a quadratic constrained least squares optimization problem as in [2]. However, since the constraint matrix is not positive definite, it is difficult to find the optimal solution. In this paper, we proposed to linearize the quadratic constraint by means of iteratively solving the optimization with the best approximation obtained in the previous iteration. A similar approach was proposed in [15] to design FIR filters, and in [12], [13] to design linear phase paraunitary filter banks. The proposed design method was applied to design prototype filters for dc-leakage free cosine modulated filter banks. Design examples were presented for both traditional cosine modulated filter banks and dc-leakage free cosine modulated filter banks with prototype filters designed using the proposed algorithms. High performance filter banks with small magnitude distortions and high stopband attenuation were obtained. In the simulations conducted with the proposed algorithm, all converge within a few iterations (six iterations at most) for a wide variety of design specifications, including the presented design examples. Noted the presented algorithm can be used to design biorthogonal cosine modulated filter banks, where the quadratic formulation of the perfect reconstruction constraint for biorthogonal filter banks can be linearized using the same technique.

REFERENCES

[1] G. Strang and T. Q. Nguyen, *Wavelets and Filter Banks*. Wellesley , MA: Wellesley Cambridge, 1996.

[2] T. Q. Nguyen, "Digital filter bank design quadratic constrained formulation," *IEEE Trans. Signal Processing*, vol. 43, pp. 2103–2108, Sept. 1995.

[3] H. Xu, W. S. Lu, and A. Antoniou, "Efficient iterative design method for cosine-modulated QMF banks," *IEEE Trans. Signal Processing*, vol. 44, pp. 1657–1668, July 1996.

[4] C. K. Goh and Y. C. Lim, "An efficient algorithm for the design of weighted minimax M-channel cosine-modulated filter banks," *IEEE Trans. Signal Processing*, vol. 46, pp. 1426–1430, May 1998.

[5] E. Abdel-Raheem, F. El-Guibaly, and A. Antoniou, "Efficient design of cosine-modulated filter banks," in *Proc. IEEE Pacific Rim Conf. on Communications, Computers, and Signal Processing*, 1995, pp. 387–390.

[6] F. Argenti and E. Del Re, "Eigenfilter design of real and complex coefficient QMF prototypes," *Proc. IEEE Trans. Circuits Syst. II*, vol. 47, pp. 787–792, Aug. 2000.

[7] R. D. Koilpillai and P. P. Vaidyanathan, "Cosine-modulated FIR filter banks satisfying perfect reconstruction," *IEEE Trans. Signal Processing*, vol. 40, pp. 770–783, Apr. 1992.

[8] T. Karp and A. Mertins, "Biorthogonal cosine-modulated filter banks without DC leakage," in *Proc. IEEE ICASSP*, 1998, pp. 1457–1460.

[9] H. Kiya and S. Yamaguchi, "FSF (Frequency Sampling Filter) bank for adaptive system identification," in *Proc. IEEE ICASSP'92*, Mar. 1992.

[10] Y. Yamada, H. Kiya, and N. Kambayashi, "A frequency domain adaptive algorithm for estimating impulse response with flat delay and dispersive response region," *IEICE Trans. Fund. Electron. Commun. Comput. Sci.*, vol. E82-A, no. 8, pp. 1559–1565, Aug. 1999.

[11] T. Saramaki and R. Bregovic, "An efficient approach for designing nearly perfect reconstruction cosine-modulated and modified DFT filter banks," in *Proc. IEEE ICASSP*, vol. 6, 2001, pp. 3617–3620.

[12] T. Nagai, C. W. Kok, M. Ikehara, and T. Q. Nguyen, "Design and lattice structure of FIR paraunitary filter banks with linear phase," *IEICE Trans. Fund. Electron. Commun. Comput. Sci.*, pp. 712–721, Apr. 1997.

[13] C. K. Chen and J. H. Lee, "Design of quadrature mirror filters with linear phase in the frequency domain," *IEEE Trans. Circuits Syst. II*, vol. 39, pp. 593–605, Sept. 1992.

[14] M. H. Er and C. K. Siew, "Design of FIR filters using quadratic programming approach," *IEEE Trans. Circuits Syst. II*, vol. 42, pp. 217–220, Mar. 1995.

[15] C. S. Burrus, J. A. Barreto, and I. W. Selesnick, "Iterative reweighted least squares design of FIR filters," *IEEE Trans. Signal Processing*, vol. 42, pp. 2926–2936, 1994.

[16] M. Rossi, J. Y. Zhang, and W. Steenarrt, "Iterative constrained least squares design of near perfect reconstruction pseudo QMF banks," in *Proc. CCECE'96*, 1996, pp. 766–769.

Design and Parallel Implementation of FIR Digital Filters With Simultaneously Variable Magnitude and Non-Integer Phase-Delay

Tian-Bo Deng

***Abstract*—Variable fractional-delay digital filters are useful in various signal processing applications. To perform both fractional-delay filtering and signal frequency selecting, variable fractional-delay filters must also have variable magnitude characteristics. This paper proposes a method for designing variable finite-impulse response (FIR) filters with both variable magnitude and variable noninteger phase-delay. First, the coefficients of a variable FIR filter are expressed as different 2-variable polynomials of a pair of spectral parameters; one is for varying magnitude response, and the other is for varying noninteger phase-delay. Then the optimal coefficients of the 2-variable polynomials are found by minimizing the total weighted squared error of the variable frequency response. Since the coefficients of the obtained variable FIR filter are the polynomials of the two spectral parameters, we can yield variable magnitude and variable noninteger phase-delay simultaneously or independently by substituting different spectral parameter values to the 2-variable polynomial coefficients. Finally, we show that the resulting variable FIR filter can be implemented in a parallel form, which is suitable for high-speed signal processing.**

***Index Terms*—Variable digital filter, variable magnitude, variable noninteger phase-delay.**

I. INTRODUCTION

Since variable digital filters have been found useful in various signal processing applications, the researches on the design and implementation of variable filters have received considerable attention in the past decade. Since the frequency response of a digital filter is specified by its magnitude and phase responses, the digital filter with variable magnitude response *and/or* variable phase response is referred to as *variable digital filter*. So far, many methods have been developed for designing variable filters with *either* variable magnitude [1]–[7] *or* variable fractional-delay (VFD) [8]–[14]. To interpolate/extrapolate a signal by using a VFD filter and simultaneously remove high-frequency noise, a variable filter with *both* adjustable fractional-delay and tunable magnitude is required such that one can get different desired variable magnitude or VFD response instantly without re-designing the filter again when processing signals with different frequency components, but such a design method is not available now. One way for achieving this objective is to cascade two variable filters; one is for tuning magnitude response [1], [2], and the other is for tuning phase response [10], [11], [13], [14]. However, the overall phase response of the cascaded system must be equalized such that the overall system has satisfactory VFD response, which is a difficult challenging problem. On the other hand, the decomposition-based method can reduce the difficult problem of designing a variable filter to constant filter designs and polynomial approximations [3], which is particularly effcient for the case where there are many parameters, and thus direct design is too time-consuming, but the decomposition-based design results are not optimal in the least-squares sense.

In this paper, we propose a weighted least-squares (WLS) method for designing variable FIR filters whose magnitude and noninteger phase-delay responses are continuously and independently tunable. The basic idea can be summarized as follows. First, we assume that the desired

Manuscript received July 24, 2001; revised January 11, 2003. This paper was recommended by Associate Editor A. Swami.

The author is with the Department of Information Science, Faculty of Science, Toho University, Chiba 274-8510, Japan (e-mail: deng@is.sci.toho-u.ac.jp).

Digital Object Identifier 10.1109/TCSII.2003.809713

variable frequency response is specified by a pair of parameters called *spectral parameters*; one is for varying magnitude response, which may represent cutoff frequency, passband width, etc., and the other is for varying fractional phase-delay. Then the variable filter coefficients are expressed as different 2-variable polynomials of the two spectral parameters. Finally, the optimal coefficients of the 2-variable polynomials are determined by minimizing the weighted squared error of the variable frequency response. This WLS approach can design VFD filters with arbitrary tunable magnitude response, which can also be regarded as the generalized version of the method [11]. Furthermore, we also show that the resulting variable FIR filter can be implemented in the parallel form that consists of *constant part* and *variable part*. In signal processing applications, only the variable part needs to be changed for obtaining different magnitude or noninteger phase-delay responses. As a result, the obtained parallel-form variable filters are suitable for high-speed signal processing. Finally, a design example is given to illustrate the effectiveness of the proposed design method.

II. Design Formulation and Parallel Implementation

Assume that the ideal variable frequency response is given by

$$H_I(\omega,\Psi,d) = M_I(\omega,\Psi)e^{j\theta_I(\omega,d)} \tag{1}$$

where ω is the normalized angular frequency and Ψ is a spectral parameter specifying the desired variable magnitude response $M_I(\omega,\Psi)$, which may be the cutoff frequency of a lowpass or highpass filter, or the center frequency of a bandpass or bandstop filter. Also, $\theta_I(\omega,d)$ represents the ideal variable linear phase

$$\theta_I(\omega,d) = -d\omega$$

with d being the desired fractional phase-delay, $d \in [-0.5, 0.5]$. Obviously, $M_I(\omega,\Psi)$ and $\theta_I(\omega,d)$ can be independently varied by using the spectral parameters Ψ and d, respectively. Our objective here is to find the optimal FIR filter

$$H(z,\Psi,d) = \sum_{k=-K/2}^{K/2} a_k(\Psi,d)z^{-k} \tag{2}$$

such that the weighted squared error

$$J_c = \int_0^{\pi}\int_{\Psi_{\min}}^{\Psi_{\max}}\int_{-0.5}^{0.5} W(\omega,\Psi,d) \times \left|H(\omega,\Psi,d) - H_I(\omega,\Psi,d)\right|^2 d_d d_\Psi d_\omega \tag{3}$$

is minimized, i.e., the variable design specification (1) is best approximated in the WLS sense, where $W(\omega,\Psi,d)$ is a nonnegative weighting function, $H(\omega,\Psi,d)$ is the actual variable frequency response of the designed filter, $a_k(\Psi,d)$ are 2-variable polynomials of the spectral parameters Ψ and d as

$$a_k(\Psi,d) = \sum_{p=0}^{P}\sum_{q=0}^{Q} b(k,p,q)\Psi^p d^q \tag{4}$$

with real-valued coefficients $b(k,p,q)$, and the weighting function

$$W(\omega,\Psi,d) = W_1(\omega)W_2(\Psi)W_3(d) \tag{5}$$

is completely separable. Substituting (4) into (2) obtains the variable transfer function

$$H(z,\Psi,d) = \sum_{k=-K/2}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q} b(k,p,q)z^{-k}\Psi^p d^q \tag{6}$$

whose frequency response is

$$H(\omega,\Psi,d) = \sum_{k=-K/2}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q} b(k,p,q)e^{-jk\omega}\Psi^p d^q. \tag{7}$$

Therefore, our objective is here to find the optimal coefficients $b(k,p,q)$ such that the weighted squared error (3) is minimized. From (7) it is known that the number of the total coefficients $b(k,p,q)$ to be determined is $(K+1)(P+1)(Q+1)$. The computational complexity required for finding $b(k,p,q)$ can be reduced by exploiting the coefficient symmetries as shown below.

Since the actual variable frequency response (7) can be rewritten as

$$\begin{aligned}H(\omega,\Psi,d) &= \sum_{p=0}^{P}\sum_{q=0}^{Q} b(0,p,q)\Psi^p d^q \\ &+ \sum_{p=0}^{P}\sum_{q=0}^{Q}\left[\sum_{k=1}^{K/2} b(k,p,q)e^{-jk\omega} + \sum_{k=-1}^{-K/2} b(k,p,q)e^{-jk\omega}\right]\Psi^p d^q\end{aligned}$$

by letting $k' = -k$, we yield

$$\sum_{k=-1}^{-K/2} b(k,p,q)e^{-jk\omega} = \sum_{k'=1}^{K/2} b(-k',p,q)e^{jk'\omega} = \sum_{k=1}^{K/2} b(-k,p,q)e^{jk\omega}$$

which leads to

$$\begin{aligned}H(\omega,\Psi,d) &= \sum_{p=0}^{P}\sum_{q=0}^{Q} b(0,p,q)\Psi^p d^q \\ &+ \sum_{k=1}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q}\left[b(k,p,q)e^{-jk\omega} + b(-k,p,q)e^{jk\omega}\right]\Psi^p d^q\end{aligned}$$

thus

$$\begin{aligned}H(\omega,\Psi,-d) &= \sum_{p=0}^{P}\sum_{q=0}^{Q}(-1)^q\cdot b(0,p,q)\Psi^p d^q + \sum_{k=1}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q}(-1)^q \\ &\cdot\left[b(k,p,q)e^{-jk\omega} + b(-k,p,q)e^{jk\omega}\right]\Psi^p d^q.\end{aligned}$$

Since the complex conjugate of $H(\omega,\Psi,d)$ is

$$\begin{aligned}H^*(\omega,\Psi,d) &= \sum_{p=0}^{P}\sum_{q=0}^{Q} b(0,p,q)\Psi^p d^q \\ &+ \sum_{k=1}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q}\left[b(k,p,q)e^{jk\omega} + b(-k,p,q)e^{-jk\omega}\right]\Psi^p d^q\end{aligned} \tag{8}$$

from the relation

$$H(\omega,\Psi,-d) = H^*(\omega,\Psi,d) \tag{9}$$

we can readily obtain

$$b(-k,p,q) = (-1)^q\cdot b(k,p,q), \quad k = 0 \sim \frac{K}{2}. \tag{10}$$

Hence, $H(\omega,\Psi,d)$ can be simplified as

$$\begin{aligned}H(\omega,\Psi,d) &= \sum_{p=0}^{P}\sum_{q=0}^{Q} b(0,p,q)\Psi^p d^q + \sum_{k=1}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q} b(k,p,q) \\ &\times\left[e^{-jk\omega} + (-1)^q\cdot e^{jk\omega}\right]\Psi^p d^q \\ &= \sum_{k=0}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q} b(k,p,q)\Omega_k(\omega)\Psi^p d^q\end{aligned}$$

where

$$\Omega_k(\omega) = \begin{cases}1, & \text{if } k = 0\\ e^{-jk\omega} + (-1)^q\cdot e^{jk\omega}, & \text{if } k \neq 0.\end{cases} \tag{11}$$

By exploiting the coefficient symmetries (10), we only need to find almost half of the coefficients $b(k,p,q)$, and thus can reduce the computational complexity required in the design process.

To minimize J_c in (3), we first sample the parameters

$$\begin{aligned}\omega &\in[0,\pi]\\ \Psi &\in[\Psi_{\min},\Psi_{\max}]\\ d &\in[-0.5,0.5]\end{aligned}\tag{12}$$

to obtain the discrete points

$$\begin{aligned}\omega_l &=\frac{(l-1)\pi}{L-1}, \qquad l=1,2,\ldots,L\\ \Psi_m &=\Psi_{\min}+\frac{(\Psi_{\max}-\Psi_{\min})(m-1)}{M-1}, \qquad m=1,2,\ldots,M\\ d_n &=-0.5+\frac{n-1}{N-1}, \qquad n=1,2,\ldots,N\end{aligned}\tag{13}$$

and get the corresponding samples of the variable design specification $H_I(\omega,\Psi,d)$, the actual variable frequency response $H(\omega,\Psi,d)$, and weighting function $W(\omega,\Psi,d)$ as

$$\begin{aligned}H_I(l,m,n) &=M_I(\omega_l,\Psi_m)e^{j\theta_I(\omega_l,d_n)}\\ H(l,m,n) &=\sum_{k=0}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q}b(k,p,q)\Omega_k(\omega_l)\Psi_m^p d_n^q\\ W(l,m,n) &=W_1(\omega_l)W_2(\Psi_m)W_3(d_n).\end{aligned}$$

Then, we want to find the optimal coefficients $b(k,p,q)$ by minimizing

$$\begin{aligned}J_d &=\sum_{l=1}^{L}\sum_{m=1}^{M}\sum_{n=1}^{N}W(l,m,n)\left|H(l,m,n)-H_I(l,m,n)\right|^2\\ &=\sum_{l=1}^{L}\sum_{m=1}^{M}\sum_{n=1}^{N}W(l,m,n)\\ &\quad\times\left|\sum_{k=0}^{K/2}\sum_{p=0}^{P}\sum_{q=0}^{Q}b(k,p,q)\Omega_k(\omega_l)\Psi_m^p d_n^q-H_I(l,m,n)\right|^2\end{aligned}$$

where J_d is the discrete version of the continuous error function J_c in (3). To simplify the design problem formulation, we perform the one-to-one index mappings

$$\begin{aligned}(l,m,n)\longrightarrow i_1, \quad i_1 &=1,2,\ldots,I_1;\ I_1=LMN\\ (k,p,q)\longrightarrow i_2, \quad i_2 &=1,2,\ldots,I_2;\ I_2=\left(\frac{K}{2}+1\right)(P+1)(Q+1)\end{aligned}\tag{14}$$

so that the three-dimensional (3-D) indices (l,m,n) and (k,p,q) can be one-to-one mapped to 1-D indices i_1 and i_2, respectively. Based on the index mappings (14), we can obtain the mappings

$$\begin{aligned}b(k,p,q) &\longrightarrow c(i_2)\\ \Omega_k(\omega_l)\Psi_m^p d_n^q &\longrightarrow \Phi(i_1,i_2)\\ H_I(l,m,n) &\longrightarrow h(i_1)\\ W(l,m,n) &\longrightarrow v(i_1).\end{aligned}\tag{15}$$

Therefore, we can write the error function J_d in a more compact form as

$$\begin{aligned}J_d &=\sum_{i_1=1}^{I_1}v(i_1)\left|\sum_{i_2=1}^{I_2}c(i_2)\Phi(i_1,i_2)-h(i_1)\right|^2\\ &=\sum_{i_1=1}^{I_1}v(i_1)\left[\sum_{i_2=1}^{I_2}c(i_2)\Phi(i_1,i_2)-h(i_1)\right]\\ &\quad\times\left[\sum_{i_2=1}^{I_2}c(i_2)\Phi^*(i_1,i_2)-h^*(i_1)\right]\end{aligned}\tag{16}$$

where $[\cdot]^*$ denotes the complex-conjugate of the complex number $[\cdot]$. To minimize the error function J_d, we differentiate J_d with respect to the i-th entry of the coefficient vector $\boldsymbol{c}$ and then set the differentiation to zero as

$$\begin{aligned}\frac{\partial J_d}{\partial c(i)} &=\sum_{i_1=1}^{I_1}v(i_1)\left\{\Phi(i_1,i)\left[\sum_{i_2=1}^{I_2}c(i_2)\Phi^*(i_1,i_2)-h^*(i_1)\right]\right.\\ &\qquad\left.+\Phi^*(i_1,i)\left[\sum_{i_2=1}^{I_2}c(i_2)\Phi(i_1,i_2)-h(i_1)\right]\right\}\\ &=0\end{aligned}$$

which leads to

$$\begin{aligned}&\sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2}v(i_1)\left[\Phi(i_1,i)\Phi^*(i_1,i_2)+\Phi^*(i_1,i)\Phi(i_1,i_2)\right]c(i_2)\\ &\quad=\sum_{i_1=1}^{I_1}v(i_1)\left[\Phi(i_1,i)h^*(i_1)+\Phi^*(i_1,i)h(i_1)\right]\end{aligned}\tag{17}$$

i.e.,

$$\begin{aligned}&\mathrm{Re}\left[\sum_{i_2=1}^{I_2}c(i_2)\sum_{i_1=1}^{I_1}v(i_1)\Phi(i_1,i)\Phi^*(i_1,i_2)\right]\\ &\quad=\mathrm{Re}\left[\sum_{i_1=1}^{I_1}v(i_1)\Phi(i_1,i)h^*(i_1)\right]\end{aligned}\tag{18}$$

where $\mathrm{Re}[\cdot]$ denotes the real part of the complex matrix $[\cdot]$. Substituting $i=1,2,\ldots,I_2$ to the above equation obtains

$$\mathrm{Re}\left[\boldsymbol{c}^t\boldsymbol{\Phi}^*\boldsymbol{V}\boldsymbol{\Phi}\right]=\mathrm{Re}\left[\boldsymbol{h}^*\boldsymbol{V}\boldsymbol{\Phi}\right]\tag{19}$$

or equally,

$$\mathrm{Re}\left[\boldsymbol{\Phi}^*\boldsymbol{V}\boldsymbol{\Phi}\right]\boldsymbol{c}=\mathrm{Re}\left[\boldsymbol{\Phi}^*\boldsymbol{V}\boldsymbol{h}\right]\tag{20}$$

where

$$\begin{aligned}\boldsymbol{V} &=\mathrm{diag}\left[v(1)v(2)\cdots v(I_1)\right]\\ \boldsymbol{\Phi} &=\begin{bmatrix}\Phi(1,1) & \Phi(1,2) & \cdots & \Phi(1,I_2)\\ \Phi(2,1) & \Phi(2,2) & \cdots & \Phi(2,I_2)\\ \vdots & \vdots & \vdots & \vdots\\ \Phi(I_1,1) & \Phi(I_1,2) & \cdots & \Phi(I_1,I_2)\end{bmatrix}\\ \boldsymbol{c} &=\begin{bmatrix}c(1)\\ c(2)\\ \vdots\\ c(I_2)\end{bmatrix}, \quad \boldsymbol{h}=\begin{bmatrix}h(1)\\ h(2)\\ \vdots\\ h(I_1)\end{bmatrix}.\end{aligned}\tag{21}$$

The (20) leads to the optimal solution

$$\boldsymbol{c}=\left\{\mathrm{Re}\left[\boldsymbol{\Phi}^*\boldsymbol{V}\boldsymbol{\Phi}\right]\right\}^{-1}\cdot\mathrm{Re}\left[\boldsymbol{\Phi}^*\boldsymbol{V}\boldsymbol{h}\right]\tag{22}$$

but the direct computation (22) is inefficient, which can be further manipulated as follows. Assume that the i_1-th column vector of $\boldsymbol{\Phi}^*$ is denoted by $\boldsymbol{e}_{i_1}$. Then

$$\begin{aligned}\boldsymbol{\Phi}^*\boldsymbol{V}\boldsymbol{\Phi} &=\sum_{i_1=1}^{I_1}v(i_1)\boldsymbol{e}_{i_1}\boldsymbol{e}_{i_1}^*\\ \boldsymbol{\Phi}^*\boldsymbol{V}\boldsymbol{h} &=\sum_{i_1=1}^{I_1}v(i_1)h(i_1)e_{i_1}.\end{aligned}\tag{23}$$

Thus the (22) can be re-written as

$$\boldsymbol{c}=\boldsymbol{\Gamma}^{-1}\boldsymbol{\alpha}$$

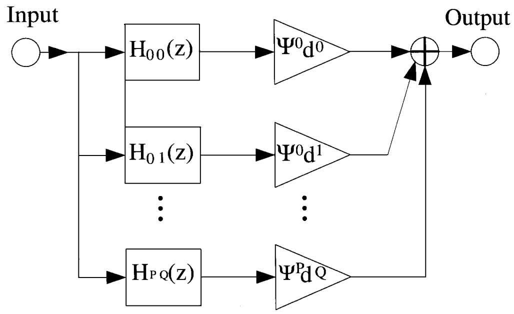

Fig. 1. Parallel implementation of variable FIR filter with constant and variable parts.

where

$$\mathbf{\Gamma} = \sum_{i_1=1}^{I_1} v(i_1)\mathrm{Re}\left[\boldsymbol{e}_{i_1}\boldsymbol{e}_{i_1}^*\right]$$
$$\boldsymbol{\alpha} = \sum_{i_1=1}^{I_1} v(i_1)\mathrm{Re}\left[h(i_1)\boldsymbol{e}_{i_1}\right].$$

Since the matrix $\mathbf{\Gamma}$ is positive definite, it can be decomposed into the form

$$\mathbf{\Gamma} = \boldsymbol{U}^T\boldsymbol{U} \tag{24}$$

by using the Cholesky decomposition, where $\boldsymbol{U}$ is an upper triangular matrix, thus

$$\mathbf{\Gamma}^{-1} = \boldsymbol{U}^{-1}\boldsymbol{U}^{-T}. \tag{25}$$

Consequently, the optimal coefficient vector $\boldsymbol{c}$ can be determined by computing

$$\boldsymbol{c} = \boldsymbol{U}^{-1}(\boldsymbol{U}^{-T}\boldsymbol{\alpha}). \tag{26}$$

The indirect inversion (25) of the matrix $\mathbf{\Gamma}$ is very important for avoiding the ill-conditioning numerical problem when the condition number of $\mathbf{\Gamma}$ is large, thus the final expression (26) provides a numerically stabilized optimal solution. Once the optimal coefficient vector $\boldsymbol{c}$ is obtained, the reverse order of (15) can be applied to map the vector $\boldsymbol{c}$ to the optimal 3-D array $b(k,p,q)$ as

$$c(i_2) \longrightarrow b(k,p,q). \tag{27}$$

Furthermore, since the resulting variable FIR filter (6) is noncausal, it cannot be applied in real-time signal procesing. However, a causal one can be easily obtained by just shifting the coefficients $b(k,p,q)$ by $K/2$ along the k axis, i.e.,

$$\beta(k,p,q) = b\left(k - \frac{K}{2}, p, q\right), \quad k = 0,1,\ldots,K \tag{28}$$

where $\beta(k,p,q)$ are the coefficients of the final causal variable filter

$$G(z,\Psi,d) = \sum_{k=0}^{K}\sum_{p=0}^{P}\sum_{q=0}^{Q}\beta(k,p,q)z^{-k}\Psi^p d^q. \tag{29}$$

It should be noted that $G(z,\Psi,d)$ has the same variable magnitude response as $H(z,\Psi,d)$, but its desired variable noninteger phase-delay is

$$D = \frac{K}{2} + d, \quad d \in [-0.5, 0.5].$$

The next problem is how to implement the designed variable filter $G(z,\Psi,d)$. From (29) we can re-arrange $G(z,\Psi,d)$ as

$$G(z,\Psi,d) = \sum_{p=0}^{P}\sum_{q=0}^{Q}\left[\sum_{k=0}^{K}\beta(k,p,q)z^{-k}\right]\Psi^p d^q. \tag{30}$$

Letting

$$H_{pq}(z) = \sum_{k=0}^{K}\beta(k,p,q)z^{-k} \tag{31}$$

we obtain

$$G(z,\Psi,d) = \sum_{p=0}^{P}\sum_{q=0}^{Q}H_{pq}(z)\Psi^p d^q \tag{32}$$

where $H_{pq}(z)$ can be regarded as constant FIR filters, and $\Psi^p d^q$ correspond to the weighting factors for $H_{pq}(z)$. Since the weighting factor $\Psi^p d^q$ are variable, thus the variable filter $G(z,\Psi,d)$ consists of *constant part* $H_{pq}(z)$ and *variable part* $\Psi^p d^q$, it can be implemented in the parallel form as Fig. 1, which is suitable for high-speed signal processing. In signal processing applications, $H_{oo}(z), H_{o1}(z), \ldots, H_{PQ}(z)$ are fixed (constant part), whereas $\Psi^0 d^0, \Psi^0 d^1, \ldots, \Psi^P d^Q$ are varied (variable part).

III. Design Example

In this section, we present a numerical example to illustrate the effectiveness of the proposed design method. The desired variable lowpass frequency response (1) is

$$M_I(\omega,\Psi) = \begin{cases} 1, & \omega \in [0, 0.26\pi + \Psi] \\ \frac{(0.50\pi+\Psi)-\omega}{0.24\pi} & \omega \in [0.26\pi+\Psi, 0.50\pi+\Psi] \\ 0, & \omega \in [0.50\pi+\Psi, \pi] \end{cases}$$
$$\Psi \in [-0.16\pi, 0.16\pi] \tag{33}$$

where the spectral parameter Ψ controls the passband width and the stopband width, whereas the transition band width is fixed (0.24π)[1], [2].

Following the aforementioned design procedures, we first get the discrete points ω_l, Ψ_m, d_n in (13) with $(L,M,N) = (51,17,11)$. Then the variable lowpass filter of order $(K,P,Q) = (30,4,4)$ is designed. It should be noted that the filter order should be chosen by designers with the tradeoff between design accuracy and computational complexity. By increasing the filter order (K,P,Q), we can reduce the design errors, but increase the computational complexity. In addition, the weighting function (5) is carefully selected as

$$W_1(\omega_l) = \begin{cases} 1, & \omega_l \notin [0.26\pi+\Psi, 0.50\pi+\Psi] \\ 0, & \omega_l \in [0.26\pi+\Psi, 0.50\pi+\Psi] \end{cases}$$
$$W_2(\Psi_m) = [210\ 200\ 45\ 30\ 25\ 20\ 0\ 15\ 5\ 0\ 0\ 0\ 30\ 1\ 20\ 40\ 25]$$
$$W_3(d_n) = 1, \qquad \text{for all } d_n \tag{34}$$

such that the frequency response errors could be almost uniformly distributed along the ω, Ψ, and d axes. Also, the normalized root-mean-squared (rms) error $E_2(m,n)$ and the maximum deviation $E_{\max}$ (both excluding the transition band) of the actual variable magnitude response

$$E_2(m,n) = \sqrt{\frac{\sum_{\omega_l}[M_I(\omega_l,\Psi_m) - M(\omega_l,\Psi_m,d_n)]^2}{\sum_{\omega_l}M_I^2(\omega_l,\Psi_m)}} \times 100\% \tag{35}$$

$$E_{\max}(m,n) = \max\Big\{|M_I(\omega_l,\Psi_m) - M(\omega_l,\Psi_m,d_n)|,\ \omega_l \in \text{ passband and stopband}\Big\} \tag{36}$$

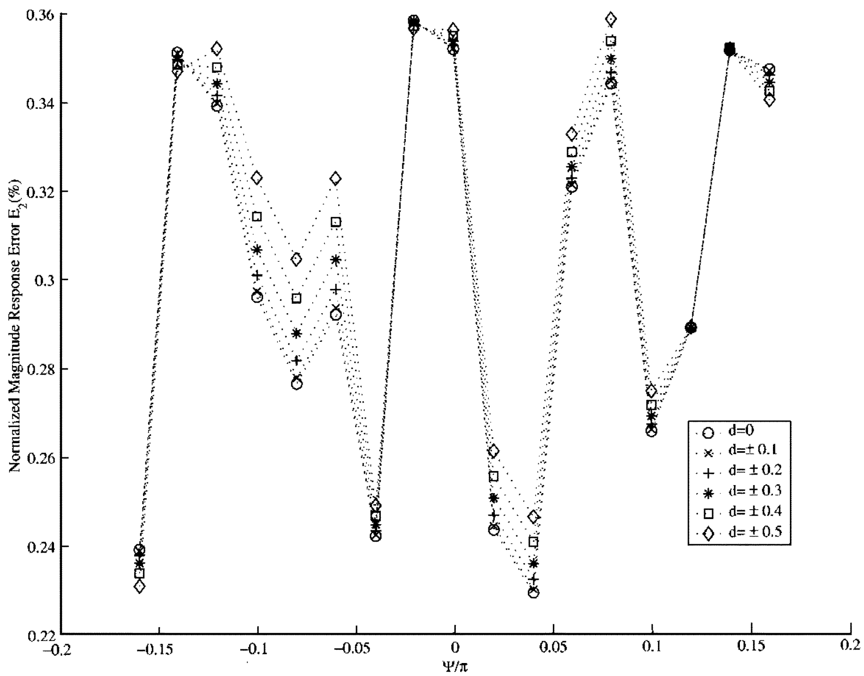

Fig. 2. Normalized rms errors of variable magnitude response for different Ψ and d.

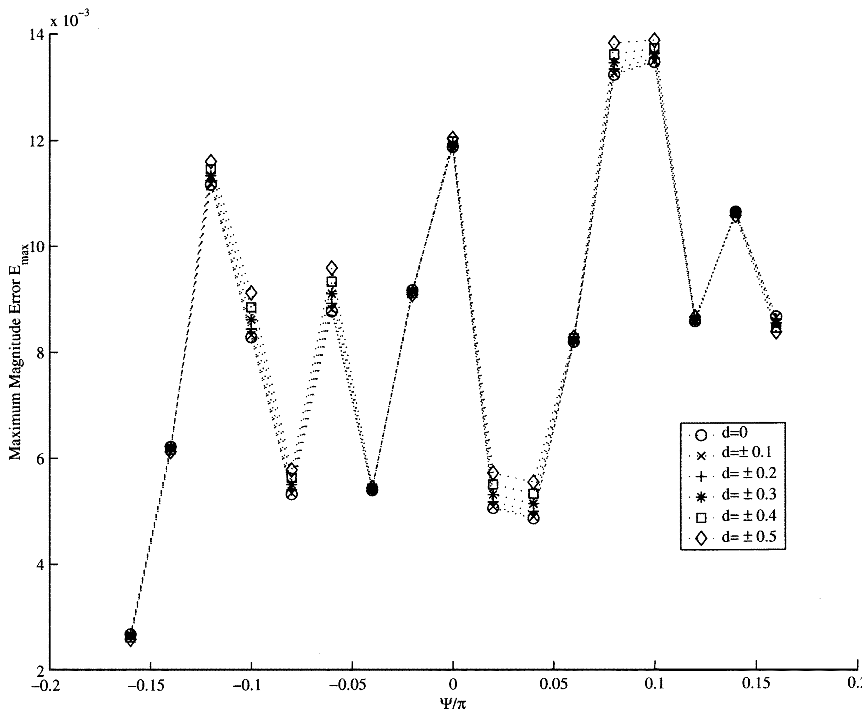

Fig. 3. Maximum deviations of variable magnitude response for different Ψ and d.

are used for evaluating the design accuracy of the variable magnitude response. For different values of Ψ_m and d_n, we get different errors $E_2(m,n)$ and $E_{\max}(m,n)$, but $M_I(\omega_l, \Psi_m)$ does not depend on d_n. It should be mentioned that we have tried a lot of values for the weighting function (34) such that the normalized rms errors $E_2(m,n)$ are almost the same (well balanced distribution) for different Ψ and d. It seems that the weighting function listed in (34) is the best among those we have tried, which leads to satisfactorily flat results as shown in Fig. 2, but it is extremely difficult to make $E_2(m,n)$ identically the same. In this case, its maximum and minimum values are 0.3582% and 0.2289%, respectively. The maximum magnitude deviations $E_{\max}(m,n)$ for different Ψ and d are depicted in Fig. 3, its maximum and minimum values are 0.0138 and 0.0026, respectively. From Figs. 2 and 3 it can also be observed that the errors $E_2(m,n)$ and $E_{\max}(m,n)$ are almost not affected by changing the fractional-delay d, but slightly depend on the spectral parameter Ψ. That is the reason why we have tried very hard to get a better weighting function as shown in (34). Fig. 4 illustrates the actual variable magnitude response for $d = 0$. As mentioned above, the

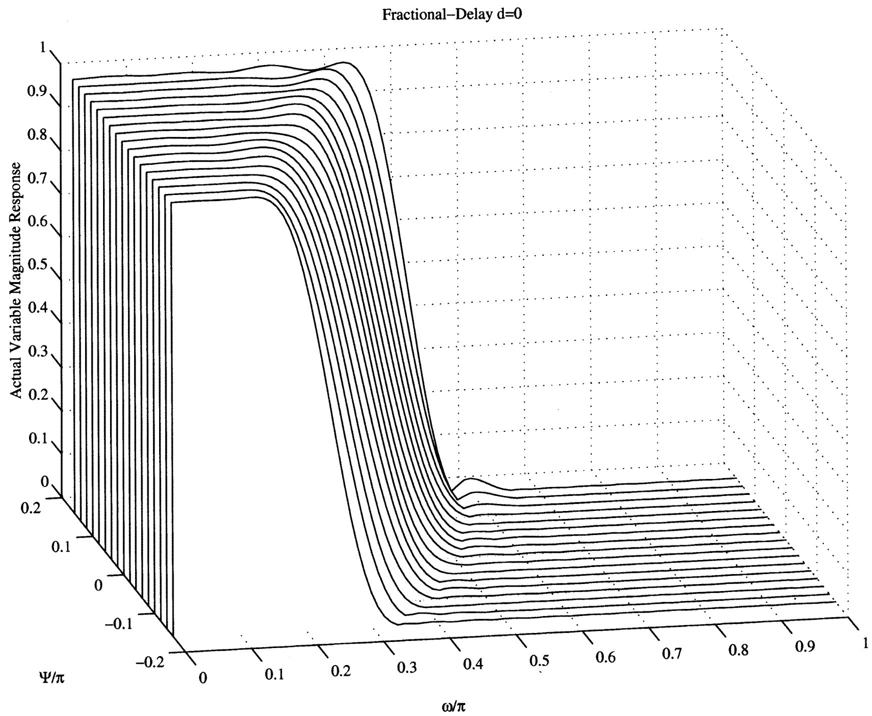

Fig. 4. Variable magnitude response for $\boldsymbol{d} = \boldsymbol{0}$.

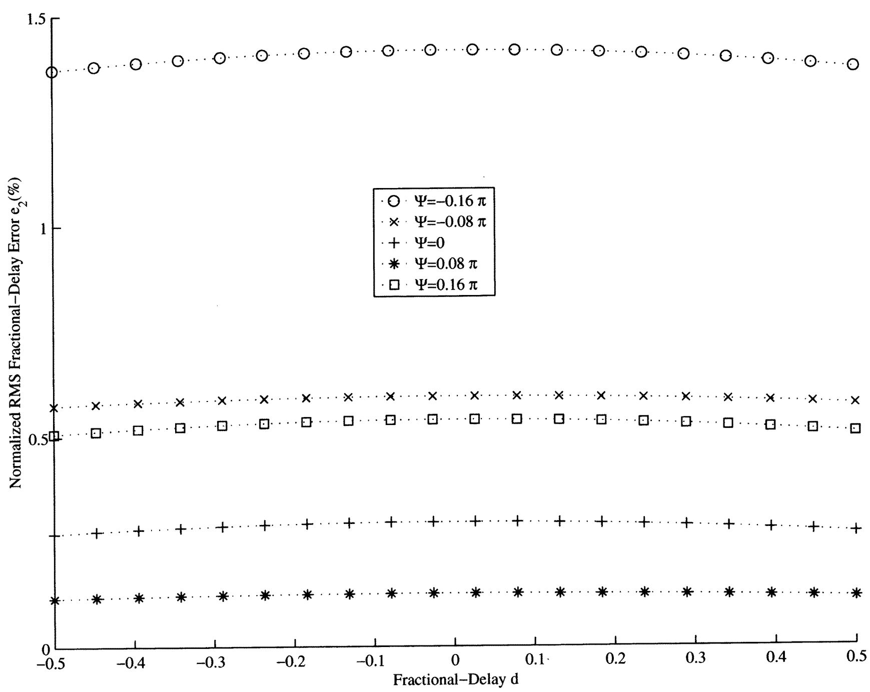

Fig. 5. Normalized rms errors of passband VFD for different $\boldsymbol{\Psi}$ and $\boldsymbol{d}$.

variable magnitude responses for other values of d are almost the same as Fig. 4 since the variable magnitude responses are almost not affected by the values of d. Moreover, to evaluate the noninteger phase-delay design accuracy, the normalized rms error $e_2(m,n)$ and the maximum deviation $e_{\max}(m,n)$ of the passband VFD response defined by

$$e_2(m,n) = \sqrt{\frac{\sum\limits_{\omega_l}[\tau_I(\omega_l,\Psi_m,d_n) - \tau(\omega_l,\Psi_m,d_n)]^2}{\sum\limits_{\omega_l}\tau_I^2(\omega_l,\Psi_m,d_n)}} \times 100\% \tag{37}$$

$$e_{\max}(m,n) = \max\{|\tau_I(\omega_l,\Psi_m,d_n) - \tau(\omega_l,\Psi_m,d_n)|\} \tag{38}$$

are used, where $\tau(\omega_l,\Psi_m,d_n)$ are the actual fractional-delay in the passband, and $\tau_I(\omega_l,\Psi_m,d_n)$ are the corresponding desired fractional-delay, evidently,

$$\tau_I(\omega_l,\Psi_m,d_n) = d_n.$$

The errors $e_2(m,n)$ and $e_{\max}(m,n)$ are the functions of Ψ_m and d_n, which are shown in Figs. 5 and 6, respectively. In this case, the maximum rms error $e_2(m,n)$ is 1.4163%, and the maximum deviation $e_{\max}(m,n)$ is 0.0128. Fig. 5 illustrates that the design errors

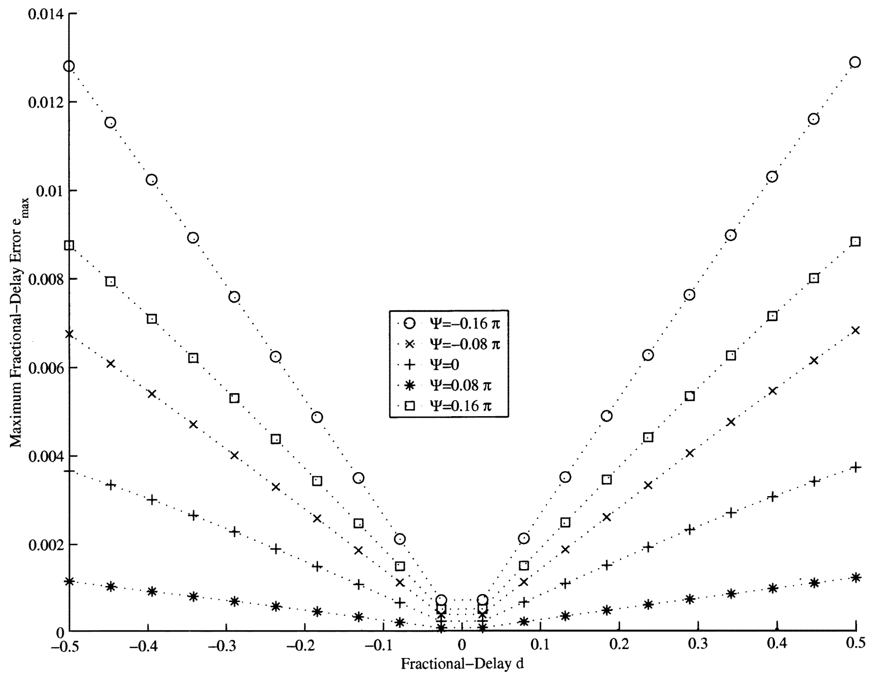

Fig. 6. Maximum deviations of passband VFD for different Ψ and $\boldsymbol{d}$.

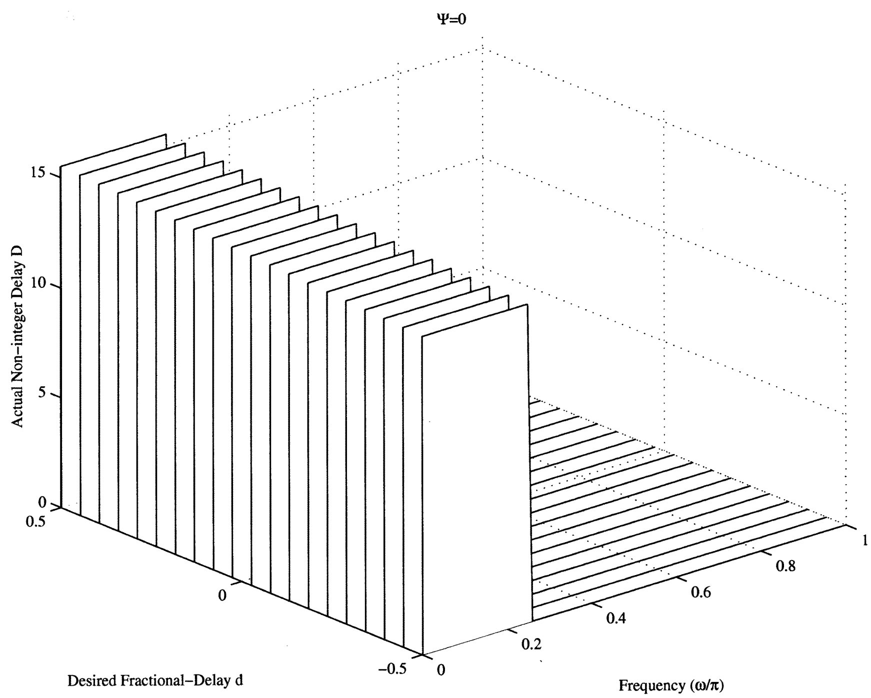

Fig. 7. Non-integer phase-delay for $\Psi = 0$.

$e_2(m,n)$ are slightly affected by Ψ, but almost not affected by the fractional-delay d since the error curves in Fig. 5 are almost flat. On the other hand, Fig. 6 shows that the maximum deviations $e_{\max}(m,n)$ almost linearly increase as d increases, the similar phenomenon can also be observed in the VFD filter designs [10]–[12], [14]. It should be noted here that our computer simulations have shown that it is very difficult to achieve a design that can balance the magnitude approximation errors and the phase-delay errors. For example, the curves of the magnitude errors $E_2(m,n)$ are flat, but the phase-delay errors $e_2(m,n)$ and $e_{\max}(m,n)$ depend on Ψ as depicted in Figs. 5 and 6, but the slight differences among $e_2(m,n)$ and $e_{\max}(m,n)$ for different values of Ψ are acceptably small. Fig. 7 shows the variable noninteger phase-delays in the passband for $\Psi = 0$, which are considerably flat (constant).

IV. Conclusion

We have proposed a WLS method for designing variable FIR filters with simultaneously variable magnitude and noninteger phase-delay responses. The filter coefficients are first expressed as 2-variable polynomials of a pair of spectral parameters. Then the optimal coefficients of the polynomials are found by minimizing the weighted squared error of the variable frequency response. In determining the final solution of the optimal filter coefficients, a Cholesky decomposition-based technique has been proposed for avoiding the numerical ill-conditioning problem that may occur due to the direct inversion of nearly singular matrices. Furthermore, we have also shown that the resulting variable FIR filter can be implemented in a parallel form that is suitable for high-speed signal processing. A design example has been given to illustrate that the WLS approach can obtain a variable filter with satisfactory variable magnitude and noninteger phase-delay responses. Since the designed variable FIR filter is much more flexible than the conventional ones with *either* variable magnitude *or* VFD responses, it is widely useful in the applications where *both* magnitude and noninteger phase-delay responses are required to be simultaneously or independently tunable.

References

[1] R. Zarour and M. M. Fahmy, "A design technique for variable digital filters," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 1473–1478, Nov. 1989.
[2] T.-B. Deng, "Design of recursive 1-D variable filters with guaranteed stability," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 689–695, Sept. 1997.
[3] ——, "Decomposition-based design of linear phase variable digital filters," *IEEE Trans. Circuits Syst. I*, vol. 46, pp. 399–402, Mar. 1999.
[4] C. S. Gargour, V. Ramachandran, and R. P. Ramachandran, "Generation of a class of two-dimensional (2-D) transfer functions yielding variable magnitude and contour characteristics," in *Proc. 2001 IEEE Int. Symp. Circuits Syst. (ISCAS 2001)*, vol. II, Sydney, Australia, May 6–9, 2001, pp. 797–800.
[5] S.-C. Chan, C. K.-S. Pun, K.-S. Yeung, and K.-L. Ho, "On the design and implementation of FIR and IIR digital filters with variable frequency characteristics," in *Proc. 2002 IEEE Int. Symp. Circuits and Systems*, vol. II, Phoenix, AZ, May 26–29, 2002, pp. 185–188.
[6] H.-K. Kwan, "High-order tunable passive digital filters," in *Proc. 2002 IEEE Int. Symp. Circuits and Systems*, vol. II, Phoenix, AZ, May 26–29, 2002, pp. 700–703.
[7] D. B. H. Tay, "Design of variable Laguerre filters," in *Proc. 2002 IEEE Int. Symp. Circuits and Systems*, vol. IV, Phoenix, Arizona, USA, May 26–29, 2002, pp. 417–420.
[8] C. W. Farrow, "A continuously variable digital delay elements," in *Proc. 1988 IEEE Int. Symp. Ciruits Syst. (ISCAS'88)*, vol. 3, Espoo, Finland, June 6–9, 1988, pp. 2641–2645.
[9] G.-S. Liu and C.-W. Wei, "A new variable fractional sample delay filter with nonlinear interpolation," *IEEE Trans. Circuits Syst. II*, vol. 39, pp. 123–126, Feb. 1992.
[10] W.-S. Lu and T.-B. Deng, "An improved weighted least-squares design for variable fractional delay FIR filters," *IEEE Trans. Circuits Syst. II*, vol. 46, pp. 1035–1040, Aug. 1999.
[11] A. Tarczynski, G. D. Cain, E. Hermanowicz, and M. Rojewski, "WLS design of variable frequency response FIR filters," in *Proc. 1997 IEEE Int. Symp. Circuits Syst. (ISCAS'97)*, Hong Kong, June 9–12, 1997, pp. 2244–2247.
[12] T.-B. Deng and W.-S. Lu, "Weighted least-squares method for designing variable fractional delay 2-D FIR digital filters," *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 114–124, Feb. 2000.
[13] C.-C. Tseng, "Design of variable fractional delay FIR filters using differentiator bank," in *Proc. 2002 IEEE Int. Symp. Circuits and Systems*, vol. IV, Phoenix, Arizona, USA, May 26–29, 2002, pp. 421–424.
[14] ——, "Design of variable fractional delay allpass filters using weighted least-squares method," in *Proc. 2002 IEEE Int. Symp. Circuits and Systems*, vol. IV, Phoenix, AZ, May 26–29, 2002, pp. 713–716.

Booth Folding Encoding for High Performance Squarer Circuits

Antonio G. M. Strollo and Davide De Caro

***Abstract*—Combined Booth encoding and Folding technique is proposed to design squarer circuits using either carry–save or Wallace Tree addition techniques. Booth-Folded technique is compared with previous state of the art squarer architecture, showing that a remarkable improvement in Timing, Power and Area performances can be gained both for carry–save and Wallace Tree cases.**

Experimental results, that use built-in-self-test for measuring on chip squarers performances, are presented. The measurements confirm the advantages of Booth-Folded architecture.

***Index Terms*—CMOS digital integrated circuits, digital integrated circuits, digital arithmetic, digital signal processors, fixed-point arithmetic, signal processing.**

I. Introduction

Specialized hardware for the computation of the square of a binary number is desirable in many applications like vector quantization [1], error correction, image compression and equalization [2], [3]. As a consequence, it is desirable to include specialized squaring units, with reduced propagation delay and power dissipation, in modern high-performance coprocessors and DSPs [4].

Many techniques have been recently proposed to increase squaring circuits performances [4]–[6]. The Folding technique, introduced in [5], uses the symmetry of the squaring partial-products matrix for archiving a 50% reduction of the number of partial products with respect to a standard multiplier. In [4] a partial products rearrangement is proposed to reduce the propagation delay of Folded squarer, whereas in [6] the number of partial products is further reduced using a divide and conquer approach.

Many recently proposed multiplier circuits [7]–[9] use the modified Booth encoding [10] to reduce the number of partial products involved in the multiplication yielding an increase in multiplier performances.

In this paper, we present a new technique [11], [12] that uses both the Folding and the Booth-encoding techniques to realize high performance squarer circuits. The proposed Booth-Folding architecture achieves an additional 50% reduction of the number of partial products with respect to the simple Folded technique, with a remarkable reduction of propagation delay and power dissipation.

The paper is organized as follows. The new Booth-Folding technique is presented in Section II while its circuit implementation is shown in Section III. Section III presents, also, two partial product generation circuits suitable for Booth-Folding. Section IV shows a comparison between the performances of 16-bit squarers designed using proposed technique, and the performances 16-bit squarer designed using the Folding technique proposed in [4].

II. Booth-Folding

A. Technique Description

Without loss of generality, let us consider an 8-bit binary signed number B, represented using two's complement notation:

$$B = -b_7 2^7 + b_6 2^6 + \ldots + b_0 2^0 = B_3 2^6 + B_2 2^4 + B_1 2^2 + B_0 2^0 \quad (1)$$

Manuscript received June 24, 2002; revised January 28, 2003. This paper was recommended by Associate Editor K. Parhi.

The authors are with the University of Naples "Federico II", Department of Electronic and Telecommunication Engineering, Naples 80125, Italy (email: dadecaro@unina.it).

Digital Object Identifier 10.1109/TCSII.2003.810574