform a low pass digital filter into another digital filter by using inverse matrices. From that, a general matrix equation is derived that is able to transform any digital filter to another.

IV. IIR Digital Filter Frequency Transformation

As discussed in Section III, a desired low pass, high pass, band pass, band stop or narrow band digital filter can be designed by transforming a given low pass digital filter. The process design is simple and uses a matrix equation as described in (3). Also from this matrix equation, a desired digital filter can be re-transformed to a given low pass digital as follows:

$$\begin{cases} a_{LP} = a_d \left[P^{tr}\right]^{-1} [T]^{-1} [T_{LP}]^{-1} \left[P_{LP}^{tr}\right]^{-1} \\ b_{LP} = b_d \left[P^{tr}\right]^{-1} [T]^{-1} [T_{LP}]^{-1} \left[P_{LP}^{tr}\right]^{-1} \end{cases} \tag{4}$$

Let subscript 'g' denote 'given' and 'd', 'desired'. From (3) and (4), the coefficients a_d and b_d of a desired digital filter can be found from a_g and b_g of a given digital filter as follows:

$$\begin{cases} a_d = a_g \left[P^{tr}\right]^{-1} \left[T_g\right]^{-1} [T_d] \left[P_d^{tr}\right] \\ a_d = a_g \left[P^{tr}\right]^{-1} \left[T_g\right]^{-1} [T_d] \left[P_d^{tr}\right] \end{cases} \tag{5}$$

The inclusion of the inverse matrices $[P^{tr}]^{-1}$ and $[T_g]^{-1}$ makes the matrix in (5) very large, so computing and hand calculation is difficult. To solve this problem, some features of matrices are considered in the next section.

A. Inverse Matrix $[P^{tr}]^{-1}$

Multiplying matrix $[P^{tr}]$ with size $(N+1,N+1)$ by itself will give a diagonal matrix in which all the numbers in the diagonal are equal to 2^N. This means that an inverse matrix $[P^{tr}]^{-1}$ is equal to matrix $[P^{tr}]$, in which all the numbers divide by 2^N as follows:

$$\left[P^{tr}\right]^{-1} = \frac{1}{2^n} P^{tr}$$

B. Inverse Matrix $[T_g]^{-1}$

$[T_x]$ and $[T_y]$ are diagonal matrices, so the inverse can be obtained by replacing each element in the diagonal with its reciprocal as follows:

$$\left[T_x\right]^{-1} = \left[\left(\frac{1}{x}\right)_{i=j}^{i-1}\right]_{(n+1;n+1)} \qquad \left[T_y\right]^{-1} = \left[\left(\frac{1}{y}\right)_{i=j}^{i-1}\right]_{(n+1;n+1)}$$

The inverses of $[T_x]$ and $[T_y]$ are used for low pass and high pass filters. In the case of band pass, band stop and narrow band filters, a matrix $[T_h]$ is used. $[T_h]$ is an upper-left triangular matrix of size $(n+1; n+1)$. If the main anti-diagonal is the first, then all the odd anti-diagonals above it are the 3rd, 5th, 7th, … and correspond to m = 1,2,3,4, … Each element in the mth anti-diagonal can be expressed in the formula as:

$$\left(T_h\right)_{(n+1-2(m-1)-j+1;j)} = K_{(n+1-2(m-1)-j+1;j)} \left(\frac{L}{U}\right)^{m-1} U^{-j+1}$$

The coefficients K can be calculated because all the K in the main anti-diagonals are equal to 1 and from m = 2, 3, 4, 5, … all the K in the first column are accordingly replaced with −2, +2, −2, +2, … ; all other K in all anti-diagonals can be found by:

$$K_{(n+1-2(m-1)-j+1;j)} = K_{(n+1-2(m-1)-j+2;j)} - K_{(n+1-2(m-1)-j+3;j)}$$

All other elements in the matrix $[T_h]$ equal to zero.

The inverse matrices $[P^{tr}]^{-1}$ and $[T_g]^{-1}$ are illustrated above without using the determinant matrix, which is very difficult to calculate due to its very large size. They are used in matrix equations to transform a digital filter to another digital filter as shown in detail as follows:

- Low pass digital to digital filters

$$\begin{cases} \left[a_d\right]_{(1;N+1)} = \left[a_{LP}\right]_{(1;n+1)} \left[P_{LP}^{tr}\right]_{(n+1;n+1)} \left[T_x^{x=t}\right]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \\ \left[b_d\right]_{(1;N+1)} = \left[b_{LP}\right]_{(1;n+1)} \left[P_{LP}^{tr}\right]_{(n+1;n+1)} \left[T_x^{x=t}\right]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \end{cases} \tag{6}$$

- High pass digital to digital filters

$$\begin{cases} \left[a_d\right]_{(1;N+1)} = \left[a_{HP}\right]_{(1;n+1)} \left[P_{HBS}^{tr}\right]_{(n+1;n+1)} \left[T_y^{y=c}\right]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \\ \left[b_d\right]_{(1;N+1)} = \left[b_{HP}\right]_{(1;n+1)} \left[P_{HBS}^{tr}\right]_{(n+1;n+1)} \left[T_y^{y=c}\right]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \end{cases} \tag{7}$$

- Band pass to digital filters

$$\begin{cases} \left[a_d\right]_{(1;N+1)} = \left(\left[a_{BP}\right]\left[P_{HBS}^{tr}\right]\right)_{(1;n+1)} \left[T_h\right]_{(n+1;n+1)} [I]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \\ \left[b_d\right]_{(1;N+1)} = \left(\left[b_{BP}\right]\left[P_{HBS}^{tr}\right]\right)_{(1;n+1)} \left[T_h\right]_{(n+1;n+1)} [I]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \end{cases} \tag{8}$$

- Band stop to digital filters

$$\begin{cases} \left[a_d\right]_{(1;N+1)} = \left(\left[a_{BS}\right]\left[P_{HBS}^{tr}\right]\right)_{(1;n+1)} \left[T_h\right]_{(n+1;n+1)} [I]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \\ \left[b_d\right]_{(1;N+1)} = \left(\left[b_{BS}\right]\left[P_{HBS}^{tr}\right]\right)_{(1;n+1)} \left[T_h\right]_{(n+1;n+1)} [I]_{(n+1;n+1)} [T]_{(n+1;N+1)} \left[P^{tr}\right]_{(N+1;N+1)} \end{cases} \tag{9}$$

In the case of converting band pass to band pass and band stop to band stop, there is no anti-diagonal unit matrix [I] in (8) and (9). In the case of a narrow band to digital filter, U and L are replaced by UQ and LQ in the matrix [Th].

Equations (6), (7), (8) and (9) can be described as general matrix equations, or Digital to Digital Filter Pascal Matrix Equations, which can transform a given digital filter (the subscript 'g') into a desired digital filter (subscript 'd') as follows:

$$\begin{cases} a_d = a_g P_g T_g T_d P_d \\ b_d = b_g P_g T_g T_d P_d \end{cases} \tag{10}$$

In (10), a_g and b_g are the coefficients of the given digital filter. Matrices [P] and [T] can be found from the Pascal's triangle and their size depends on the Nth order of the given digital filter. The only component that needs to be controlled in (10) is matrix [T_d]. If the parameters (c, t, U and L) change, the cut-off frequency of the desired low pass and high pass, or the upper and lower frequency of the band pass, band stop and narrow band will differ. Thus, the process design of a digital filter using the Digital to Digital Filter Pascal Matrix Equation will be easier for programing and hand calculation.

V. Design of a Digital Filter Using the Digital to Digital Filter Pascal Matrix Equation

This section will provide some examples of transforming a digital to another digital filter using the Digital to Digital Filter Pascal Matrix Equation with Matlab programing.

A. *Example 1: Transforming a Low Pass Digital Filter to another Low Pass Digital Filter*

Transforming a 3rd-order Butterworth low pass digital filter with the transfer function $H(z)$ at cut-off frequency f_c = 200 Hz, to a low pass digital filter with cut-off frequency 400 Hz at the sampling frequency 1000 Hz (see Fig. 2), is done as follows:

$$H(z) = \frac{0.09853 + 0.2956z^{-1} + 0.2956z^{-2} + 0.09853z^{-3}}{1 - 0.5772z^{-1} + 0.4218z^{-2} - 0.0563z^{-3}}$$

$$\begin{cases} a_g = [0.0985 \quad 0.2956 \quad 0.2956 \quad 0.0985] \\ b_g = [1 \quad -0.5772 \quad 0.4218 \quad -0.0563] \end{cases}$$

$$t = \tan\left(\pi \frac{f_c}{f_s}\right) = \tan\left(\pi \frac{200}{1000}\right) = 0.7265$$

$$c_n = \cot\left(\pi \frac{f_{cn}}{f_s}\right) = \cot\left(\pi \frac{400}{1000}\right) = 0.3249$$

$$a_d = [0.0985 \quad 0.2956 \quad 0.2956 \quad 0.0985]\begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix}$$
$$\times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.7265 & 0 & 0 \\ 0 & 0 & 0.5279 & 0 \\ 0 & 0 & 0 & 0.3835 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.3249 & 0 & 0 \\ 0 & 0 & 0.1056 & 0 \\ 0 & 0 & 0 & 0.0343 \end{bmatrix}$$
$$\times \begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix} = [0.7882 \quad 2.3647 \quad 2.3647 \quad 0.7882]$$

$$b_d = [1 \quad -0.5772 \quad 0.4218 \quad -0.0563]\begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix}$$
$$\times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.7265 & 0 & 0 \\ 0 & 0 & 0.5279 & 0 \\ 0 & 0 & 0 & 0.3835 \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.3249 & 0 & 0 \\ 0 & 0 & 0.1056 & 0 \\ 0 & 0 & 0 & 0.0343 \end{bmatrix}$$
$$\times \begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix} = [1.4940 \quad 2.6294 \quad 1.7672 \quad 0.4154]$$

$$H_d(z) = \frac{0.7882 + 2.365z^{-1} + 2.365z^{-2} + 0.7882z^{-3}}{1.494 + 2.629z^{-1} + 1.767z^{-2} + 0.4154z^{-3}}$$

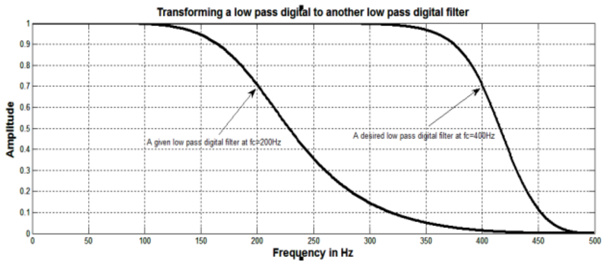

Figure 2. Transforming a digital low pass at f_c=200Hz to a digital low pass at f_c=400Hz

B. *Example 2: Transforming a High Pass Digital Filter to another High Pass Digital Filter*

Transforming a 3rd-order Butterworth high pass digital filter with the transfer function $H(z)$ at cut-off frequency f_c = 200 Hz, to a high pass digital filter with the cut-off frequency 400 Hz at the sampling frequency 1000 Hz (see Fig. 3), is done as follows:

$$H(z) = \frac{0.2569 - 0.7707z^{-1} + 0.7707z^{-2} - 0.2569z^{-3}}{1 - 0.5772z^{-1} + 0.4218z^{-2} - 0.0563z^{-3}}$$

$$\begin{cases} a_g = [0.2569 \quad -0.7707 \quad 0.7707 \quad -0.2569] \\ b_g = [1 \quad -0.5772 \quad 0.4218 \quad -0.0563] \end{cases}$$

$$c = \cot\left(\pi \frac{f_c}{f_s}\right) = \tan\left(\pi \frac{200}{1000}\right) = 1.3764$$

$$t_n = \tan\left(\pi \frac{f_{cn}}{f_s}\right) = \tan\left(\pi \frac{400}{1000}\right) = 3.0777$$

From (7):

$$a_d = [0.2569 \quad -0.7707 \quad 0.7707 \quad -0.2569]\begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix}$$
$$\times \begin{bmatrix} 0 & 0 & 0 & 2.6075 \\ 0 & 0 & 1.8944 & 0 \\ 0 & 1.3764 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 3.0777 & 0 \\ 0 & 9.4721 & 0 & 0 \\ 29.1522 & 0 & 0 & 0 \end{bmatrix}$$
$$\times \begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix} = [2.0553 \quad -6.166 \quad 6.166 \quad -2.0553]$$

$$b_d = [1 \quad -0.5772 \quad 0.4218 \quad -0.0563]\begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix}$$
$$\times \begin{bmatrix} 0 & 0 & 0 & 2.6075 \\ 0 & 0 & 1.8944 & 0 \\ 0 & 1.3764 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 3.0777 & 0 \\ 0 & 9.4721 & 0 & 0 \\ 29.1522 & 0 & 0 & 0 \end{bmatrix}$$
$$\times \begin{bmatrix} 1 & 3 & 3 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -3 & 3 & -1 \end{bmatrix} = [113.5606 \quad 199.8713 \quad 134.3300 \quad 31.5766]$$

$$H_d(z) = \frac{2.055 - 6.166z^{-1} + 6.166z^{-2} - 2.055z^{-3}}{113.6 + 199.9z^{-1} + 134.3z^{-2} + 31.58z^{-3}}$$

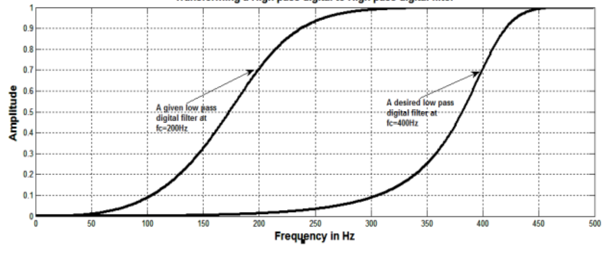

Figure 3. Transforming a digital high pass at f_c=200Hz to a digital high pass at f_c=400Hz

C. Example 3: Transforming a Band Pass Digital Filter to a Narrow Band Pass Digital Filter

Transforming a 2nd-order Butterworth band pass digital filter with the transfer function $H(z)$ with lower frequency $f_L = 100$ Hz and upper frequency $f_U = 300$ Hz, to a narrow band pass digital filter with $Q = 50$ and central frequency $f_0 = 200$ Hz (see Fig. 4), at a sampling frequency of 1000 Hz is done as follows:

$$H(z) = \frac{0.4208 - 0.4208z^{-2}}{1 - 0.4425z^{-1} + 0.1584}$$

$$\begin{cases} a_g = \begin{bmatrix} 0.4208 & 0 & -0.4208 \end{bmatrix} \\ b_g = \begin{bmatrix} 1 & -0.4425 & 0.1584 \end{bmatrix} \end{cases} \quad \begin{cases} c_U = \cot\left(\pi \dfrac{f_U}{f_s}\right) = \cot\left(\pi \dfrac{300}{1000}\right) = 0.7265 \\ t_L = \tan\left(\pi \dfrac{f_L}{f_s}\right) = \cot\left(\pi \dfrac{100}{1000}\right) = 0.3249 \end{cases}$$

$$\begin{cases} U = \dfrac{c_U}{1 - c_U t_L} = 0.9511 \\ L = \dfrac{t_L}{1 - c_U t_L} = 0.4253 \end{cases} \quad \begin{cases} U_Q = Q\cot\left(\pi \dfrac{200}{1000}\right) = 68.8191 \\ L_Q = Q\tan\left(\pi \dfrac{200}{1000}\right) = 36.3271 \end{cases}$$

From (8):

$$a_d = \begin{bmatrix} 0 & 1.6832 \end{bmatrix} \begin{bmatrix} 0 & 1.0515 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 36.3271 & 0 & 68.8191 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & -2 & 1 \end{bmatrix} = \begin{bmatrix} 1.6832 & 0 & -1.6832 \end{bmatrix}$$

$$b_d = \begin{bmatrix} 1.6008 & 1.6832 \end{bmatrix} \begin{bmatrix} 0 & 1.0515 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 36.3271 & 0 & 68.8191 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & -2 & 1 \end{bmatrix} = \begin{bmatrix} 178.6686 & -109.383 & 175.3022 \end{bmatrix}$$

$$H_d(z) = \frac{1.683 - 1.683z^{-2}}{178.7 - 109.4z^{-1} + 175.3z^{-2}}$$

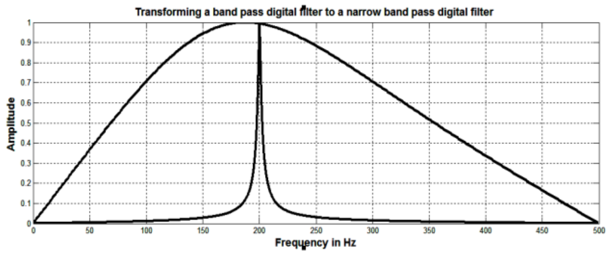

Figure 4. Transforming a digital band pass with f_L=100Hz and f_U=300Hz to a digital narrow band pass at f_0=200Hz, Q=50.

D. Example 4: Transforming a Band Stop Digital Filter to a Notch Digital Filter

Transforming a 2nd-order Butterworth band stop digital filter with transfer function $H(z)$, lower frequency $f_L = 100$ Hz and upper frequency $f_U = 300$ Hz, to a notch digital filter with $Q = 50$ and central frequency $f_0 = 200$Hz at a sampling frequency of 1000 Hz (see Fig. 5) is done as follows:

$$H(z) = \frac{0.5792 - 0.4425z^{1} + 0.5792z^{-2}}{1 - 0.4425z^{-1} + 0.1584z^{-2}}$$

$$\begin{cases} a_g = \begin{bmatrix} 0.5792 & -0.4425 & 0.5792 \end{bmatrix} \\ b_g = \begin{bmatrix} 1 & -0.4425 & 0.1584 \end{bmatrix} \end{cases}$$

From (9):

$$a_d = \begin{bmatrix} 1.6008 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1.0515 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 36.3271 & 0 & 68.8191 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & -2 & 1 \end{bmatrix} = \begin{bmatrix} 176.9854 & -109.383 & 176.9854 \end{bmatrix}$$

$$b_d = \begin{bmatrix} 1.6008 & 1.6832 \end{bmatrix} \begin{bmatrix} 0 & 1.0515 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 36.3271 & 0 & 68.8191 \end{bmatrix} \times \begin{bmatrix} 1 & 2 & 1 \\ 1 & 0 & -1 \\ 1 & -2 & 1 \end{bmatrix} = \begin{bmatrix} 178.6686 & -109.383 & 175.3022 \end{bmatrix}$$

$$H_d(z) = \frac{177 - 109.4z^{-1} + 177z^{-2}}{178.7 - 109.4z^{-1} + 175.3z^{-2}}$$

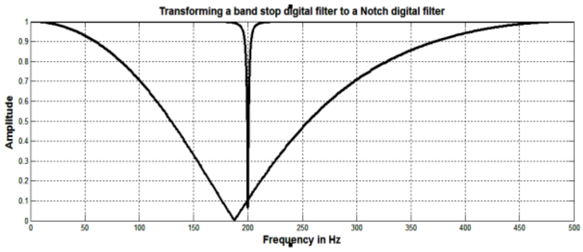

Figure 5. Transforming a digital band pass with f_L=100Hz and f_U=300Hz to a digital Notch filter at f_0=200Hz, Q=50.

VI. CONCLUSION

A new method was provided for transforming and re-transforming between any low pass digital filter with transfer function $H_g(z)$ and another digital filter with transfer function $H_d(z)$. The method can be applied to transform a given digital filter into any other digital filter (low pass, high pass, band pass, band stop or narrow band). The involvement of the Pascal's triangle in the Pascal matrix and inverse Pascal matrix equations, as demonstrated in the examples, simplifies hand calculation and computing when transforming a digital filter in the z-domain. The features of the matrices $[P]$ and $[T]$ are very helpful for finding the inverse matrix, which is not easy to do with the larger matrix size. The algorithm for this converse and inverse method is very simple because all operations employ matrix multiplication, so it is more effective for programming and calculation.

ACKNOWLEDGMENT

This is the own research of the author and he would like to share his ideas on how to apply Pascal's triangle in filter design. The author would greatly appreciate all feedback from readers to improve this research!

REFERENCES

[1] C. Emmanuel, B. Ifeacor, and W. Jervis, *Digital Signal Processing*, Addison-Wesley, 1993.

[2] J. Konopacki, "The frequency transformation by matrix operation and its application in IIR filters design," *IEEE Signal Processing Letters*, vol. 12, no. 1, pp. 5–8, 2005.

[3] P. S. Nguyen, "Frequency transformation with Pascal matrix equations," *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, vol. 10, no. 1, pp. 41-45, 2016.

[4] F. J. Garcia-Ugalde, "Z-transform by Pascal matrix and its applications in the design of IIR Filters," *Journal of Applied Research & Technology*, vol. 9, pp. 355-366, 2011.

[5] P. S. Nguyen, "Convert an analog low pass filter to a digital filter using Pascal's triangle," 2013.

[6] S. Chivapreecha, "Bilinear s-z frequency transformation using matrix Pascal operation," *IEEE Communications and Information Technology*, vol. 1, pp. 764–767, 2005.

[7] S. N. Phuoc, "Bilinear z-transformation using Pascal's triangle," *International Journal of Scientific and Engineering Research*, vol. 4, pp. 1974-1978, 2013.

Phuoc Si Nguyen received a bachelor degree in mathematics and physics from the University of Can Tho, Vietnam, and gained experience teaching mathematics and physics to high school students in his hometown. A few years later, he received a bachelor degree in electrical and electronic engineering from Victoria University of Technology (VUT) in Melbourne, Australia. Because of his outstanding achievements, he was invited to join the research group examining high-efficiency power amplifiers for mobile communication systems in the School of Communication and Informatics at VUT, where he was also a laboratory assistant and tutor. His interests are in the field of mathematics and signal processing techniques. Currently, he enjoys to research on how Pascal's triangle can be applied in the design of digital filters. This idea shows much promise in a range of applications in which the author is absorbed and hopes to share with others with similar interests.