Spectral Design of Weighted Median Filters: A General Iterative Approach

Sebastian Hoyos, *Member, IEEE*, Jan Bacca, *Student Member, IEEE*, and Gonzalo R. Arce, *Fellow, IEEE*

***Abstract*—A new design strategy for weighted median (WM) filters admitting real and complex valued weights is presented. The algorithms are derived from Mallows theory for nonlinear selection type smoothers, which states that the closest linear filter to a selection type smoother in the mean square error sense is the one having as coefficients the sample selection probabilities (SSPs) of the smoother. The new design method overcomes the severe limitations of previous approaches that require the construction of high order polynomial functions and high dimensional matrices. As such, previous approaches could only provide solutions for filters of very small sizes. The proposed method is based on a new closed-form function used to derive the SSPs of any WM smoother. This function allows for an iterative approach to WM filter design from the spectral profile of a linear filter. This method is initially applied to solve the median filter design problem in the real domain, and then, it is extended to the complex domain. The final optimization algorithm allows the design of robust weighted median filters of arbitrary size based on linear filters having arbitrary spectral characteristics.**

***Index Terms*—Mallows theory, median filters, nonlinear filters, robust signal processing, sample selection probabilities.**

I. Introduction

MALLOWS theory [1] provides a mapping between non linear smoothing functions and linear filters based on the mean square error (MSE) criteria. Mallows established that the linear filter whose output is the closest in the MSE sense to the output of a selection type nonlinear smoother is the fintie impulse response (FIR) whose coefficients are the sample selection probabilities (SSPs) of the smoother. This theory allows us to analyze some characteristics of nonlinear filters based on the characteristics of the corresponding linear filter with the SSPs as coefficients. In particular, we are interested in designing a nonlinear filter based on some frequency response requirements. In order to do that, the spectrum of a nonlinear smoother is defined as the spectral response of the linear filter whose coefficients are given by the SSPs of the smoother. Consequently, it is possible to design a nonlinear filter with a desired frequency response, using the Mallows relationship and a linear filter with that frequency response. Weighted median (WM) filters, being selection type smoothers, are encompassed by Mallows theory.

Manuscript received October 8, 2002; revised October 24, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ioan Tabus.

S. Hoyos is with the Berkeley Wireless Research Center, Electrical Enginerering and Computer Sciences, University of California, Berkeley, CA 94720 USA (e-mail: hoyos@eecs.berkeley.edu).

J. Baca and G. R. Arce are with the Department of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716 USA (e-mail: baccarod@ee.udel.edu; arce@eecis.udel.edu).

Digital Object Identifier 10.1109/TSP.2004.842206

The spectral design of WM filters can be thought of as the dual of Mallows theory. In this case, rather than seeking the FIR filter coefficients corresponding to a given set of WM filter weights, we are interested in determining the WM filter coefficients that lead to an output close in the MSE sense to the output of a linear filter with the desired spectral characteristics used as reference. In order to solve this problem, it is necessary to develop a fast and general method to find the SSPs of WM smoothers.

Some steps to the definition of a general closed-form strategy to find the SSPs of a WM smoother in the integer domain have been developed in [2], [3] as an extension of the theory of Mallows. They are based on counting principles and generating function representations. The principal characteristic of these strategies is that they obtain the sample selection probabilities through an algorithm that constructs high order polynomial functions or high dimension matrices. Since the polynomial functions can be constructed only when integer valued median weights are used, they require the scaling of partial results found through the process. The generation of these functions and matrices becomes cumbersome when the filter being transformed exceeds a few taps. Additionally, the scaling of the intermediate values can lead to handling very large numbers, making these algorithms impractical due to their computational complexity. As a result, the method in [3] is restricted to the design of filters of small length, i.e., less than nine.

As an alternative to the computational complexity of these algorithms, a new method for the calculation of the SSPs of a given WM smoother is presented. Unlike the previous approaches, this method allows positive real valued weights, making unnecessary the scaling of the weights obtained through the process that finds the median filter weights from a desired set of linear coefficients, leading to a very fast iterative optimization process. Additionally, this design method finds an easy extension to the complex domain by using the recently introduced WM filters admitting complex weights [4].

The organization of the paper is as follows. Section II provides a formal statement of the Mallows Theorem. Section III presents the definitions of WM filters admitting real and complex valued weights. In Section IV, a new closed-form derivation of the sample selection probabilities associated with a WM smoother is presented. Section V uses the SSP's closed-form solution to obtain a steepest descent algorithm that allows to find the WM filter closest in the MSE sense to a given linear filter. Section VI extends the design method to real and complex valued WM filters. Simulations are presented in Section VII and Section VIII is devoted to the conclusions.

II. Mallows Theorem

Mallows focused on the analysis of the smoothing of a random sequence $\mathbf{X}$ by a nonlinear function $\mathbf{S}$ and how this process can be approximated by a well-defined linear smoothing function, as stated on the following theorem:

Theorem 1 (Mallows [1]): Given a nonlinear smoothing function $\mathbf{S}$ operating on a random sequence $\mathbf{X}=\mathbf{Y}+\mathbf{Z}$, where $\mathbf{Y}$ is a zero mean Gaussian sequence, and $\mathbf{Z}$ is independent of $\mathbf{Y}$, we have that if $\mathbf{S}$ is stationary, location invariant, and centered (i.e., $\mathbf{S}(0)=0$), it depends on a finite number of values of $\mathbf{X}$ and $\mathrm{Var}(\mathbf{S}(\mathbf{X}))<\infty$. There exist a unique linear function $\mathbf{S^L}$ such that the MSE function

$$\mathbf{E}\left\{\left(\mathbf{S}(\mathbf{X})-\mathbf{S^L}(\mathbf{X})\right)^2\right\} \tag{1}$$

is minimized. The function $\mathbf{S^L}$ is the closest linear function to the nonlinear smoothing function $\mathbf{S}$ or its *linear part*.

In particular, median smoothers have all the characteristics required for this theorem , and as a consequence, they can be approximated by a linear function. Median smoothers are also selection type and, referring again to Mallows' theory, there is an important corollary of the previous theorem that applies to selection-type smoothers whose output is identical to one of their input samples:

Corollary 1: [1] If $\mathbf{S}$ is a selection type smoother, the coefficients of $\mathbf{S^L}$ are the sample selection probabilities of the smoother.

The sample selection probabilities are defined next for a WM smoother described by the weight vector $\mathbf{W}=\langle W_1,W_2,\cdots,W_N\rangle$ and a vector of independent and identically distributed samples $\mathbf{X}=(X_1,X_2,\cdots,X_N)$.

Definition 1: The Sample Selection Probabilities of a WM smoother $\mathbf{W}$ are the set of numbers p_j defined by

$$p_j=P\left(X_j=\mathrm{MEDIAN}[W_1\diamond X_1,\cdots,W_N\diamond X_N]\right) \tag{2}$$

Thus, p_j is the probability that the output of a WM filter is equal to the jth input sample.

Mallows' results provide a link between the linear and nonlinear domains that allows the approximation of a WM smoother by its *linear part*.

III. Median Filters Admitting Real and Complex Valued Weights

In order to propose a design algorithm for WMs, a good understanding of the different WM filter structures in the real and complex domains is necessary. Their definitions are summarized as follows.

Given a set of samples: $X_1,X_2,\cdots,X_N$, the output of the WM smoother characterized by the set of weights $W_1,W_2,\cdots,W_N$ is defined as

$$\hat{\beta}=\mathrm{MEDIAN}[W_1\diamond X_1,W_2\diamond X_2,\cdots,W_N\diamond X_N] \tag{3}$$

where $\diamond$ represents the replication operator: $W_i\diamond X_i=\overbrace{X_i,X_i,\cdots,X_i}^{W_i \text{ times}}$, and $W_i\geq 0$. In general, the WM can be computed without replicating the sample data according to the corresponding weights, as this increases the computational complexity. A more efficient method to find the WM is shown next, which not only is attractive from a computational perspective, but it also admits positive real-valued weights.

1) Calculate the threshold $T_0=(1/2)\sum_{i=1}^{N}W_i$.
2) Sort the samples in the observation vector $\mathbf{X}(n)$.
3) Sum the concomitant weights[1] of the sorted samples beginning with the maximum sample and continuing down in order.
4) The output is the sample whose weight causes the sum to become $\geq T_0$.

Under Mallows theory, the linear filter closest to this smoother is defined as $\mathbf{h}=[h_1,h_2,\cdots,h_N]$, where $h_i=p_i=P(\hat{\beta}=X_i)$ is the probability of the ith sample being chosen as the smoother's output. Since the coefficients h_i represent probabilities, they all must be non-negative, leading to linear filters with lowpass characteristics. WMs can be generalized so as to synthesize bandpass, bandstop, and highpass operations, through the use of negative weights as in the structure introduced in [5]. This structure has been recently extended to admit complex valued weights in [4].

For real-valued weights, the output of the WM filter characterized by the weights $W_1,W_2,\cdots,W_N$ is defined as

$$Y=\mathrm{MEDIAN}\left(|W_i|\diamond \mathrm{sgn}(W_i)X_i|_{i=1}^{N}\right) \tag{4}$$

where the signs of the weights are coupled to the corresponding samples, and the weight magnitudes represent the new median weights.

For complex-valued weights and samples, a simple definition of the complex WM filter is the marginal phase coupled WM [4]

$$\begin{aligned}Y=&\,\mathrm{MEDIAN}\left(|W_1|\Diamond\mathrm{Re}\{e^{-j\theta_1}X_1\}|_{i=1}^{N}\right)\\&+j\mathrm{MEDIAN}\left(|W_1|\Diamond\mathrm{Im}\{e^{-j\theta_1}X_1\}|_{i=1}^{N}\right).\end{aligned} \tag{5}$$

Here, to calculate the output of the filter, the phase of each weight is first coupled to the corresponding sample. The real and imaginary parts of the phase coupled samples are then fed separately to a WM operator, where the weights are the magnitudes of the original complex weights. The results of these operations constitute the real and imaginary parts of the final filter output, respectively.

The second definition of the complex median (the real-imaginary coupled complex WM) calculates the vectors $\mathbf{W}_R=\mathrm{Re}\{\mathbf{W}\}$, $\mathbf{W}_I=\mathrm{Im}\{\mathbf{W}\}$, $\mathbf{X}_I=\mathrm{Im}\{\mathbf{X}\}$, $\mathbf{X}_R=\mathrm{Re}\{\mathbf{X}\}$, to define $\mathbf{W}_{RI}^T=[\mathbf{W}_R^T|\mathbf{W}_I^T]$, $\mathbf{X}_{RI}^T=[\mathbf{X}_R^T|\mathbf{X}_I^T]$, $\mathbf{X}_{IR-}^T=[\mathbf{X}_I^T|-\mathbf{X}_R^T]$ and finally calculates the median as

$$\begin{aligned}Y=&\,\mathrm{MEDIAN}\left(\mathbf{W}_{RI}^T\diamond\mathbf{X}_{RI}\right)\\&+j\mathrm{MEDIAN}\left(\mathbf{W}_{RI}^T\diamond\mathbf{X}_{IR-}\right)\\=&\,\mathrm{MEDIAN}\left(|W_{R_i}|\Diamond\mathrm{sgn}\left(W_{R_i}\right)X_{R_i}|_{i=1}^{N},\right.\\&\qquad\left.\times|W_{I_i}|\Diamond\mathrm{sgn}\left(W_{I_i}\right)X_{I_i}|_{i=1}^{N}\right)\\&+j\mathrm{MEDIAN}\left(|W_{R_i}|\Diamond\mathrm{sgn}\left(W_{R_i}\right)X_{I_i}|_{i=1}^{N}\right.\\&\qquad\left.\times|W_{I_i}|\Diamond-\mathrm{sgn}\left(W_{I_i}\right)X_{R_i}|_{i=1}^{N}\right).\end{aligned} \tag{6}$$

[1]Represent the input samples and their corresponding weights as pairs of the form (X_i,W_i). If the pairs are ordered by their X variates, then the value of W associated with $X_{(m)}$, denoted by $W_{(m)}$, is referred to as the *concomitant of the m th-order statistic* [6].

The definitions of the WM filters in (4)–(6) share an important property. In order to calculate their outputs, the weights and the input samples are submitted to a series of transformations that result in the calculation of WMs having positive valued weights acting on a modified set of real-valued samples. Thus, the final calculation of the output of these filters requires a number of WM smoothing operations that can be properly designed by using Mallows' theory.

IV. Calculation of Sample Selection Probabilities for WM Smoothers

Given a WM smoother defined by the weights vector $\mathbf{W} = (W_1, W_2, \cdots, W_N)$, which is applied to the set of independent and identically distributed samples $\mathbf{X} = (X_1, X_2, \cdots, X_N)$, our first goal is to find a general closed form expression for the probability that the jth sample is chosen as the output of the WM filter, that is, to find the value $p_j = P(\hat{\beta} = X_j)$. Previous works in this area have calculated the SSPs with the aid of complicated polynomial function generators and/or high order matrices that did not allow us to write them as a simple, closed-form function. The development of a closed-form solution to this problem is described below.

The jth sample in the input vector can be ranked in N different positions in its order statistics. Since the samples are independent and identically distributed, all order statistics are equally likely to hold the sample X_j. Notice that this sample has a different probability of being the output of the median depending on where it lies in the set of ordered input samples. The final value of p_j is found as the sum of the probabilities of the sample X_j being the median for each one of the order statistics.

$$\begin{aligned} p_j &= \sum_{i=1}^{N} P\left(X_{(i)}=X_j\right) P\left(\hat{\beta}=X_{(i)}|X_{(i)}=X_j\right) \\ &= \frac{1}{N}\sum_{i=1}^{N} P\left(\hat{\beta}=X_{(i)}|X_{(i)}=X_j\right) = \frac{1}{N}\sum_{i=1}^{N} \frac{K_{ij}}{\binom{N-1}{i-1}}. \end{aligned} \tag{7}$$

The result in (7) can be explained as follows. After the sample X_j has been ranked in the ith-order statistic, there are $N-1$ samples left to occupy the remaining $N-1$-order statistics: $i-1$ before $X_j = X_{(i)}$ and $N-i$ after it. The total number of nonordered ways to distribute the remaining samples between the remaining order statistics is then equal to the number of ways in which we can distribute the set of $N-1$ samples in two subsets of $i-1$ and $N-i$ samples, leading to the denominator $\binom{N-1}{i-1}$ in (7). The order of the samples in each one of this subsets is not important since, as it will be shown shortly, only the sum of the associated weights is relevant. The term K_{ij} represents how many of these orderings will result in the output of the median being the sample X_j while it is ranked in the ith-order statistic, i.e, the number of times that $\hat{\beta} = X_j = X_{(i)}$. K_{ij} is found as the number of subsets of $N-i$ elements of the vector $\mathbf{W}$ satisfying

$$\sum_{m=i+1}^{N} W_{(m)} < T_0 \tag{8}$$

$$\sum_{m=i}^{N} W_{(m)} \geq T_0 \tag{9}$$

where $T_0 = (1/2)\sum_{m=1}^{N} W_{(m)}$, and $W_{(m)}$ is the concomitant weight associated with the mth-order statistic of the input vector. These are necessary and sufficient conditions for $X_{(i)}$ to be the median of the sample set according to [7], [8], and the method for the calculation of the WM shown in Section III.

Conditions (8) and (9) can be combined as

$$T_0 - W_j \leq \sum_{m=i+1}^{N} W_{(m)} < T_0 \tag{10}$$

where $W_{(i)}$ has been replaced by W_j since it is assumed that the jth sample of the vector is the ith-order statistic. In order to count the number of sets satisfying (10), a product of two step functions is used as follows: When the value $A = \sum_{m=i+1}^{N} W_{(m)}$ satisfies $T_0 - W_j \leq A < T_0$, the function

$$u\left(A - (T_0 - W_j)\right) u\left(T_0^- - A\right) \tag{11}$$

will be equal to one. Here, T_0^- represents a value approaching T_0 from the left in the real line, and u is the unitary step function defined as $u(x) = 1$ if $x \geq 0$ and 0 otherwise. On the other hand, (11) will be equal to zero if A does not satisfy the inequalities. Letting $T_1 = T_0 - W_j$ and adding the function in (11) over all the possible subsets of $i-1$ elements of $\mathbf{W}$ excluding W_j, the result is

$$K_{ij} = \sum_{\substack{m_1=1 \\ m_1 \neq j}}^{N} \sum_{\substack{m_2=m_1+1 \\ m_2 \neq j}}^{N} \cdots \sum_{\substack{m_s=m_{s-1}+1 \\ m_s \neq j}}^{N} u\left(A - T_1\right) u(T_0^- - A) \tag{12}$$

where $A = W_{m_1} + W_{m_2} + \cdots + W_{m_s}$, and $s = N - i$. The SSP vector is given by $\mathbf{P}(\mathbf{W}) = [p_1, p_2, \cdots, p_n]$, where p_j is defined as

$$p_j = \frac{1}{N}\sum_{i=1}^{N} \frac{K_{ij}}{\binom{N-1}{i-1}}. \tag{13}$$

This newly defined function calculates the sample selection probabilities of any WM smoother, that is, it leads to the linear filter closest to a given WM smoother in the mean square error sense. As it was stated before, the domain of this function is not restricted to integer weights. Instead, it can be evaluated for real-valued weights. Besides, the function does not require the construction of extra matrices or polynomial functions, making its calculation more straightforward.

The equation obtained in (13) calculates the linear filter coefficients (SSPs) as a function of the median smoother weights. Since the objective of this paper is to design a WM filter from a given FIR filter, this function should be inverted. However, this nonlinear function is not invertible. Before studying other alternatives to solve this problem, certain properties of the WM filters should be taken into account.

It has been demonstrated in [9] and [10] that the WM smoothers of a given window size can be divided into a finite number of *classes*. All of the smoothers in a class produce the same output when they are fed with the same set of input samples. It has also been shown that each class contains at least one integer-valued WM filter such that the sum of its components is odd. Among these filters, the one with the minimum sum of components is called the representative of the class. Table I shows the representatives of the different classes of WM smoothers available for window sizes from one to five.

TABLE I
WEIGHTED MEDIAN VECTORS AND THEIR CORRESPONDING SSPs FOR WINDOW SIZES 1 TO 5

N	WM	SSP
1	[1]	[1]
2	-	-
3	[111]	$[\frac{1}{3}\ \frac{1}{3}\ \frac{1}{3}]$
4	[2111]	$[\frac{1}{2}\ \frac{1}{6}\ \frac{1}{6}\ \frac{1}{6}]$
5	[11111]	$[\frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}\ \frac{1}{5}]$
	[22111]	$[\frac{3}{10}\ \frac{3}{10}\ \frac{2}{15}\ \frac{2}{15}\ \frac{2}{15}]$
	[31111]	$[\frac{3}{5}\ \frac{1}{10}\ \frac{1}{10}\ \frac{1}{10}\ \frac{1}{10}]$
	[32211]	$[\frac{2}{5}\ \frac{7}{30}\ \frac{7}{30}\ \frac{1}{15}\ \frac{1}{15}]$

Weighted medians obtained as the permutation of the ones shown in Table I are also representatives of other classes. Additionally, a representative of a class can be padded with zeros to form a representative of another class with larger window size. For example, for window size three, we can construct four different WM vectors [1, 1, 1] and the three permutations of [1, 0, 0].

It is also known that each WM filter has a corresponding equivalent self dual linearly separable positive Boolean function (PBF) [7], [9] and vice versa. This means that the number of different WMs of size N is the same as the number of self dual linearly separable PBF's of N variables. Equivalent WM vectors will correspond to the same PBF, and they will also have the same vector of SSPs.

Another important point is that not all integer-valued vectors represent valid WMs. Only the ones in which the sum of their components is odd (and the ones belonging to the same class) can be considered as valid WMs. The nonvalid vectors represent the so called Asymmetric WMs (AWMs). These smoothers have some undesired characteristics: They result in biased estimators of location, and they are able to filter impulses of only one polarity, either positive or negative [11].

To illustrate the consequences of all these properties, the following example shows the case for filters of length three. Here, the number of WM filters to be analyzed is reduced to include only normalized filters. These filters are included in the two-dimensional simplex $W_1+W_2+W_3=1$. According to (10) and Table I, there are four different classes of WMs for this window size. They will occupy regions in the simplex that are limited by lines of the form $W_i+W_j=(1/2)=T_0$, where $i,j \in \{1,2,3\}$, $i \neq j$. Fig. 1(a) shows the simplex with the four regions corresponding to the four classes of WMs and the representative of each class. The limits of the regions represent AWMs, and, as a consequence, they cannot be represented by a valid WM vector. Taking all this into account, we proceed to formulate the optimization process for the calculation of the WM closest to a given linear smoother in the mean square error sense.

The median weights are found to minimize the mean square error cost function given by

$$J(\mathbf{W}) = \|\mathbf{P}(\mathbf{W}) - \mathbf{h}\|^2 = \sum_{j=1}^{N} (p_j(\mathbf{W}) - h_j)^2 \qquad (14)$$

where $\mathbf{h}$ is a normalized linear smoother. Since the number of SSP vectors $\mathbf{P}(\mathbf{W})$ for a given window size is finite, a valid option to solve this problem is to list all its possible values and find between them the one that minimizes the error measure $J(\mathbf{W})$. This will lead to a division of the space of linear filters of window size N in regions: one for each SSP vector. Each one of this regions contains a certain SSP vector and all the other linear filter vectors that are closer in Euclidean distance to it than to any other SSP vector. This situation can be viewed as a quantization of the space of normalized linear smoothers, where a one-to-one correspondence can be stated between each one of the valid SSP vectors of size N and the quantization regions in the space of normalized linear smoothers. Fig. 1(b) shows the case for window size three.

All vectors in the same WM class region are mapped into the linear domain as a single point: the corresponding SSP vector. Since all WM in a class are equivalent, the associated linear filter to all of them will be the same. Therefore, there is a unique solution to the problem of finding the linear filter closest in the MSE sense to a given WM filter. On the other hand, the reverse problem, i.e., finding the WM filter closest to a given linear filter, has an infinite number of solutions. Since the linear filter domain is quantized, all the vectors in a quantization region will be associated with the SSP vector contained in the region that will be mapped into the WM domain as a class of WMs instead of as a single WM smoother. Any set of weights in that class will result in the same value of the distance measure $J(\mathbf{W})$. That is, the mapping in this case is established between a quantization region in the linear domain and a class region in the WM domain in such a way that any point in the latter can be associated with a given vector in the former. Fig. 1 illustrates the mapping between quantization regions in the linear domain and class regions in the WM domain for window size three.

This figure also shows that in some cases this mapping is trivial since the associated WM filter and linear filter could have the same set of weights. However, the trivial mappings are only special cases. For example, if a WM equivalent to the linear filter $\mathbf{h} = [1/6,\ 1/2,\ 1/6,\ 1/6]$ needs to be found, the same values cannot be used as median weights since they represent an AWM. The incidence of such cases where the associated WM filter and linear filter vector are never the same increases with larger window size. This justifies the need for an algorithm to find the right WM approximation for a given linear filter.

The procedure to transform a linear filter into its associated WM reduces to finding the region in the linear space where it belongs, finding the corresponding SSP vector, and then finding a corresponding WM vector. This is possible only if all the valid WM vectors and their corresponding SSPs for a certain window size are known. The problem of finding all the different WM vectors of size N has been subject to intensive research. However, a general closed-form solution that allows the generation of the list of PBFs, SSP vectors, or WM vectors has yet to be found. Partial solutions for the problem have been found for window sizes up to eight [9]. Even if such a general form existed, the number of possibilities grows rapidly with the window size, and the problem becomes cumbersome. For example, the number of different WMs grows from 2470 for window size eight to 175 428 for window size nine. There is no certainty about the number of vectors for window size ten and up.

Having all the possible sets of median weights for a certain window size will assure that the right solution of the problem

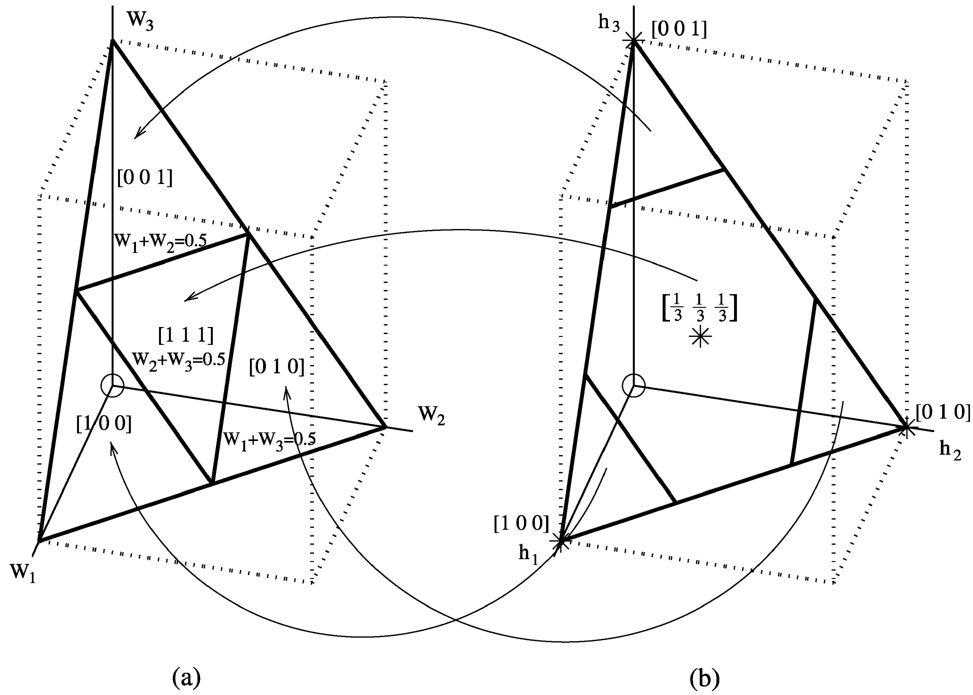


Fig. 1. Illustrative example showing the mapping between the WM class regions and the linear filter regions. (a) Simplex containing the WM vectors for window size three. The simplex is divided in four regions, and the representative of each region is also indicated. (b) Correspondence between linear filters and SSP vectors of window size three. The SSP vectors are represented by "$*$."

can be found. As it was indicated before, this option becomes unmanageable for large window sizes. This does not disqualify the method for small filters, but a faster, easier alternative is necessary to handle larger filter lengths. In the following section, an optimization algorithm for the function $J(\mathbf{W})$ is presented.

V. General Iterative Solution

The optimization process is carried out with a gradient-based algorithm, and a series of approximations to be described. The first step is to find the gradient of the cost function in (14)

$$\nabla \mathbf{J}(\mathbf{W}) = \begin{pmatrix} \frac{\partial}{\partial W_1}\mathbf{J}(\mathbf{W}) \\ \frac{\partial}{\partial W_2}\mathbf{J}(\mathbf{W}) \\ \vdots \\ \frac{\partial}{\partial W_N}\mathbf{J}(\mathbf{W}) \end{pmatrix} \tag{15}$$

where each of the terms in (15) is given by

$$\begin{aligned} \nabla_l \mathbf{J}(W) &= \frac{\partial}{\partial W_l}\mathbf{J}(W) = \frac{\partial}{\partial W_l}\left\|\mathbf{P}(\mathbf{W}) - \mathbf{h}\right\|^2 \\ &= \sum_{j=1}^{N} \frac{\partial}{\partial W_l}\left(p_j(\mathbf{W}) - h_j\right)^2 \\ &= \sum_{j=1}^{N} 2\left(p_j(\mathbf{W}) - h_j\right)\frac{\partial}{\partial W_l}p_j(W). \end{aligned} \tag{16}$$

The derivative of $p_j(W)$ is

$$\begin{aligned} \frac{\partial p_j(W)}{\partial W_l} &= \frac{\partial}{\partial W_l}\frac{1}{N}\sum_{i=1}^{N}\frac{K_{ij}}{\binom{N-1}{i-1}} \\ &= \frac{1}{N}\sum_{i=1}^{N}\frac{\frac{\partial K_{ij}}{\partial W_l}}{\binom{N-1}{i-1}}. \end{aligned} \tag{17}$$

The term K_{ij} given in (12) is not differentiable due to the discontinuities of the step functions. To overcome this situation, $u(x)$ is approximated by a smooth differentiable function $1/2(\tanh(sx) + 1)$, where s is a smoothing parameter that is assigned a value of 1 for simplicity. We choose this approximation as it has been successfully used in similar optimization procedures [5]. The last derivative in (17) can be computed as

$$\frac{\partial K_{ij}}{\partial W_l} = \frac{1}{4}\sum_{\substack{m_1=1\\ m_1\neq j}}^{N}\ \sum_{\substack{m_2=m_1+1\\ m_2\neq j}}^{N}\cdots\sum_{\substack{m_s=m_{s-1}+1\\ m_s\neq j}}^{N}\frac{\partial B}{\partial W_l} \tag{18}$$

where

$$B = \left(\tanh(A - T_1) + 1\right)\left(\tanh(T_0^- - A) + 1\right) \tag{19}$$

$$\begin{aligned} \frac{\partial B}{\partial W_l} &= C_1(W_l)\operatorname{sech}^2(A - T_1)\left(\tanh(T_0^- - A) + 1\right) \\ &\quad - C_2(W_l)(\tanh(A - T_1) + 1)\operatorname{sech}^2\left(T_0^- - A\right) \end{aligned} \tag{20}$$

where the coefficients $C_1(W_l)$ and $C_2(W_l)$ are defined by

$$\begin{aligned} C_1(W_l) &= \begin{cases} \frac{1}{2} & l = j \\ \frac{1}{2} & \exists\, i \ \mathbf{s.}t\ m_i = l \\ -\frac{1}{2} & \text{else} \end{cases} \\ C_2(W_l) &= \begin{cases} -\frac{1}{2} & \exists\, i \ \mathbf{s.}t\ m_i = l \\ \frac{1}{2} & \text{else.} \end{cases} \end{aligned} \tag{21}$$

The recursive equation for each of the median weights is

$$\begin{aligned} W_l(n+1) &= W_l(n) + \mu\left(-\nabla_l \mathbf{J}(\mathbf{W})\right) \\ &= W_l(n) + \mu\left(-\frac{\partial}{\partial W_l}\mathbf{J}(\mathbf{W})\right). \end{aligned} \tag{22}$$

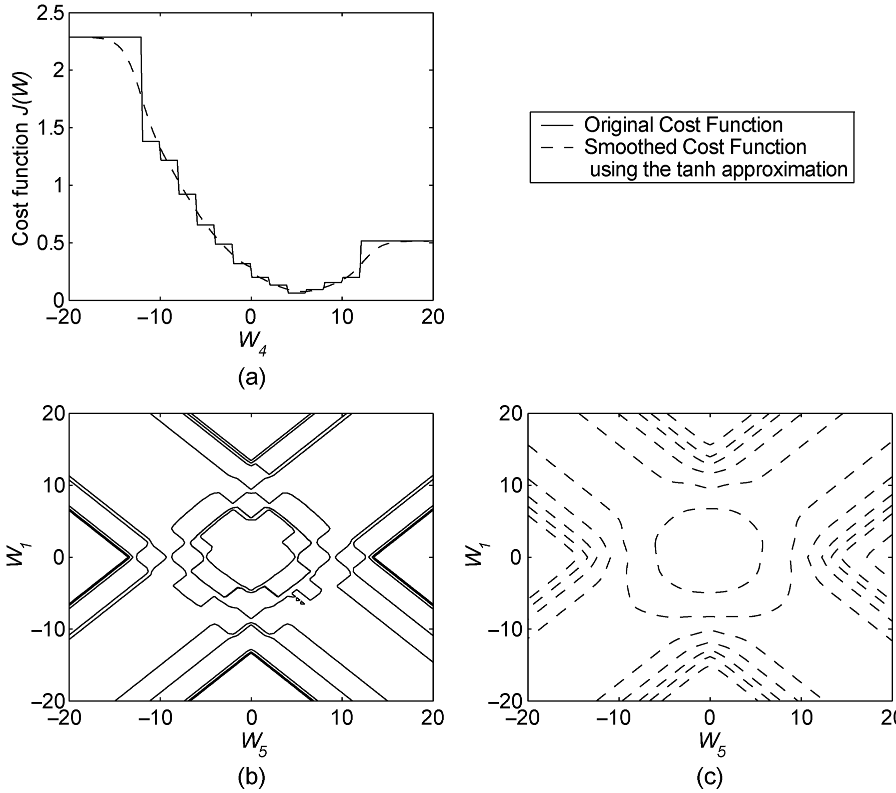


Fig. 2. Cost functions of *Mallows Iterative Algorithm* with input FIR filter $\mathrm{h} = [-0.0078\ 0.0645\ 0.4433\ 0.4433\ 0.0645\ -0.0078]$. (a) Cost functions with respect to one weight for both the original cost function (solid line) and the approximated cost function that uses the hyperbolic functions (dashed-line). (b) Contours of the cost functions with respect to two weights for the original cost function. (c) Contours of the cost functions with respect to two weights for the approximated cost function that uses the hyperbolic functions.

The iterative algorithm shown above approximates linear smoothers by WM smoothers, and it will be referred to as the *Mallows Iterative Algorithm*. A performance comparison between Mallows Iterative Algorithm and the algorithm in [3] used to design median smoothers based on linear filters showed that the former carries on the calculation faster than the latter. Both algorithms are iterative, and in each iteration step, they should find the closest linear smoother to the WM smoother calculated in (23). This WM smoother will be likely real valued (since the gradient of the cost function cannot be guaranteed to be an integer and the step size is usually rational). The method in [3] cannot calculate the SSPs of a real-valued WM, and as a consequence, these intermediate results need to be scaled until all the weights are integer. Mallows Iterative Algorithm does not require this scaling operation since it can work on any positive real valued numbers. The exact number of operations involved in both algorithms is not straightforward due to the high number of factorial operations required in each one of them, and even if we knew the complexity of each algorithm for each iteration, a fair comparison would require the convergence rate (number of iterations needed for convergence) of each algorithm, which is outside the scope of this paper. Experimental results showed that when the algorithm in [3] is used to approximate WM smoothers of size 8, it requires the same time used by Mallows Iterative Algorithm to approximate a filter of 18 taps. The exact amount of time would depend on the speed of the machine used to run the simulations.

For illustrative purposes, the original cost function in (14) and the approximated cost function for a WM filter of size 5 are shown in Fig. 2(a) as a function of W_4 and in Figs. 2(b) and 2(c) as a function of W_1 and W_5. It can be seen that the original cost function is stepwise, and as a consequence, it has an infinite number of local minima. Based on our simulation results, a smooth approximation is obtained when replacing the step functions with hyperbolic tangents, which allows the implementation of a steepest descent algorithm to minimize it. However, no formal proof of the uniqueness of the minimum of the approximated cost function has been found and remains an open mathematical problem. Experimental results show that the steepest descent algorithm converges to the global minimum of this function.

The following section presents the algorithms to use this method for the different classes of WM filters.

VI. Design of Real- and Complex-Valued WM Filters

Equations (4)–(6) show that all the real- and complex-valued WM filter definitions consist of properly modifying the input samples according to the associated weights and then using the magnitude of the weights for the calculation of positive WMs. It was stated in Theorem 1 that a nonlinear function needs to satisfy certain properties in order to be best approximated under the mean squared error sense by a linear filter. Unfortunately, the real- and complex-valued medians do not satisfy the location invariance property. However, Mallows results can be extended to cover medians of types(4)–(6) in the case of a independent, zero mean, Gaussian input sequence.

Theorem 2:

a) If the input series is Gaussian, independent, and zero centered, the coefficients of the linear part of the WM defined in (4) are defined as $h_i = \text{sgn}(W_i)p_i$, where p_i are the SSPs of the WM smoother $|W_i|$.

b) If the real and imaginary parts of the input series are Gaussian, independent, and zero centered, the coefficients of the linear part of the WM defined in (5) are defined as $h_i = e^{-j\theta_i}p_i$, where p_i are the SSPs of the WM smoother $|W_i|$.

c) For an input series as in b), the coefficients of the WM filter defined in (6) are defined as $h_{RI_i} = \text{sgn}(W_{RI_i})p_{RI_i}$, where p_{RI_i} are the SSPs of the WM smoother $|W_{RI_i}|$.

To show a), define $Y_i = \text{sgn}(W_i)X_i$. In this case, the Y_i will have the same distribution as the X_i. As a consequence

$$\mathbf{E}\left\{\left(\text{MEDIAN}(W_i \diamond X_i) - \sum h_i X_i\right)^2\right\} = \mathbf{E}\left\{\left(\text{MEDIAN}\left(|W_i| \diamond Y_i\right) - \sum q_i Y_i\right)^2\right\} \quad (23)$$

where $q_i = h_i/\text{sgn}(W_i)$. From Theorem 1, (24) is minimized when the q_i equal the SSPs of the smoother $|W_i|$, say, p_i. As a consequence

$$q_i = \frac{h_i}{\text{sgn}(W_i)} = p_i \rightarrow h_i = \text{sgn}(W_i)p_i. \quad (24)$$

A similar procedure should be used to prove b). This time, define $Y_i = e^{-j\theta_i}X_i = U_i + jV_i$.

$$\begin{aligned}\mathbf{E}&\left\{\left|\text{MEDIAN}(W_i \diamond X_i) - \sum h_i X_i\right|^2\right\}\\ &= \mathbf{E}\left\{\left|\text{MEDIAN}\left(|W_i| \diamond Y_i\right) - \sum q_i Y_i\right|^2\right\}\\ &= \mathbf{E}\left\{\left(\text{MEDIAN}\left(|W_i| \diamond U_i\right) - \sum b_i U_i + \sum c_i V_i\right)^2\right\}\\ &\quad + \mathbf{E}\left\{\left(\text{MEDIAN}\left(|W_i| \diamond V_i\right) - \sum b_i V_i + \sum c_i U_i\right)^2\right\} \quad (25)\end{aligned}$$

where $q_i = e^{j\theta_i}h_i = b_i + jc_i$. Again from Mallows' theorem, (26) is minimized when $c = 0$ and $b_i = p_i$.

To prove c), it is enough to notice that its implementation equals two realizations of the same real-valued WM filter over two different series. As a consequence

$$\begin{aligned}\mathbf{E}&\left\{\left|\text{MEDIAN}(W_i \diamond X_i) - \sum h_i X_i\right|^2\right\}\\ &= \mathbf{E}\left\{\left(\text{MEDIAN}\left(|W_{RI_i}| \diamond \text{sgn}\left(W_{RI_i}\right) X_{RI_i}\right) - \sum h_{RI_i} X_{RI_i}\right)^2\right\}\\ &\quad + \mathbf{E}\left\{\left(\text{MEDIAN}\left(|W_{RI_i}| \diamond \text{sgn}\left(W_{RI_i}\right) X_{IR_i^-}\right) - \sum h_{RI_i} X_{IR_i^-}\right)^2\right\}. \quad (26)\end{aligned}$$

Again, each one of the factors on the right-hand side is minimized when $h_{RIi} = \text{sgn}(W_{RI_i})p_{RI_i}$, and since both are non-negative, its sum is minimum when each one of them is minimum.

According to this theorem, Mallow's Iterative Algorithm can be used for the design of these filters. This can be accomplished using the following procedures.

A. WM Filter Admitting Real-Valued Weights

- Given the desired frequency characteristics, design the linear FIR filter $\mathbf{h} = (h_1, h_2, \cdots, h_N)$ using one of the traditional design tools for linear filters.
- Decouple the signs of the coefficients to form the vectors $|\mathbf{h}| = (|h_1|, |h_2|, \cdots, |h_N|)$ and $\text{sgn}(\mathbf{h}) = (\text{sgn}(h_1), \text{sgn}(h_2), \cdots, \text{sgn}(h_N))$.
- After normalizing the vector $|\mathbf{h}|$, use the proposed algorithm to find the closest WM filter to it, say, $\mathbf{W}' = (W_1', W_2', \cdots, W_N')$.
- The WM filter will be given by $\mathbf{W} = [\text{sgn}(h_i)W_i'|_{i=1}^N]$

B. Marginal Phase Coupled Complex WM Filter

- Design a linear complex-valued FIR filter $\mathbf{h} = (h_1, h_2, \cdots, h_N)$ given the desired characteristics for the filter.
- Decouple the phases of the coefficients to form the vectors $|\mathbf{h}| = (|h_1|, |h_2|, \cdots, |h_N|)$ and $\mathbf{\Theta}(\mathbf{h}) = (\theta(h_1), \theta(h_2), \cdots, \theta(h_N))$, where $\theta(h_i)$ represents the phase of h_i.
- Normalize the vector $|\mathbf{h}|$, and find the closest WM filter to it using the algorithm developed in the previous section, say, $\mathbf{W}' = (W_1', W_2', \cdots, W_N')$.
- The complex WM filter will be given by $\mathbf{W} = [e^{j\theta(h_i)}W_i'|_{i=1}^N]$

C. Real-Imaginary Coupled Complex WM Filter

- Using any available method, design a linear complex-valued FIR filter $\mathbf{h} = (h_1, h_2, \cdots, h_N)$ having the characteristics required.
- Construct the vector $\mathbf{h_{RI}}$ as follows: $\mathbf{h_{RI}^T} = [\mathbf{h_R^T}|\mathbf{h_I^T}]$, where $\mathbf{h_R}$ and $\mathbf{h_I}$ represent the vectors containing the real and imaginary parts of the components of the vector $\mathbf{h}$.
- Form the vectors $|\mathbf{h_{RI}}| = (|h_{RI_1}|, |h_{RI_2}|, \cdots, |h_{RI_{2N}}|)$ and $\text{sgn}(\mathbf{h_{RI}}) = (\text{sgn}(h_{RI_1}), \text{sgn}(h_{RI_2}), \cdots, \text{sgn}(h_{RI_{2N}}))$ containing the magnitude and sign of the components of $\mathbf{h_{RI}}$, respectively.
- Normalize the vector $|\mathbf{h_{RI}}|$, and find the closest WM filter to it using (23), say, $\mathbf{W_{RI}'} = (W_{RI_1}', W_{RI_2}', \cdots, W_{RI_{2n}}')$.
- Couple the signs of the linear weights with the obtained median weights, that is, calculate $\mathbf{W_{RI}}$ given by $\mathbf{W_{RI}} = [\text{sgn}(h_{RI_i})W_{RI_i}'|_{i=1}^{2N}]$
- To find the WM filter, use $\mathbf{W_{RI}}$ to create $\mathbf{W} = [W_{RI_i} + jW_{RI_{(N+i)}}|_{i=1}^N]$

VII. Simulation Results

Once the design algorithms for the different classes of WM filters have been presented, some examples will show their ac-

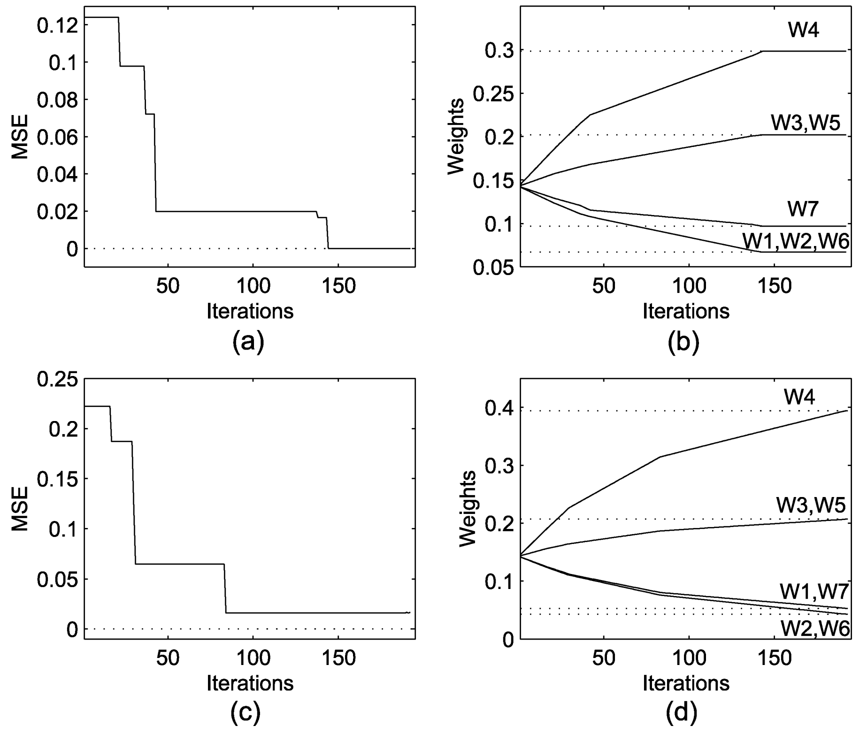


Fig. 3. (a) Learning curve of Mallows adaptive algorithm for an input that is a valid SSP vector. (b) Convergence of the weights for the same input. (c) Learning curve when the input is not a valid SSP vector. (d) Convergence of the weights for this input.

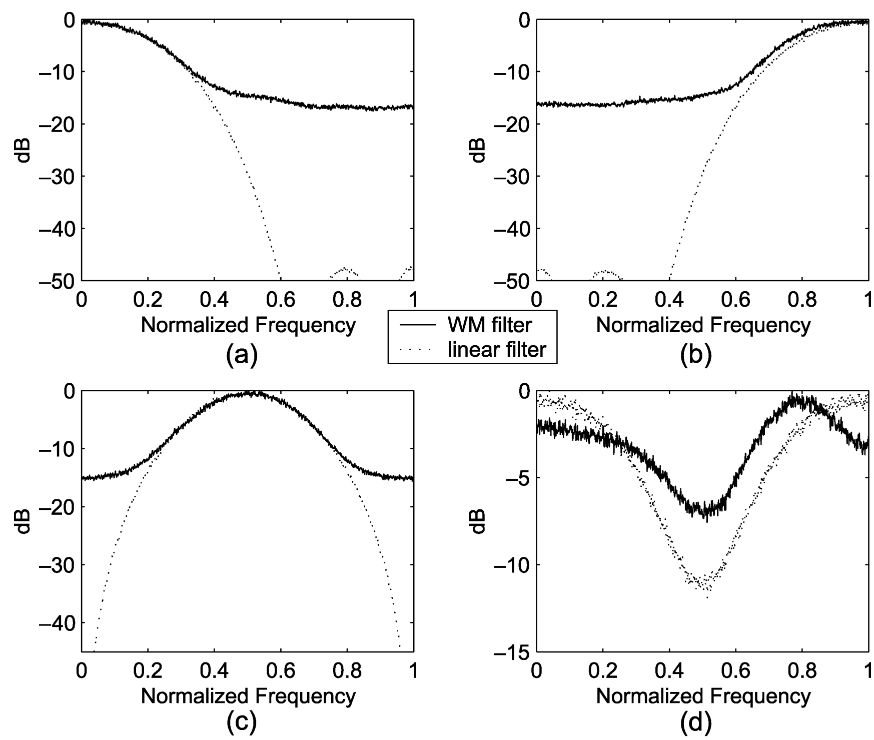


Fig. 4. Approximated frequency response of WM filters designed with Mallows Iterative Algorithm. (a) Lowpass. (b) Highpass. (c) Bandpass. (d) Bandstop.

tual behavior when they are put in practice. The main objective of this paper is to propose a strategy to design WM filters under frequency constraints. Since WM filters lack of a frequency response *per se*, we assigned to them the frequency response of their *linear part*. This assignation is accurate when the input of the filters is a white random sequence, and we use the MSE as a distance measure. It is expected that if all these requirements are met, the output of the WM filter designed with this method

TABLE II
CHARACTERISTICS OF THE FILTERS AND SIGNALS USED FOR THE FREQUENCY SELECTION EXPERIMENT

Filter	Cut-off frequencies	Sinusoidal frequencies	
Low pass	0.25	0.1	0.5
Band pass	0.35 - 0.65	0.1	0.5
High pass	0.75	0.5	0.9
Band stop	0.35 - 0.65	0.1	0.5

TABLE III
AVERAGE MSE OF THE OUTPUT OF THE REAL-VALUED WM FILTERS AND THE LINEAR FILTERS IN PRESENCE OF α-STABLE NOISE

α	Low Pass		Band Pass		High Pass		Band Stop	
	WM	Linear	WM	Linear	WM	Linear	WM	Linear
Clean	0.0211	0	0.0639	0	0.0266	0	0.0169	0
1	0.0835	77.8845	0.1411	121.4348	0.0859	77.8575	0.0935	216.1936
1.5	0.0970	0.4818	0.1352	0.7480	0.0963	0.4760	0.1089	1.3293
2	0.1086	0.0389	0.1264	0.0606	0.1070	0.0391	0.1204	0.1085

will have similar frequency characteristics to the output of the reference linear filter.

A. Convergence of the Iterative Algorithm

Initially, to test the convergence of the steepest descent algorithm, the WM filter obtained through the iterative algorithm and a true WM filter that minimizes the cost function in (14) (obtained by inspection of all the possibilities) are compared. The resultant WM filter from the iterative algorithm should ideally belong to the same class of the true WM filter. This is the case if the linear filter used as reference in the iterative algorithm is a SSP vector. Otherwise, if the reference linear filter is not a SSP vector, the algorithm is not guaranteed to converge to a WM of the same class as the true WM filter. This can be explained by looking at the derivative of the cost function in (16). Ideally, the algorithm will reach a minimum when the derivative equals zero. The second factor in (16) is the derivative of the jth SSP with respect to the lth weight. Since this term is calculated by an approximation, it cannot be guaranteed that it will reach a value of zero. The first factor is the difference between the input vector $\mathbf{h}$ and the sample selection probabilities corresponding to the previous output of the iterative algorithm. This term will never be zero unless the input vector $\mathbf{h}$ is a valid SSP vector. If it is not, the derivative may never be zero, and the adaptive algorithm may not reach the floor error. Instead, it will reach a value as close as possible to the minimum, and it will begin oscillating around it.

The following example illustrates the convergence of the adaptive algorithm when the reference filter is an SSP vector. The coefficients of the lowpass filter are given by $\mathbf{h}_1 = [-(1/35),\ (1/35),\ (8/35),\ (83/210),\ (8/35),\ (1/35),\ -(13/210)]$ with an associated SSP vector given by $|\mathbf{h}_1|$. Fig. 3(a) shows how the learning curve of the adaptive algorithm reaches a minimum when one of the WM filters associated with the SSP vector is found. The final output of the adaptive algorithm is the WM vector $W_1 = [-0.0286,\ 0.0286,\ 0.2286,\ 0.3952,\ 0.2286,\ 0.0286,\ -0.0619]$. Fig. 3(b) shows the convergence of the weights for this case. It can be seen that once the algorithm has reached the floor error, the weights maintain the values obtained, even if the adaptive algorithm is left running.

To illustrate the case when the reference linear filter is not a SSP vector, a lowpass linear filter with coefficients $\mathbf{h}_2 = [-0.0084,\ 0.0000,\ 0.2434,\ 0.4964,\ 0.2434,\ 0.0000,\ -0.0084]$ is used as the reference in the adaptive algorithm. The associated SSP vector to this linear filter is the one used in the previous example. The adaptive algorithm, whose learning curve is illustrated in Fig. 3(c), leads to the WM filter $W_2 = [-0.0429,\ 0.0429,\ 0.2095,\ 0.4095,\ 0.2095,\ 0.0429,\ -0.0429]$. This WM filter does not belong to the same class as W_1, demonstrating the potential penalty associated with using the adaptive algorithm with a reference that is not an SSP vector. However, since the adaptive algorithm tries to minimize the cost function in (14), the output of the WM filter obtained has a frequency response that resembles the one of the linear filter. Fig. 3(d) shows how, in this case, the weights will keep on changing since the algorithm does not reach a minimum.

TABLE IV
CHARACTERISTICS OF THE FILTERS AND SIGNALS USED FOR THE FREQUENCY SELECTION EXPERIMENT IN THE COMPLEX DOMAIN

Filter	Cut-off frequencies	Sinusoidal frequencies	
Low pass	-0.4, 0.7	-0.85	0.15
Band pass	-1 -0.5, 0.3 0.7	-0.75	0.9
High pass	-0.5, 0.8	-0.85	0.15
Band stop	-1 -0.4, 0.2 0.8	-0.75	0.9

B. Real-Valued WMs

In order to test the results obtained designing filters in the real domain using Mallows Iterative Algorithm, four linear FIR filters of 11 taps with lowpass, bandpass, highpass, and bandstop characteristics were designed using (22) with a step size of $\mu = 1$. The filters were submitted to the tests described below.

The frequency characteristics of the output of these filters and the linear filters used as reference were approximated as follows: 50 realizations of 10 000 samples of standard Gaussian noise were fed to the filters, and the spectra of the output was approximated using the Welch method. The results were averaged to obtain the frequency response of the signals shown in Fig. 4. The results obtained will be referred to as the approximated frequency response of the filters (median or linear). The cutoff frequencies of the linear filters used in the experiment are summarized in Table II.

The plots show that the WM filter is able to have any frequency-selection characteristics required. The characteristics of the WM filters and the linear filters are very similar in the passband, whereas the major difference is the lower attenuation provided by the WM in the stopband. The next test will show the importance of this difference when the filters are working with real signals in both noiseless and impulsive noise environments.

The frequency-selection capabilities of the filters, together with their noise rejection potential, were tested as follows: The sum of two sinusoids, with frequencies chosen to be one in the passband and one in the stopband of each filter, was contaminated with α-stable noise with different levels of impulsiveness represented in values of α of 1 (Cauchy noise, highly impulsive), 1.5, and 2 (Gaussian noise, non-impulsive) and a dispersion parameter $\gamma = 0.1$. The frequencies of the sinusoidal signals are

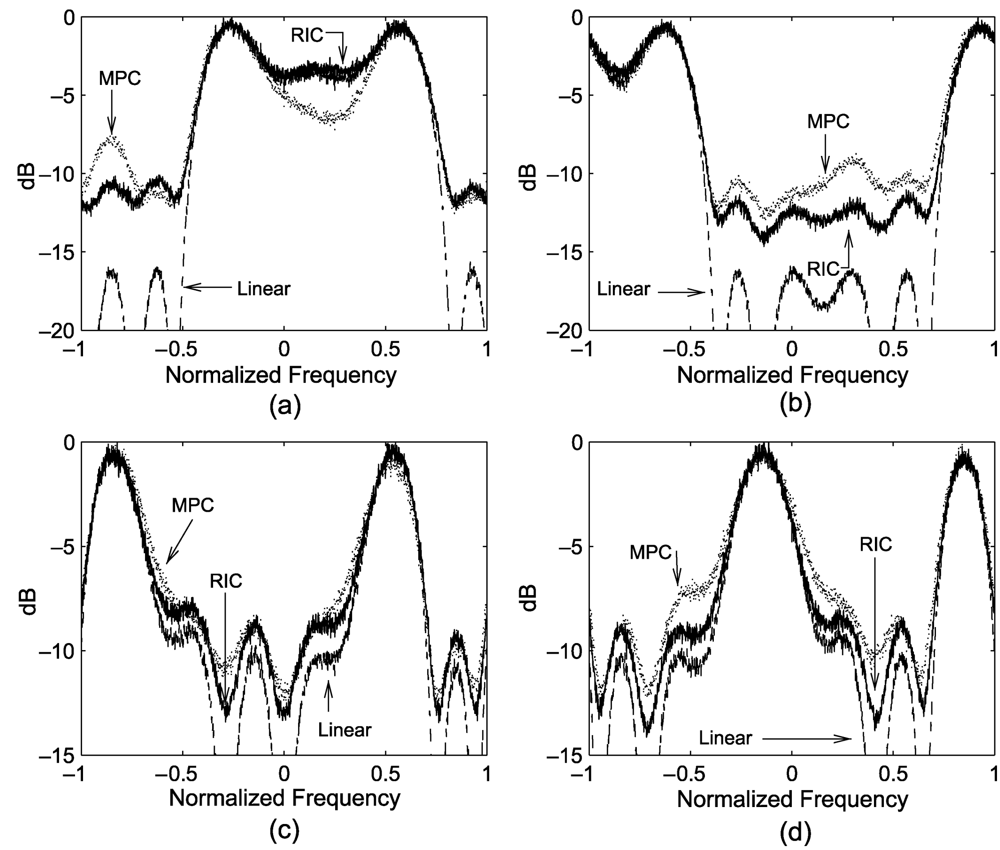


Fig. 5. Approximated frequency response of the complex WM filters designed with Mallows iterative algorithm. (a) Lowpass. (b) Highpass. (c) Bandpass. (d) Bandstop (MPC is the Marginal Phase Coupled Complex WM, and RIC is the real-imaginary coupled CWM).

TABLE V
AVERAGE MSE OF THE OUTPUT OF THE COMPLEX WM FILTERS AND THE LINEAR FILTERS IN THE PRESENCE OF α-STABLE NOISE

α	Low Pass			Band Pass		
	MPC	RIC	Linear	MPC	RIC	Linear
clean	0.5405	0.4052	0	0.2562	0.2009	0
1	0.5608	0.3988	1.4203×10^4	0.3251	0.2476	1.5160×10^4
1.5	0.5512	0.3859	2.0501	0.3181	0.2480	2.1890
2	0.5447	0.3774	0.1103	0.3153	0.2509	0.1184
α	High Pass			Band Stop		
	MPC	RIC	Linear	MPC	RIC	Linear
clean	0.1712	0.2396	0	0.5123	0.4145	0
1	0.2227	0.2938	1.0543×10^4	0.5537	0.4559	1.5151×10^4
1.5	0.2197	0.3031	1.5212	0.5443	0.4555	2.1855
2	0.2230	0.3183	0.0824	0.5427	0.4580	0.1178

shown in Table II. If an ideal linear filter were used to filter the clean signal, the output will be a sinusoidal whose normalized frequency will be the one on the passband of the corresponding filter. This signal is used as a reference to calculate the MSE of the outputs of the WM filters and a linear filter in the presence of the noise. The results obtained when the clean signals are filtered with the WM are also included for illustrative purposes. Table III shows the average MSE of 100 realizations of the filtering of 200 samples of the signal for each case.

C. Complex-Valued WMs

The tests developed in the previous section are repeated here but this time using signals and filters in the complex domain. Instead of sinusoidal signals, the sum of two exponential signals provides the input to the filters. Complex-valued FIR filters with nine taps were designed using the Matlab function cremez, and then, marginal phase-coupled and real-imaginary coupled complex WM filters with the desired characteristics were designed using the Mallows Iterative Algorithm. The characteristics of the filters and the input signals used are shown in Table IV. The approximated frequency response of the filters is shown in Fig. 5, and the values of the average mean square error obtained at the outputs of the filters are shown in Table V.

Fig. 5 shows that the frequency characteristics of the complex WM outputs are very close to the ones of their linear counterparts. The real-imaginary coupled WM filter resembles exactly the characteristics of the linear filter in the passband. The marginal phase-coupled WM filter shows slightly more ripple, yet the approximation is still excellent. The results are confirmed by the MSE values shown in Table V.

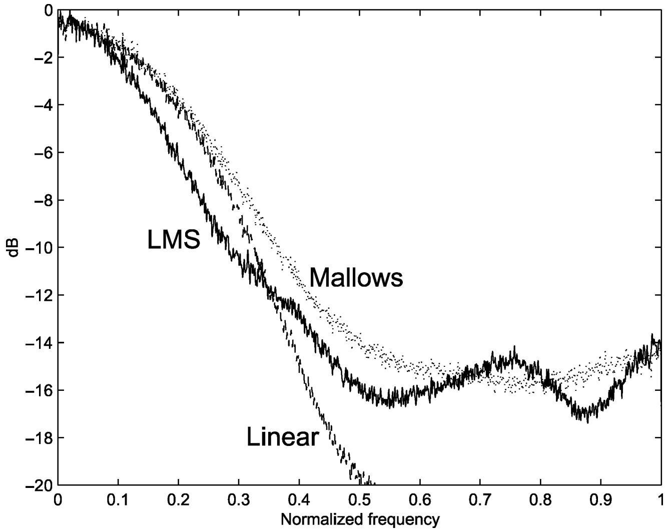


Fig. 6. Approximated frequency response of WM filters designed using the Mallows Iterative Algorithm and an LMS system-identification algorithm compared with the approximated frequency response of the model linear filter.

D. Mallows Algorithm versus Other WM Filter Design Schemes

One last test of the algorithm will be to compare it with other algorithms currently used to design WM filters. Since the final goal of the Mallows Iterative Algorithm is to approximate a linear filter with a WM filter, this algorithm was compared with the algorithm presented in [4]. This is a classical least mean squares (LMS) system-identification scheme where the adaptive filter is a WM filter that, by the end of the process, will be a good approximation of the linear filter taken as a reference. This method, as well as Mallows iterative algorithm, aims to minimize the MSE between the outputs of the median filter being designed and the linear filter used as a model. It is of interest to compare the WM filters obtained by the two algorithms. The two algorithms were used to approximate the 11-tap lowpass linear filter: $\mathbf{h} = [-0.0032,\ 0, 0.0484,\ 0.0889,\ 0.2151,$ and $0.2889,\ 0.2151,\ 0.0889,\ 0.0484, 0,\ -0.0032]$, which is a valid SSP vector. The resulting median coefficients are $W_{\text{Mallows}} = [-0.0172,\ 0,\ 0.0463,\ 0.1048,\ 0.1982,$ and $0.2564,\ 0.1982,\ 0.1048,\ 0.0463,\ 0,\ -0.0172]$ for the Mallows iterative algorithm, which is an exact median equivalent of the linear filter $\mathbf{h}(J(W_{\text{Mallows}}) = 0)$. The WM filter obtained with the LMS algorithm is $W_{\text{LMS}} = [-0.0328,\ 0.0367,\ 0.0619,\ 0.0933,\ 0.1785,$ and $0.2227,\ 0.1729,\ 0.0840,\ 0.0682,\ 0.0201,\ 0.0290]$. This filter does not belong to the same class as W_{Mallows}, and in consequence, it is not a WM equivalent of $\mathbf{h}$. For this filter, the error measure is $J(W_{\text{LMS}}) = 0.0081$.

The approximated frequency response of both median filters and the linear filter are shown in Fig. 6. It can be seen that the WM filter obtained with the system identification algorithm has a narrower passband and more ripple in the stopband than that obtained with the Mallows Iterative Algorithm. Additionally, W_{Mallows} follows the behavior of the linear filter up to an attenuation of 5–6 dB and has a sharper transition.

The difference found between the results obtained with the two methods can be understood from the fact that the LMS system identification is a stochastic algorithm that needs a random signal at the inputs of the filters in order to design the WM filter, which is a standard Gaussian sequence in this case. A consequence of this is that median filters resulting from training sequences with different statistics are different. On the other hand, Mallows' algorithm is completely deterministic, and its result will always be a WM filter in the class that is closest to the original linear filter in the mean square error sense.

VIII. Conclusions

A design strategy for real- and complex-valued WM based on Mallows' theory for nonlinear smoothers has been presented. A closed-form equation to find the sample selection probabilities of any real-valued WM smoother and a steepest descent algorithm that attempts to find the closest WM filter in the MSE sense to a given linear filter was proposed.

The proposed iterative algorithm allows for the design of a WM filter that approximates the frequency response of the linear filter used as a reference. The simulations demonstrate that it is possible to obtain any frequency characteristic required (lowpass, highpass, bandpass, and bandstop) using WM filters in the real and complex domains.

The set of simulations presented shows that the designed WM filters outperform their linear counterparts in experiments with additive α-stable noise. Thus, controlled frequency-selection capabilities have been added to the already known robustness of WM filters.

The main advantage of Mallows Iterative Algorithm, compared with LMS-type algorithms used to design WM filters, is the fact that it does not require a training sequence. Filters designed using stochastic adaptive algorithms require a training sequence, and training sequences with different statistics may result in different median filters (even if the same linear filter is used as a reference). On the other hand, Mallows Iterative Algorithm is deterministic, and it does not depend on an input signal. However, the use of either algorithm for the design of a filter depends on the specific application and the information available to design the filter. In general, when a training sequence is available, an LMS algorithm can be used, whereas Mallows Iterative Algorithm will be the alternative when the spectral characteristics required by the filter are known.

References

[1] C. L. Mallows, "Some theory of nonlinear smoothers," *Ann. Statist.*, vol. 8, pp. 694–715, Jul. 1980.
[2] M. K. Prasad and Y. H. Lee, "Stack filters and selection probabilities," *IEEE Trans. Signal Process.*, vol. 42, no. 10, pp. 2628–2642, Oct. 1994.
[3] I. Shmulevich and G. R. Arce, "Spectral design of weighted median filters admitting negative weights," *IEEE Signal Process. Lett.*, vol. 8, no. 12, pp. 313–316, Dec. 2001.
[4] S. Hoyos, Y. Li, J. Bacca, and G. Arce, "Weighted median filters admitting complex-valued weights and their optimization," *IEEE Trans. Signal Processing*, pt. 2, vol. 52, no. 10, pp. 2776–2787, Oct. 2004.
[5] G. R. Arce, "A general weighted median filter structure admitting negative weights," *IEEE Trans. Signal Process.*, vol. 46, no. 12, pp. 3195–3205, Dec. 1998.
[6] H. A. David, *Order Statistics*. New York: Wiley, 1981.
[7] J. Nieweglowski, M. Gabbouj, and Y. Neuvo, "Weighted medians-positive boolean functions conversion algorithms," *IEEE Trans. Signal Process.*, vol. 41, no. 2, pp. 149–161, Nov. 1993.
[8] L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, "Weighted median filters: a tutorial," *IEEE Trans. Circuits Syst.*, vol. 43, no. 3, pp. 157–192, Mar. 1996.
[9] S. Muroga, *Threshold Logic and Its Applications*. New York: Wiley-Interscience, 1971.

[10] O. Yli-Harja, J. Astola, and Y. Neuvo, "Analysis of the properties of median and weighted median filters using threshold logic and stack filter representation," *IEEE Trans. Signal Process.*, vol. 39, no. 2, pp. 395–410, Feb. 1991.

[11] P. D. Wendt, E. J. Coyle, and N. C. Gallagher, "Stack filters," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, no. 4, pp. 898–911, Aug. 1986.

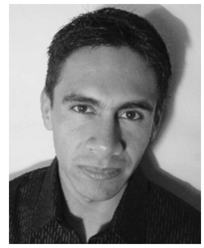
[12] S. Muroga, T. Tsuboi, and C. R. Baugh, "Enumeration of threshold functions of eight variables," *IEEE Trans. Comput.*, vol. C-19, pp. 818–825, Sep. 1970.

[13] M. K. Prasad and Y. H. Lee, "Weighted median filters: generation and properties," in *Proc. IEEE ISCAS*, vol. 1, May 1989, pp. 425–428.

[14] I. Schmulevich, V. Melnik, and K. Egiazarian, "The use of sample selection probabilities for stack filter design," *IEEE Signal Process. Lett.*, vol. 7, no. 7, pp. 189–192, Jul. 2000.

Sebastian Hoyos (S'02–M'04) was born in Cali, Colombia, in 1975. He received the B.S. degree from the Pontificia Universidad Javeriana (PUJ), Bogota, Colombia, in 2000 and the M.S. and Ph.D. degrees from the University of Delaware, Newark, all in electrical engineering, in 2002 and 2004, respectively.

He worked for Lucent Technologies, Inc. from 1999 to 2000 as Technical Manager and Sales Engineer for the Andean region in South America. Simultaneously, he was an Adjunct Professor at the PUJ University, where he lectured on microelectronics and control theory. In the Fall of 2000, he enrolled in the Department of Electrical and Computer Engineering, University of Delaware. During his master and Ph.D. studies, he worked under the PMC-Sierra Inc., the Delaware Research Partnership Program, and the Army Research Laboratory (ARL) Collaborative Technology Alliance (CTA) in Communications and Networks. At the end of the summer of 2004, he joined the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, where he is a postdoctoral researcher at the Berkeley Wireless Research Center. He has carried out industrial consulting with Conexant Systems Inc., Newport Beach, CA, for the development of the next generation of ultra-wideband communications systems. His research interests include communication systems, wireless communications, sensor network processing, robust signal processing, and mixed-signal high-speed processing and circuit design. For a complete list of his publications and patents, see www.sebastianhoyos.com.



Jan Bacca (S'03) was born in Pasto, Colombia, in 1977. He received the B.S. degree in electronic engineering from Pontificia Universidad Javeriana, Bogotá, Colombia, in 2001 and the M.E.E. degree from the Department of Electrical and Computer Engineering, University of Delaware, Newark, in 2003, where he is currently pursuing the Ph.D. degree.

Gonzalo R. Arce (F'02) received the Ph.D. degree from Purdue University, West Lafayette, IN, in 1982.

Since 1982, he has been with the faculty of the Department of Electrical and Computer Engineering, University of Delaware, Newark, where he is the Charles Black Evans Professor and Chairman of Electrical and Computer Engineering. His research interests include statistical and nonlinear signal processing, multimedia security, electronic imaging and display, and signal processing for communications.

Dr. Arce received the Whittaker RESNA and the Advanced Telecommunications and Information Distribution (ATIRP) Consortium best paper awards. He received the NSF Research Initiation Award. He was the co-chair of the 2001 EUSIPCO/IEEE Workshop on Nonlinear Signal and Image Processing (NSIP'01), the 1991 SPIE Symposium on Nonlinear Electronic Imaging, and the 2002 and 2003 SPIE ITCOM Conference. He has served as Associate Editor for the IEEE TRANSACTIONS FOR SIGNAL PROCESSING, Senior Editor of the *Applied Signal Processing Journal*, Guest Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING, and Guest Editor for *Optics Express*.