# A General Weighted Median Filter Structure Admitting Negative Weights

Gonzalo R. Arce, *Senior Member, IEEE*

***Abstract*—Weighted median smoothers, which were introduced by Edgemore in the context of least absolute regression over 100 years ago, have received considerable attention in signal processing during the past two decades. Although weighted median smoothers offer advantages over traditional linear finite impulse response (FIR) filters, it is shown in this paper that they lack the flexibility to adequately address a number of signal processing problems. In fact, weighted median smoothers are analogous to normalized FIR linear filters constrained to have only positive weights. In this paper, it is also shown that much like the mean is generalized to the rich class of linear FIR filters, the median can be generalized to a richer class of filters admitting positive and negative weights. The generalization follows naturaly and is surprisingly simple. In order to analyze and design this class of filters, a new threshold decomposition theory admitting real-valued input signals is developed. The new threshold decomposition framework is then used to develop fast adaptive algorithms to optimally design the real-valued filter coefficients. The new weighted median filter formulation leads to significantly more powerful estimators capable of effectively addressing a number of fundamental problems in signal processing that could not adequately be addressed by prior weoghted median smoother structures.**

***Index Terms*—Adaptive filters, filtering, median filters, nonlinear estimation, nonlinear filters, robustness.**

## I. Introduction

WEIGHTED median smoothers, which were introduced by Edgemore in the context of least absolute regression over 100 years ago [1], have received considerable attention in signal processing research over the last two decades [2]–[4]. Although these structures are widely known in the signal processing literature as filters, for reasons that will become apparent shortly, we will refer to these structures as *weighted median smoothers*. During the last few years, the theory behind WM smoothers has been developing quite fast. Today, due to its sound underlying theory, weighted median smoothers are increasingly being used particularly in image processing applications. The success of median smoothers in image processing is based on two intrinsic properties: edge preservation and efficient attenuation of impulsive noise—properties not shared by traditional linear filters. The applications of weighted median smoothers, however, have not significantly spread beyond image processing applications. It is often stated that there are many analogies between weighted median smoothers and linear finite impulse response (FIR) filters. In this paper, however, we show that weighted median (WM) smoothers are highly constrained and that they are significantly less powerful than linear FIR filters. In fact, WM smoothers are equivalent to normalized weighted mean filters, which are a severely constrained subset of linear FIR filters. Admitting only positive filter weights, weighted median and normalized weighted mean filters are, in essence, smoothers having "lowpass" type filtering characteristics.

A large number of engineering applications require "bandpass" or "highpass" frequency filtering characteristics. Equalization, deconvolution, prediction, beamforming, and system identification are example applications where filters having "bandpass" or "highpass" characteristics are of fundamental importance. Linear FIR equalizers admitting only positive filter weights, for instance, would lead to completely unacceptable results. Thus, it is not surprising that weighted median smoothers admitting only positive weights lead to unacceptable results in a number of applications. To overcome the limitations of WM filters, a number of generalizations have been proposed. Typically, a combination of linear and median type operations are used in tandem [4]–[9]. Hybrid methods have proven useful but are nonetheless *ad hoc* and are often too cumbersome, requiring a large number of parameters.

In this paper, based on fundamental principles of parameter estimation, a new weighted median filtering structure that admits positive and negative weights is defined. It is shown that the new WMF structure is analogous to the class of linear FIR filters, whereas the previous definition of weighted median smoothers used in the literature is analogous to constrained normalized weighted mean smoothers. The latter class is referred to as *weighted median smoothers*, and the new filter class that admits real-valued filter weights is referred to as *weighted median filters*. The generalization follows naturally and is surprisingly simple. As would be expected, smoothers reduce to weighted median smoothers whenever the filter coefficients are constrained to be positive.

Median smoothers have traditionally relied on threshold decomposition for their analysis and design [10]. Threshold decomposition is a powerful tool that exploits the following property: Median smoothing a positive signal is equivalent to decomposing the signal into several binary thresholded signals, smoothing each binary signal separately with the

Manuscript received July 8, 1997; revised June 8, 1998. This work was supported in part by the National Science Foundation under Grant MIP-9530923 and through collaborative participation in the Advanced Telecommunications/Information Distribution Research Program (ATIRP) Consortium sponsored by the U.S. Army Research Laboratory under Cooperative Agreement DAAL01-96-2-0002. The associate editor coordinating the review of this paper and approving it for publication was Prof. M. H. Er.

The author is with the Department of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716 USA (e-mail: arce@ee.udel.edu).

Publisher Item Identifier S 1053-587X(98)08709-1.

corresponding binary median smoother, and then adding the binary output signals together. The advantage provided by this *weak superposition* property is that the analysis of median smoothing binary signals is significantly easier than the analysis of median smoothing real-valued signals. Although threshold decomposition as defined in [11] admits only positive-valued signals, it can be used with real-valued input signals by adding, *a priori*, a large enough positive constant to the input signal. Once the smoothing is performed, the dc bias is subtracted. Unfortunately, this decomposition framework is not adequate for the analysis of weighted median filters with positive and negative weights. Even if a sufficiently large constant is added to the input signal, the negative weights applied to the data will map the dc-shifted samples to the negative domain. In order to analyze the new WM filter class, we first define a new threshold decomposition framework that allows real-valued inputs and negative sample weighting. The new threshold decomposition architecture overcomes the shortcomings associated with prior definitions.

In practice, the (real-valued) filter coefficients of the new WM filter must be determined in some fashion. In this paper, through the use of the new threshold decomposition architecture, an adaptive algorithm is developed for training WM filters under the mean absolute error (MAE) criterion. The obtained adaptive algorithm, which is referred to as the *fast least mean absolute (LMA) weighted median algorithm*, is simple, having an update complexity comparable with that of the ubiquitous LMS algorithm.

The organization of the paper is as follows. In Section II, the new weighted median filtering structure that admits real-valued coefficients is introduced. In Section III, the new threshold decomposition architecture is described. Adaptive algorithms to find the optimal WM filter coefficients are derived in Section IV. In Section V, several examples of adaptive and nonadaptive WM filters are shown, and their performance is compared with that of traditional WM smoothers.

## II. Weighted Median Filters with Real-Valued Weights

The sample median and sample mean have deep roots in statistical estimation theory. In particular, they are the *Maximum Likelihood* (ML) estimators of location derived from sets of independent and identically distributed (i.i.d.) samples obeying the Laplacian and Gaussian distributions, respectively. Thus, if $X_1, X_2, \cdots, X_N$ are $N$ i.i.d. Gaussian distributed samples with constant but unknown mean $\beta$, the maximum likelihood estimate of location is the value $\overline{\beta}$, which maximizes the likelihood function

$$\begin{aligned} L(x_1, \cdots, x_N; \beta) &= \prod_{i=1}^{N} f(x_i - \beta) \\ &= \left(\frac{1}{2\pi\sigma^2}\right)^{N/2} \exp\left(-\sum_{i=1}^{N} (X_i - \beta)^2 / 2\sigma^2\right). \end{aligned} \tag{1}$$

This is equivalent to minimizing the least squares sum

$$G_2(\beta) = \sum_{i=1}^{N} (X_i - \beta)^2. \tag{2}$$

The value $\overline{\beta}$ minimizing (2) is the sample mean $\overline{\beta} = \Sigma_{i=1}^{N} X_i / N$. Similarly, when the set of i.i.d. samples obey the Laplacian distribution, it can be shown that the maximum likelihood estimate of location is the value $\beta$ that minimizes the sum of least absolute deviations

$$G_1(\beta) = \sum_{i=1}^{N} |X_i - \beta|. \tag{3}$$

The value $\beta$ minimizing (3) is the sample median $\hat{\beta} = \text{MEDIAN}\,(X_1, X_2, \cdots, X_N)$. The sample mean and median thus play an analogous role in location estimation. While the mean is associated with the Gaussian distribution that often emerges naturally in practice, the median is related to the Laplacian distribution, which has heavier tails and can often provide a better model for impulsive-like processes. The robustness of the sample median is explained by the heavy tails of the Laplacian distribution.

The sample mean and median can be generalized by extending the model of maximum likelihood estimation. Let the sample set $X_1, X_2, \cdots, X_N$ be independent but not identically distributed. In particular, assume the $X_i'$s obey the same distribution, but assume that their variance is not identical for all samples. Under the Gaussian assumption, the ML estimate of location in this case can be shown to be the value $\beta$ minimizing

$$G_2(\beta) = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} (X_i - \beta)^2 \tag{4}$$

where $\sigma_i^2$ is the variance of the $i$th sample in the set. The value $\beta$ minimizing (4) is the normalized weighted average

$$\overline{\beta} = \frac{\sum_{i=1}^{N} W_i \cdot X_i}{\sum_{i=1}^{N} W_i} \tag{5}$$

with $W_i = 1/\sigma_i^2 > 0$. Likewise, under the Laplacian model, the maximum likelihood estimate of location minimizes the sum of weighted absolute deviations

$$G_1(\beta) = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} |X_i - \beta|. \tag{6}$$

The value $\tilde{\beta}$ minimizing (6) is the weighted median originally introduced over 100 years ago by Edgemore [1] and defined as

$$\tilde{\beta} = \text{MEDIAN}\,(W_1 \diamond X_1, W_2 \diamond X_2, \cdots, W_N \diamond X_N) \tag{7}$$

where $W_i = 1/\sigma_i^2 > 0$ and where $\diamond$ is the replication operator defined as $W_i \diamond X_i = \overbrace{X_i, X_i, \cdots, X_i}^{W_i \text{ times}}$. It should be noted that the weights in (5) and (7) are constrained to take on

non-negative values due to their inverse relationship to the variances of the respective observation samples.

Notably, the location parameter estimation problem just described is related to the time-series filtering problem where the output $Y(n)$, at time $n$, can be thought of as an estimate of location and where $X(n-N_1),\cdots,X(n-1),X(n),X(n+1),\cdots,X(n+N_2)$ are the set of observation samples. Although these samples, in general, exhibit temporal correlation, the independent but not identically distributed model described above can be used to capture the mutual correlation. This is possible by observing that the estimate $Y(n)$ can rely more on the sample $X(n)$ than on the other samples of $\{X(n)\}$ that are further away in time. Thus, $X(n)$ is more reliable than $X(n-1)$ or $X(n+1)$, which, in turn, are more reliable than $X(n-2)$ or $X(n+2)$, and so on. By assigning different variances (reliabilities) to the independent but not identically distributed location estimation model, the temporal correlation used in time-series filtering is captured. The weighting structure in (5) and (7) reflect the varying reliability of the samples in the observation set.

From the smoother structures described in (5) and (7), it can be seen that the class of smoothers as defined by Edgemore and, as used extensively in signal processing [2], is equivalent to the class of normalized weighted average filters. Since the former filter class is severely constrained, allowing only linear combinations of positively weighted input samples, it follows that WM smoothers are also severely limited in their structure.

Much like the sample mean can be generalized to the rich class of linear FIR filters, there must be a logical way to generalize the median to an equivalently rich class of filters that admit both positive and negative weights. We next show that this is, in fact, possible. It turns out that the extension is not only natural, leading to a significantly richer filter class, but it is simple as well. Perhaps the simplest approach to derive the class of WM filters with real-valued weights is by analogy. The sample mean $\overline{\beta}=\text{MEAN}\,(X_1,X_2,\cdots,X_N)$ can be generalized to the class of linear FIR filters as

$$\overline{\beta}=\text{MEAN}\,(W_1\cdot X_1,W_2\cdot X_2,\cdots,W_N\cdot X_N) \tag{8}$$

where $W_i\in R$. In order to apply the analogy to the median filter structure, (8) must be written as

$$\overline{\beta}=\text{MEAN}\,(|W_1|\cdot\text{sign}\,(W_1)X_1,|W_2|\cdot\text{sign}\,(W_2)X_2,\cdots |W_N|\cdot\text{sign}\,(W_N)X_N) \tag{9}$$

where the sign of the weight affects the corresponding input sample, and the weighting is constrained to be non-negative. By analogy, the class of smoothers admitting real-valued weights emerges as

$$\tilde{\beta}=\text{MEDIAN}\,(|W_1|\diamond\text{sign}\,(W_1)X_1,|W_2|\diamond\text{sign}\,(W_2)X_2\cdots,|W_N|\diamond\text{sign}\,(W_N)X_N) \tag{10}$$

with $W_i\in R$, for $i=1,2,\cdots,N$. Again, the weight signs are uncoupled from the weight magnitude values and are merged with the observation samples. The weight magnitudes play the equivalent role of positive weights in the framework of weighted median smoothers. It is simple to show that the weighted mean (normalized) and the weighted median operations shown in (9) and (10), respectively, minimize

$$G_2(\beta)=\sum_{i=1}^{N}|W_i|(\text{sign}\,(W_i)X_i-\beta)^2 \quad \text{and}$$
$$G_1(\beta)=\sum_{i=1}^{N}|W_i||\text{sign}\,(W_i)X_i-\beta|. \tag{11}$$

Although $G_2(\beta)$ is a convex continuous function, $G_1(\beta)$ is a convex but piecewise linear function whose minima is guaranteed to be one of the "signed" input samples (i.e., $\text{sign}\,(W_i)X_i$). The WM filter output for noninteger weights can be determined as follows.

1) Calculate the threshold $T_0=\frac{1}{2}\Sigma_{i=1}^{N}|W_i|$.
2) Sort the "signed" observation samples $\text{sign}\,(W_i)X_i$.
3) Sum the magnitude of the weights corresponding to the sorted "signed" samples beginning with the maximum and continuing down in order.
4) The output is the signed sample whose magnitude weight causes the sum to become $\geq T_0$.

The following example illustrates this procedure. Consider the window size 5 WM filter defined by the real-valued weights $[W_1,W_2,W_3,W_4,W_5]^T=[0.1,0.2,0.3,-0.2,0.1]^T$. The output for this filter operating on the observation set $[X_1,X_2,X_3,X_4,X_5]^T=[-2,2,-1,3,6]^T$ is found as follows. Summing the absolute weights gives the threshold $T_0=\frac{1}{2}\Sigma_{i=1}^{5}|W_i|=0.45$. The "signed" observation samples, sorted observation samples, their corresponding weight, and the partial sum of weights (from each ordered sample to the maximum) are:

| | | | | | |
|---|---|---|---|---|---|
| observation samples | $-2$, | 2, | $-1$, | 3, | 6 |
| corresponding weights | 0.1, | 0.2, | 0.3, | $-0.2$, | 0.1 |
| sorted signed observation samples | $-3$, | $-2$, | $-1$, | 2, | 6 |
| corresponding absolute weights | 0.2, | 0.1, | 0.3, | 0.2, | 0.1 |
| partial weight sums | 0.9, | 0.7, | <u>0.6</u>, | 0.3, | 0.1. |

Thus, the output is $-1$ since, when starting from the right (maximum sample) and summing the weights, the threshold $T_0=0.45$ is not reached until the weight associated with $-1$ is added. The underlined sum value above indicates that this is the first sum that meets or exceeds the threshold.

Although the four-step procedure described above to compute the weighted median is straightforward, the weighted median computation can be expressed more succinctly as follows. Let the "signed" samples $\text{sign}\,(W_i)X_i$ and their corresponding absolute valued weights be denoted as $S_i$ and $|W_i|$, respectively. The sorted "signed" samples are then denoted as $S_{(i)}$, where $S_{(1)}\leq S_{(2)}\leq\cdots\leq S_{(N)}$. The absolute valued weights corresponding to the sorted signed samples are denoted as $|W_{\ell_{(k)}}|$, where $\ell_{(k)}$ refers to the location of the $k$th-order statistic. In the previous example, the weight associated with the fourth-order statistic $S_{(4)}$ is, for instance, $|W_{\ell_{(4)}}|=|W_2|=0.2$. With this notation, the

smoother output can be written as

$$\tilde{\beta} = \left\{ S_k \colon \min_k \text{ for which } \sum_{i=0}^{k} |W_{\ell_{(N-i)}}| \geq T_0 \right\}. \quad (12)$$

The effect that negative weights have on the weighted median operation is similar to the effect that negative weights have on linear FIR filter outputs. Fig. 1 illustrates this concept where $G_2(\beta)$ and $G_1(\beta)$, which are the cost functions associated with linear FIR and WM filters, respectively, are plotted as a function of $\beta$. Recall that the output of each filter is the value minimizing the cost function. The input samples are again selected as $[X_1, X_2, X_3, X_4, X_5] = [-2, 2, -1, 3, 6]$, and two sets of weights are used. The first set is $[W_1, W_2, W_3, W_4, W_5] = [0.1, 0.2, 0.3, 0.2, 0.1]$, where all the coefficients are positive, and the second set is $[0.1, 0.2, 0.3, -0.2, 0.1]$, where $W_4$ has been changed, with respect to the first set of weights, from $0.2$ to $-0.2$. Fig. 1(a) shows the cost functions $G_2(\beta)$ of the linear FIR filter for the two sets of filter weights. Notice that by changing the sign of $W_4$, we are effectively moving $X_4$ to its new location $\text{sign}(W_4)X_4 = -3$. This, in turn, pulls the minimum of the cost function toward the relocated sample $\text{sign}(W_4)X_4$. Negatively weighting $X_4$ on $G_1(\beta)$ has a similar effect, as shown in Fig. 1(b). In this case, the minimum is pulled toward the new location of $\text{sign}(W_4)X_4$. The minimum, however, occurs at one of the samples $\text{sign}(W_i)X_i$.

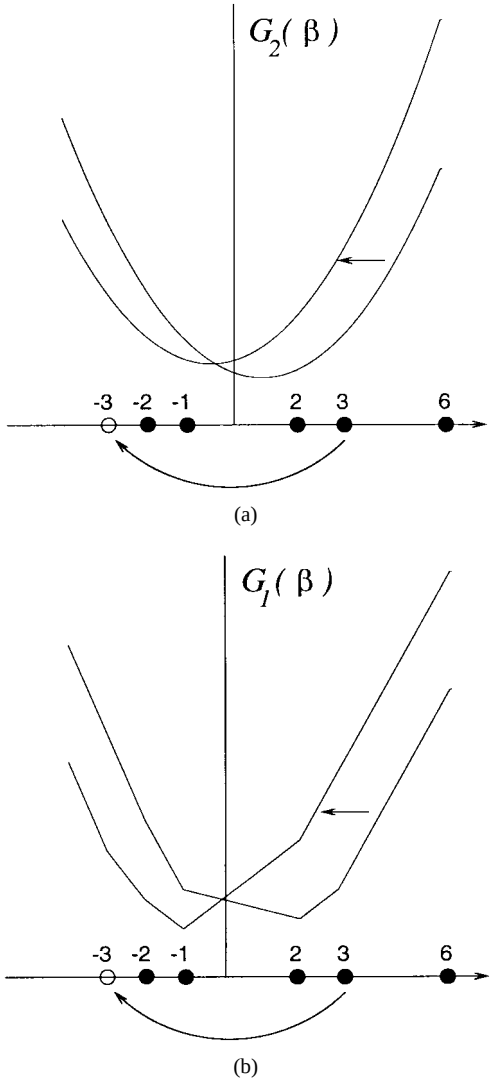


Fig. 1. Effects of negative weighting on the cost functions $G_2(\beta)$ and $G_1(\beta)$. The input samples are $[X_1, X_2, X_3, X_4, X_5]^T = [-2, 2, -1, 3, 6]^T$, which are filtered by the two set of weights $[0.1, 0.2, 0.3, 0.2, 0.1]^T$ and $[0.1, 0.2, 0.3, -0.2, 0.1]^T$, respectively.

## III. Threshold Decomposition for Real-Valued Signals

Threshold decomposition is a powerful theoretical tool used extensively in the analysis of smoothers. Introduced by Fitch *et al*. [10], threshold decomposition was originally formulated to admit signals having only a finite number of positive-valued quantization levels. Threshold decomposition was later extended to admit continuous-level positive-valued signals [11]. By adding a large enough positive constant to a signal encountered in practice, it was argued that this latter approach to threshold decomposition could be applied to an arbitrary real-valued signal. Although this approach may be adequate for the analysis of WM smoothers having only positive weights, it is not suitable for WM filters with positive and negative filter weights. Since negative weighting is equivalent to changing the sign of a weighted sample, regardless of the constant added to the signal *a priori*, the weighting operation will invariably map positive dc-shifted samples back to the negative domain.

In the following, we further extend threshold decomposition, allowing the decomposition of real-valued signals. This decomposition, in turn, can be used to analyze WM filters having real-valued weights. Consider the set of real-valued samples $X_1, X_2, \cdots, X_N$, with $X_i \in R$, and define a smoother by the corresponding real valued weights $W_1, W_2, \cdots, W_N$. Decompose each sample $X_i$ as

$$x_i^q = \text{sgn}(X_i - q) \quad (13)$$

where $-\infty < q < \infty$, and where

$$\text{sgn}(X_i - q) = \begin{cases} 1, & \text{if } X_i > q \\ 0, & \text{if } X_i = q \\ -1, & \text{if } X_i < q. \end{cases} \quad (14)$$

Thus, each sample $X_i$ is decomposed into an infinite set of binary points taking values in $[-1, 1]$ and a single point equal to 0 obtained for $X_i = q$. The original real-valued sample $X_i$ can be perfectly reconstructed from the infinite set of thresholded signals. To show this, let $\hat{X}_i = \lim_{T \to \infty} X_i^{\langle T \rangle}$, where

$$X_i^{\langle T \rangle} = \tfrac{1}{2} \int_{-T}^{-|X_i|} x_i^q \, dq + \tfrac{1}{2} \int_{-|X_i|}^{|X_i|} x_i^q \, dq + \tfrac{1}{2} \int_{|X_i|}^{T} x_i^q \, dq. \quad (15)$$

Since the first and last integrals in (15) cancel each other and since

$$\int_{-|X_i|}^{|X_i|} x_i^q \, dq = 2X_i \tag{16}$$

it follows that $X_i^{\langle T \rangle} = \hat{X}_i = X_i$. Hence, the original signal can be reconstructed from the infinite set of thresholded signals as

$$X_i = \tfrac{1}{2} \int_{-\infty}^{\infty} x_i^q \, dq. \tag{17}$$

With this threshold decomposition, the WM filter operation can be implemented as

$$\begin{aligned}\hat{\beta} &= \text{MEDIAN}\left(|W_i| \diamond \text{sgn}\,(W_i) X_i|_{i=1}^N\right) \\ &= \text{MEDIAN}\left(|W_i| \diamond \tfrac{1}{2} \int_{-\infty}^{\infty} \text{sgn}\left[\text{sgn}\,(W_i)X_i - q\right] \right. \\ &\qquad \left. \cdot \, dq|_{i=1}^N\right).\end{aligned} \tag{18}$$

The expression in (18) represents the median operation of a set of weighted integrals, each synthesizing a signed sample. Note that the same result is obtained if the weighted median of these functions, at each value of $q$, is taken first, and the resultant signal is integrated over its domain. Thus, the order of the integral and the median operator can be interchanged without affecting the result, leading to

$$\hat{\beta} = \tfrac{1}{2} \int_{-\infty}^{\infty} \text{MEDIAN}\left(|W_i| \diamond \text{sgn}\left[\text{sgn}\,(W_i)X_i - q\right]|_{i=1}^N\right) dq. \tag{19}$$

In this representation, the "signed" samples play a fundamental role; thus, we define the "signed" observation vector $\boldsymbol{S}$ as

$$\begin{aligned}\boldsymbol{S} &= [\text{sgn}\,(W_1)X_1, \text{sgn}\,(W_2)X_2, \cdots, \text{sgn}\,(W_N)X_N)]^T \\ &= [S_1, S_2, \cdots, S_N]^T.\end{aligned} \tag{20}$$

The threshold decomposed signed samples, in turn, form the vector $\boldsymbol{s}^q$ defined as

$$\begin{aligned}\boldsymbol{s}^q &= [\text{sgn}\left[\text{sgn}\,(W_1)X_1 - q\right], \cdots, \text{sgn}\left[\text{sgn}\,(W_N)X_N - q\right]]^T \\ &= [s_1^q, s_2^q, \cdots, s_N^q]^T.\end{aligned} \tag{21}$$

Letting $\boldsymbol{W}_a$ be the vector whose elements are the magnitude weights $\boldsymbol{W}_a = [|W_1|, |W_2|, \cdots, |W_N|]^T$, the WM filter operation can be expressed as

$$\hat{\beta} = \tfrac{1}{2} \int_{-\infty}^{\infty} \text{sgn}\left(\boldsymbol{W}_a^T \boldsymbol{s}^q\right) dq. \tag{22}$$

The WM filter representation using threshold decomposition is compact, although it may seem that the integral term may be difficult to implement in practice. As shown in [11], however, it can easily be simplified based on the fact that there are at most $N+1$ different binary signals for each observation vector $\boldsymbol{s}^q$. Let $S_{(i)}$ be the $i$th smallest "signed" sample; then, the $N+1$ different vectors $\boldsymbol{s}^q$ are

$$\boldsymbol{s}^q = \begin{cases} [1, 1, \cdots, 1] & \text{for } -\infty < q < S_{(1)} \\ \left[s_1^{S_{(i)}^+}, s_2^{S_{(i)}^+}, \cdots, s_N^{S_{(i)}^+}\right] & \text{for } S_{(i)} < q < S_{(i+1)} \\ & \quad 1 \le i < N-1 \\ [-1, -1, \cdots, -1] & \text{for } S_{(N)} < q < \infty \end{cases} \tag{23}$$

where $S_{(i)}^+$ denotes a value on the real line approaching $S_{(i)}$ from the right. Using these vectors in (22), we have

$$\begin{aligned}\hat{\beta} = \tfrac{1}{2} &\int_{-\infty}^{S_{(1)}} \text{sgn}\left(\boldsymbol{W}_a^T \boldsymbol{s}^{S_{(1)}^-}\right) dq \\ &+ \tfrac{1}{2} \sum_{i=1}^{N-1} \int_{S_{(i)}}^{S_{(i+1)}} \text{sgn}\left(\boldsymbol{W}_a^T \boldsymbol{s}^{S_{(i)}^+}\right) dq \\ &+ \tfrac{1}{2} \int_{S_{(N)}}^{\infty} \text{sgn}\left(\boldsymbol{W}_a^T \boldsymbol{s}^{\infty}\right) dq.\end{aligned} \tag{24}$$

The above equation reduces to

$$\begin{aligned}\hat{\beta} = \tfrac{1}{2} &\lim_{Q \to \infty} \left(S_{(1)} + Q\right) + \tfrac{1}{2} \sum_{i=1}^{N-1} \left(S_{(i+1)} - S_{(i)}\right) \\ &\cdot \text{sgn}\left(\boldsymbol{W}_a^T \boldsymbol{s}^{S_{(i)}^+}\right) - \tfrac{1}{2} \lim_{Q \to \infty} \left(Q - S_{(N)}\right)\end{aligned} \tag{25}$$

which simplifies to

$$\hat{\beta} = \frac{S_{(1)} + S_{(N)}}{2} + \frac{1}{2} \sum_{i=1}^{N-1} \left(S_{(i+1)} - S_{(i)}\right) \text{sgn}\left(\boldsymbol{W}_a^T \boldsymbol{s}^{S_{(i)}^+}\right). \tag{26}$$

The computation of WM filters with the new threshold decomposition architecture is efficient, requiring only $N-1$ threshold logic (sign) operators, allowing the input signals to be arbitrary real-valued signals and allowing positive and negative filter weights.

The filter representation in (26) also provides us with a useful interpretation of WM filters. The output $\hat{\beta}$ is computed by the sum of the midrange of the signed samples $V = (S_{(1)} + S_{(N)})/2$, which provides a coarse estimate of location, and by a linear combination of the $(i, i+1)$th spacing $V_i = S_{(i)} - S_{(i-1)}$ for $i = 1, 2, \cdots, N$. Hence

$$\hat{\beta} = V + \sum_{i=2}^{N} C\left(\boldsymbol{W}_a, \boldsymbol{s}^{S_{(i)}^+}\right) V_i. \tag{27}$$

The coefficients $C(\cdot)$ take on values $-1/2$ or $1/2$, depending on the values of the observation samples and filter weights.

## IV. Optimal Weighted Median Filtering

In many applications, it is desirable to design the weights of a filter in some optimal fashion. In this section, we develop closed-form and adaptive algorithms to find the optimal real-valued weights of WM filters. The analysis herein exploits the new threshold decomposition architecture introduced in Section III. We assume that the observed process $\{X(n)\}$ is statistically related to some desired process $\{D(n)\}$ of interest. $\{X(n)\}$ is typically a transformed or corrupted version of $\{D(n)\}$. Furthermore, we assume that these processes are jointly stationary. A window of width $N$ slides across the input process pointwise estimating the desired sequence. The vector containing the $N$ samples in the window at time $n$ is

$$\begin{aligned}\boldsymbol{X}(n) &= [X(n - N_1), \cdots, X(n), \cdots X(n + N_2)]^T \\ &= [X_1(n), X_2(n), \cdots, X_N(n)]^T\end{aligned} \tag{28}$$

with $N = N_1+N_2+1$. The running WM filter output estimates the desired signal as

$$\hat{D}(n) = \text{MEDIAN}\left[|W_i| \diamond \text{sgn}(W_i)X_i(n)|_{i=1}^{N}\right]$$

where both the weights $W_i$'s and samples $X_i(n)$ take on real values. The goal is to determine the weight values in $\boldsymbol{W} = [W_1, W_2, \cdots, W_N]^T$, which will minimize the estimation error. Under the mean absolute error (MAE) criterion, the cost to minimize is

$$J(\boldsymbol{W}) = E\{|D(n) - \hat{D}(n)|\} \tag{29}$$

$$= E\left\{\tfrac{1}{2}\left|\int_{-\infty}^{\infty} \text{sgn}(D-q) - \text{sgn}\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right) dq\right|\right\} \tag{30}$$

where the threshold decomposition representation of the signals was used. The absolute value and integral operators in (30) can be interchanged since the integral acts on a strictly positive or a strictly negative function. This results in

$$J(\boldsymbol{W}) = \tfrac{1}{2}\int_{-\infty}^{\infty} E\left\{\left|\text{sgn}(D-q) - \text{sgn}\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right)\right|\right\} dq. \tag{31}$$

Furthermore, since the argument inside the absolute value operator in (31) can only take on values in the set $\{-2, 0, 2\}$, the absolute value operator can be replaced by a properly scaled second power operator. Thus

$$J(\boldsymbol{W}) = \tfrac{1}{4}\int_{-\infty}^{\infty} E\left\{\left(\text{sgn}(D-q) - \text{sgn}\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right)\right)^2\right\} dq. \tag{32}$$

Taking the gradient of the above, we find

$$\frac{\partial}{\partial \boldsymbol{W}} J(\boldsymbol{W}) = -\frac{1}{2}\int_{-\infty}^{\infty} E\left\{e^q(n)\frac{\partial}{\partial \boldsymbol{W}}\text{sgn}\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right)\right\} dq \tag{33}$$

where $e^q(n) = \text{sgn}(D-q) - \text{sgn}(\boldsymbol{W}_a^T\boldsymbol{s}^q)$. Since the sign function is discontinuous at the origin, its derivative will introduce dirac impulse terms that are inconvenient for further analysis. To overcome this difficulty, the sign function in (33) is approximated by a differentiable function. A simple approximation is given by the hyperbolic tangent function

$$\text{sgn}(x) \approx \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{34}$$

Since $(\partial/\partial x)\tanh(x) = \text{sech}^2(x) = (2/e^x + e^{-x})$, it follows that

$$\frac{\partial}{\partial \boldsymbol{W}}\text{sgn}\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right) \approx \text{sech}^2\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right)\frac{\partial}{\partial \boldsymbol{W}}\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right). \tag{35}$$

Evaluating the derivative in (35) and after some simplifications, we find

$$\frac{\partial}{\partial \boldsymbol{W}}\text{sgn}\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right) \approx \text{sech}^2\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right)\begin{bmatrix}\text{sgn}(W_1)s_1^q\\ \text{sgn}(W_2)s_2^q\\ \vdots\\ \text{sgn}(W_N)s_N^q\end{bmatrix}. \tag{36}$$

Using (36) in (33), we obtain

$$\frac{\partial}{\partial W_j} J(\boldsymbol{W}) = -\tfrac{1}{2}\int_{-\infty}^{\infty} E\left\{e^q(n)\text{sech}^2\left(\boldsymbol{W}_a^T\boldsymbol{s}^q\right)\text{sgn}(W_j)s_j^q\right\} dq. \tag{37}$$

Using the gradient, the optimal coefficients can be found through the steepest descent recursive update

$$W_j(n+1) = W_j(n) + 2\mu\left[-\frac{\partial}{\partial W_j}J(\boldsymbol{W})\right]$$

$$= W_j(n) + \mu\left[\int_{-\infty}^{\infty} E\{e^q(n)\,\text{sech}^2(\boldsymbol{W}_a^T(n) \cdot \boldsymbol{s}^q(n))\,\text{sgn}(W_j(n))s_j^q(n)\}\,dq\right]. \tag{38}$$

Using the instantaneous estimate for the gradient, we can derive an adaptive optimization algorithm, where

$$\begin{aligned} W_j(n+1) &= W_j(n) + \mu\int_{-\infty}^{\infty} e^q(n)\,\text{sech}^2\left(\boldsymbol{W}_a^T(n)\boldsymbol{s}^q(n)\right) \cdot \text{sgn}(W_j(n))s_j^q(n)(n)\,dq \\ &= W_j(n) + \mu\int_{-\infty}^{S_{(1)}} e^{S_{(1)}}(n)\,\text{sech}^2\left(\boldsymbol{W}_a^T(n)\boldsymbol{s}^{S_{(1)}}(n)\right) \cdot \text{sgn}(W_j(n))s_j^{S_{(1)}}(n)\,dq + \mu\sum_{i=1}^{N-1}(S_{(i+1)} - S_{(i)}) \\ &\quad \cdot s_j^{S_{(i)}^+}(n)e^{S_i^+}(n)\,\text{sgn}(W_j(n))\,\text{sech}^2\left(\boldsymbol{W}_a^T(n)\boldsymbol{s}^{S_{(i)}^+}(n)\right) \\ &\quad + \mu\int_{S_{(N)}}^{\infty} e^{S_{(N)}^+}(n)\,\text{sech}^2\left(\boldsymbol{W}_a^T(n)\boldsymbol{s}^{S_{(N)}^+}(n)\right) \\ &\quad \cdot \text{sgn}(W_j(n))s_j^{S_{(N)}^+}(n)\,dq. \end{aligned} \tag{39}$$

The error term $e^q(n)$ in the first and last integrals can be shown to be zero; thus, the adaptive algorithm reduces to

$$\begin{aligned} W_j(n+1) = W_j(n) + \mu\sum_{i=1}^{N-1}(S_{(i+1)}(n) - S_{(i)}(n)) &\cdot s_j^{S_{(i)}^+}(n)e^{S_{(i)}^+}(n)\,\text{sgn}(W_j(n)) \\ &\cdot \text{sech}^2\left(\boldsymbol{W}_a^T(n)\boldsymbol{s}^{S_{(i)}^+}(n)\right) \end{aligned} \tag{40}$$

for $j = 1, 2, \cdots, N$. Since the MAE criterion was used in the derivation, the recursion in (40) is referred to as the *least mean absolute* (LMA) weighted median adaptive algorithm. This algorithm is similar to that of Yin and Neuvo's [11], except that their algorithm is applicable to the design of weighted median smoothers that do not admit negative weight values; thus, a projection operator mapping all negative weights to zero is needed in their algorithm. Moreover, updates in Yin and Neuvo's algorithm contain thresholded signals at levels determined by the sample order-statistics and not at the "signed" order statistics. In addition, a positive domain threshold decomposition architecture is used in [11]; thus, the nonlinear term in the update of their algorithm differs from the sech$^2$ nonlinearity appearing in (40).

The LMA weighted median algorithm is simple in nature, but it requires a sum of $N-1$ terms contributing to the updated of $\boldsymbol{W}(n)$, where each term is related to the contribution of the thresholded vectors $\boldsymbol{s}^q(n), q \in \mathcal{S} \equiv \{S_{(1)}(n), S_{(2)}(n), \cdots, S_{(N-1)}(n)\}$. The weight updates in (40) can be shown to have complexity $\mathcal{O}(N^2+6N), \mathcal{O}(N^2+$

1), and $\mathcal{O}(N^2+N)$ in terms of multiplications and additions, respectively. In addition, the sech$^2$ operation must be computed at each level of the decomposition.

The contribution of most of the terms in (40), however, is negligible compared with that of the vector $\boldsymbol{s}^{\hat{D}(n)}(n)$ as it will be described here. Using this fact and following the arguments used in [11], the algorithm in (40) can be simplified considerably, leading to a fast LMA WM adaptive algorithm. The contribution of each term in (40) is, to a large extent, determined by sech$^2$ $(\boldsymbol{W}_a^T\boldsymbol{s}^q)$ for $q \in \mathcal{S}$. The sech$^2$ function achieves its maximum value when its argument satisfies $\boldsymbol{W}_a^T\boldsymbol{s}^q = 0$. Its value decreases rapidly and monotonically to zero as the argument departs from zero. From the $N-1$ vectors $\boldsymbol{s}^q, q \in \mathcal{S}$, there is one for which the inner product $\boldsymbol{W}_a^T\boldsymbol{s}^q$ is closest to zero. Consequently, the update term corresponding to this vector will provide the biggest contribution in the update. Among all vectors $\boldsymbol{s}^q, q \in \mathcal{S}$, the vector providing the largest update contribution can be found through the definition of the WM filter. Since $\hat{D}(n)$ is equal to one of the signed input samples, the output of the WM filter is $S_{(k)}(n)$ if and only if three inequalities are satisfied simultaneously

$$\boldsymbol{W}_a^T\boldsymbol{s}^{S_{(k-1)}} > T_0 \tag{41}$$

$$\boldsymbol{W}_a^T\boldsymbol{s}^{S_{(k)}} \geq T_0 \tag{42}$$

$$\boldsymbol{W}_a^T\boldsymbol{s}^{S_{(k+1)}} < T_0 \tag{43}$$

where we assume $S_{(k-1)}, S_{(k)}$, and $S_{(k+1)}$ are distinct. This ensures that $\boldsymbol{s}^{S_{(k)}}$ is the vector whose inner product $\boldsymbol{W}_a^T\boldsymbol{s}^{S_{(k)}}$ is closest to zero. Since $S_{(k)}$ is the output of the WM filter at time $n$, $S_{(k)}$ is also equal to $\hat{D}(n)$. Thus, the vector contributing the most to the update in (40) is

$$\boldsymbol{s}^{\hat{D}(n)}(n) = [\operatorname{sgn}(S_1(n) - \hat{D}(n)), \cdots \operatorname{sgn}(S_N(n) - \hat{D}(n))]^T. \tag{44}$$

Using this vector as the principal contributor of the update, the algorithm in (40) is simplified leading to the recursion referred to as the fast LMA WM adaptive algorithm

$$W_j(n+1) = W_j(n) + \mu(D(n) - \hat{D}(n))\operatorname{sgn}(W_j(n)) \cdot \operatorname{sgn}(S_j(n) - \hat{D}(n)), \tag{45}$$

for $j = 1, 2, \cdots, N$.

The updates in (45) have an intuitive explanation described in Fig. 2. When the output of the WM filter is smaller than the desired output, the magnitude of the weights corresponding to the signed samples that are larger than the actual output are increased. Thus, the weight for the signed sample $-1X_i$ is decreased (larger negative value), whereas the weight for signed sample $+1X_j$ is increased. Both cases will lead to updated weights that will push the estimate higher toward $\hat{D}(n)$. Similarly, the weights corresponding to the signed samples, which are smaller than the actual output, are reduced. Thus, the weight for the signed sample $-1X_\ell$ is increased (smaller negative value), whereas the weight for signed sample $+1X_k$ is decreased. Fig. 2(b) depicts the response of the algorithm when the WM filter output is larger than the desired output. The updates of the various samples follow similar intuitive rules, as shown in Fig. 2(b).

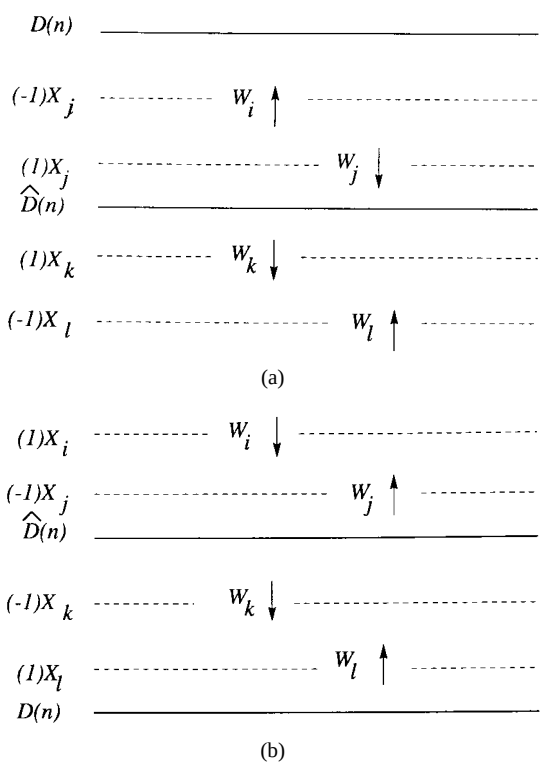


Fig. 2. Weight updates when (a) $D(n) > \hat{D}(n)$ and (b) $D(n) < \hat{D}(n)$. The signed samples are denoted as either $(-1)X_i$ or $(1)X_i$.

The order of complexity of the fast algorithm in (45) is $\mathcal{O}(3N)$ for both the number of additions and multiplications for each update of the filter weights. Since the updates only use the most significant update term in (40), it is expected that the fast algorithm requires a good initial weight vector. It has been experimentally shown that a good initial weight vector is that of the median filter. Due to the nonlinear nature of the adaptive algorithm, a convergence analysis cannot be derived. The fast algorithm, however, has worked quite well in the simulations developed in this paper. This is not surprising since the fast algorithm in (45) is similar to Yin and Neuvo's fast algorithm for the WM smoother [11], which has been extensively tested in a number of applications. Since a convergence analysis is not available for the fast LMA WM adaptive algorithm, exact bounds on the step-size $\mu$ are not available. A reliable guideline to select the step size of this algorithms is to select a step size on the order of that required for the standard LMS algorithm. The step size can then be further tuned according to the user's requirements and by evaluation of the response given by the initial step size choice.

## V. Applications of WM Filters with Real-Valued Weights

The added flexibility provided by negative weights in WM filters is illustrated in this section. First, it is shown that "frequency-selective" WM filters can be easily designed. In particular, the frequency response characteristics of a linear FIR bandpass filter and of a WM bandpass filter, both with 100 taps, is tested. We also show that the frequency response of the best equivalent weighted median smoother, whose coefficients

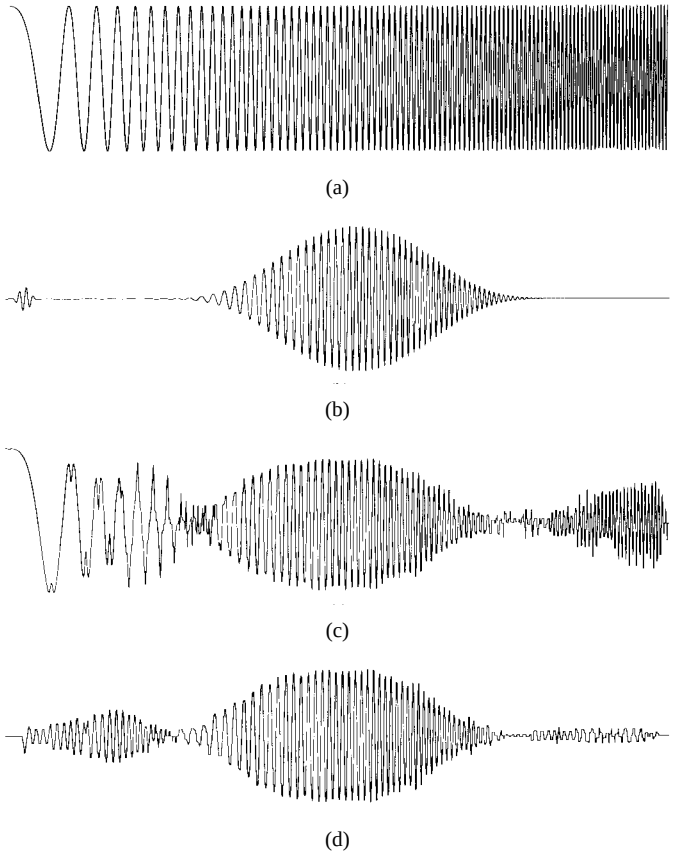

Fig. 3. Frequency selective filter outputs. (a) Chirp test signal. (b) Linear FIR filter output. (c) Weighted median smoother output. (d) Weighted median filter output with real-valued weights.

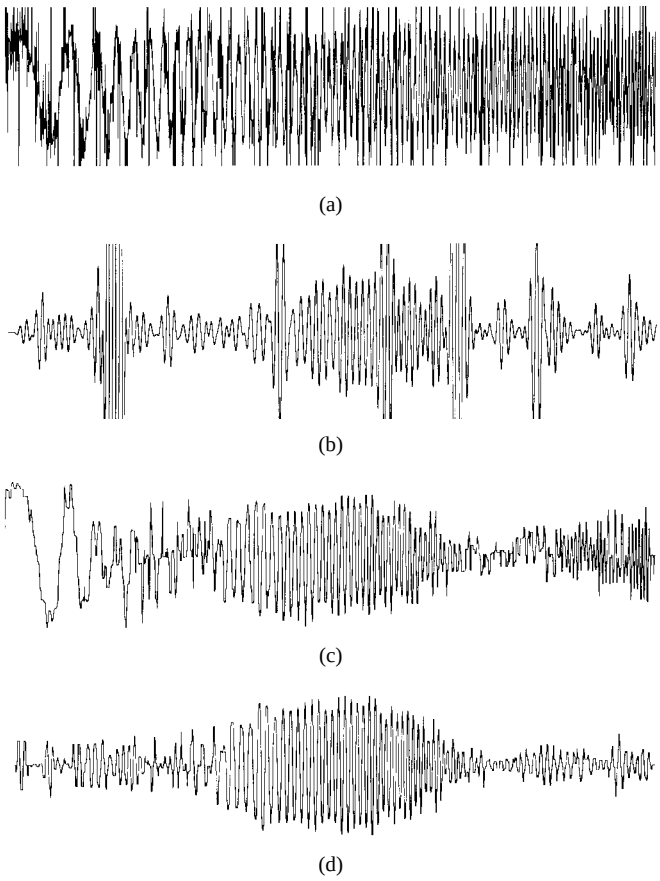

Fig. 4. Frequency selective filter outputs in noise. (a) Chirp test signal in stable noise. (b) Linear FIR filter output. (c) Weighted median smoother output. (d) Weighted median filter output with real-valued weights.

are constrained to positive values, is severely constrained. Finally, we test the new adaptive WM filter algorithm in the design of an optimal robust high-pass filter.

### *A. Frequency Selective WM Filters*

The major limitation of WM filters with positive weights is that these filter structures are smoothers in nature. A large number of important problems in signal processing require the design of *bandpass* and *highpass* filters. Channel equalization, beamformers, and predictors are example applications where flexible frequency-selective processing is needed where weighted median filter smoothers were precluded right from the start. The new WM filter structure opens the possibility of utilizing WM filters as signal processing elements in these applications.

Since linear FIR filters output the mean of a set of weighted samples, the median of an equivalently weighted sample set ought to provide a similar output. Notably, this is the case when the same set of weights as those designed for a linear FIR filter is used in the smoother structure. As we will see shortly, the frequency response characteristics of the attained WM filter follows that of the equivalent linear FIR filter, but more importantly, these are significantly more robust in the processing of signals embedded in noise.

Fig. 3(a) depicts a linearly swept-frequency cosine signal spanning instantaneous frequencies ranging from 0 to 400 Hz. Fig. 3(b) shows the chirp signal filtered by a 120-tap linear FIR filter designed by Matlab's `fir1` function with passband $0.075 \leq \omega \leq 0.125$ (normalized frequency with Nyquist $= 1$). Fig. 3(c) shows the best WM filter (smoother) output when the coefficients are constrained to positive values only. The positive coefficients are found by the method described in [12]. The WM smoother clearly fails to delete the low-frequency components, and it also introduces artifacts at higher frequencies. Fig. 3(d) depicts the output of the WM filter output where real-valued weights are allowed. The 120 median filter weights are set identically to that of the linear FIR filter. Fig. 3(d) shows the significant attenuation obtained in the low-frequency components. The high-frequency terms are cancelled almost completely as well. The small amplitude artifacts exhibited at low frequencies are due to the fact that the output of the WM filter is constrained to be equal to one of the input samples.

Fig. 4(a) depicts the chirp test signal with added $\alpha$-stable noise. The parameter $\alpha = 1.4$ was used, simulating noise with impulsive characteristics [13]. Fig. 4(a) is truncated so that the same scale is used in all plots. Fig. 4(b) shows the noisy chirp signal filtered by the 120-tap linear FIR filter. The output is affected severely by the noise components. Ringing artifacts emerge with each impulse fed into the filter. Fig. 4(c) shows the WM filter output when the coefficients are constrained to positive values only. In this case, the noise does not deteriorate the response significantly, but the response is not satisfactory

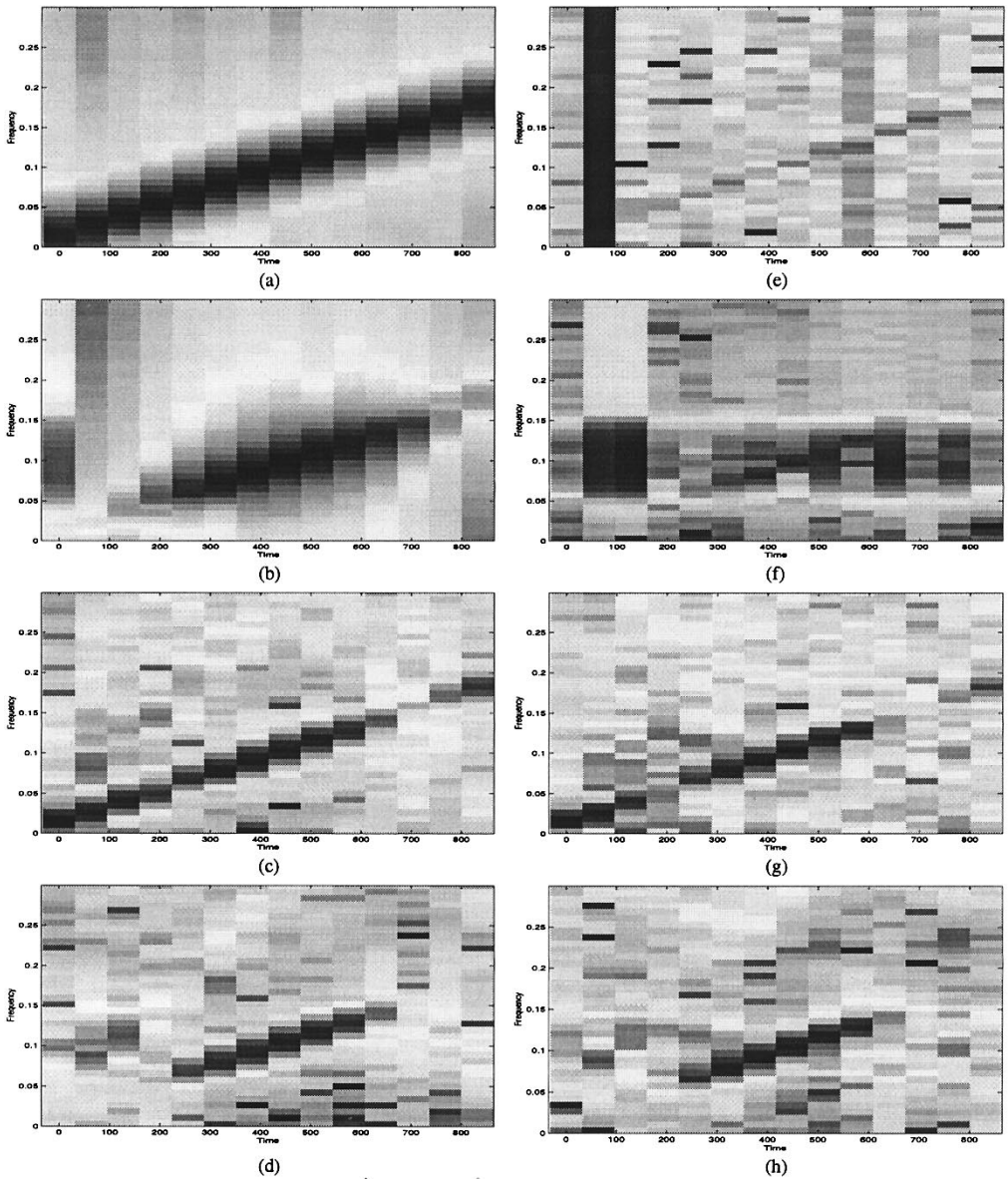


Fig. 5. Spectrograms of (a) Chirp signal. (b) Linear FIR filter output. (c) WM smoother output. (d) WM filter output with real-valued weights. (e) Chirp signal (in noise). (f) Linear FIR filter output (in noise). (g) WM smoother output (in noise). (h) WM filter output with real-valued weights (in noise).

due to the lowpass characteristics of the WM smoother. Fig. 4(d) depicts the output of the WM filter with real-valued weights, which shows a considerable improvement.

To better evaluate the frequency response of the various filters, Fig. 5 shows the time–frequency response of the various filters via the spectrogram. Fig. 5(a) and (e) show the time–frequency plot of the clean and noisy chirps, respectively. The impulses can be localized in time by their broad frequency content, and the linear component in the time–frequency plane can still be distinguished. Fig. 5(b) and (f) shows the time–frequency response of the linear filter outputs. The time–frequency localization in the noiseless case is clear, as shown in Fig. 5(b). On the other hand, the time localization is completely lost in the noisy environment. Fig. 5(c) and (g) shows the spectrograms of the weighted median smoother outputs. The time–frequency characteristics do not change significantly with noise. The undesirable low-frequency components are clearly seen in both plots. Fig. 5(d) and (h) shows the spectrograms of the WM filter admitting negative weights. The time–frequency localization is clearly seen and is not affected by noise. Notably, the median filter provides better time–frequency localization (resolution) than the linear filter, although spurious artifacts in the time–frequency plane emerge in both the noiseless and noisy case.

### *B. Design of Optimal Highpass WM Filter*

Using the weight values of a linear FIR filter in a WM filter structure leads, in general, to acceptable but suboptimal results. Significant improvements can be attained by optimally designing the WM filter weights for the application at hand. Consider the design of a "highpass" WM filter whose objective is to preserve a high-frequency tone while removing all low-frequency terms. Fig. 6(a) depicts a two-tone signal with normalized frequencies of 0.04 and 0.4 Hz. Fig. 6(b) shows

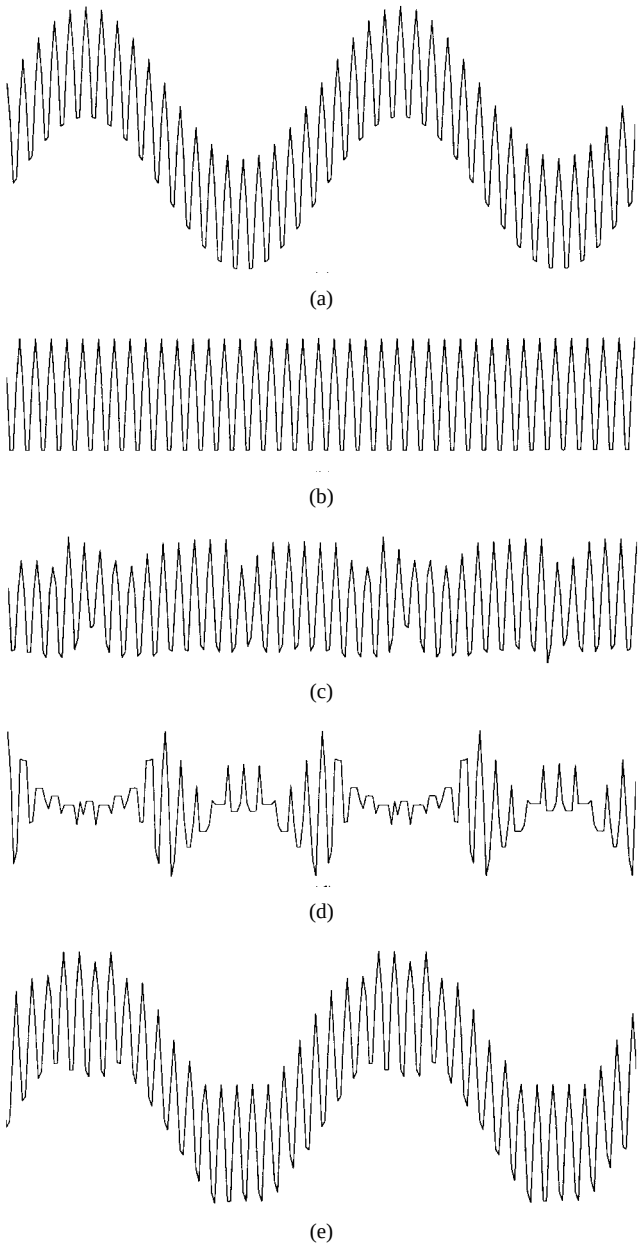


Fig. 6. (a) Two-tone input signal and output from (b) linear FIR high-pass filter. (c) Optimal WM filter. (d) WM filter using the linear FIR weight values. (e) Optimal WM smoother with non-negative weights.

the multitone signal filtered by a 28-tap linear FIR filter designed by Matlab's `fir1` function with a normalized cut-off frequency 0.2 Hz. The fast adaptive LMA algorithm was used to optimize an MW filter with 28 weights. These weights, in turn, were used to filter the multitone signal, resulting in the estimate shown in Fig. 6(c). The low-frequency components have been clearly filtered out. There are, however, some minor artifacts present due to the "selection-type" behavior of the WM filter. Fig. 6(d) depicts the WM filter output when the weights values of the linear FIR filter are used. Although the frequency content of the output signal is within the specifications, there is a significant distortion in the amplitude of the signal in Fig. 6(d). Next, Yin's fast adaptive LMA algorithm was used to optimize a MW filter (smoother) with 28 (positive) weights [11]. The filtered signal attained with

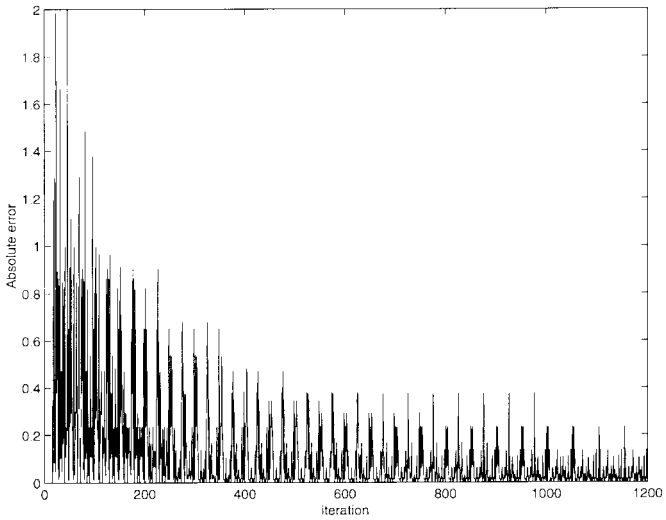


Fig. 7. Single-realization learning characteristics of the fast LMA adaptive WM filter algorithm admitting real-valued weights.

TABLE I
MEAN ABSOLUTE FILTERING ERRORS

| filter | noise free | with stable noise |
|---|---|---|
| linear FIR | 0.012 | 0.979 |
| optimal WMF smoother | 0.688 | 0.692 |
| WMF with FIR weights | 0.501 | 0.530 |
| optimal WMF (fast alg.) | 0.191 | 0.211 |
| optimal WMF | 0.190 | 0.205 |

the optimized weights is shown in Fig. 6(e). The weighted median smoother clearly fails to remove the low-frequency components, as expected. The weighted median smoother output closely resembles the input signal as it is the closest output to the desired signal it can produce.

The step size used in all adaptive optimization experiments was $10^{-3}$. The performance of the adaptive LMA algorithm in (40) and of the fast adaptive LMA algorithm in (45) were very similar. The algorithm in (40), however, proved to converge somewhat faster than the algorithm in (45). This is not surprising as the fast algorithm uses the most important, but not all, information available for the update of the adaptive LMA algorithm. Fig. 7 shows a single-realization learning curve for the fast adaptive LMA WM filter algorithm in (45). It can be seen that 400 iterations were needed for the fast adaptive LMA algorithm to converge. The algorithm in (40) required only 120 iterations; however, due to its computational load, the fast LMA algorithm would be preferred in most applications. The mean absolute error (MAE) between the desired signal and the output of the various filters is summarized in Table I. The advantage of allowing negative weights on the median filter structure is readily seen in Table I. The performance of the LMA WM optimization and of the fast implementation are equivalent. The linear filter outperforms the median structures in the noise-free case, as expected.

Having designed the various highpass filters in a noiseless environment, their performance on signals embedded in noise is tested next. Stable noise with $\alpha = 1.4$ was added to the two-tone signal. Rather than training the various filters to this noisy environment, we used the same filter coefficients as

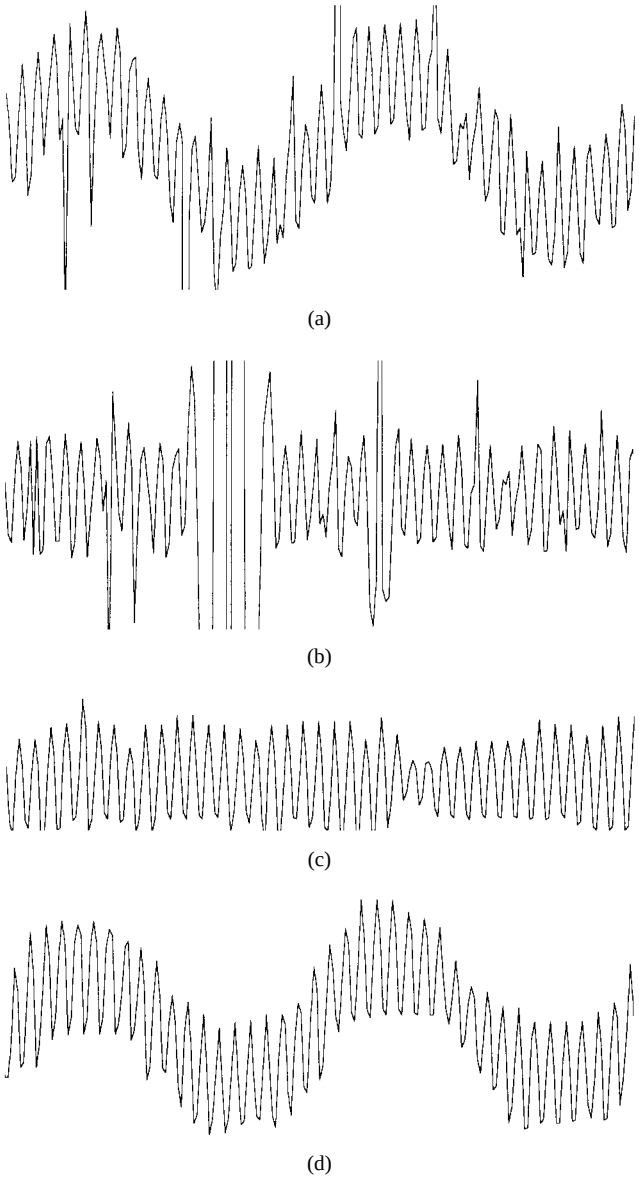

Fig. 8. (a) Two-tone signal in stable noise ($\alpha = 1.4$). (b) Linear FIR filter output. (c) WM filter output. (d) WM smoother output with positive weights.

in the noise-free simulations. Fig. 8(a) and (d) illustrates the results. The MAE for the linear, WM filter, and WM smoother were computed as 0.979, 0.211, and 0.692, respectively. As expected, the outputs of the weighted median filter and smoother are not affected, whereas the output of the linear filter is severely degraded as the linear highpass filter amplifies the high-frequency noise. Table I summarizes the MAE values attained by the various filters.

## VI. Conclusions

In this paper, it is shown that traditional weighted median smoothers are analogous to normalized FIR linear filters constrained to have only positive weights. Much like the mean is generalized to the rich class of linear FIR filters, it is shown that the median can also be generalized to a richer class of WM filters admitting positive and negative weights. The generalization follows naturally and is surprisingly simple. In order to analyze and design the general WM filter structure, the threshold decomposition architecture is extended and used to derive adaptive optimization algorithms for the new filter structure. The new structure now allows the use of WM filters in a large number of applications where "bandpass" and "highpass" type filtering is required.

Although this paper concentrates on generalizing the WM filter structure, the underlying methods are readily applicable to all filters having roots in M-estimation, including the rich class of *Myriad* filters recently introduced in [14] and [15].

## References


[1] F. Y. Edgeworth, "A new method of reducing observations relating to several quantities," *Phil. Mag. (Fifth Series)*, vol. 24, 1887.

[2] D. R. K. Brownrigg, "The weighted median filter," *Commun. Assoc. Comput. Mach.*, vol. 27, Aug. 1984.

[3] 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. Acoust., Speech, Signal Processing*, vol. 39, Feb. 1991.

[4] L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, "Weighted median filters: A tutorial," *IEEE Trans. Circuits Syst.*, vol. 41, May 1996.

[5] P. Heinonen and Y. Neuvo, "Fir-median hybrid filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, June 1987.

[6] P. Ghandi and S. A. Kassam, "Design and performance of combination filters," *IEEE Trans. Signal Processing*, vol. 39, July 1991.

[7] F. Palmieri and C. G. Boncelet, Jr., "L*l*-filters—A new class of order statistic filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, May 1989.

[8] Y.-T. Kim and G. R. Arce, "Permutation filter lattices: A general order-statistic filtering framework," *IEEE Trans. Signal Processing*, vol. 42, Sept. 1994.

[9] K. D. Lee and Y. H. Lee, "Threshold boolean filters," *IEEE Trans. Signal Processing*, vol. 42, Aug. 1994.

[10] J. P. Fitch, E. J. Coyle, and N. C. Gallagher, "Median filtering by threshold decomposition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, Dec. 1984.

[11] L. Yin and Y. Neuvo, "Fast adaptation and performance characteristics of fir-wos hybrid filters," *IEEE Trans. Signal Processing*, vol. 42, July 1994.

[12] F. Palmieri and C. G. Boncelet, Jr., "Frequency analysis and synthesis of a class of nonlinear filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, Aug. 1990.

[13] C. L. Nikias and M. Shao, *Signal Processing with Alpha-Stable Distributions and Applications*. New York: Wiley, 1995.

[14] J. G. Gonzalez and G. R. Arce, "Weighted myriad filters: A robust filtering framework derived from alpha-stable distributions," submitted for publication.

[15] S. Kalluri and G. R. Arce, "Adaptive weighted median filter algorithms for robust signal processing in alpha-stable noise environments," *IEEE Trans. Signal Processing*, vol. 46, pp. 322–334, Feb. 1998.


**Gonzalo R. Arce** (SM'93), for a photograph and biography, see p. 334 of the February 1998 issue of this TRANSACTIONS.