have taken the general rectangular Simpson's rule to evaluate the numerical integration with 100 grid points, and all simulations are done on VAX 11/780 computer in FORTRAN. In our work, we use the IMSL software package [14].

For the fourth-order full-band differentiator design, Fig. 1(a) and (b) show the magnitude response and error curves of a full-band, fourth-order differentiator with length $N = 31$, respectively. Notice that the error by the closed-form method (solid line) is smaller than that of the numerical integration method (dotted line) in all of the bands, especially near the band edge. Fig. 1(c) illustrates design time for various length fourth-order full-band differentiator design from length 11 to 49, in which the amount of time saved by the closed-form method is greatly increased with the larger filter length. For example, with length $N = 49$, the closed-form method is about four times faster than the numerical method. As for its accuracy and total error measure in the frequency band, their corresponding minimum eigenvalues for two methods are illustrated in Fig. 1(d). Obviously, the closed-form method is much faster and its error is smaller than that of the numerical method.

Another example of full-band third-order differentiators with different lengths from 10 to 50 are also designed by both the proposed closed-form method and the numerical method; results similar to the above example are obtained.

V. CONCLUSION

We have presented a closed-form method for designing arbitrary higher order differentiators by the eigenfilter approach. We have shown that the closed-form method is much faster and performs better than the numerical method. A design example has been used to illustrate the effectiveness of this approach.

REFERENCES


Prefiltering Approach for Optimal Polynomial Prediction

Timo I. Laakso and Seppo J. Ovaska

Abstract—A prefiltering approach for optimal prediction of polynomial signals is proposed. The new scheme enables the use of an arbitrary stable prefilter for which an optimal FIR postfilter is designed such that polynomial signals of given order are predicted unchanged. Additional degrees of freedom are used for noise suppression. The advantages of the approach are demonstrated with examples employing a first-order recursive prefilter.

I. INTRODUCTION

Numerous real-world data sets are comprised of samples from slowly varying analog signals, and they can be modeled well as segments of low-order polynomials [1]. This assumption is valuable when it is desired to predict future samples of such signals. Predictive filtering (i.e., filtering the signal without delaying the primary component) is important, e.g., in automatic control applications where the delay in a feedback loop must be kept as small as possible to ensure fast controllability of the system. Several types of predictors have been proposed for extrapolation of polynomial signals, e.g., optimal Heinenon–Neuvo predictors [2] and computationally efficient smoothed Newton predictors [3].

Heinenon–Neuvo predictors are nonlinear-phase FIR filters that are constructed by requiring an $L$th-order polynomial to pass the filter unaltered and, in addition, by minimizing the wideband noise gain of the filter. In this sense, the Heinenon–Neuvo predictors are optimal, and they are in fact a modification of the so-called Savitzky–Golay filters for nonpredictive filtering of polynomial signals discussed in [4] and [5]. However, as polynomial signals are very narrowband lowpass signals, a high FIR filter order is needed to get low noise gain. Instead, as is the case with conventional (nonpredictive) filtering, recursive structures are, in general, more efficient for narrowband signal processing. Furthermore, the fullband linear-phase property of conventional FIR filters (usually the main reason for using FIR filters instead of IIR filters) cannot be utilized in prediction applications since the phase can be made approximately linear only in a narrow band.

Manuscript received September 18, 1992; revised August 29, 1995. The associate editor coordinating the review of this paper and approving it for publication was Dr. Robert A. Gabel.

T. I. Laakso is currently with the School of Electronic and Manufacturing Systems Engineering, University of Westminster, London, UK, on leave from the NOKIA Research Center, Helsinki, Finland.

S. J. Ovaska is currently with the Laboratory of Telecommunications Technology, Helsinki University of Technology, Espoo, Finland, on leave from Lappeenranta University of Technology, Lappeenranta, Finland.

Publisher Item Identifier S 1053-587X/96$02.00.
Here, we propose a general prefiltering approach for optimal prediction. Assuming that the signal is to be preprocessed with an arbitrarily chosen filter, we derive a scheme to design an optimal FIR postfilter that meets the requirements of unbiased prediction of an Lth-order polynomial and optimizes the free parameters to get minimum overall noise gain.

This novel scheme offers several new possibilities to predictor design. First, it enables the use of recursive prefilters to reduce the noise gain more efficiently than is possible with the plain FIR filter. It also enables the use of any kind of a filter for preprocessing, like standard elliptic recursive filters for wideband noise reduction, or notch filters for elimination of narrowband disturbances from 50- or 60-Hz power lines. Furthermore, analog prefilters can also be accounted for to make the overall system to predict the signal as desired.

II. DERIVATION

Assume that the signal is modeled as an Lth-order polynomial of the form

\[ x(n) = a_L n^L + a_{L-1} n^{L-1} + \cdots + a_0 + e(n) \]  

(1)

where coefficients \( a_k \) are real constants, and \( e(n) \) is the additive noise component. It is desired to design a predictor with the transfer function \( G(z) = P(z)H(z) \), where \( P(z) \) is a fixed prefiler (to be freely chosen), and \( H(z) \) is the Kth-order FIR postfilter to be designed, having the z-domain transfer function

\[ H(z) = h(1) z^{-1} + h(2) z^{-2} + \cdots + h(K) z^{-K}. \]  

(2)

Note that the zeroth-order coefficient is set to zero to provide the one-step-ahead prediction property when shifting the coefficients in the actual implementation.1 We develop the derivation in two phases: First, we consider the constraints of unbiased polynomial prediction, and then, we discuss suppression of noise with the remaining degrees of freedom.

A. Conditions for Unbiased Polynomial Prediction

As derived in [2] for the case without a prefilter, the condition for unbiased prediction of an Lth-order polynomial can be expressed in terms of the filter coefficients as

\[ g_0 = \sum_{k=1}^{K} h(k) = 0 \]  

(3a)

\[ g_m = \sum_{k=1}^{K} k^m h(k), \quad m = 1, \ldots, L. \]  

(3b)

The design of nonpredictive frequency-selective filters was considered in [4], resulting in equivalent conditions. These conditions are moved into the frequency domain via Fourier transform, resulting in

\[ g_0 = H(e^{j\omega})|_{\omega = 0} = 0 \]  

(4a)

\[ g_m = D^m_o[H(e^{j\omega})]|_{\omega = 0} = 0, \quad m = 1, \ldots, L \]  

(4b)

where the operator \( D^m_o \) is defined as

\[ D^m_o = -\frac{\partial^m}{\partial (j\omega)^m}. \]  

(5)

Note that the functions \( g_m \) are always real-valued for real-coefficient transfer functions. The frequency domain form of (4) can be generalized for our prefilter structure with the transfer function \( G(z) = \)

1 Similarly, one can design a p-step-ahead predictor by setting the p-first coefficients to zero in the design and by shifting the obtained impulse response in the actual implementation.

\[ P(z)H(z) \] resulting in the constraints

\[ g_0 = P(e^{j\omega})H(e^{j\omega})|_{\omega = 0} = 0 \]  

(6a)

\[ g_m = D^m_o[P(e^{j\omega})H(e^{j\omega})]|_{\omega = 0} = 0, \quad m = 1, \ldots, L. \]  

(6b)

Applying the differentiation rule for products, it can be shown that

\[ D^m_o[G(e^{j\omega})] = \sum_{k=0}^{m} \binom{m}{k} p^{m-k}(e^{j\omega})H^{(k)}(e^{j\omega}) \]  

(7)

where

\[ H^{(k)}(e^{j\omega}) = D^k_o[H(e^{j\omega})]. \]  

(8)

Consequently, (6b) can be expressed as

\[ g_m = \sum_{k=0}^{m} p_m(k)H^{(k)}(1) = 0 \]  

(9)

where

\[ p_m(k) = \binom{m}{k} p^{m-k}(1), \quad m = 0, \ldots, L, \quad k = 0, \ldots, m. \]  

(10)

To enable further elaboration, we introduce some new notation. Let \( h = [h(1) \ h(2) \ \cdots \ h(K)]^T \) be the vector of the coefficients of the FIR postfilter, and let \( u_k = [1^k \ 2^k \ \cdots \ K^k]^T \) be the kth power sequence vector. Now, we can denote the values of the kth derivative as \( H^{(k)}(1) = h^T u_k \), and consequently, functions \( g_m \) can be expressed in the form

\[ g_m = \sum_{k=0}^{m} p_m(k)H^{(k)}(1) = h^T \left( \sum_{k=0}^{m} p_m(k)u_k \right) = h^T w_m \]  

(11)

with the definition

\[ w_m = \sum_{k=0}^{m} p_m(k)u_k, \quad m = 0, \ldots, L. \]  

(12)

Hence, (6) now becomes

\[ g_0 = h^T w_0 - 1 = 0 \]  

(13a)

\[ g_m = h^T w_m, \quad m = 1, \ldots, L. \]  

(13b)

This compact vector notation will ease the formulation of the solution, as will be seen later.

B. Minimization of Wideband Noise Gain

In [2], the prediction filter was designed to minimize the wideband noise gain \( NG \) of the filter:

\[ NG = \sum_{k=1}^{K} h^2(k) = h^T h \]  

(14)

which, according to the Parseval relation, can be expressed in the frequency domain as

\[ NG = \frac{1}{2\pi} \int_0^{2\pi} |H(e^{j\omega})|^2 d\omega. \]  

(15)

To enable a straightforward generalization for the prefilter case, we introduce a vector notation for the Fourier transform:

\[ H(e^{j\omega}) = h^T e \]  

(16)
where \( e = [e^{-j\omega} e^{-j2\omega} \cdots e^{-jK\omega}]^T \). The squared magnitude response can thus be expressed as
\[
|H(e^{j\omega})|^2 = H(e^{j\omega})[H(e^{j\omega})]^* = h^T e e^H h = h^T C h
\]  
where
- \( * \) complex conjugation
- \( ^H \) transposition with conjugation
- \( C \) real-valued Toeplitz matrix:
\[
C = \begin{bmatrix}
1 & \cos \omega & \cdots & \cos (K - 1)\omega \\
\cos \omega & 1 & \cdots & \cos (K - 2)\omega \\
\vdots & \vdots & \ddots & \vdots \\
\cos (K - 1)\omega & \cos (K - 2)\omega & \cdots & 1
\end{bmatrix}
\]  
Hence, with a fixed prefilter included, the overall noise gain can be expressed as
\[
NG = \frac{1}{\pi} \int_0^\pi |H(e^{j\omega})|^2 P(e^{j\omega})^2 d\omega
= h^H \left[ \frac{1}{\pi} \int_0^\pi C|P(e^{j\omega})|^2 d\omega \right] h
= h^H Rh
\]  
where \( R \) is a Toeplitz matrix with the elements
\[
R_{k,l} = \frac{1}{\pi} \int_0^\pi \cos (k-l)\omega |P(e^{j\omega})|^2 d\omega
\]  
which are identified as the autocorrelation coefficients of a signal with the power spectrum \(|P(e^{j\omega})|^2\).

C. Solution with Lagrange Multipliers

A standard method for solving least-squares problems with linear constraints is to use Lagrange multipliers [6]. This strategy was used in [2] and [4] and is also adopted here. We first define the cost function to be minimized as
\[
L(h, \lambda) = NG + \sum_{m=0}^L \lambda_m g_m
= h^H Rh + \lambda_0 [h^T w_0 - 1] + \sum_{m=1}^L \lambda_m h^T w_m
= h^H Rh + h^T \sum_{m=0}^L \lambda_m w_m - \lambda_0
\]  
where \( \lambda_m \) are the Lagrange multipliers. The optimal solution with the linear constraints \( g_m = 0, m = 0, \cdots, L \) is obtained by setting the derivatives with respect to \( h(k), k = 1, \cdots, K \) and \( \lambda_m, m = 0, \cdots, L \) equal to zero. By introducing a vector of Lagrange coefficients as \( \lambda = [\lambda_0 \lambda_1 \cdots \lambda_L]^T \), the sum term in (21) can be expressed as
\[
\sum_{m=0}^L \lambda_m w_m = \begin{bmatrix} \lambda_0 \\ \lambda_1 \\ \vdots \\ \lambda_L \end{bmatrix} = W \lambda
\]
where \( W \) is a \( K \times L \) matrix with \( w_m \) as its column vectors, and \( \lambda_0 \) can be expressed as
\[
\lambda_0 = v_0^T \lambda
\]
with \( v_0 = [1 \cdots 0]^T \). Hence, the cost function of (21) now becomes
\[
L(h, \lambda) = h^H Rh + h^T W \lambda - v_0^T \lambda.
\]  
Differentiating with respect to the two coefficient vectors, we obtain two sets of equations
\[
\frac{\partial L}{\partial h} = 2Rh + W \lambda = 0 \quad (25)
\]
\[
\frac{\partial L}{\partial \lambda} = W^T h - v_0 = 0 \quad (26)
\]
From (25), we get \( h = -R^{-1}W \lambda/2 \), and substituting this into (26) and solving for \( \lambda \) yields \( \lambda = -2[W^T RW]^{-1}v_0 \). Substituting this back again and solving for \( h \) yields the final solution as
\[
h = R^{-1}W[V^T V]^{-1}v_0.
\]  
This is our main result. When the prefilter \( P(e^{j\omega}) \) is specified and the corresponding \( R \) and \( W \) matrices are determined, the corresponding optimal postfilter is readily obtained in closed form via (27). With no prefiltering \( (P(e^{j\omega}) \equiv 1) \), the solution reduces to
\[
h = V[V^T V]^{-1}v_0
\]  
where
\[
V = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & 2^L \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K & \cdots & K^L \end{bmatrix}
\]  
is a Vandermonde-type matrix. As discussed in [2] and [4], the matrix solution of (28) reduces to a closed-form expression for each filter coefficient. This property is lost when a nontrivial prefilter is chosen, but it is a small price to be paid for the new possibilities the prefilter predictor offers.

Numerical difficulties may arise in matrix inversion (27) for high-order filters, but according to our experience, filter orders to 20 or 30 pose no problems. On the other hand, the main advantage of the method is to be able to use an appropriate prefilter so that the order of the FIR filter to be designed is low.

III. DESIGN EXAMPLES

Let us illustrate the predictor design using a simple first-order recursive prefilter with the transfer function
\[
P(z) = \frac{1}{1 + az^{-1}}.
\]  
We first design a tenth-order predictor \( (K = 10) \) for a second-order polynomial \( (L = 2) \) with parameter values \( a = 0, a = -0.8 \), and \( a = -0.95 \), the first choice corresponding to the Heimonen–Neuvo predictor [2]. The corresponding magnitude and group delay responses of the composite filter \( G(z) = P(z)B(z) \) are shown in Figs. 1 and 2, respectively (note that only a part of the frequency band is shown). It is seen that the use of the prefilter not only reduces the peak gain from 2.4 down to 1.9 or 1.6, but the noise gain is also reduced from \( NG = 1.4 \) down to 0.8 or 0.6. The flat region of the group delay is slightly reduced by the inclusion of the prefilter, but any second-order polynomial signal is still guaranteed to pass the filter completely unchanged.
As a second example, we set a noise gain requirement of at most \( NG = 0.3 \) and design a Heinonen–Neuvo predictor and a prefilter-predictor using the recursive prefilter of (30) with \( a = -0.99 (L = 2) \). The minimum order for the Heinonen–Neuvo predictor is \( K = 34 \), whereas the prefilter-predictor meets the requirements already with the FIR order \( K = 17 \), that is, with only half of the order of the Heinonen–Neuvo predictor. For the implementation, this means that the number of required multiplications is reduced from 34 down to 18. The magnitude and the group delay responses are shown in Fig. 2(a) and (b), respectively.

There exists an efficient structure for the implementation of the Heinonen–Neuvo predictor for second-order polynomials requiring only five multiplications and 12 additions, irrespective of the predictor length, as proposed by Campbell and Neuvo [7]. However, due to its nonmodular structure and need for additional delay variables, this implementation is not well suited, e.g., for most current-generation signal processors that are optimized for fast computation of sum-of-products type calculations. In such environments, the direct-form implementation of low-order Heinonen–Neuvo predictors is more efficient than the Campbell–Neuvo structure, and the prefilter approach is a very attractive alternative.

IV. CONCLUSIONS AND DISCUSSION

An efficient prefiltering approach was proposed for optimal prediction of polynomial form signals. Given an arbitrarily chosen fixed prefilter, a solution for an FIR postfilter was formulated that minimizes the noise gain of the overall predictor.

It was shown with examples that even using a simple first-order recursive prefilter, the noise gain can be reduced considerably as compared with the corresponding Heinonen–Neuvo design or, alternatively, the same noise gain can be achieved using a much lower order FIR postfilter with reduced computational complexity. Our scheme enables the use of any kind of prefilters to be included in the system, e.g., classical recursive filters, notch filters, or even analog filters, always retaining the property of passing polynomial form signals unaltered while reducing the harmful noise components. Selection of an appropriate prefilter is mainly an application-oriented issue. The use and design of prefilters for various applications is currently under investigation.

ACKNOWLEDGMENT

The authors are grateful to J. Huokanen for writing the MATLAB programs for the design of the proposed filters.
Optimal Matrix-Filter Design

Richard J. Vaccaro and Brian F. Harrison

Abstract—A matrix filter produces N output values given a block of N input values. Matrix filters are particularly useful for filtering short data records (e.g., N ≤ 20). In this correspondence, we introduce a new set of matrix-filter design criteria and show that the design of a matrix filter can be formulated as a convex optimization problem. Several examples are given of lowpass and bandpass designs as well as a Hilbert transformer design.

I. INTRODUCTION

Given a complex-valued input data sequence \( \{x_k, k = 1, \ldots, N\} \) whose elements are placed in a vector \( x \) and the filtered output sequence \( \{y_k, k = 1, \ldots, N\} \) whose elements are placed in a vector \( y \), the filtering operation can be expressed simply as

\[
y = Mx.
\]

(1)

Note that we assume the filter output has the same length as the input. The reason for this assumption is that we view the filtering operation as extracting the useful part of the input signal, or equivalently, attenuating the undesired part of the input signal. Thus, there is no reason for the filter to create additional samples or to delete samples. Any linear filter can be written in matrix form (1), and constraints on the filter can be expressed as constraints on the matrix \( M \). For example, a time-invariant filter imposes a Toeplitz structure on \( M \), and a causal filter is represented by a lower triangular matrix. In this correspondence, we do not place such constraints on the matrix \( M \). We simply find the matrix that optimizes some design criterion (either least-squares or mini-max) subject to constraints on the passband, transition band, or stopband. The resulting \( M \) represents a time-varying, noncausal filter. For our purposes, causality is not important because we are interested in processing blocks of data.

In the next section, we introduce a new set of matrix-filter design criteria that are formulated specifically for the finite-data-record problem. Earlier work on time-varying filter design considered the problem of finding the finite-length time-varying filter that best approximates the frequency response of a given time-invariant filter [1]. This approach requires the user to first design a time-invariant filter and then pick a set of frequency points at which to match frequency responses. No bounds on passband fidelity or stopband attenuation are given in [1].

II. MATRIX FILTER DESIGN

Consider the design of a matrix lowpass filter. The design procedure starts by specifying the passband \( P \), transition band \( T \), and stopband \( S \) of the filter. Since the input to the matrix filter is a vector \( x \in \mathbb{C}^N \), \( P, T, \) and \( S \) are just subspaces of \( \mathbb{C}^N \). In the simplest form of a matrix filter design problem, \( P, T, \) and \( S \) are parameterized by a single variable \( \omega \). For example, if we are interested in processing data that consists of a sum of complex exponentials, then we define the vector

\[
a(\omega) = \begin{bmatrix} 1 \\
e^{j\omega} \\
\vdots \\
e^{j(N-1)\omega}
\end{bmatrix}.
\]

(2)

The passband of a matrix lowpass filter may be defined as the subspace generated by vectors \( a(\omega), \omega \in \Omega_P \)

\[
P = \text{subspace spanned by } \{a(\omega), \omega \in \Omega_P\}
\]

(3)

where

\[\Omega_P = [-2\pi f_c, 2\pi f_c].\]

The transition band and stopband can each be specified in a similar manner

\[
T = \text{subspace spanned by } \{a(\omega), \omega \in \Omega_T\}
\]

(4)

where

\[\Omega_T = [2\pi f_c \leq |\omega| \leq 2\pi f_t]\]

and

\[
S = \text{subspace spanned by } \{a(\omega), \omega \in \Omega_S\}
\]

(5)

where

\[\Omega_S = [2\pi f_t \leq |\omega| \leq \pi].\]

For numerical computation, the sets of \( \omega \) corresponding to \( P, T, \) and \( S \) are evaluated at a discrete grid of points \( \omega_i \). In order to design a matrix filter, we concatenate the columns of the matrix \( M \) into a vector \( m \) and consider the following functions of the parameter vector \( m \).