DESIGN OF DIGITAL FILTERS AND FILTER BANKS BY OPTIMIZATION: APPLICATIONS

Tapio Saramäki and Juha Yli-Kaakinen
Signal Processing Laboratory, Tampere University of Technology,
P.O.Box 553, Tampere, FINLAND
Tel: +358 3 365 2930; fax: +358 3 365 3087
e-mail: ts@cs.tut.fi; juha.yli-kaakinen@tut.fi

ABSTRACT

This paper emphasizes the usefulness and the flexibility of optimization for finding optimized digital signal processing algorithms for various constrained and unconstrained optimization problems. This is illustrated by optimizing algorithms in six different practical applications. The first four applications include optimizing nearly perfect-reconstruction filter banks subject to the given allowable errors, minimizing the phase distortion of recursive filters subject to the given amplitude criteria, optimizing the amplitude response of pipelined recursive filters, and optimizing a modified Farrow structure with an adjustable fractional delay. In the last two applications, optimization algorithms are used as intermediate steps for finding the optimum discrete values for coefficient representations for various classes of lattice wave digital (LWD) filters and linear-phase finite impulse response (FIR) filters.

For the last application, linear programming is utilized, whereas for the first five ones the following two-step strategy is applied. First, a suboptimum solution is found using a simple systematic design scheme. Second, this start-up solution is improved by using a general-purpose nonlinear optimization algorithm, giving the optimum solution. Three alternatives are considered for constructing this general-purpose algorithm.

Index Terms—Optimization, nonlinear optimization, linear programming, digital signal processing, filter banks, digital filters, coefficient quantization, fractional delay filters, linear-phase recursive filters, multiplierless design, lattice wave digital filters, VLSI implementation.

1 INTRODUCTION

DURING the last two decades, the role of digital signal processing (DSP) has changed drastically. Twenty years ago, DSP was mainly a branch of applied mathematics. At that time, the scientists were aware of how to replace continuous-time signal processing algorithms by their discrete-time counterparts providing many attractive properties. These include, among others, a higher accuracy, a higher reliability, a higher flexibility, and, most importantly, a lower cost and the ability to duplicate the product with exactly the same performance.

Thanks to dramatic advances in very large scale integrated (VLSI) circuit technology as well as the development in signal processors, these benefits are now seen in the reality. More and more complicated algorithms can be implemented faster and faster in a smaller and smaller silicon area and with a lower and lower power consumption.

Due to this fact, the role of DSP has changed from theory to a "tool". Nowadays, the development of products requiring a small silicon area as well as a low power consumption in the case of integrated circuits is desired. The third important measure of the "goodness" of the DSP algorithm is the maximal sampling rate that can be used. In the case of signal processors, the code length is a crucial factor when evaluating the effectiveness of the DSP algorithm. These facts imply that the algorithms generated twenty years ago have to be re-optimized by taking into account the implementation constraints in order to generate optimized products.

Furthermore, when generating DSP products, all the subalgorithms should have the same quality. A typical example is a multirate analysis-synthesis filter bank for subband coding. If a lossy coding is used, then there is no need to use a perfect-reconstruction system due to the errors caused by coding. It is more beneficial to improve the filter bank performance in such a way that small errors are allowed in both the reconstruction and aliasing transfer functions. The goal is to make these errors smaller than those caused by coding and simultaneously either to improve the filter bank performance or to achieve a similar performance with a reduced overall delay.

In addition, there exist various synthesis problems where one of the responses is desired to be optimized in some sense while keeping some other responses, depending on the same design parameters, within the given tolerances. A typical example is the minimization of the phase distortion of a recursive filter subject to the given amplitude specifications. There are also problems where some of the design parameters are fixed or there are constraints among them.

In order to solve the above-mentioned problems effectively, in very few cases analytic or simple iterative design schemes can be used. In most cases, there is a need to use optimization. In some cases like in designing linear-phase finite-impulse-response (FIR) filters subject to some constraints, linear programming can be used. In many other cases, nonlinear optimization has to be applied to give the optimum solution.

This paper focuses on using two techniques for solving various unconstrained and constrained optimization problems for DSP systems. The first one uses linear programming for optimizing linear-phase FIR filters subject to some linear constraints, whereas the second one utilizes an efficient two-step strategy for solving other types of problems. First, a suboptimum solution is generated using a simple systematic design scheme. Second, this starting-point solution is further improved using an efficient general-purpose nonlinear optimization algorithm, giving the desired optimum solution.

Three alternatives are considered for constructing the general-purpose nonlinear optimization algorithm. The first one is generated by modifying the second algorithm of Dutta and Vidyasagar, the second one uses a transformation of the problem into a nonlinearly constrained problem, whereas the third one is based on the use of sequential quadratic programming (SQP) methods. It should be pointed out that in order to guarantee the convergence to the op-

timum solution, the first step in the overall procedure is of great importance.

The efficiency and flexibility of using optimization for finding optimized DSP algorithms is illustrated by means of six applications. The first five applications utilize the above-mentioned two-step strategy, whereas the last one is based on the use of linear programming.

In the first application, cosine-modulated multichannel analysis-synthesis filter banks are optimized such that the filter bank performance is optimized subject to the given allowable reconstruction and aliasing errors. In this case, a starting-point solution is a perfect-reconstruction filter bank generated using a systematic multi-step design scheme. Then, one of the above-mentioned general-purpose optimization algorithms is applied. It is shown that by allowing very small reconstruction and aliasing errors, the filter bank performance can be significantly improved compared to the perfect-reconstruction case. Alternatively, approximately the same filter bank performance can be achieved with a significantly reduced overall filter bank delay.

In the second application, the phase distortion of a recursive digital filter is minimized subject to the given amplitude criteria. The filter structures under consideration are conventional cascade-form realizations and lattice wave digital filters. For both cases, there exist very efficient design schemes for generating the starting-point solutions, making the further optimization with the aid of the general-purpose optimization algorithm very straightforward.

The third application concentrates on optimizing the modified Farrow structure proposed by Vesma and Saramäki to generate a system with an adjustable fractional delay. For this system, the overall delay is of the form $D_{\rm int} + \mu$, where $D_{\rm int}$ is an integer delay depending of the order of the building-block non-recursive digital filters and $\mu \in [0,\ 1)$ is the desired fractional delay. This fractional delay is a direct control parameter of the system. The goal is to optimize the overall system in such a way that for each value of μ the amplitude response stays within the given limits in the passband region, and the worst-case phase delay deviation from $D_{\rm int} + \mu$ in the given passband is minimized. Also in this case, it is easy to generate the starting-point solution for further optimization.

The fourth application addresses the optimization of the magnitude response for pipelined recursive filters. In this case, there exist several algorithms for generating a start-up filter for further optimization. It is shown that by applying one of the above-mentioned optimization algorithms, the magnitude response of the pipelined filters compared to that of the initial filter can be considerably improved.

The last two applications show how the coefficients of the digital filters can be conveniently quantized utilizing optimization techniques. The first class of filters under consideration consists of conventional lattice wave digital (LWD) filters, cascades of low-order LWD filters providing a very low sensitivity and roundoff noise, and LWD filters with an approximately linear phase in the passband. The second class of filters are conventional linear-phase FIR filters. For both filter types a similar systematic technique is applied for finding the optimized finite-precision solution.

For filters belonging to the first class, it has been observed that by first finding the largest and smallest values for both the radius and the angle of all the complex-conjugate poles, as well as the largest and smallest values for the radius of a possible real pole, in such a way that the given criteria are still met, we are able to find a parameter space which includes the feasible space where the filter specifications are satisfied. After finding this larger space, all what is needed is to check whether in this space there exist the desired discrete values for the coefficient representations. To solve these problems, one of the above-mentioned optimization algorithms is utilized. For filters belonging in the second class, the largest and smallest values for all the coefficients are determined in a similar manner in order to find the feasible space. In this case, the desired smallest and largest values can be conveniently found by using linear programming.

2 PROBLEMS UNDER CONSIDERATION

This section states several constrained nonlinear optimization problems for synthesizing filters and filter banks. They are stated in general form without specifying the details of the problem. We start with the desired form of the optimization problem. Then, it is shown how various types of problems can be converted into this desired form. The types to be considered in this section cover five out of the six applications to be considered later on.

2.1 Desired Form for the Optimization Problem

It is desired that the optimization problem under consideration in converted into the following form: Find the adjustable parameters included in the vector Φ to minimize

$$\rho(\Phi) = \max_{1 \le i \le I} f_i(\Phi) \tag{2.1}$$

subject to constraints

$$g_l(\Phi) \le 0 \quad \text{for } l = 1, 2, \ldots, L \tag{2.2}$$

and

$$h_m(\Phi) = 0 \quad \text{for } m = 1, 2, \ldots, M. \tag{2.3}$$

Section 3 considers three alternative effective techniques for solving problems of the above type. The convergence to the global optimum implies that a good start-up vector Φ can be generated using a simple systematic design scheme. This scheme depends on the problem at hand.

2.2 Constrained Problems Under Consideration

There exist several problems where one frequency response of a filter or filter bank is desired to be optimized in the minimax or least-mean-square sense subject to the given constraints. Futhermore, this contribution considers problems where a quantity depending on the unknowns is optimized subject to the given constraints. In the sequel, we use the angular frequency ω, that is related to the "real frequency" f and the sampling frequency F_s through $\omega = 2\pi f/F_s$, as the basic frequency variable. We concentrate on solving the following three problems:

Problem I: Find Φ containing the adjustable parameters of a filter or filter bank to minimize

$$\epsilon_A = \max_{\omega \in X_A} |E_A(\Phi, \omega)|, \tag{2.4a}$$

where

$$E_A(\Phi, \omega) = W_A(\omega)[A(\Phi, \omega) - D_A(\omega)], \tag{2.4b}$$

subject to the constraints to be given in the following subsection.

Problem II: Find Φ to minimize

$$\epsilon_A = \int_{\omega \in X_A} [E_A(\Phi, \omega)]^2 d\omega, \tag{2.5}$$

where $E_A(\Phi, \omega)$ is given by Eq. (2.4b), subject to the constraints to be given in the following subsection.

Problem III: Find Φ to minimize

$$\epsilon_A = \Psi_\alpha(\Phi), \tag{2.6}$$

where $\Psi_\alpha(\Phi)$ is a quantity depending on the unknowns included in Φ, subject to the constraints to be given in the following subsection.

For Problems I and II, X_A is a compact subset of $[0,\ \pi]$, $A(\Phi,\omega)$ is one of the frequency responses of the filter or filter bank under consideration, $D_A(\omega)$ is the desired function being continuous on X_A, and $W_A(\omega)$ is the weight function being positive on X_A. For Problems I and II, the overall weighted error function $E_A(\Phi,\omega)$, as given by Eq. (2.4b), is desired to be optimized in the minimax sense and in the least-mean-square sense, respectively.

2.3 Constraints under Consideration

Problems I, II, and III are desired to be solved such that some of the constraints of the following types are satisfied:

Type I Constraints: It is desired that for some frequency responses depending on the unknowns, the weighted error functions stays within the given limits on compact subsets of $[0,\ \pi]$. Mathematically, these constraints can be expressed as

$$\max_{\omega\in X_B^{(p)}} |E_B^{(p)}(\Phi,\omega)| \le \epsilon_B^{(p)} \quad \text{for } p=1,2,\ldots,P, \tag{2.7a}$$

where

$$E_B^{(p)}(\Phi,\omega) = W_B^{(p)}(\omega)[B^{(p)}(\Phi,\omega) - D_B^{(p)}(\omega)]. \tag{2.7b}$$

Type II Constraints: It is desired that for one frequency response depending on the unknowns, the weighted error function is identically equal to zero on a compact subset of $[0,\ \pi]$, that is,

$$\max_{\omega\in X_C} |E_C(\Phi,\omega)| = 0, \tag{2.8a}$$

where

$$E_C(\Phi,\omega) = W_C(\omega)[C(\Phi,\omega) - D_C(\omega)]. \tag{2.8b}$$

Type III Constraints: Some functions depending on Φ are less or equal to the given constants, that is,

$$\Theta_\beta^{(q)}(\Phi) \le \theta_\beta^{(q)} \quad \text{for } q=1,2,\ldots,Q. \tag{2.9}$$

Type IV Constraints: Some functions depending on Φ are equal to the given constants, that is,

$$\Theta_\gamma^{(r)}(\Phi) = \theta_\gamma^{(r)} \quad \text{for } r=1,2,\ldots,R. \tag{2.10}$$

2.4 Conversion of the Problems and Constraints into the Desired Form

It is straightforward to convert the above problems and constraints into the form considered in Subsection 2.2. For Problem I, the basic objective function, as given by Eq. (2.4a), can be converted into the form of Eq. (2.1) by discretizing the approximation interval X_A into the frequency points $\omega_i \in X_A$ for $i=1,2,\ldots,I$. The discretized objective function can then be written as

$$\rho(\Phi) = \max_{1\le i\le I} E_A(\Phi,\omega_i), \tag{2.11}$$

where $E_A(\Phi,\omega_i)$ is given by Eq. (2.4b). The dense is the number of grid points, the more accurate is the quantity given by Eq (2.11) to that of Eq. (2.4a).

For Problems II and III,

$$\rho(\Phi) = \begin{cases} \int_{\omega\in X_A}[E_A(\Phi,\omega)]^2 d\omega & \text{for Problem II} \\ \Psi_\alpha(\Phi) & \text{for Problem III.} \end{cases} \tag{2.12}$$

In some cases, the integral in the above equation can be expressed in a closed form. If this is not the case, a close approximation for it is obtained by replacing it by the summation $\sum_{i=1}^{I}[E_A(\Phi,\omega_i)]^2$, where the ω_i's are the grid points selected equidistantly on X_A.

What is left is to convert the constraints of Subsection 2.3 into the forms of Eqs. (2.2) and (2.3). The Type I Constraints can be converted into the desired form by discretizing the $X_B^{(p)}$'s for $p=1,2,\ldots,P$ into the points $\omega_{l^{(p)}} \in X_A$ for $l^{(p)}=1,2,\ldots,L^{(p)}$. These constraints can then be expressed as

$$E_B^{(p)}(\Phi,\omega_{l^{(p)}}) - \epsilon_B^{(p)} \le 0 \tag{2.13}$$

for $l^{(p)}=1,2,\ldots,L^{(p)}$ and $p=1,2,\ldots,P$, where $E_B^{(p)}(\Phi,\omega)$ is given by Eq. (2.7b).

Like for Type I Constraints, the Type II Constraints can be discretized by evaluating $E_C(\Phi,\omega)$, as given by Eq. (2.8b), at M points $\omega_m \in X_C$. The resulting constraints are expressible as

$$E_C(\Phi,\omega_m) = 0 \quad \text{for } m=1,2,\ldots,M \tag{2.14}$$

These constraints are directly of the form of Eq. (2.3).

Type III Constraints can be written as

$$\Theta_\beta^{(q)}(\Phi) - \theta_\beta^{(q)} \le 0 \quad \text{for } q=1,2,\ldots,Q. \tag{2.15}$$

and Type IV Constraints as

$$\Theta_\gamma^{(r)}(\Phi) - \theta_\gamma^{(r)} = 0 \quad \text{for } r=1,2,\ldots,R. \tag{2.16}$$

Hence, the Type III [Type IV] Constraints are directly of the same form as the constraints of Eq. (2.2) [Eq. (2.3)].

3 PROPOSED TWO-STEP PROCEDURE

This section shows how many constrained optimization problems can be solved using a two-step approach.

3.1 Basic Principle of the Approach

It has turned out that for solving various kinds of optimization problems the following two-step procedure is very effective. First, a suboptimum start-up solution is found in a systematic manner. Then, the optimization problem is formulated in the form of Subsection 2.1 and the problem is solved using an efficient algorithm finding at least a local optimum for this problem using the start-up solution as an initial solution. In this approach both steps are of the great importance. This is because the convergence to a good overall solution implies both a good initial solution and a computationally efficient algorithm.

In many cases, finding a good initial solution is not so trivial as it implies a good understanding and characterization of the problem. Furthermore, for each problem at hand the way of generating the start-up solution is very different. If there is a systematic approach for finding an initial solution being close to the optimum one, then this two-step procedure gives in most cases faster a solution that is better than those obtained by using simulated annealing or genetic algorithms [1–4].

However, it should be pointed out that in some cases it is easier, although more time-consuming, to use the above-mentioned other alternatives to get a good enough solution. This is especially true in those cases where a good start-up solution cannot be found or there are several local optima. In other words, the selection of a proper approach depends strongly on the problem under consideration.

3.2 Candidate Algorithms for Performing the Second Step

There exist various algorithms for solving the general constrained optimization problem stated in Subsection 2.1. This subsection concentrates on three alternatives.

3.2.1 Dutta-Vidyasagar Algorithm

A very elegant algorithm for solving the constrained optimization problem stated in Subsection 2.1 is the second algorithm of Dutta and Vidyasagar [5]. This algorithm is an iterative method which generates a sequence of approximate solutions that converges at least to a local optimal solution.

The main idea in this algorithm is to gradually find ξ and Φ to minimize the following function:

$$P(\Phi,\xi) = \sum_{i|f_i(\Phi)>\xi} [f_i(\Phi)-\xi]^2 + \sum_{l|g_l(\Phi)>0} w_l[g_l(\Phi)]^2 + \sum_{m=1}^{M} v_m[h_m(\Phi)]^2. \tag{3.1}$$

In Eq. (3.1), the first summation contains only those $f_i(\Phi)$'s that are larger than ξ. Similarly, the second summation contains only those $g_l(\Phi)$'s that are larger than zero. The w_l's and v_m's are the weights given by the user. Usually, they are selected to be equal. Their values have some effect on the convergence rate of the algorithm. If ξ is very large, then Φ can be found to make $P(\Phi,\xi)$ zero or practically zero. On the other hand, if ξ is too small, then $P(\Phi,\xi)$ cannot be made zero. The key idea is to find the minimum of ξ for which there exists Φ such that $P(\Phi,\xi)$ becomes zero or practically zero. In this case, $\rho(\Phi) \approx \xi$, where $\rho(\Phi)$ is the quantity to be minimized and is given by Eq. (2.1)

The algorithm is carried out in the following steps:

Step 1: Find a good start-up solution, denoted by $\widehat{\Phi}_0$, and set $B_{\text{low}} = 0$, $B_{\text{high}} = 10^4$, $\xi_1 = B_{\text{low}}$, and $k = 1$.

Step 2: Find $\widehat{\Phi}_k$ to minimize $P(\Phi,\xi_k)$ using $\widehat{\Phi}_{k-1}$ as an initial solution.

Step 3: Evaluate

$$M_{\text{low}} = \xi_k + \sqrt{P(\widehat{\Phi}_k,\xi_k)/n}, \tag{3.2}$$

where n is the number of the $f_i(\widehat{\Phi}_k)$'s satisfying $f_i(\widehat{\Phi}_k) > \xi_k$ and

$$M_{\text{high}} = \xi_k + \frac{P(\widehat{\Phi}_k,\xi_k)}{\sum_{i|f_i(\widehat{\Phi}_k)>\xi_k} [f_i(\widehat{\Phi}_k)-\xi_k]}. \tag{3.3}$$

Step 4: If $M_{\text{high}} \le B_{\text{high}}$, then set $\xi_{k+1} = M_{\text{high}}$. Otherwise, set $\xi_{k+1} = M_{\text{low}}$. Also set $\xi_0 = \xi_{k+1} - \xi_k$.

Step 5: Set $B_{\text{low}} = M_{\text{low}}$ and $S = P(\widehat{\Phi}_k,\xi_k)$.

Step 6: Set $k = k+1$.

Step 7: Find $\widehat{\Phi}_k$ to minimize $P(\Phi,\xi_k)$ using $\widehat{\Phi}_{k-1}$ as an initial solution.

Step 8: If $(B_{\text{high}} - B_{\text{low}})/B_{\text{high}} \le \epsilon_1$ or $\xi_0/\xi_k \le \epsilon_1$, then stop. Otherwise, go to the next step.

Step 9: If $P(\widehat{\Phi}_k,\xi_k) > \epsilon_2$, then go to Step 3. Otherwise, if $S \le \epsilon_3$, then stop. If none is true, then set $B_{\text{high}} = \xi_k$, $S = 0$, $\xi_k = B_{\text{low}}$, and go to Step 7.

In the above algorithm, we have used $\epsilon_1 = \epsilon_2 = \epsilon_3 = 10^{-14}$. A very crucial issue to arrive at least at a local optimum is to perform optimization at Steps 2 and 7 effectively. We have used the Fletcher-Powell algorithm [6]. When applying the Fletcher-Powell algorithm the partial derivatives of the objective function with respect to the unknowns are needed. The effectiveness of the above algorithm lies in the fact that at Steps 2 and 7 it exploits a criterion closely resembling the one used in the least-mean-square optimization. This guarantees that the objective function is well-behaved.

3.2.2 Transformation Method

Another method for solving the general optimization problem stated in Subsection 2.1 is to use any nonlinearly constrained optimization algorithm. In this case, the optimization problem is transformed in the following equivalent form: Find the adjustable parameters included in the vector Φ to minimize ξ subject to constraints

$$f_i(\Phi) \le \xi \quad \text{for } i = 1,2,\ldots,I, \tag{3.4a}$$

$$g_l(\Phi) \le 0 \quad \text{for } l = 1,2,\ldots,L, \tag{3.4b}$$

and

$$h_m(\Phi) = 0 \quad \text{for } m = 1,2,\ldots,M. \tag{3.4c}$$

After solving the above problem, $\rho(\Phi) = \xi$, where $\rho(\Phi)$ is the quantity to be minimized. The above optimization problem can be solved efficiently using the sequential quadratic programming (SQP) methods [7–10]. SQP methods are a popular class of methods considered to be extremely effective and reliable for solving general nonlinearly constrained optimization problems. At each iteration of an SQP method, a quadratic problem that models the constrained problem at the current iterate is solved. The solution to the quadratic problem is used as a search direction to determine the next iterate. For these methods, a nonlinearly constrained problem can often be solved in fewer iterations than the unconstrained problem. Provided that the solution space is convex, the SQP method always converges to the global optimum. An overview of SQP methods can be found in [7, 8, 11]. Again, the partial derivatives of the objective function and constraints with respect to the unknowns are needed. Note that the implementation of the gradient methods can be enhanced by using the automatic differentiation programs that compute the derivatives from user supplied programs that compute only function values (see, e.g, [12–15]). Alternatively, many algorithms provide a possibility to approximate the gradients using finite-differentiation routines [9, 10].

3.2.3 Sequential Quadratic Programming Methods

Some implementations of the SQP method can directly minimize the maximum of the multiple objective functions subject to constraints [10, 16, 17], that is, these implementations can be directly used for solving the optimization problem formulated in the Subsection 2.1. A feasible sequential quadratic programming (FSQP) algorithm solves the optimization problem stated in Subsection 2.1 using a two-phase SQP algorithm [16, 17]. This algorithm can handle both the linear and nonlinear constraints. Also, the optimization toolbox from MathWorks Inc. [10] provides a function `fminimax` which uses a SQP method for minimizing the maximum value of a set of multivariable functions subject to linear and nonlinear constraints.

4 NEARLY PERFECT-RECONSTRUCTION COSINE-MODULATED FILTER BANKS

During the past fifteen years, the subband coding by M-channel critically sampled FIR filter banks have received a widespread attention [18–20] (see also references in these textbooks). Such a system is shown in Fig. 4.1. In the analysis bank consisting of M parallel bandpass filters $H_k(z)$ for $k = 0,1,\ldots,M-1$ ($H_0(z)$ and $H_{M-1}(z)$ are lowpass and highpass filters, respectively), the input signal is filtered by these filters into separate subband signals. These signals are individually decimated by M, quantized, and encoded for transmission to the synthesis bank consisting also of M parallel filters $F_k(z)$ for $k = 0,1,\ldots,M-1$. In the synthesis bank, the coded symbols are converted to their appropriate digital quantities, interpolated by a factor of M followed by filtering by the corresponding filters $F_k(z)$. Finally, the outputs are added to produce

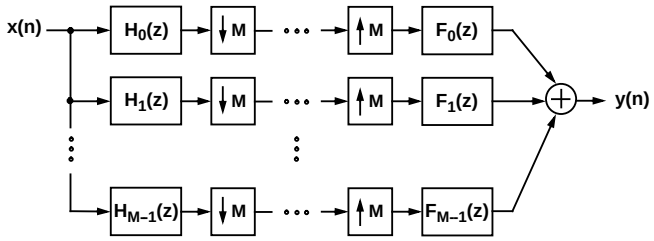

Fig. 4.1. M-channel maximally decimated filter bank.

the quantized version of the input. These filter banks are used in a number of communication applications such as subband coders for speech signals, frequency-domain speech scramblers, image coding, and adaptive signal processing [18].

The most effective technique for constructing both the analysis bank consisting of filters $H_k(z)$ for $k = 0, 1, \ldots, M-1$ and the synthesis bank consisting of filters $F_k(z)$ for $k = 0, 1, \ldots, M-1$ is to use a cosine modulation [18–29] to generate both banks from a single linear-phase FIR prototype filter. Compared to the case where all the subfilters are designed and implemented separately, the implementation of both the analysis and synthesis banks is significantly more efficient since it requires only one prototype filter and a unit performing the desired modulation operation [18–20]. Also, the actual filter bank design becomes much faster and more straightforward since the only parameters to be optimized are the coefficients of a single prototype filter.

This application shows how the two-step optimization procedure of Section 3 can be effectively used for generating prototype filters for nearly perfect-reconstruction filter banks. A starting-point solution is a perfect-reconstruction filter bank generated using systematic multi-step procedures described in [26, 29]. For the second step, the Dutta-Vidyasagar algorithm described in Subsection 3.2.1 is used. Several examples are included illustrating that by allowing small amplitude and aliasing errors, the filter bank performance can be significantly improved. Alternatively, the filter orders and the overall delay caused by the filter bank to the signal can be considerably reduced. This is very important in communication applications. In many applications such small errors are tolerable and the distortion caused by these errors to the signal is smaller than that caused by coding.

4.1 Cosine-Modulated Filter Banks

This subsection shows how M-channel critically sampled FIR filter banks can be generated using proper cosine-modulation techniques.

4.1.1 Input-Output Relation for an M-Channel Filter Bank

For the system of Fig. 4.1, the input-output relation in the z-domain is expressible as

$$Y(z) = T_0(z)X(z) + \sum_{l=1}^{M-1} T_l(z)X(ze^{-j2\pi l/M}), \qquad (4.1a)$$

where

$$T_0(z) = \frac{1}{M}\sum_{k=0}^{M-1} F_k(z)H_k(z) \qquad (4.1b)$$

and for $l = 1, 2, \ldots, M-1$

$$T_l(z) = \frac{1}{M}\sum_{k=0}^{M-1} F_k(z)H_k(ze^{-j2\pi l/M}). \qquad (4.1c)$$

Here, $T_0(z)$ is called the distortion transfer function and determines the distortion caused by the overall system for the unaliased component $X(z)$ of the input signal. The remaining transfer functions $T_l(z)$ for $l = 1, 2, \ldots, M-1$ are called the alias transfer functions and determine how well the aliased components $X(ze^{-j2\pi l/M})$ of the input signal are attenuated.

For the perfect reconstruction, it is required that $T_0(z) = z^{-N}$ with N being an integer and $T_l(z) = 0$ for $l = 1, 2, \ldots, M-1$. If these conditions are satisfied, then the output signal is a delayed version of the input signal, that is, $y(n) = x(n-N)$. It should be noted that the perfect reconstruction is exactly achieved only in the case of lossless coding. For lossy coding, it is worth studying whether it is beneficial to allow small amplitude and aliasing errors causing smaller distortions to the signal than the coding or errors that are not very noticeable in practical applications. For nearly perfect-reconstruction cases, the above-mentioned conditions should be satisfied within given tolerances.

The term $1/M$ in Eqs. (4.1b) and (4.1c) is a consequence of the decimation and interpolation processes. For simplicity, this term is forgotten in the sequel. In this case, the passband maxima of the amplitude responses of the $H_k(z)$'s and $F_k(z)$'s will become approximately equal to unity. Also the prototype filter to be considered later on can be designed such that its amplitude response has approximately the value of unity at the zero frequency. The desired input-output relation is then achieved in the final implementation by multiplying the $F_k(z)$'s by M. This is done in order to preserve the signal energy after using the interpolation filters $F_k(z)$.[1]

4.1.2 Generation of Filter Banks from a Prototype Filter Using Cosine-Modulation Techniques

For the cosine-modulated filter banks, both the $H_k(z)$'s and $F_k(z)$'s are constructed with the aid of a linear-phase FIR prototype filter of the form

$$H_p(z) = \sum_{n=0}^{N} h_p(n)z^{-n}, \qquad (4.2a)$$

where the impulse response satisfies the following symmetry property:

$$h_p(N-n) = h_p(n) \quad \text{for } n = 1, 2, \ldots, N. \qquad (4.2b)$$

One alternative is to construct the $H_k(z)$'s and $F_k(z)$'s to have the following impulse responses for $k = 0, 1, \ldots, M-1$ and $n = 0, 1, \ldots, N$ [19]:

$$h_k(n) = 2h_p(n)\cos\left[(2k+1)\frac{\pi}{2M}\left(n - \frac{N}{2}\right) + (-1)^k\frac{\pi}{4}\right] \qquad (4.3a)$$

and

$$f_k(n) = 2h_p(n)\cos\left[(2k+1)\frac{\pi}{2M}\left(n - \frac{N}{2}\right) - (-1)^k\frac{\pi}{4}\right]. \qquad (4.3b)$$

From the above equations, it follows that for $k = 0, 1, \ldots, M-1$

$$f_k(n) = h_k(N-n) \qquad (4.4a)$$

and

$$F_k(z) = z^{-N}H_k(z^{-1}). \qquad (4.4b)$$

[1] In this case, the filters in the analysis and synthesis banks of the overall system become approximately peak scaled, as is desired in many practical applications.

Another alternative is to construct the impulse responses $h_k(n)$ and $f_k(n)$ as follows [18][2]:

$$f_k(n) = 2h_p(n)\cos\left[\frac{\pi}{2M}\left(k+\frac{1}{2}\right)\left(n+\frac{M+1}{2}\right)\right] \quad (4.5a)$$

and

$$h_k(n) = 2h_p(n)\cos\left[\frac{\pi}{2M}\left(k+\frac{1}{2}\right)\left(N-n+\frac{M+1}{2}\right)\right]. \quad (4.5b)$$

The most important property of the above modulation schemes lies in the following facts. By properly designing the prototype filter transfer function $H_p(z)$, the aliased components generated in the analysis bank due to the decimation can be totally or partially compensated in the synthesis bank. Secondly, $T_0(z)$ can be made exactly or approximately equal to the pure delay z^{-N}. Hence, these modulation techniques enable us to design the prototype filter in such a way that the resulting overall bank has the perfect-reconstruction or a nearly perfect-reconstruction property.

4.1.3 Conditions for the Prototype Filter to Give a Nearly Perfect-Reconstruction Property

The above modulation schemes guarantee that if the impulse response of

$$\widehat{H}_p(z) = [H_p(z)]^2 = \sum_{n=0}^{2N} \widehat{h}_p(n) z^{-n}, \quad (4.6a)$$

where

$$\widehat{h}_p(2N-n) = \widehat{h}_p(n) \quad \text{for } n = 1, 2, \ldots, N, \quad (4.6b)$$

satisfies[3]

$$\widehat{h}_p(N) \approx 1/(2M) \quad (4.6c)$$

and

$$\widehat{h}_p(N \pm 2rM) \approx 0 \quad \text{for} \quad r = 1, 2, \ldots, \lfloor N/(2M) \rfloor, \quad (4.6d)$$

then [28][4]

$$T_0(z) = \sum_{k=0}^{M-1} F_k(z) H_k(z) \approx z^{-N}. \quad (4.7)$$

In this case, the amplitude error $|T_0(e^{j\omega}) - e^{-jN\omega}|$ becomes very small. If the conditions of Eqs. (4.6c) and (4.6d) are exactly satisfied, then the amplitude error becomes zero. It should be noted that since $T_0(z)$ is an FIR filter of order $2N$ and its impulse-response coefficients, denoted by $t_0(n)$, satisfy $t_0(2N-n) = t_0(n)$ for $n = 0, 1, \ldots, 2N$, there exists no phase distortion.

Equation (4.7) implies that $[H_p(z)]^2$ is approximately a $2M$th-band linear-phase FIR filter [30, 31]. Based on the properties of these filters, the stopband edge of the prototype filter $H_p(z)$ must be larger than $\pi/(2M)$ and is specified by

$$\omega_s = (1+\rho)\pi/(2M), \quad (4.8)$$

where $\rho > 0$. Furthermore, the amplitude response of $H_p(z)$ achieves approximately the values of unity and $1/\sqrt{2}$ at $\omega = 0$ and $\omega = \pi/(2M)$, respectively. As an example, Fig. 4.2 shows the prototype filter amplitude response for $M = 4$, $N = 63$, and $\rho = 1$ as well as the responses for the filters $H_k(z)$ and $F_k(z)$ for $k = 0, 1, 2, 4$. It is seen that the filters $H_k(z)$ and $F_k(z)$ for $k = 1, 2, \ldots, M-2$ are bandpass filters with the center frequency at $\omega = \omega_k = (2k+1)\pi/(2M)$ around which the amplitude response is very flat having approximately the value of unity. The amplitude response of these filters achieves approximately the value of $1/\sqrt{2}$ at $\omega = \omega_k \pm \pi/(2M)$ and the stopband edges are at $\omega = \omega_k \pm \omega_s$. $H_0(z)$ and $F_0(z)$ [$H_{M-1}(z)$ and $F_{M-1}(z)$] are lowpass (highpass) filters with the amplitude response being flat around $\omega = 0$ ($\omega = \pi$) and achieving approximately the value $1/\sqrt{2}$ at $\omega = \pi/M$ ($\omega = \pi - \pi/M$). The stopband edge is at $\omega = (2+\rho)\pi/(2M)$ [$\omega = \pi - (2+\rho)\pi/(2M)$]. The impulse responses for the prototype filter as well as those for the filters in the banks are shown in Figs. 4.3 and 4.4, respectively. In this case, the impulse responses for the filters in the bank have been generated according to Eq. (4.3).

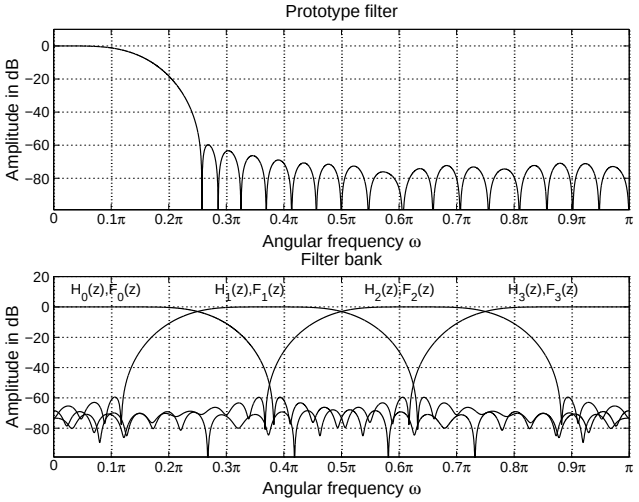

Fig. 4.2. Example amplitude responses for the prototype filter and for the resulting filters in the analysis and synthesis banks for $M = 4$, $N = 63$, and $\rho = 1$.

If $H_p(z)$ satisfies Eq. (4.6), then both of the above-mentioned modulation schemes have the very important property that the maximum amplitude values of the aliased transfer functions $T_l(z)$ for $l = 1, 2, \ldots, M-1$ are guaranteed to be approximately equal to the maximum stopband amplitude values of the filters in the bank [19], as will be seen in connection with the examples of Subsection 4.4. If smaller aliasing error levels are desired to be achieved, then additional constraints must be imposed on the prototype filter. In the case of the perfect reconstruction, the additional constraints are so strict that they dramatically reduce the number of adjustable parameters of the prototype filter [18–27, 29].

4.2 General Optimization Problems for the Prototype Filter

This subsection states two optimization problems for designing the prototype filter in such a way that the overall filter bank possesses a nearly perfect-reconstruction property. Efficient algorithms are then described for solving these problems.

[2] In [18], instead of the constant of value 2, the constant of value $\sqrt{2/M}$ has been used. The reason for this is that the prototype filter is implemented using special butterflies. The amplitude response of the resulting prototype filter approximates the value of $M\sqrt{2}$, instead of unity, at the zero frequency. For an approximately peak-scaled overall implementation, the scaling constants of values $1/(M\sqrt{2})$ and $1/\sqrt{2}$ are desired to be used in the final implementation for the $h_k(n)$'s and $f_k(n)$'s, respectively.

[3] $\lfloor x \rfloor$ stands for the integer part of x.

[4] This fact has been proven in [28] when the conditions of Eq. (4.6) are exactly satisfied.

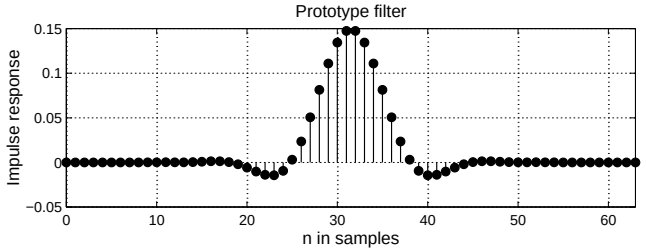

Fig. 4.3. Impulse response for the prototype filter in the case of Fig. 4.2.

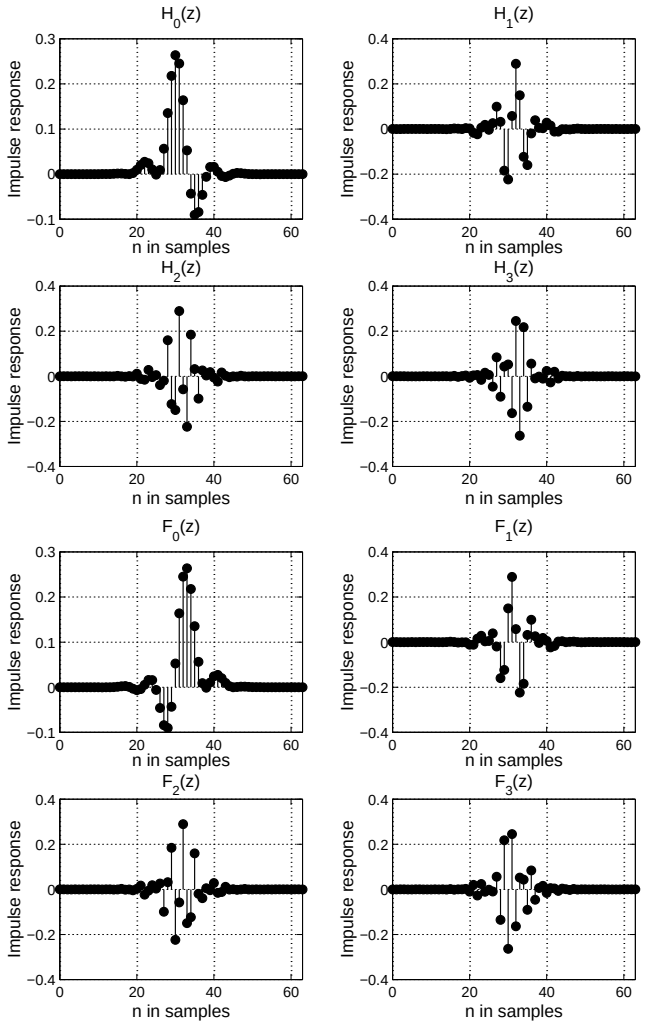

Fig. 4.4. Impulse responses for the filters in the bank in the case of Fig. 4.2.

4.2.1 Statement of the Problems

We consider the following two general optimization problems:

Problem I: Given ρ, M, and N, find the coefficients of $H_p(z)$ to minimize

$$E_2 = \int_{\omega_s}^{\pi} |H_p(e^{j\omega})|^2 d\omega, \tag{4.9a}$$

where

$$\omega_s = (1+\rho)\pi/(2M) \tag{4.9b}$$

subject to

$$1-\delta_1 \le |T_0(e^{j\omega})| \le 1+\delta_1 \quad \text{for} \quad \omega \in [0,\ \pi] \tag{4.9c}$$

and for $l = 1, 2, \ldots, M-1$

$$|T_l(e^{j\omega})| \le \delta_2 \quad \text{for } \omega \in [0,\ \pi]. \tag{4.9d}$$

Problem II: Given ρ, M, and N, find the coefficients of $H_p(z)$ to minimize

$$E_\infty = \max_{\omega \in [\omega_s,\ \pi]} |H_p(e^{j\omega})| \tag{4.10}$$

subject to the conditions of Eqs. (4.9c) and (4.9d).

4.3 Proposed Two-Step Optimization Scheme

This subsection shows how the two problems stated in the previous subsection can be conveniently solved by using the two-step optimization procedure of Section 3.

4.3.1 Algorithm for Solving Problem I

This contribution concentrates on the case where N, the order of the prototype filter, is odd (the length $N+1$ is even). This is because for the perfect-reconstruction case N is restricted to be odd [18–27, 29]. For N odd, the frequency response of the prototype filter is expressible as

$$H_p(\Phi, e^{j\omega}) = e^{-j(N-1)\omega/2} H_p^{(0)}(\omega), \tag{4.11a}$$

where

$$H_p^{(0)}(\omega) = 2 \sum_{n=1}^{(N+1)/2} h_p[(N+1)/2 - n] \cos[(n-1/2)\omega] \tag{4.11b}$$

and

$$\Phi = \big[h_p(0), h_p(1), \ldots, h_p[(N-1)/2]\big] \tag{4.11c}$$

denotes the adjustable parameter vector of the prototype filter. After some manipulations, Eq. (4.9a) is expressible as

$$E_2(\Phi) \equiv E_2 = \sum_{\mu=1}^{(N+1)/2} \sum_{\nu=1}^{(N+1)/2} \Theta(\mu,\nu)\Psi(\mu,\nu), \tag{4.12a}$$

where

$$\Theta(\mu,\nu) = h_p[(N+1)/2 - \mu] h_p[(N+1)/2 - \nu] \tag{4.12b}$$

and

$$\Psi(\mu,\nu) = \begin{cases} 2\pi - 2\omega_s - \dfrac{2\sin[(2\mu-1)\omega_s]}{2\mu-1}, & \mu = \nu \\ -\dfrac{2\sin[(\mu+\nu-1)\omega_s]}{\mu+\nu-1} \\ -\dfrac{2\sin[(\mu-\nu)\omega_s]}{\mu-\nu}, & \mu \ne \nu. \end{cases} \tag{4.12c}$$

The $|T_l(\Phi, e^{j\omega})|$'s for $l = 0, 1, \ldots, M-1$, in turn, can be written as shown in Appendix A in [29].

To solve Problem I, we discretize the region $[0,\ \pi/M]$ into the discrete points $\omega_j \in [0,\ \pi/M]$ for $j = 1, 2, \ldots, J_0$. In many cases, $J_0 = N$ is a good selection to arrive at a very accurate solution. The resulting discrete problem is to find Φ to minimize

$$\rho(\Phi) = E_2(\Phi), \tag{4.13a}$$

where $E_2(\Phi)$ is given by Eq. (4.12), subject to

$$g_j(\Phi) \le 0 \quad \text{for } j = 1, 2, \ldots, J, \tag{4.13b}$$

where

$$J = \lfloor (M+2)/2 \rfloor J_0, \tag{4.13c}$$

$$g_j(\Phi) = ||T_0(\Phi, e^{j\omega_j})| - 1| - \delta_1 \quad \text{for } j = 1, 2, \ldots, J_0, \tag{4.13d}$$

and

$$g_{lJ_0+j}(\Phi) = |T_l(\Phi, e^{j\omega_j})| - \delta_2 \tag{4.13e}$$

for $l = 1, 2, \ldots, \lfloor M/2 \rfloor$ and for $j = 1, 2, \ldots, J_0$.

In the above, the region $[0,\ \pi/M]$, instead of $[0,\ \pi]$, has been used since the $|T_l(\Phi, e^{j\omega})|$'s are periodic with periodicity equal to $2\pi/M$. Furthermore, only the first $\lfloor (M+2)/2 \rfloor$ $|T_l(\Phi, e^{j\omega})|$'s have been used since $|T_l(\Phi, e^{j\omega})| = |T_{M-l}(\Phi, e^{j\omega})|$ for $l = 1, 2, \ldots, \lfloor (M-1)/2 \rfloor$.

The above problem can be solved conveniently by using Dutta-Vidyasagar algorithm described in Subsection 3.2.1. Since the optimization problem is nonlinear in nature, a good initial starting-point solution for the vector Φ is needed. This problem will be considered in Subsection 4.3.3.

If it is desired that $|T_0(\Phi, e^{j\omega})| \equiv 1$ [28], then the resulting discrete problem is to find Φ to minimize ϵ as given by Eq. (4.13a) subject to

$$g_j(\Phi) \le 0 \quad \text{for } j = 1, 2, \ldots, J \tag{4.14a}$$

and

$$h_l(\Phi) = 0 \quad \text{for } l = 1, 2, \ldots, L, \tag{4.14b}$$

where

$$J = \lfloor M/2 \rfloor J_0, \tag{4.14c}$$

$$L = J_0, \tag{4.14d}$$

$$g_{(l-1)J_0+j}(\Phi) = |T_l(\Phi, e^{j\omega_j})| - \delta_2 \tag{4.14e}$$

for $l = 1, 2, \ldots, \lfloor M + /2 \rfloor$ and $j = 1, 2, \ldots, J_0$, and

$$h_l(\Phi) = ||T_0(\Phi, e^{j\omega_l})| - 1| \quad \text{for } l = 1, 2, \ldots, L. \tag{4.14f}$$

Again, the Dutta-Vidyasagar algorithm is used for solving this problem. As a start-up solution, the same solution as for the original problem can be used.

4.3.2 Algorithm for Solving Problem II

To solve Problem II, we discretize the region $[\omega_s,\ \pi]$ into the discrete points $\omega_i \in [\omega_s,\ \pi]$ for $i = 1, 2, \ldots, I$. In many cases, $I = 20N$ is a good selection. The resulting discrete minimax problem is to find Φ to minimize

$$\rho(\Phi) = \widehat{E}_\infty(\Phi) = \max_{1 \le i \le I} \{f_i(\Phi)\} \tag{4.15a}$$

subject to

$$g_j(\Phi) \le 0 \quad \text{for } j = 1, 2, \ldots, J, \tag{4.15b}$$

where

$$f_i(\Phi) = |H_p(\Phi, e^{j\omega_i})| \quad \text{for } i = 1, 2, \ldots, I \tag{4.15c}$$

and J and the $g_j(\Phi)$'s are given by Eqs. (4.13c), (4.13d), and (4.13e).

Again, the Dutta-Vidyasagar algorithm can be used to solve the above problem. Also, the optimization of the prototype filter for the case where $|T_0(\Phi, e^{j\omega})| \equiv 1$ can be solved like for Problem I. How to find a good initial vector Φ will be considered in the next subsection.

TABLE I
COMPARISON BETWEEN FILTER BANKS WITH $M = 32$ AND $\rho = 1$. BOLDFACE NUMBERS INDICATE THAT THESE PARAMETERS HAVE BEEN FIXED IN THE OPTIMIZATION

Criterion	K	N	δ_1	δ_2	E_∞	E_2
Least Squared	**8**	**511**	**0**	**0** $-\infty$ dB	$1.2 \cdot 10^{-3}$ -58 dB	$7.4 \cdot 10^{-9}$
Minimax	**8**	**511**	**0**	**0** $-\infty$ dB	$2.3 \cdot 10^{-4}$ -73 dB	$7.5 \cdot 10^{-8}$
Least Squared	**8**	**511**	$\mathbf{10^{-4}}$	$2.3 \cdot 10^{-6}$ -113 dB	$1.0 \cdot 10^{-5}$ -100 dB	$5.6 \cdot 10^{-13}$
Minimax	**8**	**511**	$\mathbf{10^{-4}}$	$1.1 \cdot 10^{-5}$ -99 dB	$5.1 \cdot 10^{-6}$ -106 dB	$3.8 \cdot 10^{-11}$
Least Squared	**8**	**511**	**0**	$9.1 \cdot 10^{-5}$ -81 dB	$4.5 \cdot 10^{-4}$ -67 dB	$5.4 \cdot 10^{-10}$
Least Squared	**8**	**511**	$\mathbf{10^{-2}}$	$5.3 \cdot 10^{-7}$ -126 dB	$2.4 \cdot 10^{-6}$ -112 dB	$4.5 \cdot 10^{-14}$
Least Squared	**6**	**383**	$\mathbf{10^{-3}}$	**0.00001** **-100 dB**	$1.7 \cdot 10^{-4}$ -75 dB	$8.8 \cdot 10^{-10}$
Least Squared	**5**	**319**	$\mathbf{10^{-2}}$	**0.0001** **-80 dB**	$8.4 \cdot 10^{-4}$ -62 dB	$2.7 \cdot 10^{-9}$

4.3.3 Initial Starting-Point Solutions

Good start-up solutions can be generated for Problems I and II by systematic multi-step procedures described in [26, 29] for generating perfect-reconstruction filter banks in such a way that the stopband behavior of the prototype filter is optimized in the minimax sense or in the least-mean-square sense. These procedures have been constructed in such a way that they are unconstrained optimization procedures. To achieve this, the basic unknowns have been selected such that the perfect-reconstruction property is satisfied independent of the values of the unknowns. Compared to other existing design methods, these synthesis procedures are faster and allow us to synthesize filter banks of significantly higher filter orders than the other existing design schemes.

For the perfect-reconstruction case, the order of the prototype filter is restricted to be $N = K \cdot 2M - 1$, where M is the number of filters in the analysis and synthesis banks and K is an integer. If the desired order does not satisfy this condition, then a good initial solution is found by first designing the perfect-reconstruction filter with the order of the prototype filter being selected such that K is the smallest integer making the overall order larger than the desired one. Then, the first and last impulse-response values are dropped out until achieving the desired order.

4.4 Comparisons

For comparison purposes, several filter banks have been optimized for $\rho = 1$ and $M = 32$, that is, the number of filters in the analysis and synthesis banks is 32. The stopband edge of the prototype filter is thus located at $\omega_s = \pi/32$. The results are summarized in Table I. In all the cases under consideration, the order of the prototype filter is $K \cdot 2M - 1$, where K is an integer and the stopband response is optimized in either the minimax or least-mean-square sense. δ_1 shows the maximum deviation of the amplitude response of the reconstruction error $T_0(z)$ from unity, whereas δ_2 is the maximum amplitude value of the worst-case aliasing transfer function $T_l(z)$. The boldface numbers indicate that these parameters have been fixed in the optimization. E_∞ and E_2 give the maximum stopband amplitude value of the prototype filter and the stopband energy, respectively.

The first two banks in Table I are perfect-reconstruction filter banks where the stopband performance has been optimized in the

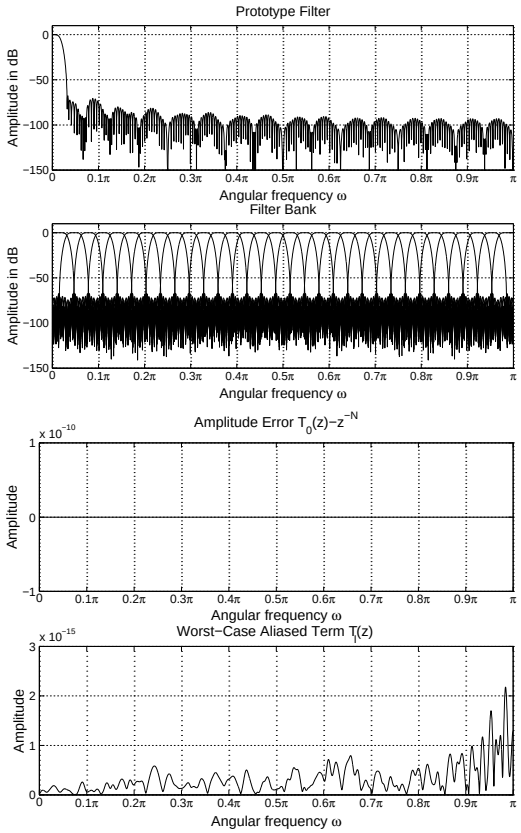

Fig. 4.5. Perfect-reconstruction filter bank of $M = 32$ filters of length $N+1 = 512$ for $\rho = 1$. The least-mean-square error design has been used.

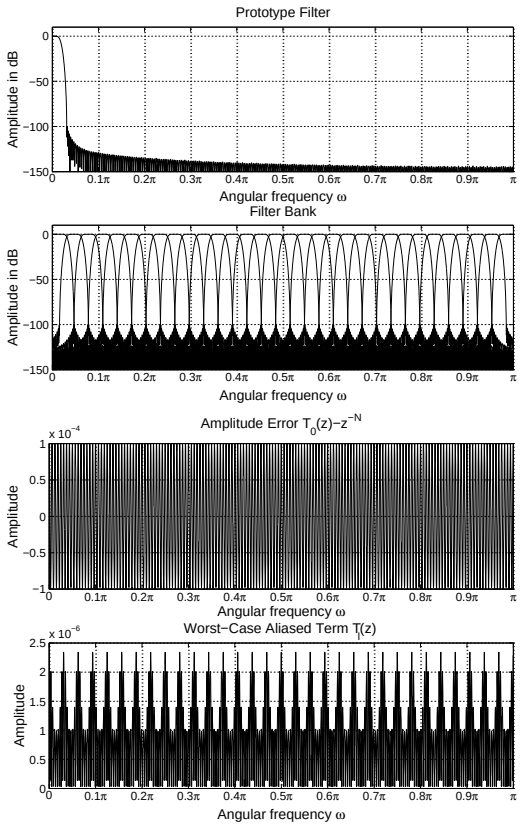

Fig. 4.6. Filter bank of $M = 32$ filters of length $N + 1 = 512$ for $\rho = 1$ and $\delta_1 = 0.0001$. The least-mean-square error design has been used.

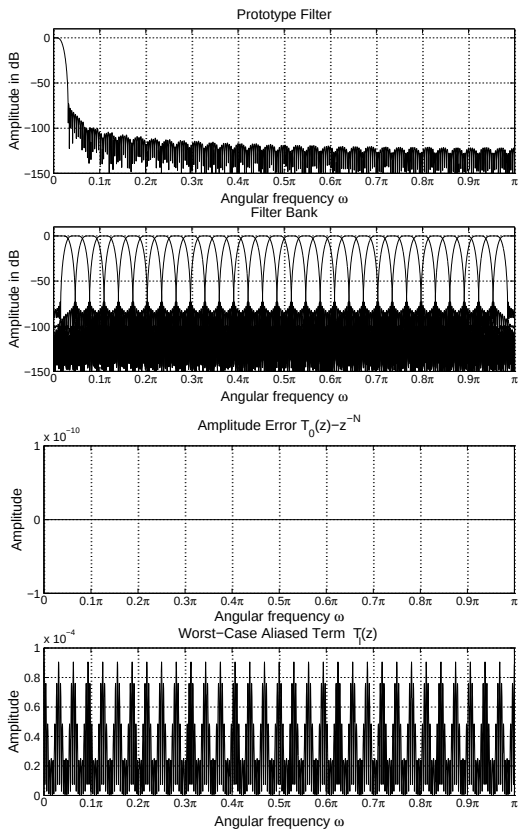

Fig. 4.7. Filter bank of $M = 32$ filters of length $N + 1 = 512$ for $\rho = 1$ and $\delta_1 = 0$. The least-mean-square error design has been used.

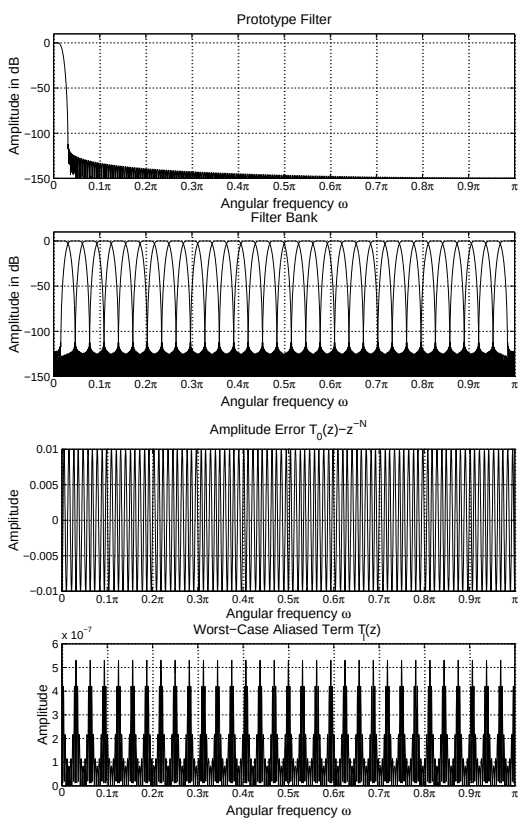

Fig. 4.8. Filter bank of $M = 32$ filters of length $N + 1 = 512$ for $\rho = 1$ and $\delta_1 = 0.01$. The least-mean-square error design has been used.

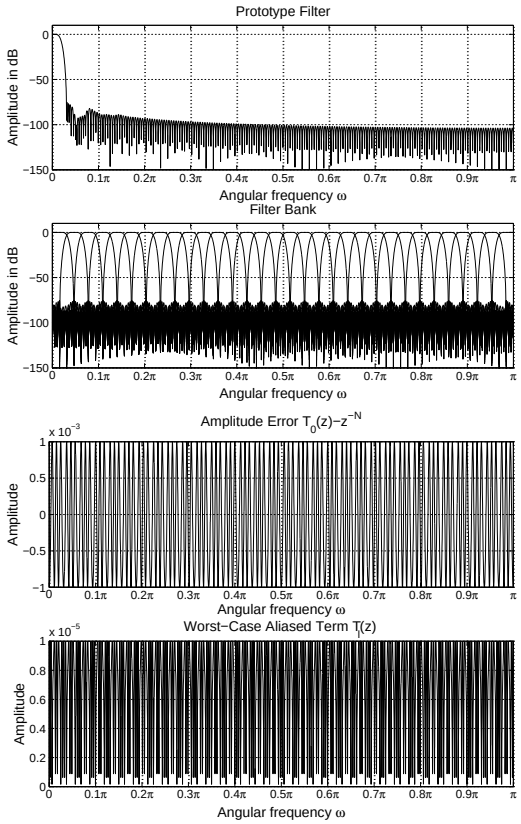

Fig. 4.9. Filter bank of $M = 32$ filters of length $N + 1 = 384$ for $\rho = 1$, $\delta_1 = 0.001$, and $\delta_2 = 0.00001$. The least-mean-square error design has been used.

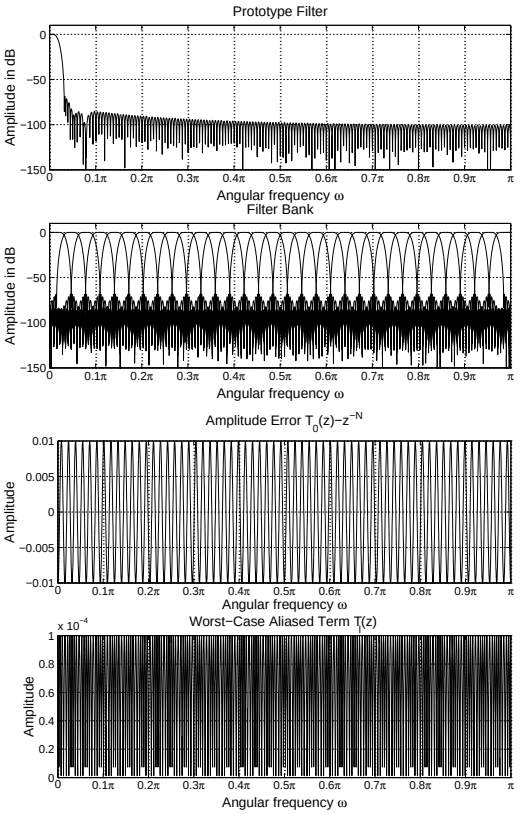

Fig. 4.10. Filter bank of $M = 32$ filters of length $N + 1 = 320$ for $\rho = 1$, $\delta_1 = 0.01$, and $\delta_2 = 0.0001$. The least-mean-square error design has been used.

least-mean-square sense and in the minimax sense. The third and fourth designs are the corresponding nearly perfect-reconstruction banks designed in such a way that the reconstruction error is restricted to be less than or equal to 0.0001. For these designs as well as for the fifth and sixth design in Table I, no constraints on the levels of the aliasing errors have been imposed. Some characteristics of the first and third designs are depicted in Figs. 4.5 and 4.6, respectively. From these figures as well as from Table I, it is seen that the nearly perfect-reconstruction filter banks provide significantly improved filter bank performances at the expense of a small reconstruction error and very small aliasing errors.

Even an optimized nearly perfect-reconstruction filter bank without reconstruction error (the fifth design in Table I) provides a considerably better performance than the perfect-reconstruction filter bank, as can be seen by comparing Figs. 4.5 and 4.7.

By comparing Figs. 4.5 and 4.8 as well as comparing the first and sixth designs in Table I, it is seen that the performance of the nearly perfect-reconstruction filter bank significantly improves when a higher reconstruction error is allowed.

For the last two designs in Table I, the orders of the prototype filters are decreased and they have been optimized subject to the given reconstruction and aliasing errors. Some of the characteristics of these designs are depicted Figs. 4.9 and 4.10. When comparing with the first perfect-reconstruction design of Table I (see also Fig. 4.5), it is observed that the same or even better filter bank performances can be achieved with lower orders when small errors are allowed.

5 DESIGN OF APPROXIMATELY LINEAR PHASE RECURSIVE DIGITAL FILTERS

One of the most difficult problems in digital filter synthesis is the simultaneous optimization of the phase and amplitude responses of recursive digital filters. This is because the phase of recursive filters is inherently nonlinear and, therefore, the amplitude selectivity and phase linearity are conflicting requirements. This application shows how the two-step approach of Section 3 can be applied in a systematic manner for minimizing the maximum passband phase deviation of recursive filters of various kinds from a linear phase subject to the given amplitude criteria. Furthermore, the benefits of the resulting recursive filters over their FIR equivalents are illustrated by means of several examples.

5.1 Background

The most straightforward approach to arrive at a recursive filter having simultaneously a selective amplitude response and an approximately linear phase response in the passband region is to generate the filter in two steps. First, a filter with the desired amplitude response is designed. Then, the phase response of this filter is made approximately linear in the passband by cascading it with an all-pass phase equalizer [32]. The main drawback in this approach is that the phase response of the amplitude-selective filter is usually very nonlinear and, therefore, a very high-order phase equalizer is needed in order to make the phase response of the overall filter very linear.

Therefore, it has turned out [33–38] to be more beneficial to implement an approximately linear phase recursive filter directly without using a separate phase equalizer. In the design techniques described in [33–38], it has been observed that in order to achieve, si-

multaneously, a selective amplitude response and an approximately linear phase performance in the passband, it is required that some zeros of the filter be located outside the unit circle.

This application considers the design of approximately linear phase recursive filters being implementable either in a conventional cascade form or as a parallel connection of two all-pass filters (lattice wave digital filters [39–41]). The selection among these two realizations depends on the practical implementation form. Cascade-form filters are usually implementable with a shorter code length in signal processors having several bits for both the coefficient and data representations. For VLSI implementations, in turn, lattice wave digital filters are preferred because of their lower coefficient sensitivity and lower output noise due to the multiplication roundoff errors.

In the case of a conventional cascade-form realization, the filter with an approximately linear phase in the passband is characterized by the fact that some zeros of the low-pass filter with transfer function of the form

$$H(z) = \gamma \prod_{k=1}^{N} (1 - \alpha_k z^{-1}) \Bigg/ \prod_{k=1}^{N} (1 - \beta_k z^{-1}) \tag{5.1}$$

lie outside the unit circle, as illustrated in Fig. 5.1(a). This figure gives a typical pole-zero plot for such a filter. Therefore, the overall filter is not a minimum-phase filter. However, the overall filter can be decomposed into a minimum-phase filter and an all-pass phase equalizer, as shown in Figs. 5.1(b) and 5.1(c). This decomposition will be used later on in this section for finding an initial filter for further optimization. Note that the poles of the all-pass filter cancel the zeros of the minimum-phase filter being located inside the unit circle, whereas the zeros of the all-pass filter generate those zeros of the overall filter that lie outside the unit circle.

In the case of a parallel connection of two all-pass filters, the transfer function is in the low-pass case of the form

$$H(z) = \frac{1}{2}[A(z) + B(z)], \tag{5.2a}$$

where

$$A(z) = \prod_{k=1}^{N_A} \frac{-\beta_k^{(A)} + z^{-1}}{1 - \beta_k^{(A)} z^{-1}}; \quad B(z) = \prod_{k=1}^{N_B} \frac{-\beta_k^{(B)} + z^{-1}}{1 - \beta_k^{(B)} z^{-1}} \tag{5.2b}$$

are stable all-pass filters of orders N_A and N_B, respectively. It is required that $N_A = N_B - 1$ or $N_A = N_B + 1$, so that the overall filter order $N = N_A + N_B$ is odd. This filter is completely characterized by the poles of $A(z)$ and $B(z)$. If $H(z)$ is desired to be determined in the form of Eq. (5.1) in a design technique, as will be done later in this section, then the following two conditions have to be satisfied in order for $H(z)$ be realizable in the form of Eq. (5.2) [42]:

1) *Condition A:* The numerator of $H(z)$ is a linear-phase FIR filter that is of even length $N+1$ and possesses a symmetric impulse response.

2) *Condition B:* For the high-pass complementary filter $G(z) = \frac{1}{2}[A(z) - B(z)]$ satisfying $H(z)H(1/z) + G(z)G(1/z) = 1$, the numerator is a linear-phase FIR filter that is of even length $N + 1$ and possesses an antisymmetric impulse response.

The first condition implies that if $H(z)$ has a real zero or a complex-conjugate zero pair outside the unit circle, then it possesses also a reciprocal real zero or a reciprocal complex-conjugate zero pair inside the unit circle. Fig. 5.2(a) shows a typical pole-zero plot for an approximately linear phase filter being realizable as a parallel connection of two all-pass filters. The main difference, compared to Fig. 5.1(a), is that now the off-the-unit-circle zeros occur in reciprocal pairs. Also in this case, the overall filter can be decomposed into a minimum-phase filter and a phase equalizer, as shown in Figs. 5.2(b) and 5.2(c). Note that the minimum-phase filter has now double zeros inside the unit circle.

The basic advantage of the filters shown in Figs. 5.1 and 5.2 is that the poles of the phase equalizer cancel the zeros of the minimum-phase filters that are located exactly at the same points. This reduces the overall filter order. For the parallel realization, the poles of the all-pass filter cancel only one each of the double zeros. When the phase of an amplitude-selective filter is equalized by using an all-pass filter, this cancellation does not take place and, consequently, the overall filter order becomes higher than in the above-mentioned decompositions.

This application shows how to design in a systematic manner selective low-pass recursive filters with an approximately linear phase using the two-step procedure of Section 3. In the first step, we utilize the decompositions of Figs. 5.1 and 5.2 and a suboptimal filter is designed iteratively until achieving the desired pole-zero cancellation as proposed in [33, 34, 38]. For performing the first step, a very efficient synthesis method described in [38] is used. The second step is carried out by using the Dutta-Vidyasagar algorithm described in Subsection 3.2.1. Several examples are included illustrating the superiority of the optimized filters over their linear-phase FIR equivalents especially in narrow-band cases.

5.2 Statement of the Approximation Problems

Before stating the approximation problems, we denote the transfer function of the filter by $H(\Phi, z)$, where Φ is the adjustable parameter vector. For the cascade-form realization,

$$\Phi = [\gamma, \alpha_1, \ldots, \alpha_N, \beta_1, \ldots, \beta_N] \tag{5.3}$$

and for the parallel-form realization,

$$\Phi = [\beta_1^{(A)}, \ldots, \beta_{N_A}^{(A)}, \beta_1^{(B)}, \ldots, \beta_{N_B}^{(B)}]. \tag{5.4}$$

Furthermore, we denote by $\arg H(\Phi, e^{j\omega})$ the unwrapped phase response of the filter.

We state the following two approximation problems:

Approximation Problem I: Given ω_p, ω_s, δ_p, and δ_s, as well as the filter order N, find Φ and ψ, the slope of a linear phase response, to minimize

$$\Delta = \max_{0 \le \omega \le \omega_p} |\arg H(\Phi, e^{j\omega}) - \psi\omega| \tag{5.5a}$$

subject to

$$1 - \delta_p \le |H(\Phi, e^{j\omega})| \le 1 \quad \text{for } \omega \in [0, \omega_p], \tag{5.5b}$$

$$|H(\Phi, e^{j\omega})| \le \delta_s \quad \text{for } \omega \in [\omega_s, \pi], \tag{5.5c}$$

and

$$|H(\Phi, e^{j\omega})| \le 1 \quad \text{for } \omega \in (\omega_p, \omega_s). \tag{5.5d}$$

Approximation Problem II: Given ω_p, ω_s, δ_p, and δ_s, as well as the filter order N, find Φ and ψ to minimize Δ as given by Eq. (5.5a) subject to the conditions of Eqs. (5.5b) and (5.5c) and

$$\frac{d|H(\Phi, e^{j\omega})|}{d\omega} \le 0 \quad \text{for } \omega \in (\omega_p, \omega_s). \tag{5.5e}$$

In the above approximation problems, the maximum deviation of the phase response from a linear phase response $\phi_{\text{ave}}(\omega) = \psi\omega$ is desired to be minimized subject to the given amplitude specifications. Note that ψ is also an adjustable parameter. In Approximation

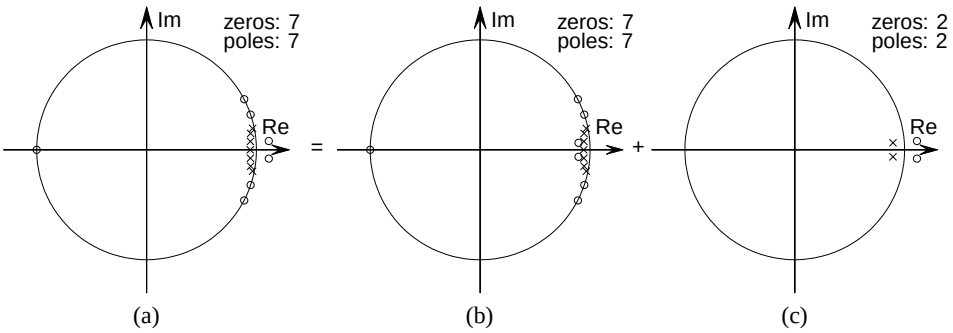

Fig. 5.1. A typical pole-zero plot for an approximately linear phase recursive filter being realizable in a cascade form and the decomposition of this filter into a minimum-phase filter and an all-pass phase equalizer.

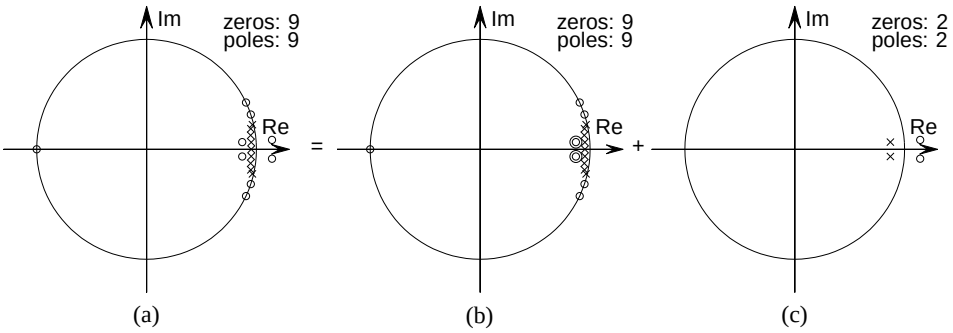

Fig. 5.2. A typical pole-zero plot for an approximately linear phase recursive filter being realizable as a parallel connection of two all-pass filters and the decomposition of this filter into a minimum-phase filter and an all-pass phase equalizer.

Problem I, it is required that the maximum value of the amplitude response be restricted to be unity in the transition band, whereas in Approximation Problem II, the amplitude response is forced to be monotonically decreasing in the transition band.

Whether to use Approximation Problem I or II depends strongly on the application. If the input-signal components within the filter transition band are not significant, then Approximation Problem I can be used. In this case, the transition band components are not amplified. In the opposite case, some attenuation in the transition band is required, and a monotonically decreasing amplitude response in this band is an appropriate selection.

5.3 Algorithms for Finding Initial Filters

This subsection describes efficient algorithms for generating good initial filters for further optimization. In these algorithms, Candidate I and Candidate II filters denote initial filters for Approximation Problems I and II, respectively.

5.3.1 Cascade-Form Filters

For the cascade-form realization, good initial filters can be found by using iteratively the following five-step procedure:

Step 1: Determine the minimum order of an elliptic filter to meet the given amplitude criteria. Denote the minimum order by $N_{\min}$ and set $k = 1$. Then, design an elliptic filter transfer function $H_{\min}^{(k)}(z)$ such that it satisfies

Condition 1: $|H_{\min}^{(k)}(e^{j\omega})|$ oscillates in the stopband $[\omega_s, \pi]$ between δ_s and 0 achieving these values at $N_{\min}+1$ points such that the value at $\omega = \omega_s$ is δ_s. Here, δ_s is the specified stopband ripple.

Condition 2: $|H_{\min}^{(k)}(e^{j\omega})|$ oscillates in the interval $[0, \Omega_p^{(k)}]$ ($\Omega_p^{(k)} \geq \omega_p$) between 1 and $1-\delta_p^{(k)}$ achieving these values at $N_{\min}+1$ points such that the value at $\omega = \Omega_p^{(k)}$ is $1-\delta_p^{(k)}$. For Candidate I, $\delta_p^{(k)} = \delta_p$ with δ_p being the specified passband ripple, whereas the passband region $[0, \Omega_p^{(k)}]$ is the widest possible to still meet the given stopband requirements. For Candidate II, $\Omega_p^{(k)} = \omega_p$ with ω_p being the specified passband edge, whereas $\delta_p^{(k)}$ is the smallest passband ripple to still meet the given stopband criteria.[5]

Step 2: Cascade $H_{\min}^{(k)}(z)$ with a stable all-pass equalizer with a transfer function $H_{\text{all}}^{(k)}(z)$ of order N_{all}. Determine the adjustable parameters of $H_{\text{all}}^{(k)}(z)$ and $\psi^{(k)}$ such that the maximum deviation of $\arg[H_{\text{all}}^{(k)}(e^{j\omega})H_{\min}^{(k)}(e^{j\omega})]$ from the average slope $\phi_{\text{ave}}(\omega) = \psi^{(k)}\omega$ is minimized in the specified passband region $[0, \omega_p]$. Let the poles of the all-pass filter be located at $z = z_1^{(k)}, z_2^{(k)}, \ldots, z_{N_{\text{all}}}^{(k)}$.

Step 3: Set $k = k+1$. Then, design a minimum-phase filter transfer function $H_{\min}^{(k)}(z)$ of order $N_{\min}+N_{\text{all}}$ such that it has N_{all} fixed zeros at $z = z_1^{(k-1)}, z_2^{(k-1)}, \ldots, z_{N_{\text{all}}}^{(k-1)}$ and it satisfies Condition 1 of Step 1 with the same number of extremal points, that is, $N_{\min}+1$, and Condition 2 of Step 1 with $N_{\min}+N_{\text{all}}+1$ extremal points, instead of $N_{\min}+1$ points.

Step 4: Like at Step 2, cascade $H_{\min}^{(k)}(z)$ with a stable all-pass filter transfer function $H_{\text{all}}^{(k)}(z)$ of order N_{all} and determine

[5]The amplitude response of this filter, as well as that of the corresponding filter obtained at Step 3, automatically has a monotonically decreasing amplitude response in the transition band. This behavior is needed for the initial filter to be used for further optimization in the case of Approximation Problem II. Similarly, the Candidate I filter satisfies the transition band restriction of Approximation Problem I.

its adjustable parameters and $\psi^{(k)}$ such that the maximum of $|\arg[H_{\rm all}^{(k)}(e^{j\omega})H_{\rm min}^{(k)}(e^{j\omega})] - \psi^{(k)}\omega|$ is minimized in $[0,\ \omega_p]$. Let the poles of the all-pass filter be located at $z = z_1^{(k)}, z_2^{(k)}, \ldots, z_{N_{\rm all}}^{(k)}$.

Step 5: If $|z_l^{(k)} - z_l^{(k-1)}| \le \epsilon$ for $l = 1, 2, \ldots, N_{\rm all}$ (ϵ is a small positive number[6]), then stop. In this case, the zeros of the minimum-phase filter being located inside the unit circle and the poles of the all-pass equalizer coincide (see Fig. 5.1), reducing the overall order of $H(z) = H_{\rm min}^{(k)}(z)H_{\rm all}^{(k)}(z)$ from $N_{\rm min} + 2N_{\rm all}$ to $N_{\rm min} + N_{\rm all}$. This filter is the desired initial filter with approximately linear-phase characteristics. Otherwise, go to Step 3.

5.3.2 Parallel-Form Filters

When designing initial filters for the parallel connection of two all-pass filters, only Steps 3 and 5 need to be modified. The basic modification of Step 3 is that the minimum-phase filter is now of order $N_{\rm min} + 2N_{\rm all}$ and it possesses double zeros at $z = z_1^{(k)}, z_2^{(k)}, \ldots, z_{N_{\rm all}}^{(k)}$ (see Fig. 5.2). Consequently, Condition 2 of Step 1 should be satisfied with $N_{\rm min} + 2N_{\rm all} + 1$ extremal points. In the case of Step 5, the algorithm is terminated when the double zeros of the minimum-phase filter being located inside the unit circle and the poles of the all-pass phase equalizer coincide (see Fig. 5.2). This reduces the overall order of $H(z) = H_{\rm min}^{(k)}(z)H_{\rm all}^{(k)}(z)$ from $N_{\rm min} + 3N_{\rm all}$ to $N_{\rm min} + 2N_{\rm all}$. The third modification in the low-pass case is that $N_{\rm min}$ should be an odd number.

The resulting $H(z)$ automatically satisfies Conditions A and B given in Subsection 5.1, thereby guaranteeing that it is implementable as a parallel connection of two all-pass filters.[7] This is based on the following facts. Condition A is satisfied since the numerator of $H(z)$ possesses one zero at $z = -1$ ($N_{\rm min}$ is odd), $N_{\rm all}$ reciprocal zero pairs off the unit circle, and $(N_{\rm min} - 1)/2$ complex-conjugate zero pairs on the unit circle. Therefore, the numerator of $H(z)$ is a linear-phase FIR filter with a symmetric impulse response of even length $N_{\rm min} + 2N_{\rm all} + 1$, as is desired. Due to facts that $N_{\rm min}$ is odd and $H(z)$ satisfies Condition 2 of step 1, with $N_{\rm min} + 2N_{\rm all} + 1$ extremal points, $|H(e^{j\omega})|$ achieves the value of unity at $\omega = 0$ and at $(N_{\rm min} - 1)/2 + N_{\rm all}$ other angular frequencies ω_l in the passband. Therefore, the numerator of the power-complementary transfer function $G(z)$ contains one zero at $z = 1$ and complex-conjugate zero pairs on the unit circle at $z = \exp(\pm j\omega_l)$ for $l = 1, 2, \ldots, (N_{\rm min} - 1)/2 + N_{\rm all}$. Hence, the numerator of $G(z)$ is a linear-phase FIR filter with an antisymmetric impulse response of even length $N_{\rm min} + 2N_{\rm all} + 1$, as is required by Condition B.

5.3.3 Subalgorithms

Steps 1 and 3 can be performed very fast by applying the algorithm described in the Appendix in [38]. For designing the phase equalizer at Steps 2 and 4, we have used the Dutta-Vidyasagar algorithm described in Subsection 3.2.1. In order to use this algorithm, the passband region is discretized into the frequency points $\omega_i \in [0,\ \omega_p]$, $i = 1, 2, \ldots, I_p$. The resulting discrete optimization problem is then to find the adjustable parameter vector Φ containing the $N_{\rm all}$ poles of the all-pass filter transfer function $H_{\rm all}^{(k)}(z)$ as well as $\psi^{(k)}$ to minimize

$$\rho(\Phi, \psi^{(k)}) = \max_{1 \le i \le I_p} e_i(\Phi, \psi)\}, \tag{5.6a}$$

where for $i = 1, 2, \ldots, I_p$

$$e_i(\Phi, \psi) = |\arg[H_{\rm all}^{(k)}(e^{j\omega_i})H_{\rm min}^{(k)}(e^{j\omega_i})] - \psi^{(k)}\omega_i|. \tag{5.6b}$$

5.4 Further Optimization

The Dutta-Vidyasagar algorithm can be applied in a straightforward manner to further reducing the phase error of the initial filter. In order to use this algorithm for solving Approximation Problem I, we discretize the passband, the transition band, and the stopband regions into the frequency points $\omega_i \in [0,\ \omega_p]$, $i = 1, 2, \ldots, I_p$, $\omega_i \in (\omega_p,\ \omega_s)$, $i = I_p + 1, \ldots, I_p + I_t$, and $\omega_i \in [\omega_s,\ \pi]$, $i = I_p + I_t + 1, \ldots, I_p + I_t + I_s$. The resulting discrete minimax problem is to find Φ and ψ to minimize

$$\Delta = \max_{1 \le i \le I_p} f_i(\Phi, \psi) \tag{5.7a}$$

subject to

$$g_i(\Phi, \psi) \le 0 \quad \text{for } i = 1, 2, \ldots, I_p + I_t + I_s\,, \tag{5.7b}$$

where

$$f_i(\Phi, \psi) = |\arg H(\Phi, e^{j\omega_i}) - \psi\omega_i| \quad \text{for } i = 1, 2, \ldots, I_p \tag{5.7c}$$

and

$$g_i(\Phi, \psi) = \begin{cases} \big||H(\Phi, e^{j\omega_i})| \\ \quad -(1 - \frac{\delta_p}{2})\big| - \frac{\delta_p}{2}, & i = 1, 2, \ldots, I_p \\ |H(\Phi, e^{j\omega_i})| - 1, & i = I_p + 1, \ldots, I_p + I_t \\ |H(\Phi, e^{j\omega_i})| - \delta_s, & i = I_p + I_t + 1, \ldots, \\ & \qquad I_p + I_t + I_s. \end{cases} \tag{5.7d}$$

For Approximation Problem II, the $g_i(\Phi, \psi)$'s for $i = I_p + 1, \ldots, I_p + I_t$ are replaced by

$$g_i(\Phi, \psi) = G(\Phi, e^{j\omega_i}), \tag{5.8a}$$

where

$$G(\Phi, e^{j\omega}) = \frac{d|H(\Phi, e^{j\omega})|}{d\omega}. \tag{5.8b}$$

5.5 Numerical Examples

This section shows, by means of examples, the efficiency and flexibility of the proposed design technique as well as the superiority of the resulting optimized approximately linear phase recursive filters over their linear-phase FIR equivalents. More examples can be found in [38].

[6] A proper value of ϵ depends on the passband bandwidth of the filter. In the case of Example 1 to be considered in Subsection 5.5, $\epsilon = 10^{-6}$ is a good selection. For filters with a very narrow passband bandwidth, a lower value should be used since the radii of the poles and zeros being located approximately at the same point increase.

[7] After knowing the poles of the filter, the problem is to implement the overall transfer function as $H(z) = [A(z) + B(z)]/2$ in such a way that the poles are properly shared with the all-pass sections $A(z)$ and $B(z)$. If the poles are distributed in the low-pass case in a regular manner, $A(z)$ can be selected to realize the real pole, the second innermost complex-conjugate pole pair, the fourth innermost complex-conjugate pole pair and so on, whereas $B(z)$ realizes the remaining poles [41]. For a very complicated pole distribution, the procedure described in [42] can be used by sharing the poles between $A(z)$ and $B(z)$.

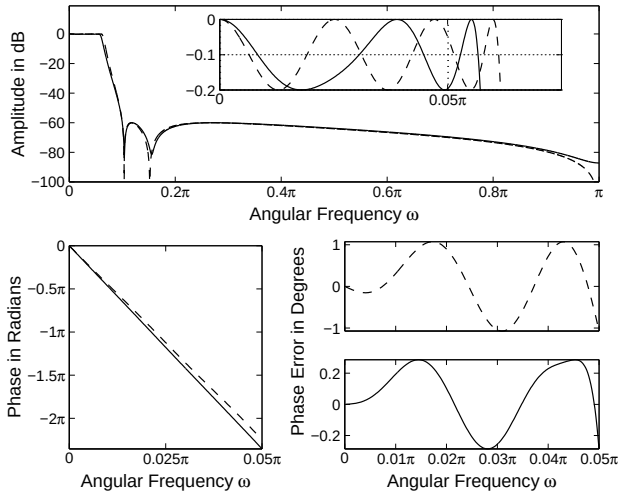

Fig. 5.3. Amplitude and phase responses for the initial filter (dashed line) and the optimized filter (solid line) for the cascade-form realization in Approximation Problem I.

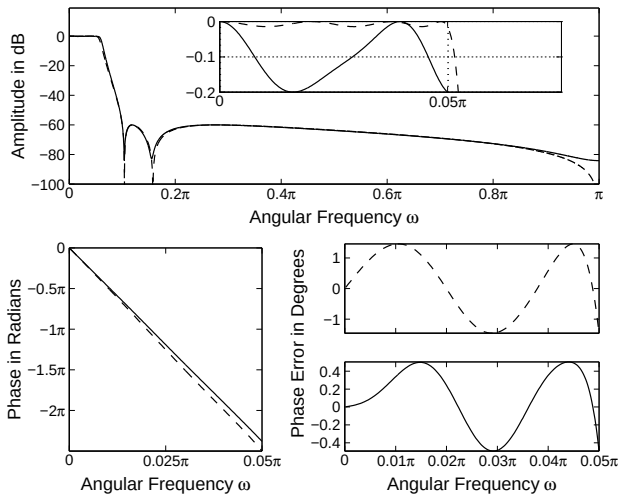

Fig. 5.4. Amplitude and phase responses for the initial filter (dashed line) and the optimized filter (solid line) for the cascade-form realization in Approximation Problem II.

5.5.1 Example 1

The filter specifications are: $\omega_p = 0.05\pi$, $\omega_s = 0.1\pi$, $\delta_p = 0.0228$ (0.2-dB passband variation), and $\delta_s = 10^{-3}$ (60-dB stopband attenuation). The minimum order of an elliptic filter to meet these criteria is five. In the case of the cascade-form realization good phase performances in the passband region are achieved by increasing the filter order by two (to seven). Figs. 5.3 and 5.4 show the amplitude and phase responses for the initial filters and the optimized filters for Approximation Problems I and II, respectively, whereas Fig. 5.1(a) shows the pole-zero plot for the initial filter for Approximation Problem I. Some of the filter characteristics are summarized in Table II. In addition to the zero and pole locations as well as the scaling constant γ [cf. Eq. (5.1)], the average phase slope $\phi_{\rm ave}(\omega) = \psi\omega$ in the passband as well as the maximum phase deviation from this curve in degrees are shown in this table.[8]

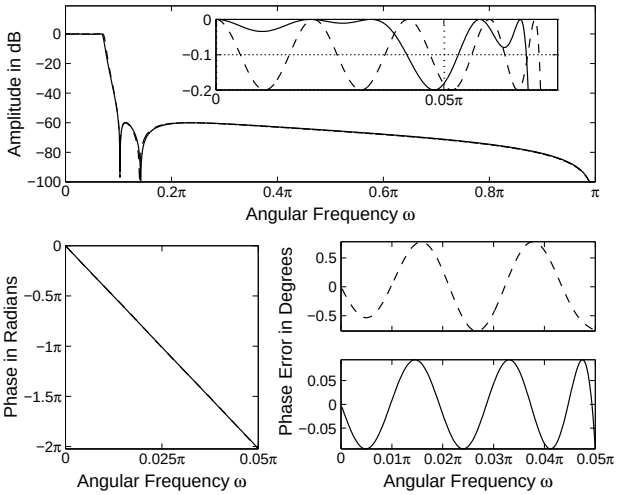

Fig. 5.5. Amplitude and phase responses for the initial filter (dashed line) and the optimized filter (solid line) for the parallel connection of two all-pass filters in Approximation Problem I.

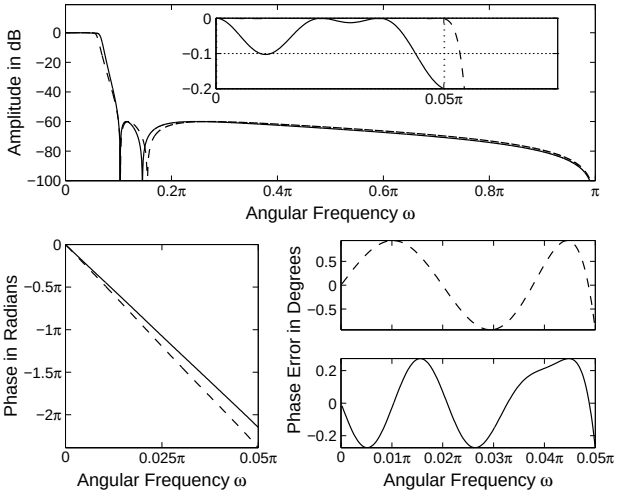

Fig. 5.6. Amplitude and phase responses for the initial filter (dashed line) and the optimized filter (solid line) for the parallel connection of two all-pass filters in Approximation Problem II.

It can be observed that the further optimization considerably reduces the phase error and, as can be expected, the error is smaller for the optimized filter in Approximation Problem I. For the optimized filters, the phase error becomes practically negligible by increasing the filter order just by two. Therefore, there is no need to further increase the filter order.

In the case of the parallel connection of two all-pass filters, excellent phase performances are obtained by increasing the filter order

[8]It has been observed that by minimizing the phase deviation, the group delay variation around the passband average is at the same time approximately minimized. Another advantage of using the phase deviation as a measure of goodness of the phase approximation lies in the fact that if the passband and stopband edges of a filter are divided by the same number and for the optimized filters the phase errors are similar, then the phase performances of these two filters can be regarded to be equally good. This will be illustrated by Example 2.

TABLE II

SOME CHARACTERISTICS FOR THE CASCADE-FORM FILTERS OF EXAMPLE 1

Approximation Problem I: Initial Filter	
$\phi_{\rm ave}(\omega) = -44.844691\omega$	$\Delta = 1.07134958$ degrees
$\gamma = 5.5746038 \cdot 10^{-4}$	
Pole Locations: 0.94259913 $0.94744290\exp(\pm j0.02578619\pi)$ $0.95792167\exp(\pm j0.04960331\pi)$ $0.98534890\exp(\pm j0.06333788\pi)$	Zero Locations: -1.00000000 $1.11625645\exp(\pm j0.02276191\pi)$ $1.00000000\exp(\pm j0.10365367\pi)$ $1.00000000\exp(\pm j0.15191514\pi)$
Approximation Problem I: Optimized Filter	
$\phi_{\rm ave}(\omega) = -47.058896\omega$	$\Delta = 0.28591762$ degrees
$\gamma = 6.1544227 \cdot 10^{-4}$	
Pole Locations: 0.93250961 $0.93372619\exp(\pm j0.02364935\pi)$ $0.94863891\exp(\pm j0.04405200\pi)$ $0.98306166\exp(\pm j0.05922065\pi)$	Zero Locations: -0.88224205 $1.10469004\exp(\pm j0.02182439\pi)$ $0.99892340\exp(\pm j0.10395934\pi)$ $0.99009291\exp(\pm j0.15551431\pi)$
Approximation Problem II: Initial Filter	
$\phi_{\rm ave}(\omega) = -49.836122\omega$	$\Delta = 1.46126328$ degrees
$\gamma = 5.8848548 \cdot 10^{-4}$	
Pole Locations: 0.92232131 $0.93570263\exp(\pm j0.02149506\pi)$ $0.94197922\exp(\pm j0.04229640\pi)$ $0.97889580\exp(\pm j0.05595920\pi)$	Zero Locations: -1.00000000 $1.09178860\exp(\pm j0.02120304\pi)$ $1.00000000\exp(\pm j0.10416179\pi)$ $1.00000000\exp(\pm j0.15780807\pi)$
Approximation Problem II: Optimized Filter	
$\phi_{\rm ave}(\omega) = -47.558734\omega$	$\Delta = 0.49831219$ degrees
$\gamma = 6.3230785 \cdot 10^{-4}$	
Pole Locations: 0.93542431 $0.93715536\exp(\pm j0.02410709\pi)$ $0.94848758\exp(\pm j0.04476666\pi)$ $0.98139390\exp(\pm j0.05911262\pi)$	Zero Locations: -0.83671079 $1.10155821\exp(\pm j0.02171602\pi)$ $0.99897232\exp(\pm j0.10395156\pi)$ $0.99153409\exp(\pm j0.15541667\pi)$

TABLE III

SOME CHARACTERISTICS FOR THE PARALLEL-FORM FILTERS OF EXAMPLE 1

Approximation Problem I: Initial Filter	
$\phi_{\rm ave}(\omega) = -40.384179\omega$	$\Delta = 0.77224193$ degrees
$\gamma = 4.9737264 \cdot 10^{-4}$	
Pole Locations: 0.95281845 $0.95400281\exp(\pm j0.02149253\pi)$ $0.95515809\exp(\pm j0.04268776\pi)$ $0.96701405\exp(\pm j0.06222002\pi)$ $0.98889037\exp(\pm j0.07255183\pi)$	Zero Locations: -1.00000000 $1.14588962\exp(\pm j0.02374703\pi)$ $0.87268440\exp(\pm j0.02374703\pi)$ $1.00000000\exp(\pm j0.10274720\pi)$ $1.00000000\exp(\pm j0.13980046\pi)$
Approximation Problem I: Optimized Filter	
$\phi_{\rm ave}(\omega) = -40.380976\omega$	$\Delta = 0.093998740$ degrees
$\gamma = 4.9584076 \cdot 10^{-4}$	
Pole Locations: 0.93884355 $0.94019623\exp(\pm j0.01990415\pi)$ $0.94572416\exp(\pm j0.03930393\pi)$ $0.96261396\exp(\pm j0.05987200\pi)$ $0.98706672\exp(\pm j0.07033312\pi)$	Zero Locations: -1.00000000 $1.15385004\exp(\pm j0.02257140\pi)$ $0.86666375\exp(\pm j0.02257140\pi)$ $1.00000000\exp(\pm j0.10291368\pi)$ $1.00000000\exp(\pm j0.14188374\pi)$
Approximation Problem II: Initial Filter	
$\phi_{\rm ave}(\omega) = -47.633016\omega$	$\Delta = 0.93893412$ degrees
$\gamma = 5.7199166 \cdot 10^{-4}$	
Pole Locations: 0.92355547 $0.93295052\exp(\pm j0.01674544\pi)$ $0.93399165\exp(\pm j0.02993278\pi)$ $0.94243024\exp(\pm j0.04768063\pi)$ $0.97884013\exp(\pm j0.05870589\pi)$	Zero Locations: -1.00000000 $1.10253033\exp(\pm j0.02169563\pi)$ $0.90700453\exp(\pm j0.02169563\pi)$ $1.00000000\exp(\pm j0.10388928\pi)$ $1.00000000\exp(\pm j0.15445551\pi)$
Approximation Problem II: Optimized Filter	
$\phi_{\rm ave}(\omega) = -42.923013\omega$	$\Delta = 0.27404701$ degrees
$\gamma = 5.2027545 \cdot 10^{-4}$	
Pole Locations: 0.94645611 $0.94596287\exp(\pm j0.02039251\pi)$ $0.94520418\exp(\pm j0.03925109\pi)$ $0.95275023\exp(\pm j0.05889096\pi)$ $0.97948758\exp(\pm j0.06612600\pi)$	Zero Locations: -1.00000000 $1.13375315\exp(\pm j0.02195365\pi)$ $0.88202622\exp(\pm j0.02195365\pi)$ $1.00000000\exp(\pm j0.10317712\pi)$ $1.00000000\exp(\pm j0.14519421\pi)$

from five to nine. Figures 5.5 and 5.6 show the amplitude and phase responses for the initial filters and the optimized filters for Approximation Problems I and II, respectively, whereas Fig. 5.2(a) shows the pole-zero plot for the initial filter for Approximation Problem I. Some characteristics of these four filters are summarized in Table III.[9] The same observations as for the cascade-form realization can be made. For the parallel connection, the phase errors are smaller due to a larger increase in the overall filter order.

The minimum order of a linear-phase FIR filter to meet the same amplitude criteria is 107, requiring 107 delay elements and 54 multipliers when exploiting the coefficient symmetry. The corresponding wave lattice filter of order nine is implementable by using only nine delays and nine multipliers [39–41]. The delay of the linear-phase FIR equivalent is 53.5 samples, whereas for the proposed recursive filters the delay is smaller, as can be seen from Tables II and III.

5.5.2 Example 2

The specifications are the same as in Example 1 except that the passband and stopband edges are divided by five, that is, $\omega_p = 0.01\pi$ and $\omega_s = 0.02\pi$. Figure 5.7 shows the amplitude and phase responses for the optimized cascade-form and parallel-form realizations in Approximation Problem I. The filter orders are the same as in Example 1. For the cascade-form and parallel-form filters, $\Delta = 0.304\,947\,65$ degrees and $\phi_{\rm ave}(\omega) = -235.742\,76\omega$; and $\Delta = 0.098\,114\,381$ degrees and $\phi_{\rm ave}(\omega) = -202.426\,00\omega$, respectively. When comparing these figures with the corresponding figures in Tables II and III and Fig. 5.7 with Figs. 5.3 and 5.5, the following observations can be made. The performances of the filters of Example 1 and 2 are practically the same in the passband region, in the transition band, and in the beginning of the stopband. The phase errors are approximately the same, whereas the slope of the average linear phase is in Example 2 very accurately five times that of Example 1, making the delay five times longer, as can be expected.

5.5.3 Basic Properties of the Proposed Filters

The above observations have been experimentally verified to be valid for the initial and optimized filters in the two approximation problems for both the cascade-form and parallel-form realizations. That is, as the passband and stopband edges are divided by the same number Λ, the resulting initial and optimized filters of the same order have approximately the same phase errors as the original filters. Furthermore, the average phase slopes or the delays are Λ times those of the original ones. For the corresponding linear-phase FIR filters, the order becomes approximately Λ times that of the original one. This shows that in very narrow band cases the proposed optimized filters become very attractive compared to their linear-phase FIR equivalents, in terms of the number of multipliers, adders, and delay elements.

[9] In this table, the poles and zero locations as well as the scaling constant are given like for the cascade-form realization in order to emphasize the zero locations. In the practical implementation, one of the all-pass sections realizes the real pole, the second innermost pole pair, and the fourth innermost pole pair, whereas the second all-pass filter realizes the remaining pole pairs.

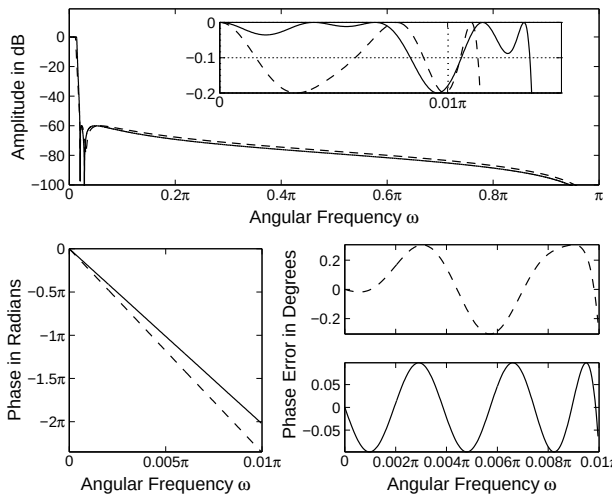

Fig. 5.7. Amplitude and phase responses for the optimized filters of Example 2 in Approximation Problem I. The dashed and solid lines show the responses for the cascade-form and parallel-form filters, respectively.

6 MODIFIED FARROW STRUCTURE WITH ADJUSTABLE FRACTIONAL DELAY

In various DSP applications, there is a need for a delay which is a fraction of the sampling interval. Furthermore, it is often desired that the delay value is adjustable during the computation. These applications include, e.g., echo cancellation, phased array antenna systems, time delay estimation, timing adjustment in all-digital receivers, and speech coding and synthesis. There exist two basic approaches to constructing such systems using FIR filters. The first one is to optimize several FIR filters for various values of the fractional delay. Another, computationally more efficient technique, is to use the Farrow structure [43] consisting several parallel fixed FIR filters. The desired fractional delay is achieved by properly multiplying the outputs of these filters with quantities depending directly on the value of the fractional delay [43–46].

This applications shows how to optimize, with the aid of the two-step procedure described in Section 3, the parameters of the modified Farrow structure introduced by Vesma and Saramäki in [47]. This structure contains a given number of fixed linear-phase FIR filters of the same even length. The attractive feature of this structure is that the fractional delay can take any value between zero and one sampling interval just by changing one adjustable parameter. Another attractive feature is that the (even) lengths of the fixed FIR filters as well as the number of filters can be arbitrarily selected. In addition to describing the optimization algorithm, some special features of the proposed structure are discussed.

6.1 Proposed Filter Structure

The proposed structure with an adjustable fractional delay μ is depicted in Fig. 6.1 [47]. It consists of $L+1$ parallel FIR filters with transfer functions of the form

$$G_l(z) = \sum_{n=0}^{N-1} g_l(n) z^{-n} \quad \text{for } l = 0, 1, \ldots, L, \tag{6.1}$$

where N is an even integer. The impulse-response coefficients $g_l(n)$ for $n = 0, 1, \ldots, N/2-1$ fulfill the following symmetry conditions:

$$g_l(n) = \begin{cases} g_l(N-1-n) & \text{for } l \text{ even} \\ -g_l(N-1-n) & \text{for } l \text{ odd.} \end{cases} \tag{6.2}$$

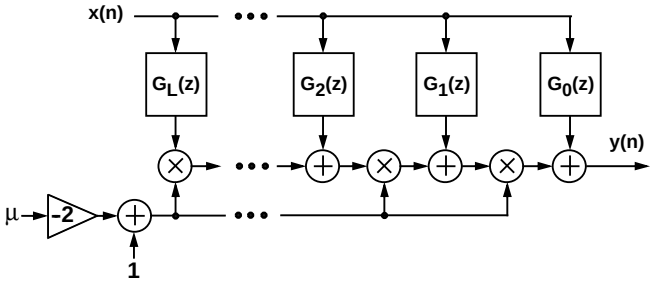

Fig. 6.1. The modified Farrow structure with an adjustable fractional delay μ.

After optimizing the above impulse-response coefficients in the manner to be described later on, the role of the adjustable parameter μ in Fig. 6.1 is to generate the delay equal to $N-1+\mu$ in the given passband region. This parameter can be varied between zero and unity. The desired delay can be obtained by multiplying the output of $G_l(z)$ by $(1-2\mu)^l$ for $l = 0, 1, \ldots, L$. For the given value μ, the overall transfer function is expressible as

$$H(\Phi, z, \mu) = \sum_{n=0}^{N-1} h(n, \Phi, \mu) z^{-n}, \tag{6.3a}$$

where

$$h(n, \Phi, \mu) = \sum_{l=0}^{L} g_l(n) (1-2\mu)^l \tag{6.3b}$$

and Φ is the adjustable parameter vector

$$\Phi = \big[g_0(0), g_0(1), \ldots, g_0(N/2-1), g_1(0), g_1(1), \ldots, g_1(N/2-1), \ldots, g_L(0), g_L(1), \ldots, g_L(N/2-1)\big]. \tag{6.3c}$$

The frequency, amplitude, and phase delay responses of the proposed Farrow structure are given by

$$H(\Phi, e^{j\omega}, \mu) = \sum_{n=0}^{N-1} h(n, \Phi, \mu) e^{-j\omega n}, \tag{6.4a}$$

$$|H(\Phi, e^{j\omega}, \mu)| = \Big| \sum_{n=0}^{N-1} h(n, \Phi, \mu) e^{-j\omega n} \Big|, \tag{6.4b}$$

and

$$\tau_p(\Phi, \omega, \mu) = -\arg(H(\Phi, e^{j\omega}, \mu))/\omega, \tag{6.4c}$$

respectively.

The original Farrow structure [43] can equally well be used for the same purpose. The main modifications are that the output of $G_l(z)$ is multiplied by $(1-\mu)^l$ and the $G_l(z)$'s are no longer linear-phase transfer functions. The equation to translate the coefficients of the original structure denoted by $\widehat{g}_l(n)$ to the coefficients of the modified structure denoted by $g_l(n)$ is given by [47]

$$g_l(n) = \sum_{k=l}^{L} (1/2)^k \binom{k}{l} \widehat{g}_k(n) \tag{6.5}$$

for $n = 0, 1, \ldots, N-1$ and $l = 0, 1, \ldots, L$. Therefore, the coefficient symmetries cannot be exploited. For the modified structure, the overall number of multipliers can be reduced to $N(L+1)/2+L$ when utilizing the symmetries of Eq. (6.2).

6.2 Optimization Problem

This subsection gives the optimization problem for the proposed overall system with an adjustable fractional delay. Furthermore, it shows that given N and the passband region, there exists a lower limit for the achievable amplitude distortion independent of L.

6.2.1 Statement of the Problem

If the frequency band of interest is

$$\Omega_p = [0,\ \omega_p], \quad \omega_p < \pi, \tag{6.6}$$

then the design problem is to determine the parameter vector Φ such that for each value of μ within $0 \le \mu < 1$, the amplitude response $|H(\Phi, e^{j\omega}, \mu)|$, as given by Eq. (6.4b), approximates unity on Ω_p and the phase delay $\tau_p(\Phi, \omega, \mu)$, as given by Eq. (6.4c), approximates $N/2 - 1 + \mu$. We state the following optimization problem:

Optimization Problem: Given L, N, Ω_p, and ϵ, find the adjustable parameter vector Φ to minimize

$$\delta_p = \max_{0\le\mu<1}\left[\max_{\omega\in\Omega_p} |\tau_p(\Phi,\omega,\mu) - (N/2 - 1 + \mu)|\right] \tag{6.7a}$$

subject to

$$\delta_a = \max_{0\le\mu<1}\left[\max_{\omega\in\Omega_p} ||H(\Phi, e^{j\omega}, \mu)| - 1|\right] \le \epsilon. \tag{6.7b}$$

6.2.2 Lower Limit for the Amplitude Distortion

When optimizing the overall system, the following fact should be taken into consideration. Given N and Ω_p, there exists the lower achievable value for δ_a, denoted by δ_{lower}, independent of the value of L. This occurs for $\mu = 1/2$. In this case,

$$h(n, \Phi, 1/2) = h(N - 1 - n, \Phi, 1/2) \tag{6.8}$$

for $n = 0, 1, \ldots, N/2 - 1$, that is, $h(n, \Phi, 1/2)$ for $n = 0, 1, \ldots, N-1$ is the impulse response of a linear-phase filter which possesses an even symmetry. δ_{lower} can be determined by designing a linear-phase FIR filter with impulse response $e(n)$ of even length N using the Remez multiple exchange algorithm. Since the filter length is even, the corresponding amplitude response has a single zero at $\omega = \pi$. The desired impulse-response coefficients can be simply determined by using the desired and weight functions equal to unity on Ω_p. The corresponding $h(n, \Phi, 1/2)$'s are given for the worst-case situation by

$$h(n, \Phi, 1/2) = h(N - 1 - n, \Phi, 1/2) = e(n) \tag{6.9}$$

for $n = 0, 1, \ldots, N/2 - 1$.

6.3 Optimization Procedure

This subsection describes an algorithm for finding the optimum solution to the problem stated in the previous subsection. In addition, it is shown how good initial values for the filter parameters can be determined in order to guarantee the convergence of the algorithm to the optimum solution.

6.3.1 Optimization Algorithm

In order to solve the optimization problem stated in the previous subsection we discretize the passband region into the frequency points $\omega_i \in [0,\ \omega_p]$, $i = 1, 2, \ldots, I = 20N$ and the range $0 \le \mu \le 1/2$ [10] into the points $\mu_j \in [0,\ 1/2]$, $j = 1, 2, \ldots, J = 100$. The resulting discrete minimax problem is to find the adjustable parameter vector Φ to minimize

$$\delta_p = \max_{1\le i\le I, 1\le j\le J} f_{i,j}(\Phi) \tag{6.10a}$$

[10] Since for μ and $1 - \mu$ the amplitude and phase delay distortions are the same, only the values of μ in the range $[0,\ 1/2]$ have to be considered.

TABLE IV
MAXIMUM WORST-CASE PHASE DELAY DEVIATIONS IN EXAMPLE 1 FOR SOME VALUES OF N AND L

N	L	δ_p
8	2	0.07809
8	3	0.00402
8	4	0.00342
8	5	0.00324
10	2	0.03825
10	3	0.00179
10	4	0.00094

subject to

$$c_{i,j}(\Phi) \le 0 \quad \text{for } i = 1, 2, \ldots, I \text{ and } j = 1, 2, \ldots, J, \tag{6.10b}$$

where

$$f_{i,j}(\Phi) = |\tau_p(\Phi, \omega_i, \mu_j) - (N/2 - 1 + \mu_j)| \tag{6.10c}$$

and

$$c_{i,j}(\Phi) = ||H(\Phi, e^{j\omega_i}, \mu_j)| - 1| - \epsilon. \tag{6.10d}$$

Again, the Dutta-Vidyasagar algorithm is exploited for solving this problem.

6.3.2 Initial Parameters for Optimization

The convergence of the above algorithm to the optimum solution implies rather good initial values for the unknown $g_l(n)$'s. For $L = 2$, good initial values are given by

$$g_0(n) = e(n),\ g_1(n) = \alpha,\ g_2(n) = \alpha - e(n), \tag{6.11}$$

where $\alpha = 0$ for $n = 0, 1, \ldots, N/2 - 2$ and $\alpha = 1/2$ for $n = N/2 - 1$. With these values, $h(n, \Phi, 0)$ is equal to unity for $n = N/2 - 1$ and equal to zero at the remaining values of n. $h(n, \Phi, 1)$ is equal to unity for $n = N/2$ and equal to zero at the remaining values of n. Furthermore, $h(n, \Phi, 1/2) = e(n)$.

For $L > 2$, the optimum solution can be obtained conveniently by first optimizing the filter for $L = 2$. The second step is to optimize the filter for $L = 3$. A good starting point filter is obtained by selecting $g_L(n) = 0$ for $n = 0, 1, \ldots, N/2 - 1$ and using for the other unknowns the optimized values obtained for $L = 2$. Then, the optimization is performed in the same manner for $L = 4$. After that, the process is repeated by gradually increasing L by one until reaching the desired value for it.

6.4 Examples

This subsection illustrates, by means of examples, the flexibility and effectiveness of the proposed optimization scheme. It also gives guidelines on how to select N and L in a proper manner.

6.4.1 Example 1

It is required that $\Omega_p = [0,\ 0.75\pi]$, $\epsilon = 0.025$, and $\delta_p \le 0.01$. The two lowest even lengths for a linear-phase FIR filter to give an amplitude response with maximum deviation from unity on Ω_p being less than ϵ are $N = 8$ and $N = 10$. The corresponding deviations are 0.0235 and 0.0095.

Table IV gives for $N = 8$ and $N = 10$ the optimized values of δ_p for some values L. $N = 8$ and $L = 3$ gives an acceptable worst-case phase delay deviation. The amplitude and phase delay

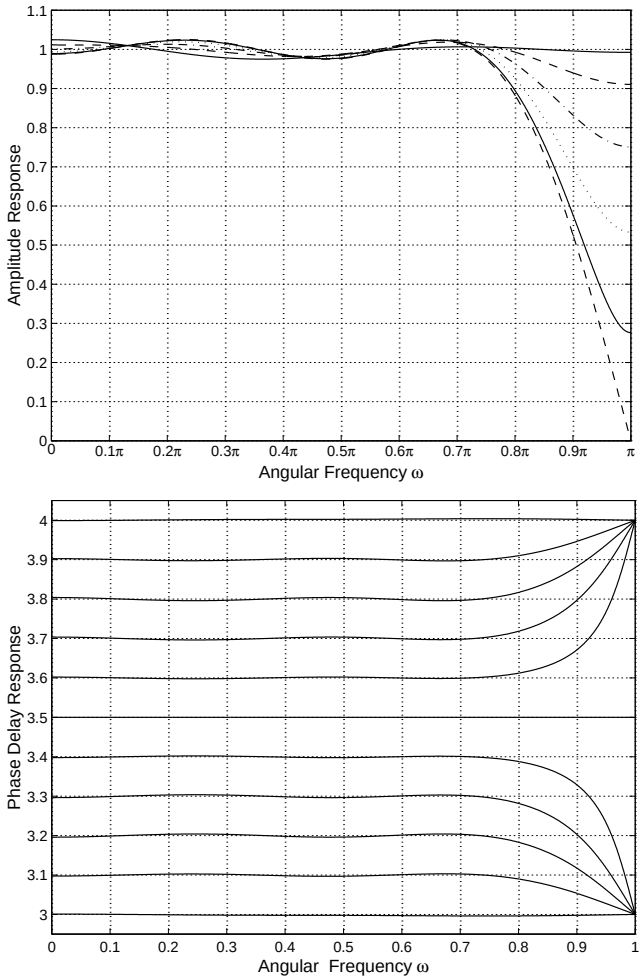

Fig. 6.2. Amplitude and phase delay responses for a filter of Example 1 ($N = 8$ and $L = 3$) for $\mu = 0,\ 0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9,\ 1.0$. In the case of the amplitude response, the dashed line with a lower value at $\omega = \pi$; solid line a lower value at $\omega = \pi$; dotted line; dot-dashed line; dashed line with a higher value $\omega = \pi$; and solid line with a higher value $\omega = \pi$ give the responses for $\mu = 0.5$; for both $\mu = 0.4$ and $\mu = 0.6$; for both $\mu = 0.3$ and $\mu = 0.7$; for both $\mu = 0.2$ and $\mu = 0.8$; for both $\mu = 0.1$ and $\mu = 0.9$; and for both $\mu = 0$ and $\mu = 1.0$; respectively.

responses of this Farrow structure are depicted in Fig. 6.2 for $\mu = 0,\ 0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9,\ 1.0$.

The following observations can be made based on the results of Table IV. For $N = 8$, δ_p decreases very slowly when L is increased. This is due to the fact that the lowest achievable amplitude deviation of 0.0235 is very close to $\epsilon = 0.025$. For $N = 10$, in turn, δ_p decreases very fast when L is increased. This shows that if the given criteria are desired to met by a small value of L, then it is advisable to select N to be the smallest integer for which the lowest achievable amplitude deviation is not very close to the given value of ϵ. Another observation from Table IV is that L should be larger than 2 in order to provide a small value for δ_p.

6.4.2 Example 2

It is required that $\Omega_p = [0,\ 0.9\pi]$, $\epsilon = 0.01$, and $\delta_p \leq 0.001$. In this case, the lowest achievable amplitude deviation for $N = 26$ is 0.0075 that is not very close to $\epsilon = 0.01$. Therefore, $N = 26$ can be used. The given criteria are met by $L = 4$. The amplitude and phase delay responses for $\mu = 0,\ 0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9,\ 1.0$ are depicted in Fig. 6.3 for this design.

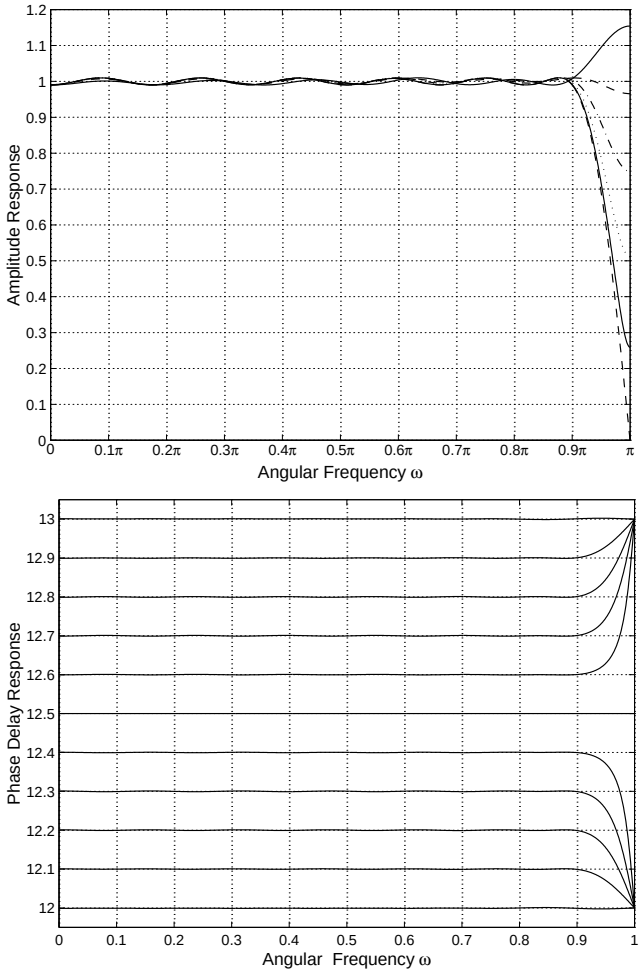

Fig. 6.3. Amplitude and phase delay responses for a filter of Example 2 for $\mu = 0,\ 0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5,\ 0.6,\ 0.7,\ 0.8,\ 0.9,\ 1.0$. In the case of the amplitude responses, the same line types as in Fig. 6.2 are used.

7 OPTIMIZATION OF PIPELINED RECURSIVE DIGITAL FILTERS

In general, recursive digital filters require much lower orders to achieve the given magnitude specifications than their non-recursive counterparts, especially in cases requiring narrow transition bands [48, 49]. However, the maximum achievable sampling frequency of the recursive filters is limited due their feedback connections. The maximal sampling frequency of a recursive filter is determined by the ratio between the number of delay elements and the latency of the arithmetic operations in the feedback loops [50]. Consequently, there exists two methods for increasing the maximal sample frequency. The first one is to increase the number of delay elements while the second one is to decrease the operation latency. One approach to obtain recursive filters with several delay elements in their feedback loop is to use transformation techniques [51–58]. In these techniques, the original transfer functions is transformed into a new transfer function by adding several cancelling zero-pole pairs. The number of cancelling zero-pole pairs depends upon the transformation technique employed.

This application shows how to optimize, with the aid of the two-step procedure described in Section 3, the magnitude response of the pipelined recursive filters obtained using the transformation al-

gorithms [53, 59–74]. The main advantage of the optimization approach lies in the fact that it allows many degrees of freedom in the design of these filters. The design margin introduced by coefficient transformation can be allocated, e.g., for maximizing the stopband attenuation. Alternatively, it is possible to minimize the radius of the outermost pole resulting in a reduced roundoff noise due to multiplication errors. In addition, this approach does not rely on the pole-zero cancellation. In finite wordlength implementations, the inexact pole-zero cancellation will lead to errors in the realized spectrum and a time-variant behavior [75–77].

7.1 Transformation Techniques

High-speed recursive filters can be obtained through transformation techniques. These techniques can be broadly classified into two categories, namely, *clustered look-ahead* (CLA) [59–61, 66, 72] and *scattered look-ahead* (SLA) [59, 75, 78] transformations. The former technique will generally yield to a lower computational complexity. However, it is required to use direct-form structures, that generally share some poor finite-wordlength properties, such as a reasonably high coefficient sensitivity and a high roundoff noise. In addition, the inexact pole-zero cancellation may increase the coefficient sensitivity even further. In the sequel, we concentrate only on the optimization of the recursive filters obtained using CLA transformations. However, the proposed algorithm can easily be modified for optimizing the recursive filters obtained using SLA transformations as well.

In general, CLA transformations techniques involve the augmentation of an unpipelined filter described by

$$\widehat{H}(z) = \frac{A(z)}{B(z)} = \frac{\sum_{k=0}^{N} a_k z^{-k}}{1 + \sum_{k=1}^{N} b_k z^{-k}} \tag{7.1}$$

into the form

$$H(z) = \frac{C(z)}{D(z)} = \frac{A(z)Q(z)}{B(z)Q(z)} = \frac{\sum_{k=0}^{N+M} c_k z^{-k}}{1 + \sum_{k=M}^{N+M} d_k z^{-k}}, \tag{7.2}$$

where M is the number of pipeline stages in the feedback loop. In other words, by adding cancelling zero-pole pairs that are roots of the augmentation polynomial, $Q(z)$, the transfer function of Eq. (7.1) can be converted into the pipelined filter of Eq. (7.2) with M-stages of pipelining. Since the first $M-1$ coefficients are zero-valued, the speedup factor of the pipelined filter is roughly equal to M.

7.2 Optimization Problem

This subsection states the optimization problem for designing pipelined recursive filters in such a way that the stopband attenuation is minimized. Efficient algorithms are then described for solving this problem.

7.2.1 Statement of the problem

For optimization purposes it is attractive to rewrite the transfer function of Eq. (7.2) as a cascade of first- and second-order sections as follows:

$$H(z) = k_0 \prod_{k=1}^{K_0} \frac{1 + a_{0k} z^{-1}}{1 + b_{0k} z^{-1}} \prod_{k=1}^{K_1} \frac{1 + a_{1k} z^{-1} + a_{2k} z^{-2}}{1 + b_{1k} z^{-1} + b_{2k} z^{-2}}, \tag{7.3}$$

where K_0 and K_1 are the number of first- and second order sections, respectively, and k_0 is a multiplicative constant for adjusting the desired level for the passband.

If $C(z)$ possesses K_0 real zeros at $z = r_k^{(\mathrm{zr})}$ for $k = 1, 2, \ldots, K_0$ and K_1 complex-conjugate zero pairs at $r_k^{(\mathrm{zc})} \exp(\pm j\theta_k^{(\mathrm{zc})})$ for $k = 1, 2, \ldots, K_1$ and $D(z)$ possesses K_0 real poles at $z = r_k^{(\mathrm{pr})}$ for $k = 1, 2, \ldots, K_0$ and K_1 complex-conjugate poles at $r_k^{(\mathrm{pc})} \exp(\pm j\theta_k^{(\mathrm{pc})})$ for $k = 1, 2, \ldots, K_1$, then

$$a_{0k} = r_k^{(\mathrm{zr})} \qquad \text{for } k = 1, 2, \ldots, K_0, \tag{7.4a}$$
$$b_{0k} = r_k^{(\mathrm{pr})} \qquad \text{for } k = 1, 2, \ldots, K_0, \tag{7.4b}$$
$$a_{1k} = -2r_k^{(\mathrm{zc})} \cos\theta_k^{(\mathrm{zc})} \qquad \text{for } k = 1, 2, \ldots, K_1, \tag{7.4c}$$
$$a_{2k} = (r_k^{(\mathrm{zc})})^2 \qquad \text{for } k = 1, 2, \ldots, K_1, \tag{7.4d}$$
$$b_{1k} = -2r_k^{(\mathrm{pc})} \cos\theta_k^{(\mathrm{pc})} \qquad \text{for } k = 1, 2, \ldots, K_1, \tag{7.4e}$$

and

$$b_{2k} = (r_k^{(\mathrm{pc})})^2 \qquad \text{for } k = 1, 2, \ldots, K_1. \tag{7.4f}$$

The squared-magnitude response of $H(z)$ can be derived by using substitution $z = e^{j\omega}$ in $H(z)H(1/z)$. After some manipulations, this response can be expressed as

$$|H(\Phi, e^{j\omega})|^2 = k_0^2 \frac{|F_1(\omega)|^2 |F_2(\omega)|^2}{|G_1(\omega)|^2 |G_2(\omega)|^2}, \tag{7.5a}$$

where

$$|F_1(\omega)|^2 = \prod_{k=1}^{K_0} \left[1 + 2a_{0k}\cos\omega + a_{0k}^2\right], \tag{7.5b}$$

$$|F_2(\omega)|^2 = \prod_{k=1}^{K_1} \Big[\big(\cos\omega(1 + a_{2k}) + a_{1k}\big)^2 + \big(\sin\omega(1 - a_{2k})\big)^2\Big], \tag{7.5c}$$

$$|G_1(\omega)|^2 = \prod_{k=1}^{K_0} \left[1 + 2b_{0k}\cos\omega + b_{0k}^2\right], \tag{7.5d}$$

$$|G_2(\omega)|^2 = \prod_{k=1}^{K_1} \Big[\big(\cos\omega(1 + b_{2k}) + b_{1k}\big)^2 + \big(\sin\omega(1 - b_{2k})\big)^2\Big], \tag{7.5e}$$

and Φ is the adjustable parameter vector containing the adjustable parameters of the filter

$$\begin{aligned}\Phi = \big[&k_0, a_{0k}, \ldots, a_{0K_0}, a_{1k}, \ldots, a_{1K_1}, a_{2k}, \ldots, a_{2K_1},\\ &b_{0k}, \ldots, b_{0K_0}, b_{1k}, \ldots, b_{1K_1}, b_{2k}, \ldots, b_{2K_1}\big].\end{aligned} \tag{7.6}$$

The amplitude specifications for the filter are stated as follows:

$$1 \le |H(\Phi, e^{j\omega})| \le 1 + \delta_p \quad \text{for } \omega \in [0, \omega_p] \tag{7.7a}$$
$$|H(\Phi, e^{j\omega})| \le \delta_s \qquad \text{for } \omega \in [\omega_s, \pi], \tag{7.7b}$$

where

$$|H(\Phi, e^{j\omega})| = \sqrt{|H(\Phi, e^{j\omega})|^2}. \tag{7.8}$$

In order to the ensure the stability all the poles must lie inside the unit circle. This, implies that

$$|b_{0k}| \le 1 \quad \text{for } k = 1, 2, \ldots, K_0 \tag{7.9a}$$

and b_{1k} and b_{2k} for $k = 1, 2, \ldots, K_1$ lie in a triangular region defined by

$$b_{1k} - b_{2k} \le 1, \tag{7.9b}$$
$$-b_{1k} - b_{2k} \le 1, \tag{7.9c}$$

and

$$b_{2k} \leq 1. \tag{7.9d}$$

Note that these constraints are linear with respect to the design parameters.

The denominator polynomial $D(z)$ can be obtained by convolving the denominator polynomials of the first- and second-order sections as follows:

$$\begin{aligned} D(z) &= 1 + d_1 z^{-1} + \cdots + d_{K_0+2K_1+2} z^{K_0+2K_1+2} \\ &= (1 + b_{01} z^{-1}) * \cdots * (1 + b_{0K_0} z^{-1}) \\ &\quad * (1 + b_{11} z^{-1} + b_{21} z^{-2}) * \cdots \\ &\quad * (1 + b_{1K_1} z^{-1} + b_{2K_1} z^{-2}). \end{aligned} \tag{7.10}$$

The optimization problem under consideration is the following.

Optimization problem: Given ω_p, ω_s, and δ_p, as well as M, the number of pipelining stages in the feedback loop, find the adjustable parameter vector Φ, as given by Eq. (7.6), to minimize on $[\omega_s, \pi]$ the peak absolute value of $H(\Phi, \omega)$, as given by

$$\delta_s = \max_{\omega_s \leq \omega \leq \pi} |H(\Phi, e^{j\omega})| \tag{7.11a}$$

subject to conditions of Eqs. (7.7a), (7.9), and

$$d_k = 0 \quad \text{for } k = 1, 2, \ldots, M-1. \tag{7.11b}$$

7.2.2 Optimization Algorithm

To solve this problem, we discretize the passband and stopband region into the frequency points $\omega_i \in [0, \omega_p]$ for $i = 1, 2, \ldots, L_p$ and $\omega_i \in [\omega_s, \pi]$ for $i = L_p + 1, L_p + 2, \ldots, L_p + L_s$. The resulting discrete optimization problem is to find Φ to minimize

$$\widehat{\delta}_s = \max_{L_p+1 \leq i \leq L_p+L_s} f_i(\Phi), \tag{7.12a}$$

subject to conditions

$$h_m(\Phi) \leq 0 \quad \text{for } m = 1, 2, \ldots, L_p, \tag{7.12b}$$

$$g_l(\Phi) = 0 \quad \text{for } l = 1, 2, \ldots, M-1, \tag{7.12c}$$

and

$$e_k(\Phi) = 0 \quad \text{for } k = 1, 2, \ldots, 2K_0 + 3K_1, \tag{7.12d}$$

where

$$f_i(\Phi) = \left| H(\Phi, e^{j\omega_i}) \right|, \tag{7.12e}$$

$$h_m(\Phi) = \left| \left| H(\Phi, e^{j\omega_i}) \right| - \left(1 + \frac{\delta_p}{2}\right) \right| - \frac{\delta_p}{2}, \tag{7.12f}$$

$$g_l(\Phi) = d_k, \tag{7.12g}$$

and

$$e_k(\Phi) = \begin{cases} b_{0k} - 1 & \text{for } k = 1, \ldots, K_0 \\ -b_{0k} - 1 & \text{for } k = K_0 + 1, \ldots, 2K_0 \\ b_{1k} - b_{2k} - 1 & \text{for } k = 2K_0 + 1, \ldots, 2K_0 + K_1 \\ -b_{1k} - b_{2k} - 1 & \text{for } k = 2K_0 + K_1 + 1, \ldots, \\ & \qquad 2K_0 + 2K_1 \\ -b_{2k} - 1 & \text{for } k = 2K_0 + 2K_1 + 1, \ldots, \\ & \qquad 2K_0 + 3K_1. \end{cases} \tag{7.12h}$$

The above-mentioned optimization problem can be solved using any of the three alternatives considered in Section 3. However, the algorithm based on the use of SQP methods is very efficient due to linearity of the constraints of Eq. (7.9). The convergence of the above algorithm to the optimum solution implies rather good initial values for the unknowns. A good initial starting-point filter for further optimization can be generated by using the algorithm proposed in [66].

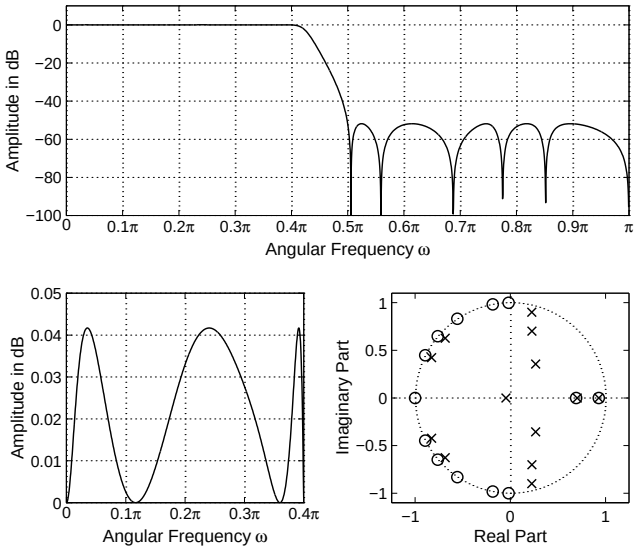

Fig. 7.1. Some responses for the optimized pipelined recursive filter.

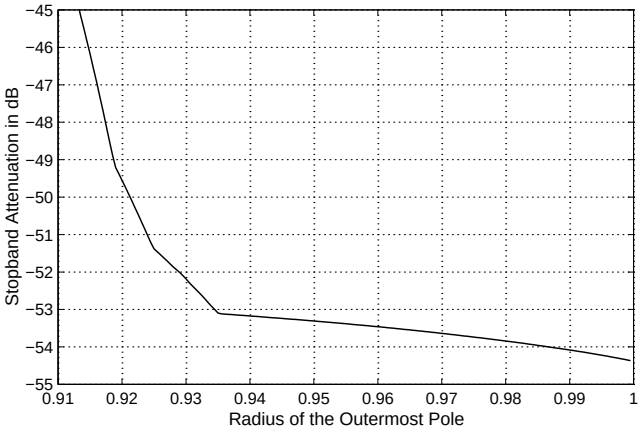

Fig. 7.2. Stopband attenuation as a function of the radius of the outermost pole.

7.3 Numerical Example

In order to illustrate the applicability of the proposed optimization algorithm, we consider the optimization of a sixth-order elliptic low-pass filter with five-stages of pipelining used by Lim *et al.* [70] to introduce their minimum order augmentation technique. The passband ripple and stopband attenuation of the prototype filter are $A_p = 0.0419$ dB and $A_s = 47.6$ dB, respectively. The passband edge is at $\omega_p = 0.4\pi$ and the stopband edge at $\omega_s = 0.5\pi$.

The amplitude response, as well as the zero-pole plot for the optimized filter is shown in Fig. 7.1. In this case, the optimization has been performed in such a manner that the stopband attenuation has been maximized, while the radius of the outermost pole is restricted to be less or equal to $0.927\,93$, the largest magnitude of the poles of the prototype filter. The stopband attenuation for the optimized filter is approximately 51.829 dB. The values for the optimized coefficients are given in the Table V. Figure 7.2 shows the stopband attenuation as a function of the radius of the outermost pole in the case when the passband ripple is restricted to be less or equal to 0.0419 dB.

TABLE V
OPTIMIZED INFINITE-PRECISION COEFFICIENTS FOR THE PIPELINED RECURSIVE FILTER

$C(z)$	$D(z)$
$c_0 = 0.029\,424\,34$	$d_0 = 1$
$c_1 = 0.123\,882\,74$	$d_1 = 0$
$c_2 = 0.272\,120\,96$	$d_2 = 0$
$c_3 = 0.374\,994\,15$	$d_3 = 0$
$c_4 = 0.313\,462\,63$	$d_4 = 0$
$c_5 = 0.063\,186\,02$	$d_5 = -0.945\,573\,14$
$c_6 = -0.240\,862\,90$	$d_6 = 0.135\,921\,17$
$c_7 = -0.403\,317\,38$	$d_7 = 0.008\,794\,80$
$c_8 = -0.339\,392\,32$	$d_8 = -0.026\,715\,29$
$c_9 = -0.139\,651\,81$	$d_9 = -0.101\,912\,70$
$c_{10} = 0.033\,067\,31$	$d_{10} = 0.272\,599\,89$
$c_{11} = 0.092\,213\,87$	$d_{11} = -0.123\,288\,91$
$c_{12} = 0.062\,358\,02$	$d_{12} = 0.038\,495\,87$
$c_{13} = 0.018\,870\,16$	$d_{13} = 0.002\,034\,03$

8 DESIGN OF LATTICE WAVE DIGITAL FILTERS WITH SHORT COEFFICIENT WORDLENGTH

Among the best structures for implementing recursive digital filters are lattice wave digital (LWD) filters (parallel connections of two all-pass filters). They are characterized by many attractive properties, such as a reasonably low coefficient sensitivity, a low roundoff noise level, and the absence of parasitic oscillations. The main drawback is that if the stopband attenuation is very high, then many bits are required for the coefficient representations.

In order to get around this problem, a structure consisting of a cascade of LWD filters has been introduced [79, 80]. The main advantage of this structure, compared with the direct LWD filter, is that the poles of the new structure are further away from the unit circle. Consequently, the number of bits required for both the data and coefficient representations are significantly reduced. The price paid for these reductions is a slight increase in the overall filter order. By properly selecting the number of LWD filters and their orders and optimizing them, their coefficients are implementable by using a few powers of two. Filters of this kind are very attractive in VLSI implementations, where a general multiplier is very costly.

This application shows how the coefficients of the various classes of LWD filters can be conveniently quantized utilizing the two-step approach of Section 3. The filters under consideration consist of conventional LWD filters, cascades of low-order LWD filters providing a very low sensitivity and roundoff noise, and LWD filters with an approximately linear phase in the passband considered in Section 5.

8.1 Proposed Filter Class

Let the transfer function of a recursive digital filter be given by

$$H(z) = \prod_{k=1}^{K} H_k(z), \tag{8.1a}$$

where

$$H_k(z) = \alpha_k A_k(z) + \beta_k B_k(z). \tag{8.1b}$$

Here, the $A_k(z)$'s and $B_k(z)$'s for $k = 1, 2, \ldots, K$ are stable all-pass filters of orders M_k and N_k, respectively. An implementation of the above transfer function is depicted in Fig. 8.1. This contribution concentrates on synthesizing low-pass filters. In this case, $M_k = N_k - 1$ or $M_k = N_k + 1$, so that $M_k + N_k$, the overall order of $H_k(z)$, is odd. If the $A_k(z)$'s and $B_k(z)$'s are implemented

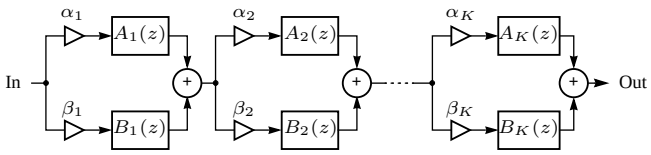

Fig. 8.1. Proposed recursive filter structure. The $A_k(z)$'s and $B_k(z)$'s are stable all-pass filters.

as a cascade of first- and second-order wave digital all-pass structures and M_k and N_k are assumed to be odd and even, respectively, then the $A_k(z)$'s and $B_k(z)$'s are expressible in terms of the adaptor coefficients as follows (see, e.g., [41] or [81]):

$$A_k(z) = \frac{-\gamma_0^{(k)} + z^{-1}}{1 - \gamma_0^{(k)} z^{-1}} \cdot \prod_{l=1}^{m_k} \frac{-\gamma_{2l-1}^{(k)} + \gamma_{2l}^{(k)}(\gamma_{2l-1}^{(k)} - 1)z^{-1} + z^{-2}}{1 + \gamma_{2l}^{(k)}(\gamma_{2l-1}^{(k)} - 1)z^{-1} - \gamma_{2l-1}^{(k)} z^{-2}} \tag{8.2a}$$

and

$$B_k(z) = \prod_{l=1}^{n_k} \frac{-\widehat{\gamma}_{2l-1}^{(k)} + \widehat{\gamma}_{2l}^{(k)}(\widehat{\gamma}_{2l-1}^{(k)} - 1)z^{-1} + z^{-2}}{1 + \widehat{\gamma}_{2l}^{(k)}(\widehat{\gamma}_{2l-1}^{(k)} - 1)z^{-1} - \widehat{\gamma}_{2l-1}^{(k)} z^{-2}}, \tag{8.2b}$$

where

$$m_k = (M_k - 1)/2 \quad \text{and} \quad n_k = N_k/2. \tag{8.2c}$$

If $A_k(z)$ possesses a real pole at $z = r_0^{(k)}$ and m_k complex-conjugate pole pairs at $z = r_l^{(k)} \exp(\pm j\theta_l^{(k)})$ for $l = 1, 2, \ldots, m_k$ and $B_k(z)$ possesses n_k complex-conjugate pole pairs at $z = \widehat{r}_l^{(k)} \exp(\pm j\widehat{\theta}_l^{(k)})$ for $l = 1, 2, \ldots, n_k$, then

$$\gamma_0^{(k)} = r_0^{(k)}, \tag{8.3a}$$

$$\gamma_{2l-1}^{(k)} = -(r_l^{(k)})^2 \quad \text{for } l = 1, 2, \ldots, m_k, \tag{8.3b}$$

$$\gamma_{2l}^{(k)} = \frac{2r_l^{(k)} \cos(\theta_l^{(k)})}{1 + (r_l^{(k)})^2} \quad \text{for } l = 1, 2, \ldots, m_k, \tag{8.3c}$$

$$\widehat{\gamma}_{2l-1}^{(k)} = -(\widehat{r}_l^{(k)})^2 \quad \text{for } l = 1, 2, \ldots, n_k, \tag{8.3d}$$

and

$$\widehat{\gamma}_{2l}^{(k)} = \frac{2\widehat{r}_l^{(k)} \cos(\widehat{\theta}_l^{(k)})}{1 + (\widehat{r}_l^{(k)})^2} \quad \text{for } l = 1, 2, \ldots, n_k. \tag{8.3e}$$

8.2 Optimization Problem

Before stating the optimization problem, we denote the transfer function of the filter by $H(\Phi, z)$, where Φ is the adjustable parameter vector

$$\begin{aligned}\Phi = [&\alpha_1, \beta_1, r_0^{(1)}, \ldots, r_{m_1}^{(1)}, \theta_1^{(1)}, \ldots, \theta_{m_1}^{(1)}, \widehat{r}_1^{(1)}, \ldots, \\ &\widehat{r}_{n_1}^{(1)}, \widehat{\theta}_1^{(1)}, \ldots, \widehat{\theta}_{n_1}^{(1)}, \ldots, \alpha_K, \beta_K, r_0^{(K)}, \ldots, r_{m_K}^{(K)}, \\ &\theta_1^{(K)}, \ldots, \theta_{m_K}^{(K)}, \widehat{r}_1^{(K)}, \ldots, \widehat{r}_{n_K}^{(K)}, \widehat{\theta}_1^{(K)}, \ldots, \widehat{\theta}_{n_K}^{(K)}].\end{aligned} \tag{8.4}$$

The amplitude specifications for the filter are stated as follows:

$$1 - \delta_p \le |H(\Phi, e^{j\omega})| \le 1 \quad \text{for } \omega \in [0, \omega_p] \tag{8.5a}$$

$$|H(\Phi, e^{j\omega})| \le \delta_s \quad \text{for } \omega \in [\omega_s, \pi]. \tag{8.5b}$$

Alternatively, these criteria are expressible as

$$|E(\Phi,\omega)| \le 1 \quad \text{for } \omega \in [0,\omega_p] \cup [\omega_s,\pi] \tag{8.6a}$$
$$E(\Phi,\omega) \le 0 \quad \text{for } \omega \in [0,\omega_p], \tag{8.6b}$$

where

$$E(\Phi,\omega) = W(\omega)[|H(\Phi,e^{j\omega})| - D(\omega)] \tag{8.6c}$$

with

$$D(\omega) = \begin{cases} 1 & \text{for } \omega \in [0,\omega_p] \\ 0 & \text{for } \omega \in [\omega_s,\pi], \end{cases} \tag{8.6d}$$

and

$$W(\omega) = \begin{cases} 1/\delta_p & \text{for } \omega \in [0,\omega_p] \\ 1/\delta_s & \text{for } \omega \in [\omega_s,\pi]. \end{cases} \tag{8.6e}$$

This application concentrates on the coefficient quantization in fixed-point arithmetic. In many implementations, it is attractive to carry out the multiplication of a data sample by a filter coefficient value using a sequence of shifts and adds. For such a purpose, it is desirable to express the coefficient values in the form

$$\sum_{r=1}^{R} a_r 2^{-P_r}, \tag{8.7}$$

where each of the a_r's is either 1 or -1 and the P_r's are positive integers in the increasing order. The target is to find all the coefficient values included in Φ, as given by Eq. (8.4), in such a way that: 1) R, the number of powers of two, is made as small as possible and 2) P_R, the number of fractional bits, is made as small as possible.

The optimization problem under considerations is the following.

Optimization Problem: Find K, the number of subfilters, the M_k's and N_k's, as well as the adjustable parameter vector Φ, as given by Eq. (8.4), in such a way that we have the following.

1. $H(\Phi,z)$ meets the criteria given by Eq. (8.5) or (8.6).
2. The coefficients included in Φ are quantized to achieve the above-mentioned target for their representations.

8.3 Filter Optimization

The solution to the stated optimization problem can be found in two steps. In the first step, a filter with infinite-precision coefficients is determined in such a way that it exceeds the given amplitude criteria to provide some tolerance for the coefficient quantization. The second step involves finding a filter meeting the given criteria with simple coefficient representation forms.

8.3.1 Optimization of Infinite-Precision Filters

The problem is to find the adjustable parameter vector Φ to minimize on $[0,\omega_p] \cup [\omega_s,\pi]$ the peak absolute value of $E(\Phi,\omega)$, as given by Eqs. (8.6c)–(8.6e), subject to the condition

$$E(\Phi,\omega) \le 0 \quad \text{for } \omega \in [0,\omega_p]. \tag{8.8}$$

To solve this problem, we discretize the passband and stopband regions into the frequency points $\omega_i \in [0,\omega_p]$ for $i = 1,2,\ldots,L_p$ and $\omega_i \in [\omega_s,\pi]$ for $i = L_p+1, L_p+2, \ldots, L_p+L_s$. The resulting discrete minimax problem is to find Φ to minimize

$$\epsilon = \max_{1 \le i \le L_p+L_s} |E(\Phi,\omega_i)| \tag{8.9a}$$

subject to

$$E(\Phi,\omega_i) \le 0 \quad \text{for } i = 1,2,\ldots,L_p. \tag{8.9b}$$

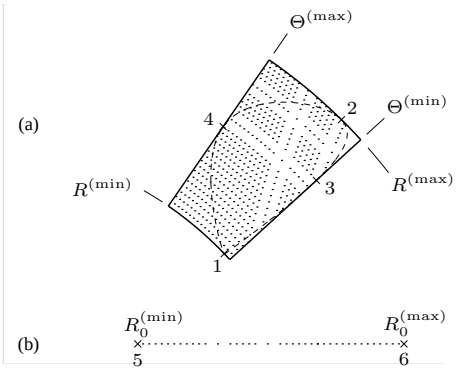

Fig. 8.2. Typical search spaces for the poles when three powers of two with 7 fractional bits ($R = 3$ and $P_R = 7$) are used for the adaptor coefficients. (a) Upper-half-plane pole for the complex-conjugate pole pair. (b) Real pole.

In order to meet the criteria of Eq. (8.5) or (8.6), K as well as the M_k's and N_k's for $k = 1,2,\ldots,K$, have to be selected such that the minimized ϵ becomes less than or equal to unity.

The above problem can be solved in a straightforward manner by using the Dutta-Vidyasagar algorithm described in Section 3.2.1. For this nonlinear optimization algorithm, the convergence to the global optimum cannot be assured. Hence, a good guess for the initial filter has an extensive effect on the convergence of the algorithm to the optimal solution. It has turned out that, in many cases, it is beneficial to select all the α_k's and β_k's to be equal to $1/2$, as for the conventional LWD filters. Furthermore, it is in most cases advantageous to select all the $A_k(z)$'s and the $B_k(z)$'s to be of the same order, respectively. Only in cases where K, the number stages, is large and the required passband ripple is relatively low, it is beneficial to give other values for α_k and β_k only in one section. For the case where $\alpha_k = \beta_k = 1/2$ for $k = 1,2,\ldots,K$, the starting point filter for further optimization can be determined by using several identical copies of the same subfilter. The passband and stopband ripples for this subfilter should be approximately equal to δ_p/K and $\sqrt[K]{\delta_s}$, respectively.[11] For the case where for one section α_k and β_k have values different from $1/2$, a good initial filter for further optimization can be achieved by using the procedure described in Appendix B in [80].

8.3.2 Optimization of Finite-Precision Filters

It has turned out that a very straightforward quantization scheme for the filter coefficients is obtained as follows in the case where all the α_k's and β_k's are equal to $1/2$. For each complex-conjugate pole pair, the largest and smallest values for both the radius and angle are determined in such a way that by reoptimizing the remaining pole parameter and the locations of the remaining poles, the overall criteria, as given by Eq. (8.5) or (8.6), can still be met. For each real pole, the smallest and largest values for the radius are found in a similar manner.

The above procedure gives for the upper-half-plane pole for each complex-conjugate pole pair $r_l^{(k)} \exp(\pm j\theta_l^{(k)})$ for $l = 1,2,\ldots,m_k$ and for $k = 1,2,\ldots,K$ and $\widehat{r}_l^{(k)} \exp(\pm j\widehat{\theta}_l^{(k)})$ for $l = 1,2,\ldots,n_k$ and for $k = 1,2,\ldots,K$ the region $R\exp(j\Theta)$

[11] There is clearly a tradeoff between the number of subfilters, K, and the order of the subfilter; the higher is the value of K, the lower is the order of the subfilter. However, since the subfilter order is restricted to be an odd integer, there are only few practical combinations for the subfilter order and K. It is not necessary for the subfilter being an odd order elliptic filter to exactly meet the ripple requirements. This is due to the fact that further optimization makes the subfilters different and simultaneously improves the overall filter performance.

where $R^{(\min)} \le R \le R^{(\max)}$ and $\Theta^{(\min)} \le \Theta \le \Theta^{(\max)}$, as illustrated in Fig. 8.2(a). The crosses numbered by $1, 2, 3$, and 4 correspond, respectively, to the points where the smallest radius $R^{(\min)}$, the largest radius $R^{(\max)}$, the smallest angle $\Theta^{(\min)}$, and the largest angle $\Theta^{(\max)}$ are reached. Inside this region, there is the feasible region given by dashed line in Fig. 8.2(a), where the pole can be located such that by relocating the remaining poles, the given overall criteria are still met by using infinite-precision arithmetic. For each real pole $r_0^{(k)}$ for $k = 1, 2, \ldots, K$, there is the corresponding region $R_0^{(\min)} \le R \le R_0^{(\max)}$ that is simultaneously the feasible region. In Fig. 8.2(b), the crosses numbered by 5 and 6 indicate $R_0^{(\min)}$ and $R_0^{(\max)}$, respectively.

The next step is to find in the above regions those pole locations which are achievable by implementing the adaptor coefficients in the form of Eq. (8.7) with the given R and the given largest value for P_R. The dots in Fig. 8.2 indicate these pole locations. Note that the distributions are very irregular due to the desired representation form. For the complex-conjugate pole pairs, the larger region is used since it can be found by applying the algorithm to be described later only four times. All what is still needed is to check whether there exists a combination of the discrete pole positions with which the given overall criteria are met. More details on how to effectively find the desired finite-precision filter have been described in [82] and [79].

The above-mentioned infinite-precision regions can be determined conveniently with the aid of the Dutta-Vidyasagar algorithm. In this case, there are $\sum_{k=1}^{K}(M_k + N_k)$ problems of the form: Find Φ to minimize ψ subject to

$$|E(\Phi, \omega_i)| - 1 \le 0 \quad \text{for } i = 1, 2, \ldots, L_p + L_s \tag{8.10a}$$

$$E(\Phi, \omega_i) \le 0 \quad \text{for } i = 1, 2, \ldots, L_p. \tag{8.10b}$$

For these problems, ψ is $r_l^{(k)}$, $1 - r_l^{(k)}$ for $l = 0, 1, \ldots, m_k$, $k = 1, 2, \ldots, K$; $\theta_l^{(k)}$, $\pi - \theta_l^{(k)}$ for $l = 1, 2, \ldots, m_k$, $k = 1, 2, \ldots, K$; and $\widehat{r}_l^{(k)}$, $1 - \widehat{r}_l^{(k)}$, $\widehat{\theta}_l^{(k)}$, $\pi - \widehat{\theta}_l^{(k)}$ for $l = 1, 2, \ldots, n_k$, $k = 1, 2, \ldots, K$, respectively.[12]

If for one section α_k and β_k are not equal to $1/2$, then, in addition to the poles or equivalently the adaptor coefficients, these parameters are included in the above quantization scheme.

The proposed quantization scheme provides significant advantages over those based on the use of simulated annealing or genetic algorithms. First of all, it is always guaranteed that the optimum solution can be found provided that it exists. Second, the computational workload to arrive at the optimum discrete-valued solution is in most cases significantly smaller than in the two above-mentioned algorithms.

8.4 Numerical Examples

This section shows, by means of examples, the efficiency and flexibility of the quantization scheme described in the previous section as well as the superiority of the cascaded filters over direct LWD filters in finite wordlength implementations. More examples can be found in [80, 82–84].

8.4.1 Example 1

It is desired to design a filter with the passband and stopband edges at $\omega_p = 0.1\pi$ and at $\omega_s = 0.2\pi$, respectively. The maximum allowable passband ripple and the required stopband attenuation are 0.5 dB ($\delta_p = 0.0559$) and 100 dB ($\delta_s = 10^{-5}$), respectively.

[12] In these problems, the optimization is performed, using special arrangements, in such a manner that the above-mentioned infinite-precision regions are not allowed to completely overlap, thus reducing the computational complexity of the overall quantization scheme.

TABLE VI
OPTIMIZED FINITE-PRECISION ADAPTOR COEFFICIENTS FOR THE DIRECT LWD FILTER IN EXAMPLE 1

$A(z)$	$B(z)$
$\gamma_0 = 1 - 2^{-3} + 2^{-6}$	
$\gamma_1 = -1 + 2^{-3} + 2^{-6} + 2^{-9}$	$\widehat{\gamma}_1 = -1 + 2^{-2} - 2^{-4} + 2^{-9}$
$\gamma_2 = 1 - 2^{-5}$	$\widehat{\gamma}_2 = 1 - 2^{-6} + 2^{-9}$
$\gamma_3 = -1 + 2^{-5} - 2^{-7} - 2^{-9}$	$\widehat{\gamma}_3 = -1 + 2^{-4} + 2^{-6}$
$\gamma_4 = 1 - 2^{-4} - 2^{-8}$	$\widehat{\gamma}_4 = 1 - 2^{-4} + 2^{-6} - 2^{-8}$

TABLE VII
OPTIMIZED FINITE-PRECISION ADAPTOR COEFFICIENTS FOR THE CASCADE OF FOUR LWD FILTERS IN EXAMPLE 1

$A(z)$	$B(z)$
$\gamma_0^{(1,2)} = 2^{-1} + 2^{-3}$	$\widehat{\gamma}_1^{(1,2)} = -1 + 2^{-2} - 2^{-5}$
	$\widehat{\gamma}_2^{(1,2)} = 1 - 2^{-3} + 2^{-5}$
$\gamma_0^{(3)} = 2^{-1} + 2^{-3} + 2^{-5}$	$\widehat{\gamma}_1^{(3)} = -1 + 2^{-2}$
	$\widehat{\gamma}_2^{(3)} = 1 - 2^{-3} + 2^{-5}$
$\gamma_0^{(4)} = 1 - 2^{-2} + 2^{-5}$	$\widehat{\gamma}_1^{(4)} = -1 + 2^{-2} - 2^{-4}$
	$\widehat{\gamma}_2^{(4)} = 1 - 2^{-4}$

The minimum order of a direct LWD filter to meet the given amplitude criteria is seven.[13] However, this filter meets just the given criteria. Therefore, to allow some tolerance for the coefficient quantization, the filter order has to be increased to nine. Using the quantization scheme described above, the given criteria are met by a filter with $K = 1$ and $\alpha_1 = \beta_1 = 1/2$. The optimized discrete-valued adaptor coefficients are given in Table VI. In this case, nine fractional bits are needed for the adaptor coefficients. Among the solutions satisfying the given amplitude specifications with nine fractional bits, the one with the smallest ϵ, as given by Eq. (8.10), has been selected.

For $K = 4$, the given criteria are met by $\alpha_k = \beta_k = 1/2$, $M_k = 1$, and $N_k = 2$ for $k = 1, 2, 3, 4$. Table VII gives the optimized finite-precision adaptor coefficients. In this case, all the coefficients can be represented as two or three powers of two and only five fractional bits are needed . A total of only 6 adders[14] are required to implement all the filter coefficients. Note that two sections are identical. The solution has been selected as for the $K = 1$ case.

Figure 8.3 shows for the $K = 1$ design the amplitude response. In addition, the passband details of the amplitude response are shown. The pole-zero plot for the overall design is depicted in Fig. 8.4. Similar characteristics for $K = 4$ are shown in Figs. 8.5 and 8.6, respectively.

The above cascade of four low-order LWD filter sections is very attractive for VLSI implementations, since no general multipliers are needed. The price paid for this is a slight increase in the overall filter order compared to the direct LWD filter. For $K = 4$, the order increases from nine to twelve.

[13] It is well known that the odd order elliptic low-pass filter is the most selective filter being implementable as a parallel connection of two all-pass filters (see, e.g., [4]).

[14] When the adaptors shown in Fig. 9 in [4] are used, the actual multiplier to be implemented is always positive and less than or equal to half. Therefore, in this case, the number of adders required for implementing the adaptor coefficients becomes smaller.

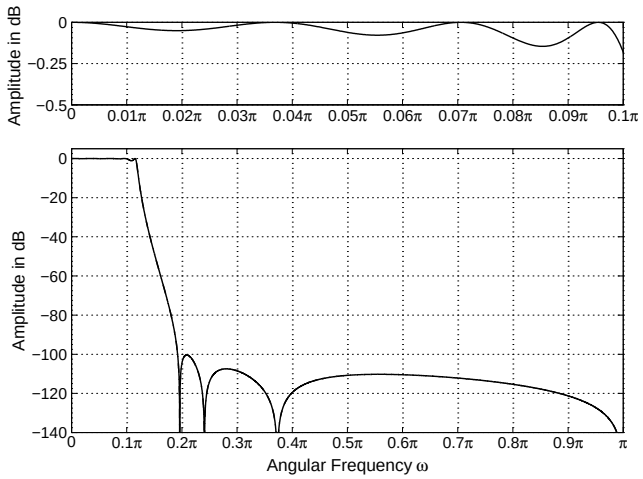

Fig. 8.3. Some amplitude responses for the optimized finite-precision direct LWD filter in Example 1.

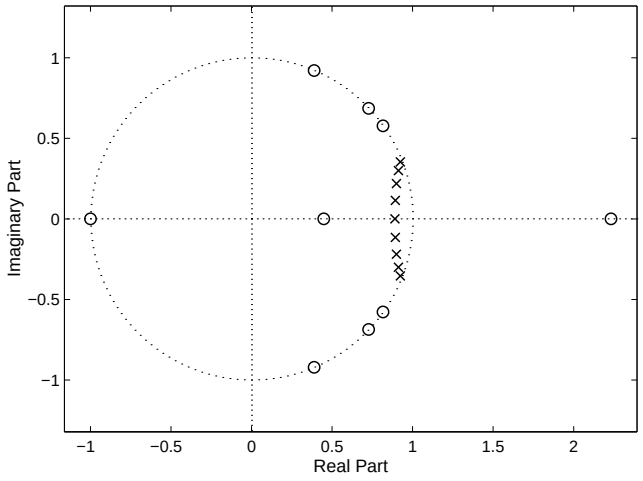

Fig. 8.4. Pole-zero plot for the optimized finite-precision direct LWD filter in Example 1.

Another advantage of the proposed filters compared to the direct LWD filter is the fact that the radius of the outermost complex-conjugate pole pair is significantly smaller. For $K = 1$ and $K = 4$, these values are $0.989\,20$, and $0.901\,38$, respectively. When using the adaptors shown in Fig. 9 in [4], the output noise gains are 31.9 dB and 21.8 dB for $K = 1$ and $K = 4$, respectively. This shows that for $K = 4$ roughly two fewer bits are required for the data representation to arrive at approximately the same output noise level as with the corresponding direct LWD filter. It should be pointed out that lower output noise values can be achieved by using other adaptor structures (see, e.g., [81]).

8.4.2 Example 2

The proposed algorithm can also be modified for quantizing the coefficients of the parallel-form approximately linear phase recursive digital filters of Section 5. In this case, the resulting discrete problems are the following [cf. Eq. (8.10)]: Find Φ, as well as τ, the slope of the linear-phase response, to minimize ψ subject to

$$|E(\Phi, \omega_i)| - 1 \leq 0 \quad \text{for } i = 1, 2, \ldots, L_p + L_s, \tag{8.11a}$$

$$E(\Phi, \omega_i) \leq 0 \quad \text{for } i = 1, 2, \ldots, L_p, \tag{8.11b}$$

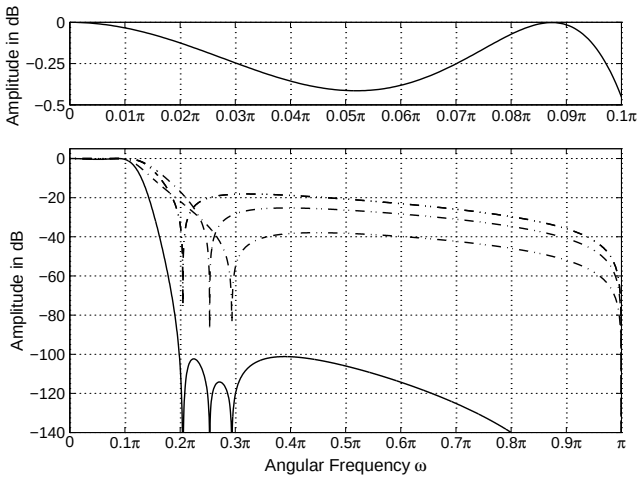

Fig. 8.5. Some amplitude responses for the cascade of four optimized finite-precision LWD filters in Example 1. The solid and dot-dashed lines show the responses for the overall filter and the subfilters, respectively. Two subfilters are identical (the dot-dashed line with the lowest attenuation).

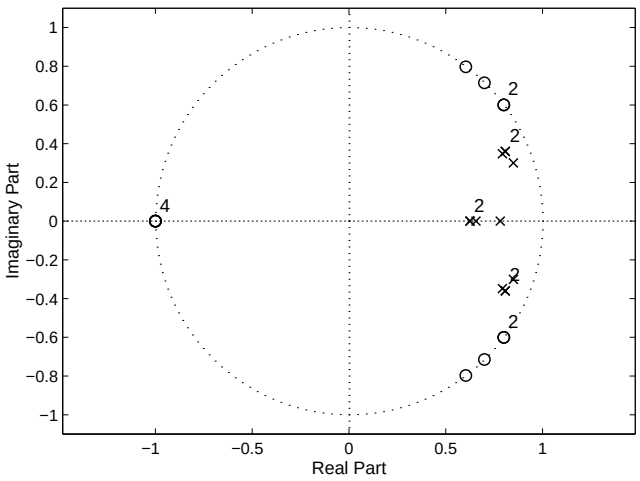

Fig. 8.6. Pole-zero plot for the cascade of four optimized finite-precision LWD filters in Example 1.

and

$$|\arg[H(\Phi, e^{j\omega_i})] - \tau\omega| - \Delta \leq 0 \quad \text{for } i = 1, 2, \ldots, L_p, \tag{8.11c}$$

where $\arg[H(\Phi, e^{j\omega})]$ denotes the unwrapped phase response of the filter and Δ is the maximum allowable ripple from the linear-phase response. Again, the Dutta-Vidyasagar algorithm has been exploited for solving the above problem. Note that in the case of approximately linear phase recursive filters, the cascade implementation does not offer considerable advantages compared to direct LWD filters.

The specifications are the same as in Example 1 in Section 5. That is, it is desired to design a low-pass filter with passband and stopband edges at $\omega_p = 0.05\pi$ and at $\omega_s = 0.1\pi$, respectively. The maximum allowable passband ripple is $\delta_p = 0.0228$ (0.2-dB passband ripple) and the stopband ripple is $\delta_s = 10^{-3}$ (60-dB stopband attenuation). As shown in Section 5, an excellent phase performance is obtained by using a ninth-order LWD filter. For this optimal infinite-precision filter, the phase error from the average phase slope is $\Delta = 0.093\,99$ degrees. In order to allow some toler-

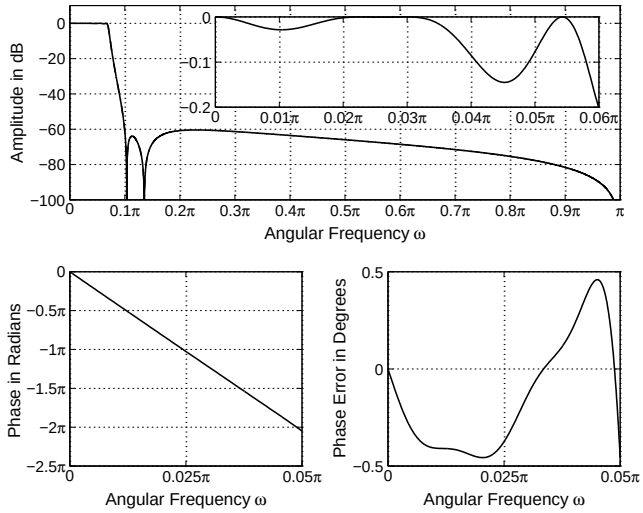

Fig. 8.7. Amplitude and phase responses for the optimized finite-precision approximately linear-phase LWD filter.

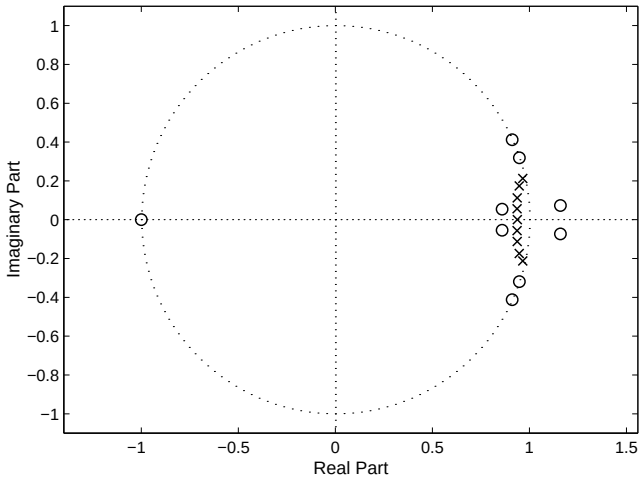

Fig. 8.8. Pole-zero plot for the optimized finite-precision approximately linear-phase LWD filter.

TABLE VIII
OPTIMIZED FINITE-PRECISION ADAPTOR COEFFICIENTS FOR THE APPROXIMATELY LINEAR-PHASE LWD FILTER

$A_1(z)$	$A_2(z)$
$\gamma_0 = \ 1 - 2^{-4}$	
$\gamma_1 = -1 + 2^{-5} - 2^{-7}$	$\gamma_5 = -1 + 2^{-4} + 2^{-7} + 2^{-9}$
$\gamma_2 = \ 1 - 2^{-5} + 2^{-7}$	$\gamma_6 = \ 1 - 2^{-6} - 2^{-9} + 2^{-11}$
$\gamma_3 = -1 + 2^{-3} - 2^{-6} + 2^{-10}$	$\gamma_7 = -1 + 2^{-3} - 2^{-8}$
$\gamma_4 = \ 1 - 2^{-7} - 2^{-10}$	$\gamma_8 = \ 1 - 2^{-8}$

ance for the coefficient quantization, the maximum allowable phase error is increased to $\Delta = 0.5$ degrees.

The filter specifications are met if the adaptor coefficient are represented using four powers of two with eleven fractional bits. A total of ten adders are required to implement all the adaptor coefficients. The phase error for the quantized filter is $\Delta = 0.458\,549$ degrees. The minimum order of a linear-phase FIR filter to meet the same amplitude specifications is 107, requiring 107 delay elements and 54 multipliers when exploiting coefficient symmetry. The delay of the linear-phase FIR equivalent is 53.5 samples, whereas for the proposed recursive filter the delay is only 40.9 samples.

Figure 8.7 shows the amplitude and phase responses as well as the passband details for the optimized multiplierless filter, whereas Fig. 8.8 shows the pole-zero plot for the filter. The optimized adaptor coefficients are given in the Table VIII.

9 DESIGN OF FIR FILTERS WITH POWER-OF-TWO COEFFICIENTS

In VLSI implementations a general multiplier element is very costly. To get around this problem, it is attractive to carry out the multiplication of a data sample by a filter coefficient value by using a sequence of shifts and adds. The shifts can be implemented by using hard-wired shifters and hence they are essentially free.

During the past two decades numerous algorithms have been proposed for designing multiplierless FIR filters [48, 85–112]. The great variety of methods include the mixed-integer linear programming (MILP) [86, 87, 93], weighted least-squares methods [88, 98, 103], stochastic optimization, e.g., genetic algorithms [99, 106], quantization by coefficient sensitivity [97, 112], time-domain approximations [96, 102, 111], and local search techniques [91, 92].

In this application we propose a method for designing FIR filters with a minimum number of adders required to meet the specifications. The method is based on the observation that by first finding the largest and smallest values for the coefficients of filter in such a way that the given criteria are still met by reoptimizing the remaining coefficient values, we are able to find a parameter space which includes the feasible space where the filter specifications are satisfied. After finding this larger space, all what is needed is to check whether in this space there exist the desired discrete values for the coefficient representations.

The main advantage of the proposed algorithm compared with the other existing algorithms is that it offers more flexibility for allocating the adders in such a manner that the overall complexity is minimized. The drawback of the algorithm is that for high order filters the search space can be very large resulting in excessive computation times. However, there are several "tricks" to speed up the algorithm. In addition, the algorithm is highly parallel and the subproblems can be solved independently using parallel processing.

9.1 Introduction

The zero-phase frequency response of a linear-phase Nth-order FIR filter can be expressed as

$$H(\omega) = \sum_{n=0}^{M} h(n)\,\mathrm{Trig}(\omega, n), \tag{9.1}$$

where the $h(n)$'s are the filter coefficients and $\mathrm{Trig}(\omega, n)$ is an appropriate trigonometric function depending on whether N is odd or even and whether the impulse response is symmetrical or antisymmetrical [48, 113]. Here, $M = N/2$ if N is even, and $M = (N+1)/2$ if N is odd.

The general form for expressing the FIR filter coefficients as a sums and differences of signed-powers-of-two (SPT) terms is

$$h(n) = \sum_{k=1}^{W_n+1} a_{k,n} 2^{-P_{k,n}} \quad \text{for } n = 1, 2, \ldots, M, \tag{9.2}$$

where $a_{k,n} \in \{-1, 1\}$ and $P_{k,n} \in \{1, 2, \ldots, L\}$ for $k = 1, 2, \ldots, W+1$, that is, each coefficient $h(n)$ has W_n adders and an L-bit coefficient wordlength.

9.2 Statement of the Problem

The amplitude criteria for the FIR filter can be written as

$$1 - \delta_p \le \beta^{-1}|H(\omega)| \le 1 + \delta_p \quad \text{for } \omega \in [0, \omega_p] \tag{9.3a}$$

$$\beta^{-1}|H(\omega)| \le \delta_s \quad \text{for } \omega \in [\omega_s, \pi], \tag{9.3b}$$

where

$$\beta = \frac{1}{2}\left[\max|H(\omega)| + \min|H(\omega)|\right] \quad \text{for } \omega \in [0, \omega_p] \tag{9.3c}$$

is the average passband gain. These criteria are used when the filter coefficients are quantized on a highly nonuniform discrete grid as in the case of the power-of-two coefficients. In this case, is it beneficial to take the passband gain of the filter as an optimization variable together with the filter coefficients. Obviously, this is not a problem, since in many applications the major concern is the relative stopband attenuation (i.e., relative to passband attenuation) [91–93, 111].

To estimate the implementation costs for the filter, we consider the number of adders and subtractors required to implement all the coefficients as well as the number of adders inside the delay line for transposed-form implementations. This is because the shifts can be implemented almost for free using hardwired shifters and the number adders required to implement all the coefficients depends on the filter implementation, e.g., whether the coefficient symmetry is exploited and whether the redundancy within the coefficients is utilized.

If the coefficient symmetry is exploited, then a reasonable estimate for the filter costs can be expressed as

$$N - 2Q + \sum_{n=0}^{M} W_n, \tag{9.4}$$

where Q is the number of zero-valued coefficients requiring no implementation.

The optimization problem under consideration is the following:

Optimization Problem: Given ω_p, ω_s, δ_p, and δ_s, as well as L, the number of fractional bits, and maximum allowed number of SPT terms per coefficient, find the filter coefficients $h(n)$ for $n = 1, 2, \ldots, M$ as well as β to minimize implementation cost, as given by Eq. (9.4), in such a manner that first the criteria of Eq. (9.3) are met and then the normalized peak ripple (NPR), given by

$$\delta_{\text{NPR}} = \max\{\widehat{\delta}_p w_p/\beta, \widehat{\delta}_s w_s/\beta\}, \tag{9.5}$$

is minimized. Here, $\widehat{\delta}_p$ and $\widehat{\delta}_s$, w_p, and w_s are the realized passband and stopband ripples and error weightings, respectively.

9.3 Filter Optimization

The solution to the stated optimization problem can be found in two steps. In the first step, a filter with infinite-precision coefficients is determined in such a way that it exceeds the given amplitude criteria to provide some tolerance for the coefficient quantization. The second step involves finding a filter meeting the given criteria with the simplest coefficient representation forms.

9.3.1 Optimization of Infinite-Precision Coefficients

It has turned out that a very straightforward quantization scheme for the FIR filter coefficients is obtained as follows. For filter coefficient $h(n)$ for $n = 0, 1, \ldots, M-1$ the largest and smallest values of the coefficient are determined in such a manner that the given amplitude criteria are met and $h(M) = 1$. Therefore, there are $2M$ problems of the following form. Find the filter coefficients $h(n)$ for $n = 0, 1, \ldots, M-1$ as well as β to minimize ψ subject to the conditions of Eq. (9.3). For these problems ψ is $-h(n)$ or $h(n)$ where $h(n)$ is one among the filter coefficients $h(n)$ for $n = 0, 1, \ldots, M-1$. To solve these problems, we discretize the passband and stopband regions into the frequency points $\omega_i \in [0, \omega_p]$ for $i = 1, 2, \ldots, L_p$ and $\omega_i \in [\omega_s, \pi]$ for $i = L_p+1, L_p+2, \ldots, L_p+L_s$. The resulting discrete minimization problem is to find $h(n)$ for $n = 0, 1, \ldots, M-1$ as well as β to minimize ψ subject to conditions given by Eq. (9.6). The above optimization problems can be solved conveniently by using linear optimization.

TABLE IX
OPTIMIZED INFINITE-PRECISION COEFFICIENTS FOR THE 10TH-ORDER FILTER IN EXAMPLE

$h(n)^{(\min)}$	$h(n)^{(\max)}$
$h(0) = \ 0.037\,986\,8$	$h(0) = \ 0.069\,715\,7$
$h(1) = -0.175\,829\,4$	$h(1) = -0.126\,536\,9$
$h(2) = -0.155\,314\,5$	$h(2) = -0.106\,186\,3$
$h(3) = \ 0.210\,734\,5$	$h(3) = \ 0.250\,780\,1$
$h(4) = \ 0.691\,666\,7$	$h(4) = \ 0.799\,256\,3$
$h(5) = \ 1.0$	$h(5) = \ 1.0$

9.3.2 Optimization of Finite-Precision Coefficients

It has been experimentally observed that the parameter space defined above forms the feasible space where the filter specifications are satisfied. After finding this larger space, all what is needed is to check whether in this space there exists a combination of the discrete coefficient values with which the given overall criteria are met. However, since the average passband gain of the filter is not fixed, the search must be performed for one octave of the scale factor α in order to find the optimal solution.[15]

The search can be done in a straightforward manner by first generating a look-up table containing all the possible power-of-two numbers for a given wordlength and a given maximum number of adders per coefficient. The largest and smallest values of the infinite-precision coefficients are then scaled for each power-of-two number $\alpha \equiv h(M) \in [1/3, 2/3]$ in a look-up table as

$$\widehat{h}(n)^{(\min)} = \alpha h(n)^{(\min)} \quad \text{for } n = 0, 1, \ldots, M \tag{9.7a}$$

$$\widehat{h}(n)^{(\max)} = \alpha h(n)^{(\max)} \quad \text{for } n = 0, 1, \ldots, M \tag{9.7b}$$

and the magnitude response is evaluated for each combination of the power-of-two numbers in the ranges $[\widehat{h}(n)^{(\min)}, \widehat{h}(n)^{(\max)}]$ for $n = 0, 1, \ldots, M$ to check whether the filter meets the amplitude criteria.

To facilitate the above discussion, we use a simple example that illustrates the proposed design procedure. Suppose that we need to design a 10th-order lowpass filter whose passband and stopband edges are at 0.25π and at 0.5π, respectively. The desired passband ripple and the desired stopband attenuation are 0.2 dB and 20 dB, respectively. The largest and smallest values of the infinite-precision coefficients after infinite-precision optimization are given in Table IX.

When two SPT terms per coefficient with seven fractional bits are used for coefficient representations, there exist ten scaling factors, as shown in Table X, for which there are discrete coefficient values between the smallest and largest values of the infinite-precision coefficients. In addition, Table X shows the absolute minimum number of adders required the implement the overall filter. As can be seen from this table, the minimum number of adders is achieved when the scaling factor α is equal to 0.5.

[15] Only one octave of the scale factors needs to be searched since the scaling by a factor of two does not affect the quantization process.

$$\sum_{n=0}^{M} h(n)\,\mathrm{Trig}(\omega_i, n) - \beta[\delta_p + D(\omega_i)] \le 0 \quad \text{for } i = 1, 2, \ldots, L_p, \tag{9.6a}$$

$$-\sum_{n=0}^{M} h(n)\,\mathrm{Trig}(\omega_i, n) - \beta[\delta_p - D(\omega_i)] \le 0 \quad \text{for } i = 1, 2, \ldots, L_p, \tag{9.6b}$$

$$\sum_{n=0}^{M} h(n)\,\mathrm{Trig}(\omega_i, n) - \beta[\delta_s + D(\omega_i)] \le 0 \quad \text{for } i = L_p + 1, L_p + 2, \ldots, L_p + L_s, \tag{9.6c}$$

$$-\sum_{n=0}^{M} h(n)\,\mathrm{Trig}(\omega_i, n) - \beta[\delta_s - D(\omega_i)] \le 0 \quad \text{for } i = L_p + 1, L_p + 2, \ldots, L_p + L_s, \tag{9.6d}$$

and

$$h(M) = 1. \tag{9.6e}$$

When the largest and the smallest values of the infinite-precision coefficients are scaled according to Eq. (9.7) with $\alpha = 0.5$, the permissible discrete values within the limits $[\widehat{h}(n)^{(\min)}, \widehat{h}(n)^{(\max)}]$ for $n = 0, 1, \ldots, M$ are shown in Table XI. As can be seen from the Table XI, there are two permissible discrete values for $h(0)$ and $h(1)$, three permissible values for $h(2)$ and $h(3)$, and one for $h(4)$ and $h(5)$. Therefore, the overall number of combinations to be checked is $2 \cdot 2 \cdot 3 \cdot 3 \cdot 1 \cdot 1 = 36$. In order to reduce the computational complexity in finding the best discrete solution, it is beneficial to start with the combinations requiring the smallest number of additions. If one of these combinations meets the criteria, then there is no need to consider other combinations. A simple branch-and-bound type algorithm has been developed for the evaluation of the combinations.

9.4 Numerical Examples

9.4.1 Example 1

Our first example considers Example 1 in [108]. The passband and stopband edge frequencies are 0.3π and 0.5π, respectively. In [108] it was observed that the minimum number of SPT terms, required to achieve a -60 dB normalized peak ripple value for a 37th-order low-pass filter with twelve fractional bits is 40 when the maximum allowed number of SPT terms per a coefficient is three. Using the proposed algorithm, only 34 SPT terms and 48 adders, with twelve fractional bits and three SPT terms per coefficient, are required to implement the overall filter. The passband ripple and the stopband attenuation for the quantized filter are 0.00822 dB and 60.50 dB, respectively. The optimized finite-precision coefficients are given in Table XIII.

The characteristics of the filters designed using various algorithms are summarized in Table XII. Here, δ_{NPR} gives the normalized peak ripple in decibels, NSP and NA give the number of signed-powers-of-two terms and number adders required to implement the overall filter, respectively.

It should be pointed out that in this applications it is desired to minimize the number of adders required to implement the overall filter when coefficient symmetry is utilized. However, in many other papers, the optimization goal is to minimize the number of SPT terms required to implement all the coefficients. [16] Therefore, in order to compare the proposed algorithm with the other existing methods, the number of SPT terms required to implement all the coefficients is also given.

9.4.2 Example 2

In this example we concentrate on the specifications considered in [87, 91, 92, 102], that is, it is desired to design 24th-order low-pass filter. The passband and stopband edges are 0.3π and 0.5π,

[16] This is because if the coefficient symmetry is not utilized, the number of SPT terms corresponds to the number adders required to implement the overall filter.

TABLE X
PERMISSIBLE SCALING FACTORS AND THE CORRESPONDING MINIMUM NUMBER OF ADDERS REQUIRED TO IMPLEMENT THE OVERALL FILTER

Scaling Factor α		Min. No. Adds
0.375	$= 2^{-1} - 2^{-3}$	4
$0.437\,5$	$= 2^{-1} - 2^{-4}$	3
$0.484\,375$	$= 2^{-1} - 2^{-6}$	3
$0.492\,1875$	$= 2^{-1} - 2^{-7}$	3
0.5	$= 2^{-1}$	2
$0.507\,8125$	$= 2^{-1} + 2^{-7}$	3
$0.515\,625$	$= 2^{-1} + 2^{-6}$	3
$0.531\,25$	$= 2^{-1} + 2^{-5}$	3
$0.562\,5$	$= 2^{-1} + 2^{-4}$	3
0.625	$= 2^{-1} + 2^{-3}$	5

TABLE XI
FINITE-PRECISION COEFFICIENT VALUES FOR $\alpha = 0.5$

Finite-Precision Values			No. Adds
$h(0)^{(0)} =$	$0.023\,437\,5$	$= 2^{-5} - 2^{-7}$	1
$h(0)^{(1)} =$	$0.031\,25$	$= 2^{-5}$	0
$h(1)^{(0)} =$	$-0.078\,125$	$= -2^{-4} - 2^{-6}$	1
$h(1)^{(1)} =$	$-0.070\,312\,5$	$= -2^{-4} - 2^{-7}$	1
$h(2)^{(0)} =$	$-0.070\,312\,5$	$= -2^{-4} - 2^{-7}$	1
$h(2)^{(1)} =$	$-0.062\,5$	$= -2^{-4}$	0
$h(2)^{(2)} =$	$-0.054\,687\,5$	$= -2^{-4} + 2^{-7}$	1
$h(3)^{(0)} =$	$0.109\,375$	$= 2^{-3} - 2^{-6}$	1
$h(3)^{(1)} =$	$0.117\,187\,5$	$= 2^{-3} - 2^{-7}$	1
$h(3)^{(2)} =$	0.125	$= 2^{-3}$	0
$h(4)^{(0)} =$	0.375	$= 2^{-1} - 2^{-3}$	1
$h(5)^{(0)} =$	0.5	$= 2^{-1}$	0

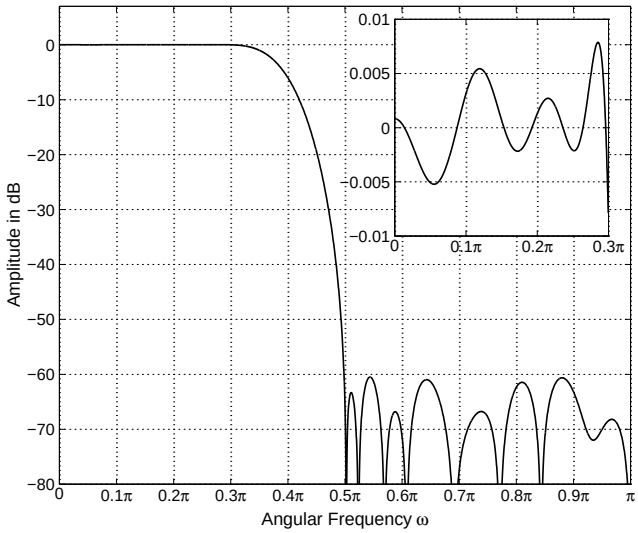

Fig. 9.1. Magnitude response for the multiplierless FIR filter in Example 1.

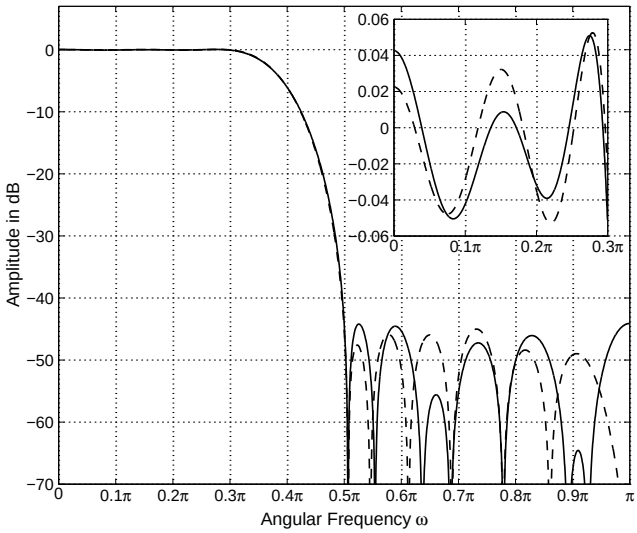

Fig. 9.2. Magnitude responses for the multiplierless FIR filters in Examples 2 and 3. The solid and dashed lines show the responses in Examples 2 and 3, respectively.

TABLE XII
SUMMARY OF FILTER DESIGNS IN EXAMPLE 1

Method	δ_{NPR} (dB)	NSP	NA
Lim *et al.* [87]	-62.08	43	–
Chen *et al.* [108]	-60.87	40	–
Proposed	-60.48	34	48

TABLE XIII
OPTIMIZED FINITE-PRECISION COEFFICIENT VALUES FOR THE FIR FILTER IN EXAMPLE 1

$h(0) = h(37) = -2^{-11}$
$h(1) = h(36) = 0$
$h(2) = h(35) = +2^{-9} - 2^{-12}$
$h(3) = h(34) = +2^{-9}$
$h(4) = h(33) = -2^{-9} - 2^{-11}$
$h(5) = h(32) = -2^{-7} + 2^{-9} - 2^{-11}$
$h(6) = h(31) = 0$
$h(7) = h(30) = +2^{-6} - 2^{-8}$
$h(8) = h(29) = +2^{-7} + 2^{-9}$
$h(9) = h(28) = -2^{-6} + 2^{-8} - 2^{-10}$
$h(10) = h(27) = -2^{-5} + 2^{-8} + 2^{-12}$
$h(11) = h(26) = 0$
$h(12) = h(25) = +2^{-4} - 2^{-6} - 2^{-9}$
$h(13) = h(24) = +2^{-5} + 2^{-8} + 2^{-10}$
$h(14) = h(23) = -2^{-4} + 2^{-6} - 2^{-10}$
$h(15) = h(22) = -2^{-3} + 2^{-6} + 2^{-8}$
$h(16) = h(21) = 0$
$h(17) = h(20) = +2^{-2} + 2^{-6}$
$h(18) = h(19) = +2^{-1}$

respectively. The desired ripples for the ideal equiripple filter are $\delta_{\mathrm{p}} = \delta_{\mathrm{s}} = 0.005$ ($A_{\mathrm{s}} = -46$) dB. In this case, three SPT terms with nine fractional bits are used for coefficient representations. In this case, 30 adders are required to implement the overall filter. Table XIV summarizes the characteristics of the filters designed using various methods. The amplitude response for the optimized filter as well as the passband details are shown using the solid line in Fig. 9.2.

TABLE XIV
SUMMARY OF FILTER DESIGNS IN EXAMPLE 2

Method	δ_{NPR} (dB)	NSP	NA
Samueli [92]	-42.17	24	35
Li *et al.* [96]	-43.33	24	–
Chen *et al.* [102]	-43.97	24	33
Proposed	-44.09	21	30

9.4.3 Example 3

The proposed algorithm also enables the use of the common subexpression reduction algorithms (see, e.g, [110,114–117]) for the evaluation of the filter costs. For example, if the filter specifications are the same as in Example 2, then the overall filter can be implemented using only 26 adders with a 23th-order filter, compared to 30 adders required for the conventional implementation. In this case, the subexpression reduction algorithm based on the method described in [117] is carried out for all the filters satisfying the amplitude specifications. The normalized peak ripple magnitude for the optimized filter is -44.34 dB. The optimized coefficients are given in Table XV. The amplitude response for the optimized filter as well as the passband details are shown using the dashed line in Fig. 9.2.

REFERENCES

[1] L. Davis and M. Streenstrup, "Genetic algorithms and simulated annealing: An overview," in *Algorithms and Simulated Annealing*, L. Davis, Ed., chapter 1, pp. 1–11. Morgan Kaufmann, 1987.

[2] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[3] D. B. Fogel, *Evolutionary Computation*. IEEE Press, Piscataway, 1995.

[4] P. J. M. van Laarhoven and E. H. L. Aarts, *Simulated annealing: Theory and applications*. Kluwer Academic Publishers, 1989.

TABLE XV
OPTIMIZED FINITE-PRECISION COEFFICIENT VALUES FOR THE FIR FILTER IN EXAMPLE 3

$h_a = 2^{-7},\ h_b = 2^{-4} - 2^{-6},\ h_c = h_b + 2^{-2}, h_d = 2^{-9}$		
$h(0)$	$= h(23) = +2^{-7}$	$= h_a$
$h(1)$	$= h(22) = +2^{-7}$	$= h_a$
$h(2)$	$= h(21) = -2^{-6} + 2^{-8}$	$= -2^{-2}h_b$
$h(3)$	$= h(20) = -2^{-5} + 2^{-7}$	$= -2^{-1}h_b$
$h(4)$	$= h(19) = 0$	$= 0$
$h(5)$	$= h(18) = +2^{-4} - 2^{-6} - 2^{-9}$	$= h_b - h_d$
$h(6)$	$= h(17) = +2^{-5} + 2^{-7} - 2^{-9}$	$= 2^{-3}h_c$
$h(7)$	$= h(16) = -2^{-4} + 2^{-6} - 2^{-8}$	$= -h_b - 2^{-8}$
$h(8)$	$= h(15) = -2^{-3} + 2^{-7} + 2^{-9}$	$= -2^{-3} + h_a + h_d$
$h(9)$	$= h(14) = 0$	$= 0$
$h(10)$	$= h(13) = +2^{-2} + 2^{-4} - 2^{-6}$	$= h_c$
$h(11)$	$= h(12) = +2^{-1} + 2^{-4}$	$= 2^{-1} + 2^{-4}$

[5] S. R. K. Dutta and M. Vidyasagar, "New algorithms for constrained minimax optimization," *Math. Programming*, vol. 13, pp. 140–155, 1977.

[6] R. Fletcher and M. Powell, "A rapidly convergent descent method for minimization," *Comput. J.*, vol. 6, no. 2, pp. 163–168, 1963.

[7] R. Fletcher, *Practical Methods of Optimization*. Jonh Wiley & Sons, New York, NY, 2 edition, 1989.

[8] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1995.

[9] P. Spellucci, "An SQP method for general nonlinear programs using only equality constrained subproblems," *Math. Prog.*, vol. 82, pp. 413–448, 1998.

[10] T. Coleman, M. A. Branch, and A. Grace, *Optimization Toolbox User's Guide*, The MathWorks, Inc, Jan. 1999, Version 2.

[11] P. M. Gill, W. Murray, and M. H. Wright, *Practical Optimization*. Academic Press, London, 1981.

[12] N. Iwata and Masao Iri, "Computation of the gradient of a function with many variables," in *Papers of the Special Interest Group on Numerical Analysis*, pp. 1–10. Information Processing Society of Japan, 7-1, 1983.

[13] H. Fischer, "Automatic differentiation: Parallel computation of function, gradient and Hessian matrix," *Parallel Computing*, vol. 13, pp. 101–110, 1990.

[14] J. C. Gilbert, "Automatic differentation and iterative processes," *Optimization Methods and Software*, vol. 1, pp. 13–21, 1992.

[15] G. Corliss M. Berz, C. Bischof and A. Griewank, Eds., *Computational Differentiation: Techniques, Applications, and Tools*. SIAM, Philadelphia, Penn., 1996.

[16] J. L. Zhou and A. L. Tits, "An SQP algorithm for finely discretized continuous minimax problems and other minimax problems with many objective functions," *SIAM J. Num. Anal.*, vol. 6, no. 2, pp. 461–487, May 1996.

[17] J. L. Zhou, A. L. Tits, and C. T. Lawrence, "User's guide for FFSQP version 3.7: A fortran code for solving optimization programs, possibly minimax, with general inequality constraints and linear equality constraints, generating feasible iterates," Tech. Rep. SRC-TR-92-107r5, Institute for Systems Research, University of Maryland, College Park, MD 20742, 1997.

[18] H. S. Malvar, *Signal Processing with Lapped Transforms*. Artech House, Boston, MA, USA, 1992.

[19] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*. Prentice Hall, Englewood Cliffs, NJ, 1992.

[20] N. J. Fliege, *Multirate Digital Signal Processing*. John Wiley & Sons, Chichester, 1994.

[21] H. S. Malvar, "Modulated QMF filter banks with perfect reconstruction," *Electron. Lett.*, vol. 26, pp. 906–907, June 1990.

[22] T. A. Ramstad and J. P. Tanem, "Cosine-modulated analysis-synthesis filter bank with critical sampling and perfect reconstruction," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991, pp. 1789–1792.

[23] R. D. Koilpillai and P. P. Vaidyanathan, "New results on cosine-modulated FIR filter banks satisfying perfect reconstruction," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991, pp. 1793–1796.

[24] H. S. Malvar, "Extended lapped transforms: Fast algorithms and applications," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991, pp. 1797–1800.

[25] R. D. Koilpillai and P. P. Vaidyanathan, "Cosine-modulated FIR filter banks satisfying perfect reconstruction," *IEEE Trans. Signal Processing*, vol. 40, no. 4, pp. 770–783, Apr. 1992.

[26] T. Saramäki, "Designing prototype filters for perfect-reconstruction cosine-modulated filter banks," in *Proc. IEEE Int. Symp. Circuits Syst.*, San Diego, CA, May 1992, pp. 1605–1608.

[27] H. S. Malvar, "Extended lapped transforms: Properties, applications and fast algorithms," *IEEE Trans. Signal Processing*, vol. 40, no. 11, pp. 2703–2714, Nov. 1992.

[28] T. Q. Nguyen, "Near-perfect-reconstruction pseudo-QMF banks," *IEEE Trans. Signal Processing*, vol. 42, no. 1, pp. 63–76, Jan. 1994.

[29] T. Saramäki, "A generalized class of cosine-modulated filter banks," in *Transforms and Filter Banks: Proceedings of the First International Worshop on Transforms and Filter Banks*, K. Egiazarian, T. Saramäki, and J. Astola, Eds., pp. 336–365. Tampere, Finland, 1998.

[30] F. Minzer, "On half-band, third-band, and Nth-band filters FIT filters and their design," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, pp. 734–738, Oct. 1982.

[31] T. Saramäki and Y. Neuvo, "A class of FIR (Nth-band) Nyquist filters with zero intersymbol interference," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 1182–1190, Oct. 1987.

[32] L. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[33] F. Leeb, "Lattice wave digital filters with simultaneous conditions on amplitude and phase," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Toronto, Canada, May 1991, pp. 1645–1648.

[34] J. Földvári-Orosz, T. Henk, and E. Simonyi, "Simultaneous amplitude and phase approximation for lumped and sampled filters," *Int. J. Circuit Theory Applicat.*, vol. 19, pp. 77–100, 1991.

[35] B. Jaworski and T. Saramäki, "Linear phase IIR filters composed of two parallel allpass sections," in *Proc. IEEE Int.*

Symp. Circuits Syst., London, England, May 1994, pp. 537–540.

[36] A. Jones, S. Lawson, and T. Wicks, "Design of cascaded allpass structures with magnitude and delay constraint using simulated annealing and quasi-Newton methods," in *Proc. IEEE Int. Symp. Circuits Syst.*, Singapore, June 1991, pp. 2439–2442.

[37] S. S. Lawson and T. Wicks, "Design of efficient digital filters satisfying arbitrary loss and delay specifications," *Proc. Inst. Elect. Eng., Part. G*, vol. 139, pp. 611–620, Oct. 1992.

[38] K. Surma-aho and T. Saramäki, "A systematic approach for designing approximately linear phase recursive digital filters," *IEEE Trans. Circuits Syst. II*, vol. 46, no. 7, pp. 956–962, July 1999.

[39] R. Nouta, "The Jaumann structure in wave-digital filters," *Int. J. Circuit Theory Applicat.*, vol. 2, pp. 163–174, June 1974.

[40] A. Fettweis, H. Levin, and A. Sedlmeyer, "Wave digital lattice filters," *Int. J. Circuit Theory Applicat.*, vol. 2, pp. 203–211, June 1974.

[41] L. Gazsi, "Explicit formulas for lattice wave digital filters," *IEEE Trans. Circuits Syst.*, vol. CAS-32, pp. 68–88, Jan. 1985.

[42] T. Saramäki, "On the design of digital filters as a sum of two all-pass filters," *IEEE Trans. Circuits Syst.*, vol. CAS-32, no. 11, pp. 1191–1193, Nov. 1985.

[43] C. W. Farrow, "A continuously variable digital delay element," in *Proc. IEEE Int. Symp. Circuits Syst.*, Espoo, Finland, June 1991, pp. 2641–2645.

[44] J. Vesma and T. Saramäki, "Interpolation filters with arbitrary frequency response for all-digital receivers," in *Proc. IEEE Int. Symp. Circuits Syst.*, Atlanta, Georgia, May 1996, pp. 568–571.

[45] F. M. Gardner, "Interpolation in digital modems — Part I: Fundamentals," *IEEE Trans. Commun.*, vol. 41, pp. 501–507, Mar. 1993.

[46] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, "Splitting the unit delay," *IEEE Signal Processing Magazine*, vol. 13, no. 1, pp. 30–60, Jan. 1996.

[47] J. Vesma and T. Saramäki, "Design and properties of polynomial-based fractional delay filters," in *Proc. IEEE Int. Symp. Circuits Syst.*, Geneva, Switzerland, May 2000, vol. 1, pp. 104–107.

[48] T. Saramäki, "Finite impulse response filter design," in *Handbook for Digital Signal Processing*, S. K. Mitra and J. F. Kaiser, Eds., chapter 4, pp. 155–277. John Wiley and Sons, New York, 1993.

[49] E. C. Ifeachor and B. W. Jervis, *Digital Signal Processing, A Practical Approach*. Addison-Wesley, 1993.

[50] M. Renfors and Y. Neuvo, "The maximum sampling rate of digital filters under hardware speed constraints," *IEEE Trans. Circuits Syst.*, vol. CAS-28, no. 3, pp. 196–202, Mar. 1981.

[51] K. K. Parhi and D. G. Messerschmitt, "Pipelined VLSI recursive filter architectures using scattered look-ahead and decomposition," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Apr. 11–14 1988, vol. 4, pp. 2120–2123.

[52] S. C. Knowles, R. F. Woods, J. G. McWhirter, and J. V. McCanny, "A high performance systolic IIR filter architecture," in *IEE Colloquium on Digital Signal Processing for VLSI*, 1988, pp. 2/1–2/4.

[53] K. K. Parhi and D. G. Messerschmitt, "Pipeline interleaving and parallelism in recursive digital filters — Part I: Pipelining using scattered look-ahead and decomposition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, no. 7, pp. 1099–1117, July 1989.

[54] S. C. Knowles, J. G. McWhirter, R. F. Woods, and J. V. McCanny, "Bit-level systolic architectures for high-speed pipelined recursive filters," *J. VLSI Signal Processing*, vol. 1, no. 1, pp. 9–24, Sept. 1989.

[55] H. G. Martinez and T. W. Parks, "Design of recursive digital filters with optimum magnitude and attenuation poles on the unit circle," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, no. 3, pp. 150–156, Mar. 1978.

[56] M. A. Richards, "Application of Deczky's program for recursive filter design to the design of recursive decimators," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, no. 5, pp. 811–814, Oct. 1982.

[57] C.-P. Lan and C.-W. Jen, "Efficient time domain synthesis of pipelined recursive filters," *IEEE Trans. Circuits Syst. II*, vol. 41, no. 9, pp. 618–622, Sept. 1994.

[58] P. Boonyanant and S. Tantaratana, "Direct form-based pipelined IIR filter realization," in *Proc. 1998 IEEE Asia-Pacific Conf. on Circuits and Systems*, Nov. 24–27 1998, pp. 85–88.

[59] H. B. Voelcker and E. E. Hartquist, "Digital filtering via block recursion," *IEEE Trans. Audio Electroacoust.*, vol. AU-18, no. 6, pp. 169–176, June 1970.

[60] P. M. Kogge and H. S. Stone, "A parallel algorithm for the efficient solution of a general class of recursive equations," *IEEE Trans. Comput.*, vol. C-22, no. 8, pp. 786–793, Aug. 1973.

[61] H. H. Loomis and B. Sinha, "High-speed recursive digital filter realization," *IEEE Trans. Circuit, Syst., Signal Process.*, vol. 3, no. 3, pp. 267–294, Sept. 1984.

[62] M. A. Soderstrand, K. Chopper, and B. Sinha, "Comparison of three new techniques for pipelining IIR digital filters," in *Proc. 18th Asilomar Conf. on Circuits, Systems and Computers*, Pacific Grove, CA, Nov. 1984, pp. 439–443.

[63] K. K. Parhi and D. G. Messerschmitt, "Pipeline interleaving and parallelism in recursive digital filters — Part II: Pipelined incremental block filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-37, no. 7, pp. 1118–1134, July 1989.

[64] K. K. Parhi, C. Y. Wang, and A. P. Brown, "Synthesis of control circuits in folded pipelined DSP architectures," *IEEE J. Solid-State Circuits*, vol. 27, no. 1, pp. 29–43, Jan. 1992.

[65] M. A. Soderstrand, H. H. Loomis, and R. Gnanasekaran, "Pipelining techniques for IIR digital filters," in *Proc. IEEE Int. Symp. Circuits Syst.*, New Orleans, LA, May 1990, pp. 121–124.

[66] Y. C. Lim and B. Liu, "Pipelined recursive filter with minimum order augmentation," *IEEE Trans. Signal Processing*, vol. 40, no. 7, pp. 1643–1651, July 1992.

[67] E. Q. Wong, M. A. Soderstrand, and H. H. Loomis, "Computer-aided design of pipelined IIR digital filters," in *Proc. of the 1992 IEEE Midwest Symposium on Circuits and Systems*, Washington, D.C., Aug. 1992, IEEE Circuits and Systems Society.

[68] M. A. Soderstrand and A. E. de la Serna, "Stability of IIR digital filters constructed with time-domain pipelining technique," in *Proc. 1993 IEEE Midwest Symposium on Circuits and Systems*, Detroit, MI, Aug. 1993.

[69] A. E. de la Serna, "Stability of time-domain pipelined IIR digital filters," Dipl. Eng. thesis, University of California, Davis, CA, 95616, Sept. 1993.

[70] Y. C. Lim, "Parallel and pipelined implementations of injected numerator lattice digital filters," *IEEE Trans. Circuits Syst. II*, vol. 42, no. 7, pp. 480–486, July 1995.

[71] Y. C. Lim, "A new approach for deriving scattered coefficients of pipelined IIR filters," *IEEE Trans. Signal Processing*, vol. 43, no. 10, pp. 2405–2406, Oct. 1995.

[72] M. A. Soderstrand and A. E. de la Serna, "Minimum denominator-multiplier pipelined recursive digital filters," *IEEE Trans. Circuits Syst. II*, vol. 42, no. 10, pp. 666–672, Oct. 1995.

[73] Z. Jiang and A. N. Willson, Jr, "Design and implementation of efficient pipelined IIR digital filters," *IEEE Trans. Signal Processing*, vol. 43, no. 3, pp. 579–590, Mar. 1995.

[74] K. Chang, "Improved clustered look-ahead pipeling algorithm with minimum order augmentation," *IEEE Trans. Signal Processing*, vol. 45, no. 10, pp. 2575–2579, Oct. 1997.

[75] K. K. Parhi, "Finite word effects in pipelined recursive filters," *IEEE Trans. Signal Processing*, vol. 39, no. 6, pp. 1450–1454, June 1991.

[76] K. S. Arun and D. R. Wagner, "High-speed digital filtering: Structures and finite wordlength effects," *J. VLSI Signal Processing*, vol. 4, no. 4, pp. 355–370, Nov. 1992.

[77] M. Lapointe, H. T. Huynh, and P. Fortier, "Systematic design of pipelined recursive filters," *IEEE Trans. Comput.*, vol. 42, no. 4, pp. 413–426, 1993.

[78] K. K. Parhi, "Pipelining in dynamic programming archetectures," *IEEE Trans. Signal Processing*, vol. 39, no. 6, pp. 1442–1450, June 1991.

[79] J. Yli-Kaakinen and T. Saramäki, "Design of low-sensitivity and low-noise recursive digital filters using a cascade of low-order wave lattice filters," in *Proc. 1998 IEEE Int. Symp. Circuits Syst.*, Monterey, CA, vol. V, pp. 404–408.

[80] J. Yli-Kaakinen and T. Saramäki, "Design of low-sensitivity and low-noise recursive digital filters using a cascade of low-order wave lattice filters," *IEEE Trans. Circuits Syst. II*, vol. 46, no. 7, pp. 906–914, July 1999.

[81] M. Renfors and E. Zigouris, "Signal processor implementation of digital all-pass filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 714–729, May 1988.

[82] J. Yli-Kaakinen, "Optimization of recursive digital filters for practical implementation," Dipl. Eng. thesis, Dept. of Elect. Eng., Tampere Univ. of Tech., Finland, June 1998, [Online]. Available HTTP: http://alpha.cc.tut.fi/˜ylikaaki/thesis.

[83] J. Yli-Kaakinen and T. Saramäki, "An efficient algorithm for the design of lattice wave digital filters with short coefficient wordlength," in *Proc. IEEE Int. Symp. Circuits Syst.*, Orlando, FL, May 30–June 2 1999, pp. 443–448.

[84] J. Yli-Kaakinen and T. Saramäki, "An algorithm for the design of multiplierless approximately linear-phase lattice wave digital filters," in *Proc. IEEE Int. Symp. Circuits Syst.*, Genava, Switzerland, May 28–31 2000, vol. 2, pp. 77–80.

[85] J. W. Adams and A. N. Willson, Jr., "A new approach to FIR digital filters with fewer multipliers and reduced sensitivity," *IEEE Trans. Circuits Syst.*, vol. CAS-30, pp. 277–283, May 1983.

[86] Y. C. Lim, S. R. Parker, and A. G. Constantinides, "Finite word length FIR filter design using integer programming over a discrete coefficient space," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-30, no. 4, pp. 661–?, 1982.

[87] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete powers-of-two coefficient space," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, no. 3, pp. 583–?, 1983.

[88] Y. C. Lim and S. R. Parker, "Discrete coefficient FIR digital filter design based upon LMS criteria," *IEEE Trans. Circuits Syst.*, vol. CAS-30, pp. 723–739, Oct. 1983.

[89] Y. C. Lim and B. Liu, "The design of cascade form FIR filters for discrete space implementation," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 1988, vol. 3, pp. 1814–1817.

[90] Y. C. Lim and B. Liu, "Design of cascade form FIR filters with discrete valued coefficients," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 11, pp. 1735–1739, Nov. 1988.

[91] Q. Zhao and Y. Tadokoro, "A simple design of FIR filters with powers-of-two coefficients," *IEEE Trans. Circuits Syst.*, vol. 35, no. 5, pp. 566–570, May 1988.

[92] H. Samueli, "A low-complexity multiplierless half-band recursive digital filter design," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 3, pp. 442–444, Mar. 1989.

[93] Y. C. Lim and S. R. Parker, "Design of discrete-coefficient-value linear phase FIR filters with optimum normalized peak ripple magnitude," *IEEE Trans. Circuits Syst.*, vol. 37, pp. 1480–1486, Dec. 1990.

[94] H. Shaffeu, M. M. Jones, H. D. Griffiths, and J. T. Taylor, "Improved design procedure for multiplierless FIR digital filters," *Electron. Lett.*, vol. 27, no. 13, pp. 1142–1144, 1991.

[95] C. Young and D. L. Jones, "Improvement in finite wordlength FIR digital filter design by cascading," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 1992, vol. 5, pp. 109–112.

[96] D. Li, J. Song, and Y. C. Lim, "A polynomial-time algorithm for designing digital filters with power-of-two coefficients," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1993, vol. 1, pp. 84–87.

[97] D. Ait-Boudaoud and R. Cemes, "Modified sensitivity criterion for the design of powers-of-two FIR filters," *Electron. Lett.*, vol. 29, no. 16, pp. 1467–1469, 1993.

[98] J.-H. Lee, C.-K. Chen, and Y. C. Lim, "Design of discrete coefficient FIR digital filters with arbitrary amplitude and phase responses," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 444–448, July 1993.

[99] R. Cemes and D. Ait-Boudaoud, "Genetic approach to design of multiplierless FIR filters," *Electron. Lett.*, vol. 29, pp. 2090–2091, Nov. 1993.

[100] G. Wade, A. Roberts, and G. Williams, "Multiplier-less FIR filter design using a genetic algorithm," *IEE Proc.-Vis. Image Signal Process.*, vol. 141, no. 3, pp. 175–180, June 1994.

[101] W. J. Oh and Y. H. Lee, "Cascade/parallel form FIR filters with powers-of-two coefficients," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1994, vol. 2, pp. 545–548.

[102] C. L. Chen, K.-Y. Khoo, and A. N. Willson, Jr., "An improved polynomial-time algorithm for designing digital filters with power-of-two coefficients," in *Proc. 1995 IEEE Int. Symp. Circuits Syst.*, Apr. 28–May 3 1995, vol. 1, pp. 223–226.

[103] J.-J. Shyu and Y.-C. Lim, "A new approach to the design of discrete coefficient FIR digital filters," *IEEE Trans. Signal Processing*, vol. 43, pp. 310–314, Jan. 1995.

[104] Y.-C. Lim, R. Yang, and B. Liu, "The design of cascaded FIR filters," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1996, vol. 2, pp. 181–184.

[105] A. G. Dempster and M. D. Macleod, "Comparison of fixed-point FIR digital filter design techniques," *IEEE Trans. Circuits Syst. II*, vol. 44, no. 7, pp. 591–593, July 1997.

[106] D. W. Redmill and D. R. Bull, "Design of low complexity fir filters using genetic algorithms and directed graphs," in *Second International Conference On Genetic Algorithms in Engineering Systems: Innovations and Applications*, 1997, pp. 168–173.

[107] L. M. Smith, "Decomposition of FIR digital filters for realization via the cascade connection of subfilters," *IEEE Trans. Signal Processing*, vol. 46, no. 6, pp. 1681–1684, June 1998.

[108] C.-L. Chen and A. N. Willson Jr., "High order Σ-Δ modulation encoding for design of multiplierless FIR filters," *Electron. Lett.*, vol. 34, no. 24, pp. 2298–2300, 1998.

[109] W.-S. Lu, A. Antoniou, and S. Saab, "Sequential design of FIR digital filters for low-power DSP applications," in *Conference Record of the Thirty-First Asilomar Conference on Signals, Systems & Computers*, 1998, vol. 1, pp. 701–704.

[110] R. Păsko, P. Schaumont, V. Derudder, S. Vernalde, and D. Ďuračková, "A new algorithm for elimination of common subexpressions," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 1, pp. 58–68, 1999.

[111] M. Chen, J. Jon, and H.-M. Lin, "An efficient algorithm for the multiple constant multiplication problem," in *International Symposium on VLSI Technology, Systems, and Applications*, 1999, pp. 119–122.

[112] Y. C. Lim, R. Yang, D. Li, and J. Song, "Signed power-of-two term allocation scheme for the design of digital filters," *IEEE Trans. Circuits Syst. II*, vol. 46, no. 5, pp. 577–584, May 1999.

[113] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Prentice Hall, Englewood Cliffs, NJ, 1989.

[114] A. G. Dempster and M. D. Macleod, "General algorithms for reduced-adder integer multiplier design," *Electron. Lett.*, vol. 31, pp. 1800–1802, Oct. 1995.

[115] R. I. Hartley, "Subexpression sharing in filters using canonic signed digit multipliers," *IEEE Trans. Circuits Syst. II*, vol. 43, no. 10, pp. 677–688, Oct. 1996.

[116] M. Potkonjak, M. B. Srivastava, and A. P. Chandrakasan, "Multiple constant multiplications: Efficient and versatile framework and algorithms for exploring common subexpression elimination," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 15, no. 2, pp. 151–165, Feb. 1996.

[117] A. Matsuura, M. Yukishita, and A. Nagoya, "A hierarchical clustering method for the multiple constant multiplication problem," *IEICE Trans. Fundamentals*, vol. E80–A, no. 10, pp. 1767–1773, oct 1997.