Computing Frequency Transformations for Lattice Digital All-Pass Filters

Phillip A. Regalia

Département Electronique et Communications, Institut National des Télécommunications
9, rue Charles Fourier, 91011 Evry cedex FRANCE

Résumé:

Une méthode numérique est décrite pour le calcul des transformations spectrales des filtres passe-tout en treillis. Elle peut être appliquée aux fonctions de transfert de deux circuits passe-tout en parallèle, pour réaliser un filtre accordé, par exemple. La méthode s'appuie sur la configuration Hessenburg orthogonal de la description de la variable d'état d'un filtre en treillis, et l'algorithme ainsi obtenu démontre une excellente stabilité numerique.

Abstract:

A numerical procedure is described for computing spectral transformations of all-pass filters realized in lattice form. The procedure can be applied, for example, to transfer functions implemented as the parallel connection of two all-pass filters, thus yielding a tunable filter realization. The method exploits the orthogonal Hessenburg structure of the state space description of a lattice filter, and the algorithm so obtained exhibits excellent numerical stability.

I Introduction

This paper describes a procedure to compute first-order spectral transformations for a class of recursive digital filters. Such transformations are useful in spectral analysis applications where one wants, for instance, a lowpass filter with variable cutoff frequency, or a bandpass filter with variable bandwidth. These transformations are also useful in general purpose filter design routines. For example, if a satisfactory filter design is known for one set of frequency specifications, the design can be transformed to an alternate set of specifications without having to solve a new approximation problem.

We treat here the so-called lowpass-to-lowpass transformation [1]:

$$z^{-1} \rightarrow \beta(z) = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}, \qquad |\alpha| < 1. \tag{1}$$

The constraint $|\alpha| < 1$ ensures that a stable filter maps to a stable filter. If $F(z)$ is a given lowpass filter with cutoff frequency ω_1, and the transformed filter $F(1/\beta(z))$ is desired to have a cutoff frequency ω_2, the parameter α is chosen according to

$$\alpha = \frac{\sin[(\omega_1 - \omega_2)/2]}{\sin[(\omega_1 + \omega_2)/2]}. \tag{2}$$

To implement $F(1/\beta(z))$, one is first tempted to use a realization of $F(z)$ and replace each delay element (denoted by "z^{-1}" in a signal flow graph) with the allpass filter $\beta(z)$. In practice though, this leads to delay-free loops, thereby violating the computability condition of the filter. In general, a new set of parameters must instead be computed for the transformed filter $F(1/\beta(z))$.

A useful class of transfer functions, including Butterworth, Chebyshev, and elliptic types, can be written as the sum of two all-pass functions $A_1(z)$ and $A_2(z)$ [2], [3]:

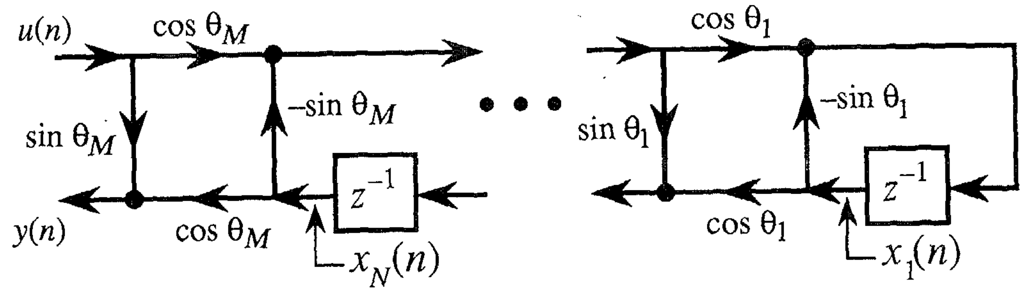
$$F(z) = \tfrac{1}{2}[A_1(z) + A_2(z)]. \tag{3}$$

The utility of this decomposition comes in noting that the transformed filter $F(1/\beta(z))$ can then be written in the form

$$F(1/\beta(z)) = \tfrac{1}{2}[A_1(1/\beta(z)) + A_2(1/\beta(z))], \tag{4}$$

where $A_1(1/\beta(z))$ and $A_2(1/\beta(z))$ remain allpass functions. Hence the problem reduces to implementing two tunable allpass filters. This approach has merit because the allpass functions can be implemented with structures that exhibit very low roundoff noise independent of the pole locations of the filter, a desirable property in tunable filter realizations. Moreover, an N-th order allpass filter can be completely described in terms of N parameters (as opposed to $2N+1$ for general N-th order recursive filters), which can be exploited to obtain a computationally efficient realization.

Figure 1: Normalized Lattice Filter



II Lattice Allpass Filters

In this section we review briefly the good numerical properties of allpass filters from a state space viewpoint. We then examine in greater detail the normalized lattice filter, and derive the algorithm to compute the transformed filter parameters. We begin with the following property, which is a consequence of the Discrete-Time Bounded Real Lemma [4]:

Property 2.1. *Let $A(z) = Y(z)/U(z)$ be a stable allpass filter. Then $A(z)$ admits a realization*

$$\begin{bmatrix} \mathbf{x}(n+1) \\ y(n) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^t & d \end{bmatrix} \begin{bmatrix} \mathbf{x}(n) \\ u(n) \end{bmatrix}, \tag{5}$$

such that the system matrix

$$\mathbf{R} \triangleq \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{c}^t & d \end{bmatrix} \tag{6}$$

is orthogonal. Conversely, any system with $\mathbf{R}$ *orthogonal is an allpass filter.*

The good numerical properties of allpass filters can be understood as a consequence of the orthogonal nature of the state computations. For example, consider the reachability and observability Grammian matrices $\mathbf{K}$ and $\mathbf{W}$ associated with the system:

$$\mathbf{K} = \sum_{n=0}^{\infty} \mathbf{A}^n \mathbf{b} (\mathbf{A}^n \mathbf{b})^t = \mathbf{A}\mathbf{K}\mathbf{A}^t + \mathbf{b}\mathbf{b}^t, \tag{7a}$$

$$\mathbf{W} = \sum_{n=0}^{\infty} (\mathbf{c}^t \mathbf{A}^n)^t \mathbf{c}^t \mathbf{A}^n = \mathbf{A}^t \mathbf{W} \mathbf{A} + \mathbf{c}\mathbf{c}^t. \tag{7b}$$

From orthogonality of $\mathbf{R}$ in (6), we have $\mathbf{R}^t\,\mathbf{R} = \mathbf{R}\mathbf{R}^t = \mathbf{I}$, which implies that

$$\mathbf{A}\mathbf{A}^t + \mathbf{b}\mathbf{b}^t = \mathbf{I}, \tag{8a}$$

$$\mathbf{A}^t\mathbf{A} + \mathbf{c}\mathbf{c}^t = \mathbf{I}. \tag{8b}$$

Comparing these with (7) we identify $\mathbf{K} = \mathbf{W} = \mathbf{I}$, independent of the pole locations of the filter $G(z)$. Such a system is inherently "balanced" in the system theoretic sense [5], and accordingly satisfies the conditions in [6] for minimum roundoff noise appearing at the filter output $y(n)$. Thus, if $A(z)$ can be implemented such that the orthogonality of $\mathbf{R}$ in (6) is obtained for any set of parameters, then robust numerical performance is obtained over the entire tuning range of the filter.

One such filter structure with this property is the normalized lattice filter [7], sketched in Fig. 1. In effect, with the delay elements removed, the filter is seen to be an interconnection of Givens rotations. Hence for any set of $\{\theta_m\}_{m=1}^{M}$ parameters, the corresponding system matrix $\mathbf{R}$ of (6) is orthogonal.

The synthesis procedure relates to the two multiplier lattice form [7] by identifying

$$\sin\theta_m = k_m, \qquad m = 1, 2, \ldots, M, \tag{9}$$

where k_m is the reflection coefficient of the m^{th} lattice stage. The well known stability condition $|k_m| < 1$ translates to $\cos\theta_m \neq 0$, and in particular, stability is trivial to ensure in a tunable environment. It can be shown that $\cos\theta_m$ can be replaced by $-\cos\theta_m$ (or equivalently θ_m can be replaced by $\pi - \theta_m$), without affecting the overall all-pass transfer function. By convention, however, one usually chooses $\cos\theta_m > 0$, or equivalenty, $|\theta_m| < \pi/2$.

For the normalized lattice filter, the matrix $\mathbf{R}$ takes the form

$$\mathbf{R} = \mathbf{Q}_1\,\mathbf{Q}_2 \cdots \mathbf{Q}_N \tag{10}$$

where

$$\mathbf{Q}_k = \begin{bmatrix} \mathbf{I}_k & & & \\ & -\sin\theta_k & \cos\theta_k & \\ & \cos\theta_k & \sin\theta_k & \\ & & & \mathbf{I}_{N-k} \end{bmatrix}. \tag{11}$$

Upon multiplying the factors, if is found that $\mathbf{R}$ takes an upper Hessenburg form, and upon partitioning $\mathbf{R}$ as per (6) it is found that $\mathbf{A}$ and $\mathbf{c}$ take the forms

$$\mathbf{A} = \begin{bmatrix} a_{11} & x & \cdots & & x \\ \cos\theta_1 & a_{22} & & & x \\ 0 & \cos\theta_2 & & & \vdots \\ \vdots & 0 & \ddots & & x \\ 0 & \cdots & 0 & \cos\theta_{M-1} & x \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \cos\theta_M \end{bmatrix} \tag{12a}$$

and that the diagonal elements of $\mathbf{A}$ are given by

$$a_{kk} = \begin{cases} -\sin\theta_1 & k = 1; \\ -\sin\theta_{k-1}\,\sin\theta_k & k \neq 1. \end{cases} \tag{12b}$$

Following the nomenclature in [8], the system configuration of (12*a*) will be called the "observable Hessenburg" form. Note that the subdiagonal elements of $\mathbf{A}$ can never be zero if the filter is stable.

Property 2.2. *Let a given realization of $F(z)$ have the state space parameters $\{\mathbf{A}, \mathbf{b}, \mathbf{c}^t, d\}$, and let $\beta(z)$ represent the first-order (all-pass) lowpass-to-lowpass transformation of (1). The transformed filter $F(1/\beta(z))$ admits the state space parameters $\{\mathbf{A}_\beta, \mathbf{b}_\beta, \mathbf{c}_\beta^t, d_\beta\}$, given by*

$$\begin{aligned} \mathbf{A}_\beta &= (\mathbf{I} + \alpha\mathbf{A})^{-1}(\alpha\mathbf{I} + \mathbf{A}), \\ \mathbf{b}_\beta &= (1-\alpha^2)^{1/2}(\mathbf{I} + \alpha\mathbf{A})^{-1}\mathbf{b}, \\ \mathbf{c}_\beta^t &= (1-\alpha^2)^{1/2}\mathbf{c}^t(\mathbf{I} + \alpha\mathbf{A})^{-1}, \\ d_\beta &= d - \alpha\mathbf{c}^t(\mathbf{I} + \alpha\mathbf{A})^{-1}\mathbf{b}. \end{aligned} \tag{14}$$

Moreover, the Grammian matrices of (7) are invariant under this transformation.

Note that if the original system has an orthogonal system matrix, the transformed filter retains this orthogonality. This transformation, however, does *not* map a Hessenburg system into a Hessenburg system, and hence the new system matrix does not correspond to a lattice filter. This suggest that to obtain the system representation corresponding to a lattice filter, one can apply an orthogonal similarity transformation to obtain $\{\mathbf{T}^t\mathbf{A}_\beta\mathbf{T}, \mathbf{T}^t\mathbf{b}_\beta, \mathbf{c}_\beta^t\mathbf{T}, d_\beta\}$, such that the new system is in Hessenburg observable form. The uniqueness of this transformation is established in the following:

Property 2.3. *Let* $(\mathbf{A}_\beta, \mathbf{c}_\beta)$ *form an observable pair. Then there exists a unique orthogonal matrix* $\mathbf{T}$ *and a unique Hessenburg observable pair* $(\mathbf{A}, \mathbf{c})$ *such that*

$$(\mathbf{T}^t\mathbf{A}_\beta\mathbf{T}, \mathbf{T}^t\mathbf{c}_\beta) = (\mathbf{A}, \mathbf{c}). \tag{15}$$

This result follows by duality of Proposition 3.2 in [8]. Thus if $(\mathbf{A}, \mathbf{c})$ is a Hessenburg observable pair which satisfies ($8b$), the pair describes a lattice filter.

III Algorithm Description

We discuss here the practical considerations to implement the frequency transformation described above.

The procedure begins by constructing the Hessenburg system matrix $\mathbf{R}$ of (6), using for example the "factored" description of (10). The final dimensions of $\mathbf{R}$ are $M{+}1 \times M{+}1$, where M is the order of the allpass filter to be transformed.

The next step is to obtain the transformed system description of (14). Note that the matrix $(\mathbf{I}+\alpha\mathbf{A})^{-1}$ appears in all the expressions. As a practical point, the matrix $\mathbf{I}+\alpha\mathbf{A}$ ceases to be invertible if and only if $1/\alpha = \lambda(\mathbf{A})$, where $\lambda(\mathbf{A})$ is any eigenvalue of $\mathbf{A}$. By construction, the given allpass filter is stable provided $\theta_m \neq \pi/2$ for any m, which then implies $|\lambda(\mathbf{A})| < 1$. Thus, with $|\alpha| < 1$, the condition $1/\alpha = \lambda(\mathbf{A})$ becomes impossible, and $(\mathbf{I}+\alpha\mathbf{A})^{-1}$ exists. Rather than try to invert $\mathbf{I}+\alpha\mathbf{A}$, an $\mathbf{LU}$ decomposition is performed [9], where $\mathbf{L}$ is lower triangular and $\mathbf{U}$ is upper triangular. Note from ($12a$) that $\mathbf{A}$ is in upper Hessenburg form, and hence so is $\mathbf{I}+\alpha\mathbf{A}$. The Gaussian elimination routine used in the $\mathbf{LU}$ decomposition becomes rather simple in this case, since only one comparison is needed in the partial pivoting strategy at each stage of the reduction. Moerover, the roundoff error growth factor is bounded above by M [9], where M is the dimension of the problem.

The solution of equations (14) begins with solving

$$(\mathbf{I}+\alpha\mathbf{A})\mathbf{x} = \mathbf{b}. \tag{16}$$

With $\mathbf{I}+\alpha\mathbf{A} = \mathbf{LU}$, the solution $\mathbf{x}$ is obtained from of

$$\mathbf{Ux} = \mathbf{L}^{-1}\mathbf{b}, \tag{17}$$

using back substitution. Next, d_β is formed according to

$$d_\beta = d - \alpha\mathbf{c}^t\mathbf{x}. \tag{18}$$

Note from ($12a$) that $\mathbf{c}$ has only one nonzero element; accordingly the inner product $\mathbf{c}^t\mathbf{x}$ in (18) requires only one multiplication. Next, $\mathbf{b}_\beta$ in (14) is obtained from

$$\mathbf{b}_\beta = (1-\alpha^2)^{1/2}\mathbf{x}. \tag{19}$$

The matrix $\mathbf{A}_\beta$ in (14) is obtained as the solution of

$$(\mathbf{I}+\alpha\mathbf{A})\mathbf{A}_\beta = \alpha\mathbf{I}+\mathbf{A}, \tag{20}$$

which may be obtained from M backsolves of the form (17), upon substituting the appropriate column of $\alpha\mathbf{I}+\mathbf{A}$ as the right-hand side vector.

The term $\mathbf{c}_\beta$ in (14) is not the solution of a column-oriented problem as described above; rather $\mathbf{c}_\beta$ may be obtained by solving

$$\mathbf{x}^t(\mathbf{I}+\alpha\mathbf{A}) = \mathbf{c}^t. \tag{21}$$

Note that this is a row-oriented, rather than a column-oriented, problem. That is, substituting $\mathbf{I}+\alpha\mathbf{A} = \mathbf{LU}$ in (21), the solution $\mathbf{x}^t$ is obtained by solving the upper triangular system

$$(\mathbf{x}^t\,\mathbf{L})\mathbf{U} = \mathbf{c}^t. \tag{22}$$

With $\mathbf{U}$ upper triangular, and with $\mathbf{c}$ in the form of ($12a$), we find that $\mathbf{c}$ is a left eigenvector of $\mathbf{U}$ with eigenvalue $[\mathbf{U}]_{M,M}$ [the (M,M)-th element of $\mathbf{U}$]. Hence no explicit back solve is necessary; one has

$$\mathbf{x}^t\,\mathbf{L} = \mathbf{c}^t/[\mathbf{U}]_{M,M}\,. \tag{23}$$

The solution $\mathbf{x}$ is obtained using the information of $\mathbf{L}$, and $\mathbf{c}_\beta$ is obtained as $\mathbf{c}_\beta = (1-\alpha^2)^{1/2}\mathbf{x}$.

The final step is then to transform the system to upper Hessenburg form using a sequence of Householder transformations. The first step becomes

$$\begin{bmatrix} \mathbf{T}^t & \mathbf{0} \\ \mathbf{0}^t & 1 \end{bmatrix} \begin{bmatrix} \mathbf{A}_\beta & \mathbf{b}_\beta \\ \mathbf{c}_\beta^t & d_\beta \end{bmatrix} \begin{bmatrix} \mathbf{T} & \mathbf{0} \\ \mathbf{0}^t & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{T}^t\mathbf{A}_\beta\mathbf{T} & \mathbf{T}^t\mathbf{b}_\beta \\ \mathbf{c}_\beta^t\mathbf{T} & d_\beta \end{bmatrix}. \tag{24}$$

The matrix $\mathbf{T}$ is chosen as

$$\mathbf{T} = \mathbf{I} - \frac{2}{\mathbf{u}^t\mathbf{u}}\mathbf{u}\mathbf{u}^t, \tag{25a}$$

with

$$\mathbf{u} = \mathbf{c}_\beta + \mathrm{sgn}(\gamma_M)\,\|\mathbf{c}_\beta\|_2\,\mathbf{e}_M\,, \tag{25b}$$

where M is the dimension of $\mathbf{c}_\beta$, γ_M is its last element, and $\mathbf{e}_M$ is the unit vector with a "1" in the last position. This results in $\mathbf{c}_\beta^t\mathbf{U} = -\mathrm{sgn}(\gamma_N)\,\|\mathbf{c}_\beta\|_2\,\mathbf{e}_M^t$, thereby introducing the desired zeros. The next step of the reduction then introduces $M{-}2$ zeros in the second-to-last row, and so on.

Finally, having effected the Householder transformation, the resulting upper Hessenburg system matrix may be "factored" into the new rotation parameters $\{\theta_m\}$ by clever inspection of the matrices of (12). One sees that the subdiagonal elements of $\mathbf{A}$ contain $\cos\theta_m$, which give information about the magnitudes of the rotation parameters, but not their signs, since $\cos\theta_m = \cos(-\theta_m)$. The diagonal elements can then be used to infer the signs of the rotation parameters. For example, refering to ($12b$) one sees that the $(1,1)$ element of $\mathbf{A}$ is $-\sin\theta_1$, from which the sign of θ_1 is easily determined. Since θ_1 and $\cos\theta_2$ are known, the sign of θ_2 can be inferred from the sign of the $(2,2)$ element of $\mathbf{A}$, and the procedure continues along the diagonal to obtain the sign of the $\{\theta_m\}$ parameters in succession. Note that successive division of the diagonal elements to obtain

$\sin\theta_m$ is ill-advised, since one might by chance encounter $\sin\theta_m \approx 0$, which would result in poor conditioning in the next division operation.

IV Simulation Example

We illustrate the above freuqency transformation applied to a ninth order elliptic lowpass filter $F(z)$ decomposed as

$$F(z) = \frac{1}{2}[A_1(z) + A_2(z)], \qquad (28)$$

where $A_1(z)$ is a fifth order allpass filter, $A_2(z)$ fourth order, corresponding to the rotation parameters listed here:

	$A_1(z)$	$A_2(z)$
θ_1	−0.359090	−0.501757
θ_2	1.336602	1.022503
θ_3	−0.637535	−0.566953
θ_4	0.675644	0.280158
θ_5	−0.262467	—

A FORTRAN subroutine was written to implement the above described algorithm. The new rotation parameters corresponding to a different cutoff frequency are obtained using two subroutine calls, one for each allpass filter. Figure 2 shows some frequency response examples obtained for various choices of the parameter α.

V Concluding Remarks

This paper has described a program to compute first order frequency transformations on a class of recursive digital filters. By considering filters which may be decomposed as the sum of two allpass functions, the tunability problem reduces to implementing two tunable allpass filters. This has important practical utility, since allpass filter structures are known for which the roundoff noise is very low independent of the pole locations. Moreover, the coefficient sensitivities are known to be very low in this class of filters [2], [3].

This report used the normalized lattice filter as the allpass filter model, for the which system matrix is orthogonal and assumes an upper Hessenburg form. Thus, upon applying the desired frequency transformation to the state space system description, the resulting description is converted back to Hessenburg form using a sequence of (rearranged) Householder transformations. The overall program may be understood as transforming an orthogonal matrix to an orthogonal matrix, which results in excellent numerical stability of the algorithm.

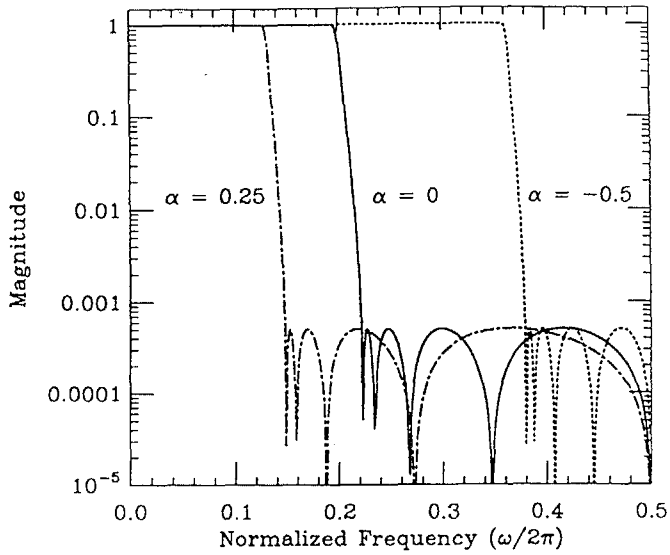


Figure 2: Example frequency responses.

References

[1] A. G. Constantinides, "Spectral Transformations for Digital Filters," *Proc. IEE*, vol. 117, pp. 1585–1590, August 1970.

[2] A. Fettweis, H. Levin, and A. Sedlmeyer, "Wave Digital Lattice Filters," *Int. J. Circuit Theory and Applications*, vol. 2, pp. 203–211, 1974.

[3] T. Saramäki, "On the Design of Digital Filters as the Sum of Two All-Pass Filters," *IEEE Trans. Circuits and Systems*, vol. CAS-32, pp. 1191–1193, November 1985.

[4] P. P. Vaidyanathan, "The Discrete-Time Bounded-Real Lemma in Digital Filtering," *IEEE Trans. Circuits and Systems*, vol. CAS-32, pp. 918–924, September 1985.

[5] B. C. Moore, "Principle Component Analysis in Linear Systems: Controllability, Observability, and Model Reduction," *IEEE Trans. Automatic Control*, vol. AC-26, pp. 17–32, February 1981.

[6] C. T. Mullis and R. A. Roberts, "Roundoff Noise in Digital Filters: Frequency Transformations and Invariants," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP-24, pp. 538–550, December 1986.

[7] A. H. Gray, Jr., and J. D. Markel, "A Normalized Filter Structure," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP-23, pp. 268–270, 1975.

[8] A. J. Laub and A. Linnemann, "Hessenburg and Hessenburg/Triangular Forms in Linear System Theory," *Int. J. Control*, vol. 44, pp. 1523–1527, 1986.

[9] G. W. Stewart, *Introduction to Matrix Computations*, Academic Press: New York, 1973.