Fast Algorithms for Weighted Myriad Computation by Fixed-Point Search

Sudhakar Kalluri, *Member, IEEE,* and Gonzalo R. Arce, *Fellow, IEEE*

***Abstract*—This paper develops fast algorithms to compute the output of the *weighted myriad filter*. Myriad filters form a large and important class of nonlinear filters for robust non-Gaussian signal processing and communications in impulsive noise environments. Just as the weighted mean and the weighted median are optimized for the Gaussian and Laplacian distributions, respectively, the *weighted myriad* is based on the class of α-stable distributions, which can accurately model impulsive processes.**

The weighted myriad is an M-estimator that is defined in an implicit manner; no closed-form expression exists for it, and its direct computation is a nontrivial and prohibitively expensive task. In this paper, the weighted myriad is formulated as *one* of the fixed points of a certain mapping. An iterative algorithm is proposed to compute these fixed points, and its convergence is proved rigorously. Using these *fixed-point iterations*, fast algorithms are developed for the weighted myriad. Numerical simulations demonstrate that these algorithms compute the weighted myriad with a high degree of accuracy at a relatively low computational cost.

***Index Terms*—Iterative methods, median filters, nonliear filters, optmization.**

I. Introduction

A LARGE number of real-world processes are impulsive in nature, containing sharp spikes or occasional outliers. Examples of impulsive signals include low-frequency atmospheric noise, underwater acoustic signals, radar clutter, and multiple-access interference in wireless communication systems [1]–[3]. The performance of traditional *linear signal processing,* which is optimal under the Gaussian model for the signal statistics, is inadequate in an impulsive environment. Impulsive signals are more accurately modeled by distributions whose density functions have heavier tails than the Gaussian distribution [4]. In recent years, there has been considerable interest in the development of robust techniques for signal processing and communications, based on heavy-tailed distributions for the signal statistics.

Weighted median filters, along with other filters based on *order statistics* [5], [6], have been widely used for robust image processing due to their ability to reject outliers while preserving edges and fine detail in images. These nonlinear filters are optimal under the Laplacian noise model, whose distribution is more heavy-tailed than the Gaussian distribution. However, their applications have not spread significantly beyond the field of image processing, largely because they are constrained to be *selection filters* (the filter output is always, by definition, one of the input samples). Although hybrid techniques combining linear and median filtering have been developed, they tend to be *ad hoc* in nature and prohibitively complex.

Weighted myriad filters (WMyF's) have been proposed as a class of nonlinear filters for robust non-Gaussian signal processing in impulsive noise environments [7]–[10]. These filters have been derived based on maximum likelihood location estimation from samples following the so-called α-stable distributions [3], [4]. The attractive features of α-stable distributions are that they include the Gaussian distribution as a special limiting case while possessing heavier tails than the Gaussian as well as Laplacian distributions. As a result, WMyF's constitute a robust generalization of linear filtering that is at the same time inherently more powerful than weighted median filters. Myriad filters have been successfully employed in robust communications and image processing applications [11]–[13].

The class of WMyF's is derived from the *sample myriad*, which is an M-estimator of location [14] for the class of α-stable distributions. Given a set of samples $\{x_i\}_{i=1}^N$, an M-estimator of location is given by $\hat{\theta} \triangleq \arg\min_\theta \sum_{i=1}^N \rho(x_i - \theta)$, where $\rho(\cdot)$ is called the *cost function* of the M-estimator. Maximum likelihood location estimators are special cases of M-estimators with $\rho(x) \sim -\log f(x)$, where $f(x)$ is the density function of the samples. Using the Gaussian and Laplacian density functions, we obtain the cost functions for the *sample mean* and the *sample median* as $\rho(x) = x^2$ and $\rho(x) = |x|$, respectively. The sample myriad is defined using the cost function $\rho(x) = \log(K^2 + x^2)$, where the so-called *linearity parameter* K controls the impulse-resistance (outlier-rejection capability) of the estimator; a more detailed description is given in Section II. Table I shows the cost functions and the outputs for the linear (mean), median, and myriad filters. In each row of the table, the filter output is the value that minimizes the associated cost function. These filters are generalized to their weighted versions by introducing non-negative weights $\{w_i\}_{i=1}^N$ in the cost function expressions. The notations $w_i \diamond x_i$ (for the weighted median) and $w_i \circ x_i$ (for the weighted myriad), shown in the last column of the table, reflect these weighting operations. In the case of the weighted median with *integer* weights, the expression $w_i \diamond x_i$ has the added significance of

Manuscript received February 4, 1998; revised June 16, 1999. This work was supported in part by the National Science Foundation under Grant MIP-9530923 and 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 Dr. Ali H. Sayed.

S. Kalluri was with the Department of Electrical and Computer Engineering, University of Delaware, Newark, DE 19716 USA. He is now with Level One Communications, Inc., Sacramento, CA 95827 USA (e-mail: skalluri@level1.com).

G. R. Arce 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(00)00109-4.

TABLE I
M-ESTIMATOR COST FUNCTIONS AND FILTER OUTPUTS FOR VARIOUS FILTER FAMILIES

Filter	Cost Function	Filter Output
Linear	$\sum_{i=1}^{N}(x_i-\theta)^2$	mean $\{x_1, x_2, \ldots, x_N\} = \sum_{i=1}^{N} x_i \;/\; N$
Median	$\sum_{i=1}^{N}\lvert x_i-\theta\rvert$	median $\{x_1, x_2, \ldots, x_N\}$
Myriad	$\sum_{i=1}^{N}\log\left[K^2+(x_i-\theta)^2\right]$	myriad $\{x_1, x_2, \ldots, x_N; K\}$
Weighted Mean	$\sum_{i=1}^{N} w_i(x_i-\theta)^2$	$\sum_{i=1}^{N} w_i x_i \;/\; \sum_{i=1}^{N} w_i$
Weighted Median	$\sum_{i=1}^{N} w_i\lvert x_i-\theta\rvert$	median $\{w_i \diamond x_i\}_{i=1}^{N}$
Weighted Myriad	$\sum_{i=1}^{N}\log\left[K^2+w_i(x_i-\theta)^2\right]$	myriad $\{w_1 \circ x_1, w_2 \circ x_2, \ldots, w_N \circ x_N; K\}$

denoting the *replication* of the sample x_i by the integer w_i; the filter output is then the (unweighted) median of a modified set of observations, where each sample x_i appears w_i times.

As Table I shows, it is trivial to compute the weighted mean. The weighted median can also be determined directly; however, it requires sorting the input samples, making it a computationally expensive task. There has, therefore, been considerable research to develop fast algorithms to compute the weighted median. The weighted myriad, on the other hand, is not even available in explicit form. A direct computation of the weighted myriad is therefore a nontrivial and prohibitively expensive task since it involves the minimization of the associated cost function shown in the last row of Table I. In this paper, we first define a certain *mapping* having several *fixed points* and show that the weighted myriad is one of these fixed points. It is the particular fixed point that minimizes the weighted myriad cost function of Table I. We propose an iterative algorithm to compute these fixed points. We then develop fast algorithms, incorporating these *fixed point iterations* for the computation of the weighted myriad. The performance of these algorithms is evaluated using a numerical example. It is shown that these algorithms achieve a high degree of accuracy in approximating the weighted myriad at a relatively low cost of computation. Using these algorithms, the full potential of the class of WMyF's can now be realized in robust signal processing and communications applications.

The paper is organized as follows. Section II introduces the weighted myriad. In Section III, we present iterative algorithms for fixed-point computation, including a proof of their convergence. Fast algorithms for weighted myriad computation are developed in Section IV. Computer simulations illustrating these algorithms are presented in Section V.

II. THE WEIGHTED MYRIAD

This section briefly introduces the *weighted myriad* and develops some of its properties that will be useful later in the paper. For a more detailed treatment, see [7]–[9] and [11].

The class of WMyF's is derived from the so-called sample *myriad*, which is defined as the maximum likelihood estimate (MLE) of the location parameter of data following the Cauchy distribution. Consider N independent and identically distributed (i.i.d.) random variables $\{X_i\}_{i=1}^{N}$, each following a Cauchy distribution with location parameter θ and scaling factor $K > 0$. Thus, $X_i \sim \text{Cauchy}(\theta, K)$ with the density function

$$f_{X_i}(x_i;\theta,K) = \frac{K}{\pi}\cdot\frac{1}{K^2+(x_i-\theta)^2} = \frac{1}{K}f\left(\frac{x_i-\theta}{K}\right) \tag{1}$$

where $f(v) \triangleq 1/\pi \cdot 1/(1+v^2)$ is the density function of a *standard* Cauchy random variable: $(X_i-\theta)/K \sim \text{Cauchy}(0, 1)$. Given a set of observations $\{x_i\}_{i=1}^{N}$, the sample myriad $\hat{\theta}_K$ maximizes the likelihood function $\prod_{i=1}^{N} f_{X_i}(x_i;\theta,K)$. Equivalently, using (1) and some manipulation, we obtain

$$\hat{\theta}_K = \arg\min_{\theta}\prod_{i=1}^{N}\left[1+\left(\frac{x_i-\theta}{K}\right)^2\right]. \tag{2}$$

Notably, the sample myriad reduces to the sample mean as $K \to \infty$ [8].

By assigning non-negative weights to the input samples (observations), based on their varying levels of reliability, the weighted myriad is derived as a generalization of the sample myriad. This is done by assuming that the observations are drawn from N independent Cauchy distributed random variables, all having the same location parameter but varying scale factors. Given N observations $\{x_i\}_{i=1}^{N}$ and weights $\{w_i \geq 0\}_{i=1}^{N}$ define the input vector $\boldsymbol{x} \triangleq [x_1, x_2, \cdots, x_N]^T$ and the weight vector $\boldsymbol{w} \triangleq [w_1, w_2, \cdots, w_N]^T$. For a given *nominal* scale factor K, the underlying random variables $\{X_i\}_{i=1}^{N}$ are assumed to be Cauchy distributed with location parameter θ and scale factors $\{S_i\}_{i=1}^{N}$: $X_i \sim \text{Cauchy}(\theta, S_i)$, where

$$S_i \triangleq \frac{K}{\sqrt{w_i}} > 0, \qquad i = 1, 2, \cdots, N. \tag{3}$$

Increasing the weight w_i (thus decreasing the scale S_i) causes the distribution of X_i to be more concentrated around θ, making

X_i a more reliable sample. Note that the sample myriad is included as a special case: When all the weights are equal to unity, the scale factors all reduce to $S_i = K$, leading to the sample myriad at the nominal scale factor K.

The weighted myriad $\hat{\theta}_K(\boldsymbol{w}, \boldsymbol{x})$ maximizes the likelihood function $\prod_{i=1}^{N} f_{X_i}(x_i; \theta, S_i)$. Using (1) for $f_{X_i}(x_i; \theta, S_i)$, the weighted myriad can be expressed as

$$\begin{aligned}\hat{\theta}_K(\boldsymbol{w}, \boldsymbol{x}) &= \arg\min_{\theta} P(\theta) \\ &\triangleq \arg\min_{\theta} \prod_{i=1}^{N} \left[1 + \left(\frac{x_i - \theta}{S_i}\right)^2\right] \\ &= \arg\min_{\theta} \prod_{i=1}^{N} \left[1 + w_i \left(\frac{x_i - \theta}{K}\right)^2\right]. \end{aligned} \tag{4}$$

By rewriting (4), the weighted myriad $\hat{\theta}_K(\boldsymbol{w}, \boldsymbol{x}) \equiv \hat{\theta}$ can also be expressed as

$$\begin{aligned}\hat{\theta} &= \arg\min_{\theta} \log(P(\theta)) \triangleq \arg\min_{\theta} Q(\theta) \\ &= \arg\min_{\theta} \sum_{i=1}^{N} \log\left[1 + \left(\frac{x_i - \theta}{S_i}\right)^2\right] \end{aligned} \tag{5}$$

since $\log(\cdot)$ is a strictly increasing function. We refer to the function

$$Q(\theta) = \log(P(\theta)) = \sum_{i=1}^{N} \log\left[1 + \left(\frac{x_i - \theta}{S_i}\right)^2\right] \tag{6}$$

as the *weighted myriad objective function* since it is minimized by the weighted myriad. Note that when $w_i = 0$ ($S_i = \infty$), the corresponding term in $P(\theta)$ or $Q(\theta)$ drops out; the sample x_i is thus effectively ignored when its weight is zero.

The weighted myriad is an M-estimator [14]. To see this, introduce the function

$$\rho(v) \triangleq \log\left(1 + v^2\right). \tag{7}$$

We can then express the weighted myriad $\hat{\theta}_K(\boldsymbol{w}, \boldsymbol{x}) \equiv \hat{\theta}$ from (5) as

$$\hat{\theta} = \arg\min_{\theta} Q(\theta) = \arg\min_{\theta} \sum_{i=1}^{N} \rho\left(\frac{x_i - \theta}{S_i}\right) \tag{8}$$

which defines an M-estimator of location from samples of varying scale [14].

The computation of the weighted myriad $\hat{\theta}$ is complicated by the fact that the objective function $Q(\theta)$ can have several local minima, as we shall see presently. To derive some basic properties of $\hat{\theta}$, we examine $Q(\theta)$ further. First, use (6) to write the derivative of $Q(\theta)$ as

$$Q'(\theta) = \frac{P'(\theta)}{P(\theta)} = 2\sum_{i=1}^{N} \frac{\left(\frac{\theta - x_i}{S_i^2}\right)}{1 + \left(\frac{x_i - \theta}{S_i}\right)^2}. \tag{9}$$

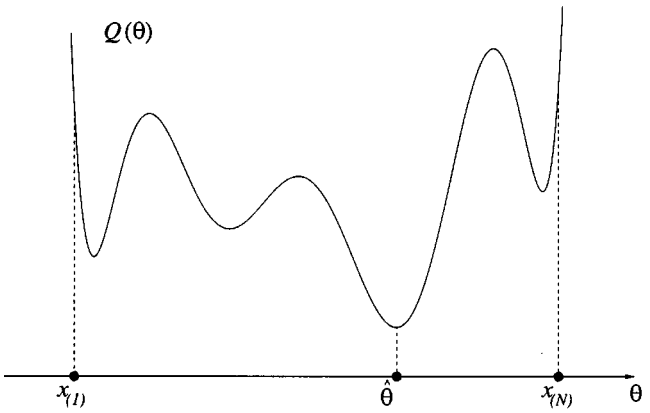


Fig. 1. Sketch of a typical weighted myriad objective function $Q(\theta)$.

The following proposition brings together a few key properties of $Q(\theta)$ and $\hat{\theta}$ that will be used in the later sections on the computation of $\hat{\theta}$. The properties described below are illustrated by Fig. 1, which shows the form of a typical objective function $Q(\theta)$.

Proposition 2.1: Let $\{x_{(j)}\}_{j=1}^{N}$ signify the order statistics (samples sorted in increasing order of amplitude) of the input vector $\boldsymbol{x}$, with $x_{(1)}$ the smallest and $x_{(N)}$ the largest. The following statements hold:

a) The objective function $Q(\theta)$ has a finite number [at most $(2N-1)$] of local extrema.
b) The weighted myriad $\hat{\theta}$ is one of the local minima of $Q(\theta)$: $Q'(\hat{\theta}) = 0$.
c) $Q'(\theta) > 0$ [$Q(\theta)$ strictly increasing] for $\theta > x_{(N)}$, and $Q'(\theta) < 0$ [$Q(\theta)$ strictly decreasing] for $\theta < x_{(1)}$.
d) All the local extrema of $Q(\theta)$ lie within the range $[x_{(1)}, x_{(N)}]$ of the input samples.
e) The weighted myriad is in the range of input samples: $x_{(1)} \leq \hat{\theta} \leq x_{(N)}$.

Proof:

a) We have $Q(\theta) = \log(P(\theta))$ from (6). The function $P(\theta)$, given from (4) by

$$P(\theta) = \prod_{i=1}^{N} \left[1 + \left(\frac{x_i - \theta}{S_i}\right)^2\right] \tag{10}$$

is a polynomial in θ of degree $2N$ with well-defined derivatives of all orders. Its derivative $P'(\theta)$ is a polynomial of degree $(2N-1)$ with at most $(2N-1)$ real roots. Now, $Q'(\theta) = P'(\theta)/P(\theta)$ from (9), and it is clear from (10) that $P(\theta) \neq 0$ for any θ. Hence, the roots of $Q'(\theta)$ and $P'(\theta)$ are identical. Therefore, $Q'(\theta)$ also has at most $(2N-1)$ real roots, that is, $Q(\theta)$ has at most $(2N-1)$ local extrema.

b) From a), it is clear that $Q(\theta)$ is a sufficiently smooth function that is defined for all real θ and having derivatives of all orders. In addition, from (6), $Q(\pm\infty) = +\infty$. It follows that the global minimum of $Q(\theta)$ (which is the weighted myriad $\hat{\theta}$) must occur at one of its local minima.

c) Let $\theta > x_{(N)} \Rightarrow (\theta - x_{(N)}) > 0$. Then, since $x_i \leq x_{(N)}\ \forall i \in \{1, 2, \cdots, N\}$, we have $(\theta - x_i) \geq (\theta - x_{(N)}) > 0\ \forall i \in \{1, 2, \cdots, N\}$. Using this in (9), we

obtain $Q'(\theta) > 0$ strictly. Similarly, $\theta < x_{(1)} \Rightarrow (\theta - x_i) \leq (\theta - x_{(1)}) < 0 \; \forall i \in \{1, 2, \cdots, N\}$. Then, from (9), $Q'(\theta) < 0$ strictly.

d) From c), we see that $Q'(\theta) \neq 0$ if $\theta > x_{(N)}$ or $\theta < x_{(1)}$. Thus, for real θ, $Q'(\theta) = 0 \Rightarrow \theta \in [x_{(1)}, x_{(N)}]$. That is, the real roots of $Q'(\theta)$, which are the local extrema of $Q(\theta)$, lie in the range $[x_{(1)}, x_{(N)}]$ of the input samples.

e) This follows from b) and d). This completes the proof of the proposition.

The weighted myriad $\hat{\theta}$ is a solution of the equation $Q'(\theta) = 0$. Referring to (7), define

$$\psi(v) \triangleq \rho'(v) = \frac{2v}{1+v^2} \tag{11}$$

which is called the *influence function* of an M-estimator. Then, we can use (9) to write the following equation for the local extrema of $Q(\theta)$:

$$Q'(\theta) = -\sum_{i=1}^{N} \frac{1}{S_i} \cdot \psi\left(\frac{x_i - \theta}{S_i}\right) = 0. \tag{12}$$

As a final note, we can use (3) and (12) to show that when $K \to \infty$ with the weights w_i held constant, there is a *single* local extremum, and $\hat{\theta}_K \to \hat{\theta}_\infty = \sum_{i=1}^{N} w_i x_i / \sum_{i=1}^{N} w_i$, which is the (linear) *weighted mean*. Hence, we have the name *linearity parameter* for the nominal scale factor K.

III. Fixed Point Iterations for Weighted Myriad Computation

The weighted myriad is one of the *real* roots of the function $Q'(\theta)$ of (9). In this section, these roots are formulated as fixed points of a mapping, and an iterative algorithm is presented for their computation.

Referring to (12), introduce the *positive* functions

$$h_i(\theta) \triangleq \frac{1}{S_i^2} \cdot \varphi\left(\frac{x_i - \theta}{S_i}\right) > 0, \qquad i = 1, 2, \cdots, N \tag{13}$$

where

$$\varphi(v) \triangleq \frac{\psi(v)}{v} = \frac{2}{1+v^2}. \tag{14}$$

We can then recast (12) as

$$Q'(\theta) = -\sum_{i=1}^{N} h_i(\theta) \cdot (x_i - \theta) = 0 \tag{15}$$

for the local extrema of $Q(\theta)$. This formulation implies that the *sum of weighted deviations* of the samples is zero with the (positive) weights themselves being functions of θ.

A. Fixed-Point Formulation

Rewriting (15) as

$$\theta = \frac{\sum_{i=1}^{N} h_i(\theta) \cdot x_i}{\sum_{i=1}^{N} h_i(\theta)} \tag{16}$$

we see that each local extremum of $Q(\theta)$, including the weighted myriad $\hat{\theta}$, can be written as a *weighted mean* of the input samples x_i. Since the weights $h_i(\theta)$ are always positive, the right-hand side of (16) is in the interval $(x_{(1)}, x_{(N)})$, confirming that all the local extrema lie within the range of the input samples. By defining the mapping

$$T(\theta) \triangleq \frac{\sum_{i=1}^{N} h_i(\theta) \cdot x_i}{\sum_{i=1}^{N} h_i(\theta)} \tag{17}$$

the local extrema of $Q(\theta)$, or the roots of $Q'(\theta)$, are seen to be the *fixed points* of $T(\cdot)$

$$\theta^* = T(\theta^*). \tag{18}$$

We propose the following *fixed-point iteration* algorithm to compute these fixed points:

$$\theta_{m+1} \triangleq T(\theta_m) = \frac{\sum_{i=1}^{N} h_i(\theta_m) \cdot x_i}{\sum_{i=1}^{N} h_i(\theta_m)}. \tag{19}$$

In the classical literature, this is also called the *method of successive approximation* for the solution of the equation $\theta = T(\theta)$ [15]. In Section III-B, we prove that the iterative scheme of (19) converges to a fixed point of $T(\cdot)$; thus

$$\lim_{m \to \infty} \theta_m = \theta^* = T(\theta^*). \tag{20}$$

Note that there can be as many as $(2N-1)$ fixed points θ^*; the initial value θ_0 chosen in (19) determines the particular fixed-point obtained.

A different perspective on (19) can be obtained by using (15) to rewrite the recursion as

$$\theta_{m+1} = \theta_m - \frac{Q'(\theta_m)}{H(\theta_m)} \tag{21}$$

where $H(\theta_m) \triangleq \sum_{i=1}^{N} h_i(\theta_m) > 0$. To interpret (21), consider the *tangent* of $Q(\theta)$ at $\theta = \theta_m$: $Y(\theta) \triangleq Q(\theta_m) + Q'(\theta_m)(\theta - \theta_m)$. Then, we have $Y(\theta_{m+1}) = Q(\theta_m) - \{[Q'(\theta_m)]^2 / H(\theta_m)\} \leq Q(\theta_m)$. Thus, considering $Y(\theta)$ as a linear approximation of $Q(\theta)$ around the point θ_m, the update *attempts* to reduce $Q(\cdot)$: $Q(\theta_{m+1}) \approx Y(\theta_{m+1}) \leq Q(\theta_m)$. This does not guarantee that $Q(\theta_{m+1}) \leq Q(\theta_m)$; however, it is shown in Section III-B that (21) does in fact decrease $Q(\theta)$ at each iteration.

We can contrast the recursion of (19) with the update in Newton's method [15] for the solution of the equation $Q'(\theta) = 0$:

$$\theta'_{m+1} \triangleq \theta_m - \frac{Q'(\theta_m)}{Q''(\theta_m)} \tag{22}$$

which is interpreted by considering the *tangent* of $Q'(\theta)$ at $\theta = \theta_m$: $Z(\theta) \triangleq Q'(\theta_m) + Q''(\theta_m)(\theta - \theta_m)$. Here, $Z(\theta)$ is used

as a linear approximation of $Q'(\theta)$ around the point θ_m, and θ'_{m+1} is the point at which the tangent $Z(\theta)$ crosses the θ axis: $Q'(\theta'_{m+1}) \approx Z(\theta'_{m+1}) = 0$.

Although Newton's method can have fast (quadratic) convergence, the major disadvantage of this method is that it may converge only if the initial value θ_0 is sufficiently close to the solution θ^* [15]. Thus, only local convergence is guaranteed. The conditions for convergence [15] are also difficult to verify in practice, especially since they have to be determined for each specific function $Q(\theta)$. Further, Newton's method can converge to either a local minimum or a local maximum of $Q(\theta)$ (depending on the initial value θ_0), whereas we need only the local minima in order to compute the weighted myriad. On the other hand, as we shall prove in Section III-B, the fixed-point iteration scheme of (19) decreases the objective function $Q(\theta)$ continuously at each step, leading to *global convergence* (convergence from an arbitrary starting point) to the local *minima* of $Q(\theta)$.

B. Convergence of Fixed-Point Iterations

In this section, we prove that the sequence $\{\theta_m\}$ of (19) converges to one of the local extrema of the objective function $Q(\theta)$. We show, in fact, that except for a degenerate case with zero probability of occurrence, $\{\theta_m\}$ always converges to a local *minimum*. As a first step, we show in the following theorem (Theorem 3.1) that the recursion (19) *decreases* $Q(\theta)$. The proof of this theorem uses the following lemma, which reveals the updated value θ_{m+1} as the solution to a weighted least-squares problem at each iteration.

Lemma 3.1: The value θ_{m+1} in (19) is the global minimizer of the function

$$B_m(\theta) \triangleq \sum_{i=1}^{N} h_i(\theta_m) \cdot (x_i - \theta)^2. \tag{23}$$

Proof: Since $B_m(\theta)$ is quadratic in θ, its global minimizer is the unique solution of the derivative equation $B'_m(\theta) = 0$. From (23), $B'_m(\theta_{m+1}) = -2\sum_{i=1}^{N} h_i(\theta_m) \cdot (x_i - \theta_{m+1}) = 0$; the second equality is easily derived from the definition of θ_{m+1} in (19). Hence, we have the result.

Theorem 3.1: Consider the sequence $\{Q_m \triangleq Q(\theta_m)\}$, with $\{\theta_m\}$ given by (19). Let $R \triangleq x_{(N)} - x_{(1)}$ be the range of the input samples. Then, we have the following.

a) $Q_{m+1} < Q_m$ strictly, if $\theta_{m+1} \neq \theta_m$. If $\theta_{m+1} = \theta_m$, then $Q'(\theta_m) = 0$, and θ_m is a local extremum of $Q(\theta)$.

b)

$$\begin{aligned} Q_m - Q_{m+1} &\geq \left(\tfrac{1}{2}\sum_{i=1}^{N} h_i(\theta_m)\right) \cdot (\theta_{m+1} - \theta_m)^2 \\ &= \left(\sum_{i=1}^{N} \frac{1}{S_i^2 + (x_i - \theta_m)^2}\right) \cdot (\theta_{m+1} - \theta_m)^2 \\ &\geq \left(\sum_{i=1}^{N} \frac{1}{S_i^2 + R^2}\right) \cdot (\theta_{m+1} - \theta_m)^2 \end{aligned} \tag{24}$$

where the last inequality holds if $\theta_m \in [x_{(1)}, x_{(N)}]$.

Proof: See Appendix A.

Remarks:

i) If $\theta_{m+1} = \theta_m$, the sequence $\{\theta_m\}$ evidently converges since all subsequent values $\{\theta_{m+2}, \cdots\}$ are also equal to θ_m.

ii) The condition $\theta_m \in [x_{(1)}, x_{(N)}]$ in b) is not restrictive since even if the initial value θ_0 is chosen outside $[x_{(1)}, x_{(N)}]$, (19) shows that θ_m, $m > 0$ will all lie within this interval.

iii) This theorem exploits parts of [14, Sec. 7.8], which deals with the computation of regression M-estimates.

Corollary 3.1.1: The sequence $\{Q_m\}$ converges; $Q_m \downarrow Q^*$, where $Q^* \triangleq \inf(\{Q_m\}_{m=1}^{\infty})$.

Proof: It is evident from the theorem that $Q_{m+1} \leq Q_m$. Thus, $\{Q_m = Q(\theta_m)\}$ is a decreasing sequence, and it is bounded below since $Q(\theta) \geq 0$ from (6). Hence, the sequence converges to its infimum: $Q_m \downarrow Q^*$, where $Q^* \triangleq \inf(\{Q_m\}_{m=1}^{\infty})$.

Corollary 3.1.2: $|\theta_{m+1} - \theta_m| \to 0$ and the sequence of derivatives $Q'(\theta_m) \to 0$.

Proof: Using (24), we can write

$$|\theta_{m+1} - \theta_m| \leq \eta\sqrt{Q_m - Q_{m+1}} \tag{25}$$

where

$$\eta \triangleq \left(\sum_{i=1}^{N} \frac{1}{S_i^2 + R^2}\right)^{-(1/2)} > 0.$$

Since $\{Q_m\}$ converges (Corollary 3.1.1), we have $(Q_m - Q_{m+1}) = |Q_{m+1} - Q_m| \to 0$. It then follows from (25) that $|\theta_{m+1} - \theta_m| \to 0$, proving the first part of the corollary. Now, using (21), we have

$$Q'(\theta_m) = -\left(\sum_{i=1}^{N} h_i(\theta_m)\right) \cdot (\theta_{m+1} - \theta_m) \tag{26}$$

which, together with (13), leads to

$$|Q'(\theta_m)| \leq \left(\sum_{i=1}^{N} \frac{2}{S_i^2}\right) \cdot |\theta_{m+1} - \theta_m|. \tag{27}$$

Since $|\theta_{m+1} - \theta_m| \to 0$, it follows from (27) that $Q'(\theta_m) \to 0$. This completes the proof.

Fig. 2 illustrates the behavior of the sequence $\{\theta_m\}$ by depicting the two possible scenarios that are described in Theorem 3.1. In the first case, we have a sequence of distinct elements ($\theta_{m+1} \neq \theta_m$ for any m), which always decreases $Q(\theta)$. We will show later in this section that the sequence in this case converges to a local *minimum*, which is shown as θ_1^* in the figure. The second case in the figure depicts a situation where the sequence terminates at a particular iteration m when $\theta_{m+1} = \theta_m = \theta_2^*$, where θ_2^* is a local extremum of $Q(\theta)$ (it happens to be a local *maximum* in this figure). Note that in both cases, the sequence stays within the range of the input samples. In addition, the sequence proceeds always in such a way that $(\theta_m - \theta_{m+1})$ has the same sign as the derivative $Q'(\theta_m)$ at the current iteration.

Corollary 3.1.2 is not enough to ensure the convergence of the sequence $\{\theta_m\}$ to a solution of the equation $Q'(\theta) = 0$. It is instrumental, however, in establishing the next theorem (Theorem 3.2), which states that after a finite number of iterations, *all* subsequent values of θ_m are confined between the two local maxima adjacent to *one* of the local minima of $Q(\theta)$. The choice of the initial value θ_0 will determine the particular local minimum around which the sequence $\{\theta_m\}$ is ultimately localized.

Theorem 3.2: For each iteration m, let a_m and b_m ($a_m < b_m$) be the two adjacent local maxima of $Q(\theta)$ such that $\theta_m \in [a_m, b_m]$. Let θ_m^* denote the local minimum of $Q(\theta)$ lying within this interval: $\theta_m^* \in (a_m, b_m)$. Assume that $\{\theta_m\}$ is such that $\forall\ m, \theta_{m+1} \neq \theta_m$. Then, $\exists M$ such that $\theta_m \in (a_M, b_M)\ \forall m \geq M$.

Proof: First, since $\theta_{m+1} \neq \theta_m$, Theorem 3.1 shows that θ_m is not a local extremum for any m; thus, θ_m is within the open interval (a_m, b_m). Now, from Corollary 3.1.2, $|\theta_{m+1} - \theta_m| \to 0$. Therefore, given any $\epsilon > 0$, $\exists M(\epsilon)$ such that $|\theta_{m+1} - \theta_m| < \epsilon\ \forall m \geq M(\epsilon)$. In particular, choose $\epsilon = \epsilon_0 \triangleq \min\{|\theta^* - \phi^*|\colon Q'(\theta^*) = Q'(\phi^*) = 0\}$; then ϵ_0 is also the smallest distance between *adjacent* local extrema of $Q(\theta)$. Letting $M \triangleq M(\epsilon_0)$, it follows that

$$\forall m \geq M, \quad \begin{array}{c} |\theta_{m+1} - \theta_m| < \epsilon_0 \leq b_M - \theta_M^* \\ \text{and} \\ |\theta_{m+1} - \theta_m| < \epsilon_0 \leq \theta_M^* - a_M. \end{array} \tag{28}$$

Now, let $m = M \triangleq M(\epsilon_0)$. We shall show that $\theta_{M+1} \in (a_M, b_M)$. Referring to Fig. 3, consider separately the following two cases:

Case I—$a_M < \theta_M < \theta_M^* < b_M$: Fig. 3(a) depicts this situation, where $\theta_M < \theta_M^*$. Now, it is clear from Section II [in particular, the proof of Proposition 2.1(a)] that $Q(\theta)$ is a sufficiently smooth function, having continuous derivatives of all orders. It easily follows that its derivative $Q'(\theta) < 0$ for θ between any local maximum and the local minimum immediately to the right of it. In particular, we have $Q'(\theta_M) < 0$. Referring to (26) and knowing that $(\sum_{i=1}^{N} h_i(\theta_m)) > 0$, we then have $\theta_{M+1} > \theta_M$, as shown in the figure. Using this fact and (28), we obtain $\theta_{M+1} - \theta_M = |\theta_{M+1} - \theta_M| < b_M - \theta_M^* < b_M - \theta_M$; we do the final step because $\theta_M < \theta_M^*$. This implies $\theta_{M+1} < b_M$. Together with the fact that $\theta_{M+1} > \theta_M$, this shows that $\theta_{M+1} \in (\theta_M, b_M) \subset (a_M, b_M)$.

Case II—($a_M < \theta_M^* < \theta_M < b_M$: This case is illustrated in Fig. 3(b). Arguing as in Case I, we can show that $Q'(\theta_M) > 0$ for $\theta_M^* < \theta_M < b_M$. Using (26) again, we now have $\theta_{M+1} < \theta_M$. Note that in this case, we have shown θ_{M+1} in the figure to be on the same side of θ_M^* as θ_M, unlike in Fig. 3(a). In fact, in both cases I and II, either one of $\theta_{M+1} < \theta_M^*$ and $\theta_{M+1} > \theta_M^*$ could be true. Now, using $\theta_{M+1} < \theta_M$, together with (28), leads to $\theta_M - \theta_{M+1} = |\theta_{M+1} - \theta_M| < \theta_M^* - a_M < \theta_M - a_M$, where the last step is due to $\theta_M^* < \theta_M$. Thus, $\theta_{M+1} > a_M$. Combining this with $\theta_{M+1} < \theta_M$, we obtain $\theta_{M+1} \in (a_M, \theta_M) \subset (a_M, b_M)$.

We have thus shown that $\theta_{M+1} \in (a_M, b_M)$. This implies that $a_{M+1} = a_M$ and $b_{M+1} = b_M$, that is, θ_{M+1} is between the same adjacent local maxima, as is θ_M. We can continue this process, using the same arguments as in Cases I and II above, to show inductively that $\theta_m \in (a_M, b_M)\ \forall m \geq M$, where $M \triangleq M(\epsilon_0)$. This completes the proof of the theorem.

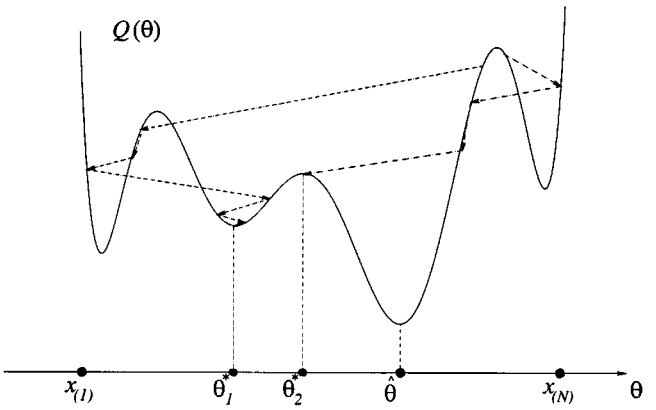


Fig. 2. Typical scenarios in the behavior of the sequence $\{\theta_m\}$ of fixed-point iterations.

Remark: The proof holds, with very trivial modifications, for the cases where θ_m lies: i) between $x_{(1)}$ and the local maximum immediately to the right of $x_{(1)}$ or ii) between $x_{(N)}$ and the local maximum immediately to its left. These two situations can be tackled by defining $a_m = x_{(1)}$ in i) and $b_m = x_{(N)}$ in ii). This remark also applies when the same situations arise in other theorems and proofs of this paper.

We have thus established that the sequence $\{\theta_m\}$ eventually stays within the interval (a_M, b_M) around the local minimum θ_M^*. This is a key result that is used in the proof of the following theorem, which supplies the limit of the sequence of values $\{Q_m = Q(\theta_m)\}$ of the objective function.

Theorem 3.3: The limit of the sequence $\{Q_m = Q(\theta_m)\}$ is $Q(\theta_M^*)$; $Q_m \downarrow \inf(\{Q_m\}_{m=1}^{\infty}) = Q(\theta_M^*)$, where θ_M^* is defined in the proof of Theorem 3.2.

Proof: See Appendix B.

Using Theorem 3.3 and the smoothness properties of $Q(\theta)$, the convergence of the sequence $\{\theta_m\}$ is finally established in the following theorem.

Theorem 3.4: Consider the sequence $\{\theta_m\}$ of fixed-point iterations defined in (19). Then, we have the following.

a) If (and only if) $\theta_{m+1} = \theta_m$ for some $m = m_0$, then $Q'(\theta_{m_0}) = 0$ and θ_{m_0} is a local extremum of $Q(\theta)$. The sequence then terminates at θ_{m_0}, i.e., $\theta_m = \theta_{m_0}\ \forall m \geq m_0$.

b) If $\{\theta_m\}$ is such that $\forall\ m, \theta_{m+1} \neq \theta_m$, then the sequence converges to a local *minimum* θ^* of $Q(\theta)$: $\theta_m \to \theta^* \triangleq \theta_M^*$, where θ_M^* is defined in the proof of Theorem 3.2.

Proof:

a) See Theorem 3.1 and Remark i), which follows it.

b) We shall prove the convergence of the subsequence $\{\theta_m\colon m \geq M\}$, which implies the convergence of $\{\theta_m\}_{m=1}^{\infty}$. Let $Q^* \triangleq Q(\theta_M^*)$. Given any $\epsilon > 0$, define the quantities $\delta_1 > 0$, $\delta_2 > 0$ such that $Q(\theta^* - \epsilon) = Q^* + \delta_1$ and $Q(\theta^* + \epsilon) = Q^* + \delta_2$. These are shown in Fig. 4, which shows the objective function $Q(\theta)$ in the interval

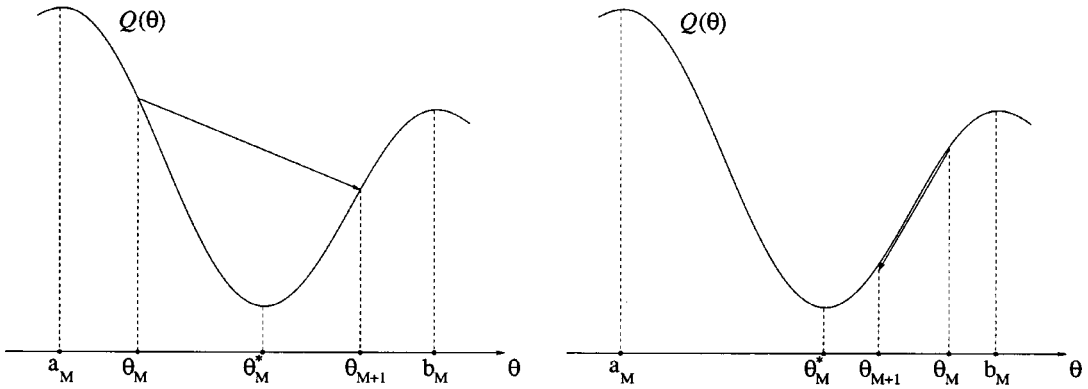


Fig. 3. Two cases in the proof of Theorem 3.2: (i) Case I ($\theta_M < \theta_M^*$) and (ii) Case II ($\theta_M > \theta_M^*$).

$[a_M, b_M]$ within which the subsequence $\{\theta_m : m \geq M\}$ is confined. It is assumed (without prejudice to the veracity of the theorem) that ϵ is small enough to ensure that $\theta^* - \epsilon$ and $\theta^* + \epsilon$ fall within the interval (a_M, b_M). Let $\delta = \delta(\epsilon) \triangleq \min\{\delta_1, \delta_2\}$. Assume, without loss of generality, that $\delta = \delta_2$; this is the situation represented in the figure. Now, define $\tilde{\epsilon} > 0$ such that $Q(\theta^* - \tilde{\epsilon}) = Q(\theta^* + \epsilon) = Q^* + \delta_2$. Note that $\tilde{\epsilon} \leq \epsilon$, as shown in the figure. That is, $\theta^* - \epsilon \leq \theta^* - \tilde{\epsilon}$, since $Q(\theta^* - \epsilon) = Q^* + \delta_1 \geq Q^* + \delta_2 = Q(\theta^* - \tilde{\epsilon})$, and $Q(\theta)$ is a decreasing function ($Q'(\theta) < 0$) for $a_M < \theta < \theta^*$. Now, we have $Q_m \downarrow Q^*$ from Theorem 3.3. Therefore, given the value $\delta = \delta(\epsilon) > 0$ defined above, $\exists L \equiv L(\delta(\epsilon))$ such that $\forall\, m > L$, $|Q_m - Q^*| = Q_m - Q^* < \delta \Rightarrow Q_m < Q^* + \delta \Rightarrow \theta_m \in (\theta^* - \tilde{\epsilon}, \theta^* + \epsilon) \Rightarrow \theta_m \in (\theta^* - \epsilon, \theta^* + \epsilon)$. Thus, we have shown that for any $\epsilon > 0$, $\exists L$ such that $\forall\, m > L$, $|\theta_m - \theta^*| < \epsilon$. Hence, $\theta_m \to \theta^* \triangleq \theta_M^*$; this completes the proof of the theorem.

Note that the degenerate case corresponding to Theorem 3.4(a) is an event that occurs with zero probability. Clearly then, the sequence $\{\theta_m\}$ of fixed-point iterations defined in (19) converges with probability one to a local *minimum* of the objective function $Q(\theta)$. Exploiting this property to find algorithms to compute the weighted myriad is the subject of the next section.

IV. Fast Weighted Myriad Computation Algorithms

The weighted myriad $\hat{\theta}$ globally minimizes the objective function $Q(\theta)$ or, equivalently, the *polynomial objective function* $P(\theta)$, which is given from (4) by

$$P(\theta) = \exp(Q(\theta)) = \prod_{i=1}^{N} \left[1 + \left(\frac{x_i - \theta}{S_i}\right)^2\right]. \qquad (29)$$

For computational purposes, it is more economical to use the polynomial version $P(\theta)$ rather than using the function $Q(\theta)$. From Proposition 2.1, $\hat{\theta}$ is *one* of the local minima of $P(\theta)$ or one of the *real* roots of the derivative function $P'(\theta)$. Further, all these roots lie within the range $[x_{(1)}, x_{(N)}]$ of the input samples.

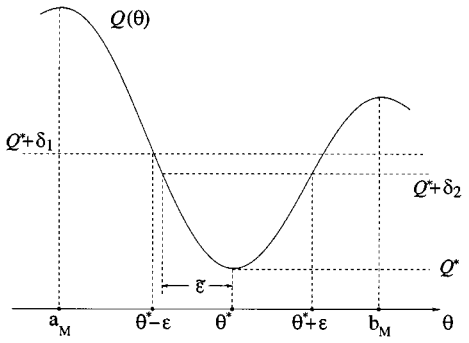


Fig. 4. Depiction of the proof of Theorem 3.4.

The fixed-point iterations $\theta_{m+1} = T(\theta_m)$ proposed in Section III [see (19)] converge to the real roots of $P'(\theta)$ for any initial value θ_0. In fact, these recursions converge almost surely to the local *minima* rather than the local *maxima* of $P(\theta)$ (see Theorem 3.4). Based on these observations, we can use the following *generic* approach to compute the weighted myriad:

- Step 1) Choose a finite set $\mathcal{C}_0$ of initial values θ_0 with $\mathcal{C}_0 \subset [x_{(1)}, x_{(N)}]$.
- Step 2) For each $\theta_0 \in \mathcal{C}_0$, implement the fixed point recursion of (19) for a desired number of iterations L: $\theta_{m+1} = T(\theta_m)$, $m = 0, 1, \cdots, L-1$. This forms the set $\tilde{\mathcal{C}}^* \triangleq T^{(L)}(\mathcal{C}_0)$ of *estimates* $\tilde{\theta}^*$ of the local minima of $P(\theta)$, where $T^{(L)}(\cdot)$ denotes the mapping $T(\cdot)$ applied L times. The elements of $\tilde{\mathcal{C}}^*$ are the *candidates* for the weighted myriad.
- Step 3) The weighted myriad is then computed as the element of $\tilde{\mathcal{C}}^*$ that minimizes the polynomial objective function $P(\theta)$: $\hat{\theta} \approx \arg\min_{\tilde{\theta}^* \in \tilde{\mathcal{C}}^*} P(\tilde{\theta}^*)$.

The choice of the set $\mathcal{C}_0$ in Step 1) above leads to different versions of the generic algorithm, with varying complexity and accuracy, which also depends on the choice of the number of iterations L. Recall from Proposition 2.1 that there are at most $(2N-1)$ local extrema of $P(\theta)$ with at most N local minima. One way of improving the accuracy of the algorithm could be to choose a large number of values for the initial set $\mathcal{C}_0$ by a fine sampling of the interval of interest $[x_{(1)}, x_{(N)}]$. This approach might increase the likelihood that the final candidate set $\tilde{\mathcal{C}}^*$ (of Step 2) includes *all* the local minima of $P(\theta)$, although this is not guaranteed to be the case. However, this is computationally

TABLE II
COMPLEXITIES OF WEIGHTED MYRIAD ALGORITHMS (WINDOW SIZE N, L FIXED—POINT ITERATIONS); $\nu(m)$ IS THE COMPLEXITY IN FINDING THE ROOTS OF A POLYNOMIAL OF DEGREE m AND VARIES WITH THE PARTICULAR ROOT FINDING METHOD USED

Algorithm	Multiplications	Additions	Other Operations
Polynomial Root Finding	$(9N^2+7N-7)$	$(6N^2+3N-6)$	$\nu(2N-1)$ $+\mathcal{O}(2N-1)$
Algorithm I	$L(4N^2+N)+(3N^2+1)$	$L(4N^2-2N)+2N^2$	$\mathcal{O}(N)$
Algorithm II	$L(4N+1)+(3N^2+1)$	$L(4N-2)+2N^2$	$\mathcal{O}(N)$

very expensive and involves finding the order statistics $x_{(1)}$ and $x_{(N)}$. As a tradeoff between the demands of speed and accuracy, we propose the following algorithm, which chooses $\mathcal{C}_0$ to be the *set of input samples* $\{x_i\}_{i=1}^N$.

Fixed-Point Search Weighted Myriad Algorithm I (FPS–WMyI)

Step 1) Using each of the input samples $\{x_i\}_{i=1}^N$ as an initial value, perform L iterations of the fixed-point recursion $\theta_{m+1}=T(\theta_m)$ of (19). Denote the resulting final values as $\{y_i=T^{(L)}(x_i)\}_{i=1}^N$.

Step 2) The weighted myriad is chosen as the element of $\{y_i\}_{i=1}^N$ that minimizes the polynomial objective function $P(\theta)$ of (29): $\hat{\theta}_{\text{FPS–WMyI}}=\arg\min_{y_i} P(y_i)$.

The algorithm can be described compactly as

$$\hat{\theta}_{\text{FPS–WMyI}}=\arg\min_{T^{(L)}(x_i)} P\left(T^{(L)}(x_i)\right). \tag{30}$$

A much faster algorithm can be obtained by realizing that most of the N recursions in Step 1 above will converge to values y_i [local minima of $P(\theta)$] that are far from the weighted myriad. An input sample x_i that is close to the weighted myriad is likely to converge to the weighted myriad itself. Motivated by this fact, we define the *selection weighted myriad* $\hat{\theta}_s$ [16] as the *input sample* that minimizes the weighted myriad objective function $Q(\theta)$ or, equivalently, the polynomial objective function $P(\theta)$:

$$\hat{\theta}_s \triangleq \arg\min_{x_i} Q(x_i)=\arg\min_{x_i} P(x_i). \tag{31}$$

Using $\hat{\theta}_s$ as an initial value in the fixed-point recursion of (19), we obtain the following fast algorithm.

Fixed-Point Search Weighted Myriad Algorithm II (FPS–WMyI)

Step 1) Compute the selection weighted myriad: $\hat{\theta}_s=\arg\min_{x_i} P(x_i)$.

Step 2) Using $\hat{\theta}_s$ as the initial value, perform L iterations of the fixed-point recursion $\theta_{m+1}=T(\theta_m)$ of (19). The final value of these iterations is then chosen as the weighted myriad $\hat{\theta}_{\text{FPS–WMyII}}=T^{(L)}(\hat{\theta}_s)$.

This algorithm can be compactly written as

$$\hat{\theta}_{\text{FPS–WMyII}}=T^{(L)}\left(\arg\min_{x_i} P(x_i)\right). \tag{32}$$

Note that for the special case $L=0$ (meaning that no fixed point iterations are performed), both the above algorithms compute the selection weighted myriad $\hat{\theta}_{\text{FPS–WMyI}}=\hat{\theta}_{\text{FPS–WMyII}}=\hat{\theta}_s$. Now, compare the two algorithms for the same number of iterations $L\geq 1$. Suppose $\hat{\theta}_s$ happens to be the input sample x_k. Then, Algorithm II yields $\hat{\theta}_{\text{FPS–WMyII}}=T^{(L)}(\hat{\theta}_s)=T^{(L)}(x_k)$. On the other hand, from Step 1 of Algorithm I, $\{y_i=T^{(L)}(x_i)\}_{i=1}^N$; in particular, $y_k=T^{(L)}(x_k)$. Therefore, $\hat{\theta}_{\text{FPS–WMyII}}=y_k$. Now, if $\hat{\theta}_s=x_k$ is close to the weighted myriad $\hat{\theta}$, then y_k will be close to $\hat{\theta}$, and we will have $\hat{\theta}_{\text{FPS–WMyI}}=\arg\min_{y_i} P(y_i)=y_k=\hat{\theta}_{\text{FPS–WMyII}}$. In this case, both algorithms yield the same result, with Algorithm II being much faster. Suppose, however, that $\hat{\theta}_s$ is close to a local minimum of $P(\theta)$ that is different from the weighted myriad $\hat{\theta}$. Then, y_k will not be close to $\hat{\theta}$, and Algorithm I will choose some other y_i, $i\neq k$. In this case, Algorithm I will give a more accurate result than Algorithm II.

Computational Complexity: A direct computation of the weighted myriad requires finding the real roots of the $(2N-1)$-degree derivative polynomial $P'(\theta)$ (see Proposition 2.1). It is easy to show that the coefficients of $P'(\theta)$ can be found using $(3N^2+11N-9)$ multiplications and $(2N^2+5N-6)$ additions. The weighted myriad is the real root that minimizes the polynomial objective function $P(\theta)$ of (29). The computation of $P(\theta)$ for any θ requires $(3N-1)$ multiplications and $(2N)$ additions. Choosing the minimum out of a set of m values is an $\mathcal{O}(m)$ task. Each fixed-point iteration step of (19) requires $(4N+1)$ multiplications and $(4N-2)$ additions. Let $\nu(m)$ denote the number of operations (multiplications, additions, etc.) required to determine the real roots of a polynomial of degree m having real coefficients. Based on these observations, we can derive expressions for the complexities of the different algorithms; these are shown in Table II. For large window sizes N and a significant number of fixed-point iterations L, the number of required operations is approximately $\mathcal{O}(9N^2)+\nu(2N-1)$ for the polynomial root finding (PRF) based algorithm: $\mathcal{O}(4N^2L)$ for Algorithm I and $\mathcal{O}(4NL+3N^2)$ for Algorithm II. The complexity of the PRF-based method is typically dominated by the term $\nu(2N-1)$ involving root finding; for example, the EISPACK routines [17] for root finding have complexity proportional to $\mathcal{O}(m^3)$ for a polynomial of degree m, making $\nu(2N-1)\sim\mathcal{O}(8N^3)$.

V. Numerical Examples

The fixed-point iterations of Section III and the fast weighted myriad computation Algorithms I and II of Section IV are illustrated in this section with two examples. In the first example, a single input vector $\boldsymbol{x}$ is chosen, along with a weight vector $\boldsymbol{w}$ and linearity parameter K. The fixed-point iteration sequences of (19) are computed with several different initial values, and their convergence is demonstrated. In the second example, a long input signal is filtered with a sliding-window weighted myriad filter using different algorithms, including the polynomial root finding (PRF) based algorithm. The speed and accuracy of the algorithms are evaluated for several window sizes N and different values of K. The PRF-based algorithm requires finding the roots of the derivative $P'(\theta)$ of the *polynomial objective function* $P(\theta)$ (see Proposition 2.1). These roots are found using a root finding algorithm described in [18], which is apparently superior in speed and accuracy to the best previously known root finding methods.

Example 1: In order to demonstrate the fixed-point iterations of Section III, a single input vector of length $N = 9$ was generated, with the N samples chosen to be independent and uniformly distributed over [0, 1]. The weight vector was also generated randomly with the weights following a uniform distribution over [0, 1]. The linearity parameter was chosen to be $K = 0.03$. Fig. 5 shows the weighted myriad objective function $Q(\theta)$ of (6) for this example. Recall from Proposition 2.1 that all the local extrema of $Q(\theta)$, including the weighted myriad $\hat{\theta}$, lie within the range of the input samples. For our example, as the figure shows, $Q(\theta)$ has four local minima and three local maxima, and the input samples range from the smallest $x_{(1)} = 0.13$ to the largest $x_{(N)} = 0.99$. The four local minima, which are computed using the root finding method of [18], are at 0.17, 0.27, 0.38, and 0.93, with the weighted myriad (the global minimum point) being $\hat{\theta} = 0.93$.

The fixed-point iteration scheme of (19) was implemented with $L = 10$ iterations for this example using different initial values in order to compute all the local minima of $Q(\theta)$. The initial values for these iterations were the set of N input samples $\{x_i\}_{i=1}^{N}$. Fig. 6(a) shows the $N = 9$ curves representing the different fixed-point iteration sequences obtained. The figure clearly demonstrates the convergence of the fixed-point iterations. We see that the iteration sequences form four sets, each set of curves converging to one of the four local minima of $Q(\theta)$. Although all the sequences happen to be monotonic in this example, it should be mentioned that this may not always be the case. The curve in the figure that starts at the selection weighted myriad $\hat{\theta}_s$ will correspond to the outputs of Algorithm II for different iterations. In this example, the selection weighted myriad happens to be $\hat{\theta}_s = x_6 = x_{(7)} = 0.95$. This is quite close to the weighted myriad $\hat{\theta} = 0.93$, as expected, and Algorithm II thus succeeds in converging to the right value of $\hat{\theta}$.

Fig. 6(b) shows the output of Algorithm I as a function of the number of iterations. This is obtained by picking, at each iteration, the value out of the N curves of Fig. 6(a) that minimizes the objective function $Q(\theta)$. The initial value of the output of Algorithm I is the same as the selection weighted myriad $\hat{\theta}_s = 0.95$. In addition, in the figure , we show (horizontal dashed line) the weighted myriad $\hat{\theta} = 0.93$. As seen from the figure, the output of Algorithm I is very close to the weighted myriad after just a few iterations. The corresponding curve for Algorithm II has been omitted since it happens to be identical to that of Algorithm I in this example.

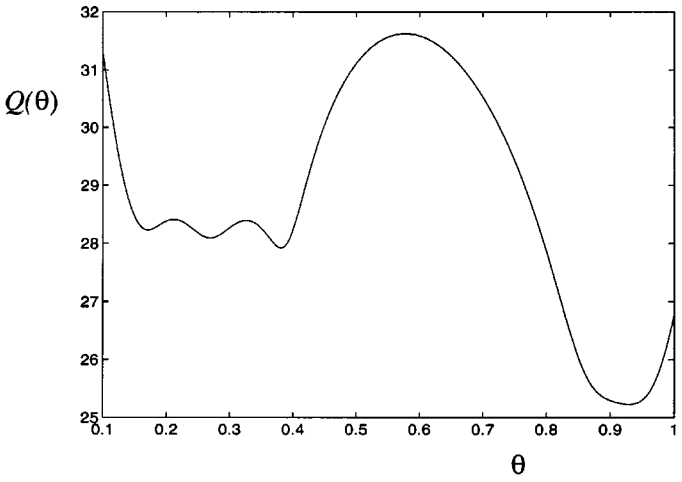
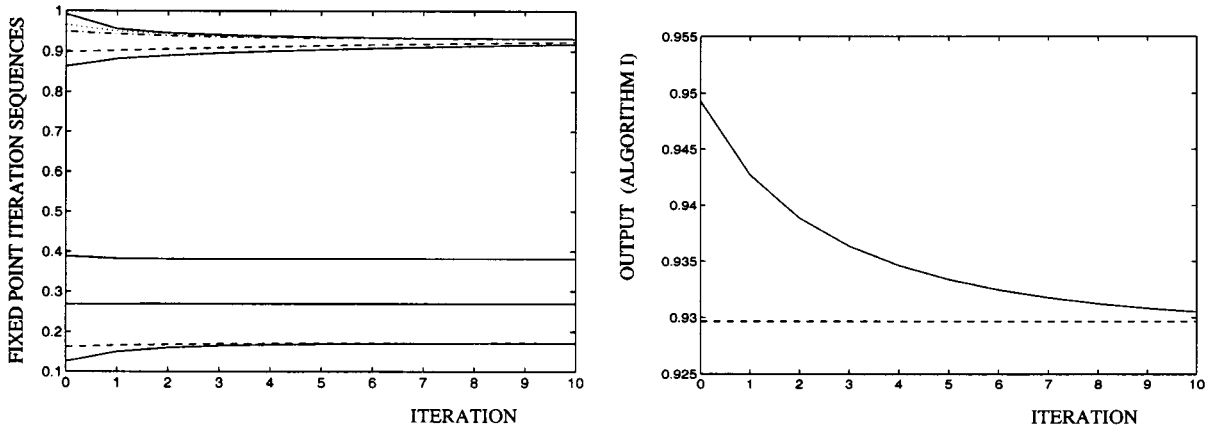


Fig. 5. Weighted myriad objective function $Q(\theta)$ in Example 1. Input vector $\boldsymbol{x} = [0.13, 0.86, 0.39, 0.99, 0.27, 0.95, 0.97, 0.16, 0.90]$, weight vector $\boldsymbol{w} = [0.70, 0.36, 0.94, 0.22, 0.39, 0.04, 0.26, 0.60, 0.02]$, linearity parameter $K = 0.03$.

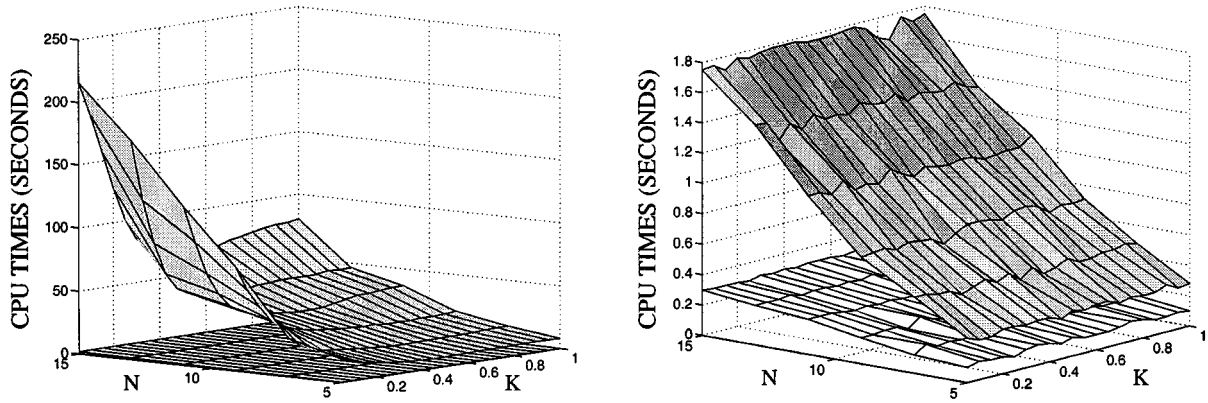
Example 2: In this example, the speed and accuracy of Algorithms I and II are investigated by filtering a long input signal using several window sizes N and different values of the linearity parameter K. The input signal consisted of 5000 randomly generated samples following a uniform distribution over [0, 1]. The window sizes used were $N = 5$, 7, 9, 11, 13, and 15. For each N, the N filter weights were generated randomly, again following the uniform distribution over [0, 1]. The same weight vector was used to filter the input signal with several values of K varying from $K = 0.05$ to $K = 1.0$ in steps of 0.05. Three algorithms were used for the filtering: the PRF-based algorithm using the root finding method of [18] and Algorithms I and II. The fixed-point search Algorithms I and II were implemented for iterations ranging from $L = 0$ to $L = 5$. All the computations were performed in C on a Sun Ultra2 Enterprise workstation (SUN4U/170 Ultra-2/1170).

Fig. 7(a) shows the amount of CPU time (in seconds) that is spent by the PRF-based algorithm in filtering the 5000-long input signal for different values of N and K. The corresponding CPU times used by Algorithm I (with $L = 5$ iterations) are shown for comparison. It is clear from the figure that Algorithm I is consistently faster than the PRF-based algorithm; the contrast in speeds becomes especially evident for small K and large N. The figure also shows that for a given N, the CPU time is largely independent of K, provided K is not too small. The higher execution times for very low values of K are due to the typically larger number of local extrema of the weighted myriad objective function for small K. The PRF-based algorithm, which involves finding and testing all these local extrema, will therefore need more computations for very small K. Fig. 7(b) shows the CPU times for Algorithms I and II. Note that



(a) (b)

Fig. 6. (a) Fixed-point iteration sequences in Example 1 with initial values at the input samples and (b) weighted myriad (solid line) in Example 1 computed using Algorithm I with different numbers of fixed-point iterations. The dashed line is the weighted myriad computed using root finding.



(a) (b)

Fig. 7. CPU times (in seconds) used by various algorithms for different window sizes N and linearity parameters K. (a) PRF-based algorithm (top surface) and Algorithm I with $L = 5$ iterations (bottom mesh) and (b) Algorithms I (top surface) and II (bottom mesh) with $L = 5$.

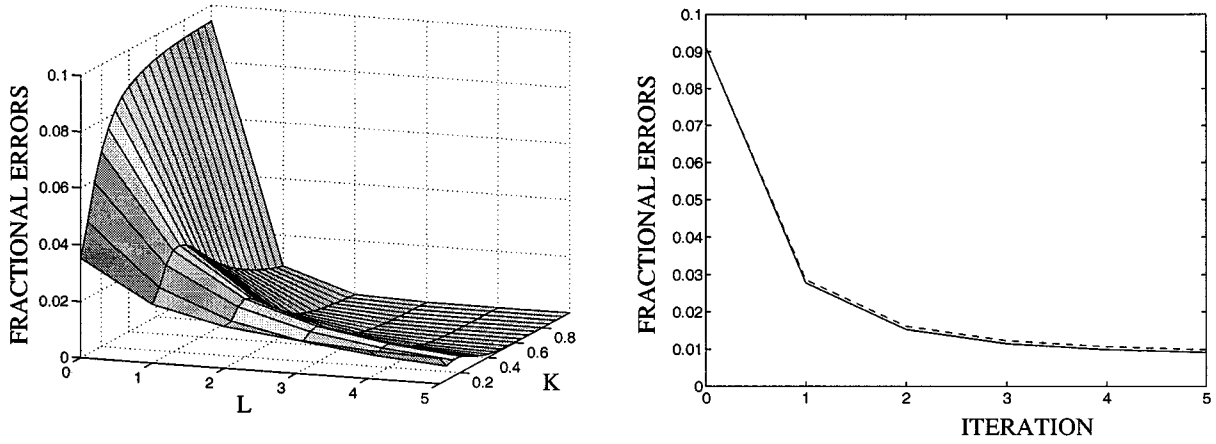
although both algorithms have very low execution times, algorithm II is faster for all values of N and K, whereas the CPU times for Algorithm I increase much more rapidly with N.

Fig. 8(a) shows the fractional absolute error (absolute error divided by the PRF-based value) of Algorithm I for window size $N = 9$. This is calculated as an average in filtering the entire input signal and is plotted for different K and different numbers of iterations L. The plot shows that the fractional error decays rapidly to well below 0.02 (2%) for all K after just a few iterations. The corresponding plot for Algorithm II is not shown since it turns out to be only marginally different from Fig. 8(a). The fractional absolute errors of Algorithms I and II are averaged over all N and K and plotted in Fig. 8(b) as functions of the number of iterations L. This figure again confirms that both algorithms converge rapidly to very low errors (less than 2%) after just two to three iterations, with Algorithm II having only a slightly higher error. Note that the curves in the figure have the same value for $L = 0$; this is expected since both algorithms compute the selection weighted myriad of (31) when $L = 0$.

Finally, the execution times of the different algorithms, and the fractional errors of Algorithms I and II with $L = 5$ iterations, are averaged over K and shown in Table III for different window sizes N. Algorithm I is seen to be about 40–50 times faster than the PRF-based algorithm for all values of N. Algorithm II is even faster; it varies from being faster than the PRF-based algorithm by a factor of about 90 for $N = 5$ to a factor of about 300 for $N = 15$. The average errors of the algorithms are not more than 1% for most values of N becoming slightly larger (3%) only when $N = 15$. Algorithm II is recommended for use in practical applications since it is the fastest algorithm while also yielding accurate results.

VI. Conclusion

The problem of computation of the output of the *weighted myriad filter* was addressed in this paper. The direct computation of the weighted myriad is a nontrivial and prohibitively expensive task. Instead, this paper recast the computation problem, formulating the weighted myriad as *one* of the fixed points of a certain mapping. An iterative algorithm was then proposed to compute these fixed points, and the convergence of these *fixed-point iterations* was rigorously established. Fast *iterative fixed-point search* algorithms to compute the weighted myriad were then derived, incorporating these fixed-point iterations. Two numerical examples were presented, involving



(a) (b)

Fig. 8. (a) Fractional absolute error of Algorithm I as a function of the number of iterations L with window size $N = 9$ and varying linearity parameter K. (b) Mean fractional absolute errors (fractional errors averaged over N and K) of Algorithms I (solid) and II (dashed) plotted against the number of iterations L.

TABLE III
CPU TIMES IN SECONDS (AVERAGED OVER K) FOR DIFFERENT ALGORITHMS, AND FRACTIONAL ERRORS AS PERCENTAGES (AVERAGED OVER K) FOR ALGORITHMS I AND II WITH $L = 5$ ITERATIONS

Window Size	CPU Times (secs.)			Fractional Errors (%)	
N	PRF	Algo. I	Algo. II	Algo. I	Algo. II
5	9.572	0.284	0.110	0.145	0.309
7	14.862	0.476	0.143	0.145	0.241
9	28.735	0.714	0.178	0.188	0.252
11	43.926	1.024	0.216	0.733	0.779
13	58.285	1.381	0.258	0.984	1.040
15	94.200	1.749	0.295	3.270	3.285

filtering randomly generated input signals with a weighted myriad filter, with the weighted myriad computed using different algorithms. The convergence of the fixed-point iterations was demonstrated through the first example. The speed and accuracy of the different algorithms were statistically analyzed in the second example. These fixed-point search algorithms were shown to compute the weighted myriad with a very high degree of accuracy at a relatively low computational cost. With the computational bottleneck of weighted myriad filters removed as a result of this paper, the full potential of this important class of nonlinear filters can now be realized in several applications in robust signal processing and communications.

APPENDIX A
PROOF OF THEOREM 3.1

We will prove the theorem using a *quadratic comparison function* $C^{(m)}(\theta)$, which is defined at each iteration m of the sequence $\{\theta_m\}$, and satisfying a set of conditions in relation to the objective function $Q(\theta)$. Let

$$C^{(m)}(\theta) \equiv C(\theta) \triangleq \sum_{i=1}^{N} C_i(v_i) \tag{33}$$

where the *scale-normalized deviations* v_i are given by $v_i \equiv v_i(\theta) \triangleq (x_i - \theta)/S_i$, $i = 1, 2, \cdots, N$. The functions $C_i(\cdot)$ are to be chosen so that $C(\theta)$ *is quadratic in* θ and satisfies the conditions

$$\begin{aligned} &\text{i)} \quad C(\theta) \geq Q(\theta) \quad \forall \theta \\ &\text{ii)} \quad C(\theta_m) = Q(\theta_m) \\ &\text{iii)} \quad C'(\theta_m) = Q'(\theta_m) \end{aligned}$$

and

$$\text{iv)} \quad \theta_{m+1} = \arg\min_{\theta} C(\theta). \tag{34}$$

Note from (8) that the objective function $Q(\theta)$ can be written as

$$Q(\theta) = \sum_{i=1}^{N} \rho(v_i) \tag{35}$$

where $\rho(\cdot)$ is defined in (7); compare (35) with (33). In order to achieve the desired properties for $C(\theta)$, we choose the functions $C_i(\cdot)$ to be quadratic as in

$$C_i(v) \triangleq c_i + \tfrac{1}{2} d_i v^2, \qquad i = 1, 2, \cdots, N \tag{36}$$

with c_i and d_i chosen, at each iteration, so that the following holds:

$$\begin{aligned} &\text{i)} \quad C_i(v) \geq \rho(v) \quad \forall v \\ &\text{ii)} \quad C_i\left(v_i^{(m)}\right) = \rho\left(v_i^{(m)}\right) \end{aligned}$$

and

$$\text{iii)} \quad C_i'\left(v_i^{(m)}\right) = \rho'\left(v_i^{(m)}\right) = \psi\left(v_i^{(m)}\right) \tag{37}$$

where the values $\{v_i^{(m)}\}_{i=1}^{N}$ are the scale-normalized deviations at the current iteration

$$v_i^{(m)} \triangleq v_i(\theta_m) = \frac{x_i - \theta_m}{S_i} \tag{38}$$

and $\psi(\cdot)$ is defined in (11). Note that we have two parameters (c_i and d_i), whereas there are three conditions in (37). We determine the values satisfying the conditions ii) and iii) in (37); these can easily be derived to be

$$c_i = \rho\left(v_i^{(m)}\right) - \tfrac{1}{2} v_i^{(m)} \psi\left(v_i^{(m)}\right)$$

and

$$d_i = \varphi\left(v_i^{(m)}\right) = \frac{\psi\left(v_i^{(m)}\right)}{v_i^{(m)}} = S_i^2 h_i(\theta_m) \tag{39}$$

where $h_i(\cdot)$ and $\varphi(\cdot)$ are defined in (13) and (14), respectively. It turns out that the resulting functions $C_i(\cdot)$ satisfy condition i) of (37) automatically; the proof of this fact is relegated to the end of this Appendix. Using (36) and (39) in (33), we obtain the following expression for the comparison function:

$$C(\theta) = \sum_{i=1}^{N} c_i + \tfrac{1}{2} \sum_{i=1}^{N} h_i(\theta_m)(x_i - \theta)^2 \tag{40}$$

which is evidently quadratic in θ, as required. Now, with conditions i)–iii) of (37) satisfied, it immediately follows from (33) and (35) that $C(\theta)$ satisfies conditions i)–iii) of (34). Further, referring to (23) of Lemma 3.1, we see that $C(\theta)$ can also be written as $C(\theta) = \sum_{i=1}^{N} c_i + (1/2) B_m(\theta)$. From Lemma 3.1, the updated value θ_{m+1} is the *unique* global minimizer of $B_m(\theta)$. It evidently follows that $C(\theta)$ satisfies condition iv) of (34); in addition,

$$C(\theta) = C(\theta_{m+1}) \Longleftrightarrow \theta = \theta_{m+1}. \tag{41}$$

We finally have the quadratic comparison function $C(\theta)$ with the desired properties; we use this in proving the two parts a) and b) of Theorem 3.1:

a) From (34), using (in order) condition i) with $\theta = \theta_{m+1}$, condition iv) with $\theta = \theta_m$, and, finally, condition ii), we obtain

$$Q(\theta_{m+1}) \le C(\theta_{m+1}) \le C(\theta_m) = Q(\theta_m). \tag{42}$$

Thus, $Q_{m+1} = Q(\theta_{m+1}) \le Q(\theta_m) = Q_m$. Further, if $\theta_{m+1} \ne \theta_m$, then $C(\theta_{m+1}) \ne C(\theta_m)$ [using (41)]; hence, $Q_{m+1} < Q_m$ strictly. On the other hand, (18) and (19) imply that θ_m is a local extremum of the objective function $Q(\theta)$ if (and only if) $\theta_{m+1} = \theta_m$.

b) From (42), we have

$$\begin{aligned} Q(\theta_m) - Q(\theta_{m+1}) &= C(\theta_m) - Q(\theta_{m+1}) \\ &\ge C(\theta_m) - C(\theta_{m+1}). \end{aligned} \tag{43}$$

Now, from condition iv) of (34), θ_{m+1} is the global minimizer of the quadratic function $C(\theta)$, which can therefore be expressed in the form $C(\theta) = C(\theta_{m+1}) + \lambda(\theta - \theta_{m+1})^2$. To determine λ, note that it is simply the coefficient of θ^2 in $C(\theta)$. This is readily obtained by examining the expression for $C(\theta)$ in (40); thus, $\lambda = (1/2)\sum_{i=1}^{N} h_i(\theta_m)$. Substituting the value of λ and setting $\theta = \theta_m$, we can write $C(\theta_m) = C(\theta_{m+1}) + ((1/2)\sum_{i=1}^{N} h_i(\theta_m)) \cdot (\theta_{m+1} - \theta_m)^2$. Using this in (43), we finally obtain

$$Q(\theta_m) - Q(\theta_{m+1}) \ge \left(\tfrac{1}{2} \sum_{i=1}^{N} h_i(\theta_m)\right) \cdot (\theta_{m+1} - \theta_m)^2. \tag{44}$$

Suppose now that $\theta_m \in [x_{(1)}, x_{(N)}]$. Then, we have $(x_i - \theta_m)^2 \le R^2$, where $R = x_{(N)} - x_{(1)}$. Using this fact and the definition of $h_i(\cdot)$ in (13), we can easily show that

$$\begin{aligned} \frac{1}{2} \sum_{i=1}^{N} h_i(\theta_m) &= \sum_{i=1}^{N} \frac{1}{S_i^2 + (x_i - \theta_m)^2} \\ &\ge \sum_{i=1}^{N} \frac{1}{S_i^2 + R^2}. \end{aligned} \tag{45}$$

and the truth of (24) follows. This completes the proof of Theorem 3.1.

Remark: Parts of our proof use ideas from [14, Sec. 7.8] on the computation of joint regression M-estimates of location and scale.

As promised earlier, we now verify that the functions $C_i(\cdot)$ of (36) satisfy condition i) of (37): $C_i(v) \ge \rho(v)\ \forall v$. For a given $i \in \{1, 2, \cdots, N\}$, define the difference function $\Delta(v) \triangleq C_i(v) - \rho(v)$, where $\rho(v) = \log(1 + v^2)$ from (7). We need to show that $\Delta(v) \ge 0\ \forall v$. Using (39) and letting $v_0 \triangleq v_i^{(m)}$ for convenience, we can write $\Delta(v) = [\rho(v_0) - \rho(v)] + (1/2)\varphi(v_0)[v^2 - v_0^2]$, where $\varphi(v) = 2/(1 + v^2)$ from (14). Substituting for $\rho(\cdot)$ and $\varphi(\cdot)$, we obtain

$$\Delta(v) = \log\left(\frac{1 + v_0^2}{1 + v^2}\right) + \frac{v^2 - v_0^2}{1 + v_0^2}. \tag{46}$$

Using the transformation $z \triangleq (1 + v^2)/(1 + v_0^2) > 0$, we can write $\Delta(v) = \Omega(z) \triangleq -\log(z) + z - 1$. The problem is now reduced to showing that $\Omega(z) \ge 0$ over $(0, +\infty)$. It is a simple exercise to verify that $\Omega(z)$ has a unique minimum at $z = 1$. Consequently, $C_i(v) - \rho(v) = \Delta(v) = \Omega(z) \ge \Omega(1) = 0$, and the proof is complete.

Appendix B
Proof of Theorem 3.3

From Corollary 3.1.1 and Theorem 3.2, it is evident that $\inf(\{Q_m\}_{m=1}^{\infty}) \ge Q(\theta_M^*)$, $Q(\theta_M^*)$ being the minimum value of $Q(\theta)$ in (a_M, b_M). We will prove the theorem by contradiction. Suppose then that $Q^* \triangleq \inf(\{Q_m\}_{m=1}^{\infty}) > Q(\theta_M^*)$. This situation is shown in Fig. 9. Now, define $\xi_1 < \theta_M^*$ and $\xi_2 > \theta_M^*$ such that $Q(\xi_1) = Q(\xi_2) \triangleq Q^*$. Since $Q(\theta_m) \ge Q^*\ \forall m$, it is evident from the figure that $\theta_m \notin (\xi_1, \xi_2)$ for any m. Further, considering $m \ge M$ only and using the arguments employed in Cases I and II of the proof of Theorem 3.2, we can show that $\theta_m < \theta_{m+1} < b_M$ when $a_M < \theta_m < \theta_M^*$ and $a_M < \theta_{m+1} < \theta_m$ when $\theta_M^* < \theta_m < b_M$. This means that succeeding values of the sequence $\{\theta_m\}$ move further away from the endpoints a_m and b_M. Consequently, the subsequence $\{\theta_m : m \ge M\}$ is bounded away from a_M and b_M. Therefore,

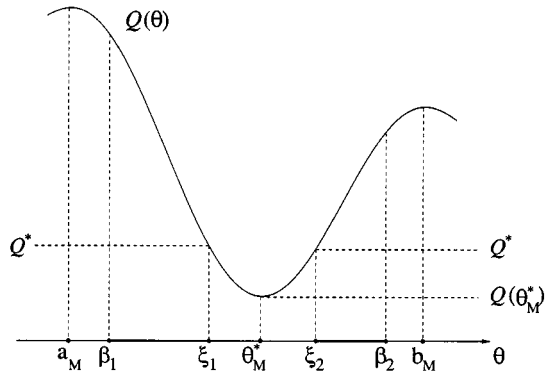


Fig. 9. Illustration of the proof of Theorem 3.3.

we can find values β_1 and β_2 with $a_M < \beta_1 < \xi_1$ and $\xi_2 < \beta_2 < b_M$ such that $\forall m \geq M$, either $\theta_m \in [\beta_1, \xi_1]$ or $\theta_m \in [\xi_2, \beta_2]$. This is illustrated in the figure, with bold lines showing the intervals $[\beta_1, \xi_1]$ and $[\xi_2, \beta_2]$, which are the regions within which the subsequence $\{\theta_m: m \geq M\}$ is confined. A consequence of this is that $\{\theta_m: m \geq M\} \subset (a_M, b_M)$ is bounded away from a_M, b_M and θ_M^*, which are the only points in $[a_M, b_M]$ at which the derivative function $Q'(\theta) = 0$. We can then conclude, using the continuity of $Q'(\theta)$, that the subsequence $\{Q'(\theta_m): m \geq M\}$ is bounded away from 0. As a result, $Q'(\theta_m) \not\to 0$, which contradicts Corollary 3.1.2. Hence, the assumption that $\inf(\{Q_m\}_{m=1}^{\infty}) > Q(\theta_M^*)$ is false. Thus, $\inf(\{Q_m\}_{m=1}^{\infty}) = Q(\theta_M^*)$, and the theorem is proved.

REFERENCES

[1] M. P. Shinde and S. N. Gupta, "Signal detection in the presence of atmospheric noise in tropics," *IEEE Trans. Commun.*, vol. COMM-22, Aug. 1974.

[2] D. Middleton, "Statistical-physical models of electromagnetic interference," *IEEE Trans. Electromagn. Compat.*, vol. EMC-19, pp. 106–127, 1977.

[3] J. Ilow, "Signal processing in alpha-stable noise environments: Noise modeling, detection and estimation," Ph.D. dissertation, Univ. Toronto, Toronto, Ont., Canada, Dec. 1995.

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

[5] L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, "Weighted median filters: A tutorial," *IEEE Trans. Circuits Syst. II*, vol. 43, Mar. 1996.

[6] I. Pitas and A. Venetsanopoulos, "Order statistics in digital image processing," *Proc. IEEE*, vol. 80, Dec. 1992.

[7] J. G. Gonzalez and G. R. Arce, "Weighted myriad filters: A robust filtering framework derived from α-stable distributions," in *Proc. IEEE ICASSP*, Atlanta, GA, 1996.

[8] J. G. Gonzalez, "Robust techniques for wireless communications in non-Gaussian environments," Ph.D. thesis, Univ. Delaware, Newark, Dec. 1997.

[9] S. Kalluri, "Nonlinear adaptive algorithms for robust signal processing and communications in impulsive environments," Ph.D. dissertation, Univ. Delaware, Newark, Dec. 1998.

[10] S. Kalluri and G. R. Arce, "Robust frequency—Selective filtering using weighted myriad filters admitting real-valued weights," , submitted for publication.

[11] ——, "Adaptive weighted myriad filter algorithms for robust signal processing in α-stable noise environments," *IEEE Trans. Signal Processing*, vol. 46, pp. 322–334, Feb. 1998.

[12] J. G. Gonzalez, D. W. Griffith, and G. R. Arce, "Matched myriad filtering for robust communications," in *Proc. CISS*, Princeton, NJ, 1996.

[13] P. Zurbach, J. G. Gonzalez, and G. R. Arce, "Weighted myriad filters for image processing," in *Proc. IEEE Int. Symp. Circuits Syst.*, Atlanta, GA, 1996.

[14] P. J. Huber, *Robust Statistics*. New York: Wiley, 1981.

[15] D. G. Luenberger, *Optimization by Vector Space Methods*. New York: Wiley, 1969.

[16] J. G. Gonzalez, D. L. Lau, and G. R. Arce, "Toward a general theory of robust nonlinear filtering: Selection filters," in *Proc. IEEE ICASSP*, Munich, Germany, 1997.

[17] B. T. Smith *et al.*, "Matrix eigensystem routines—EISPACK guide," in *Lecture Notes in Computer Science*. New York: Springer-Verlag, 1976, vol. 6.

[18] M. Lang and B. Frenzel, "Polynomial root finding," *IEEE Signal Processing Lett.*, vol. 1, Oct. 1994.

Sudhakar Kalluri (S'90–M'99) was born in Vijayawada, India, in 1968. He received the B.Tech degree in electronics and communication engineering from the Indian Institute of Technology, Madras, in 1989, the M.S.E. degree in electrical engineering from Princeton University, Princeton, NJ, in 1991, and the Ph.D. degree in electrical engineering from the University of Delaware, Newark, in 1998.

He has worked at the IBM T. J. Watson Research Center, Yorktown Heights, NY, in the Video and Image Technologies Department. Since 1998, he has been a Postdoctoral Research Associate with the Department of Electrical and Computer Engineering, University of Delaware. He has consulted with the Army Research Laboratories in the area of communications. His research interests include robust signal processing and communications, statistical signal processing, communication theory, signal processing for communications, nonlinear adaptive algorithms, and nonlinear filter theory. In 1999, he joined Level One Communications, Inc., Sacramento, CA.

Gonzalo R. Arce (F'00) was born in La Paz, Bolivia. He received the B.S.E.E. degree (with highest honors) from the University of Arkansas, Fayetteville, in 1979 and the M.S. and Ph.D. degrees in electrical engineering from Purdue University, West Lafayette, IN, in 1980 and 1982, respectively.

Since 1982, he has been with the Department of Electrical and Computer Engineering, University of Delaware, Newark, where he is currently Professor, Interim Chair, and Fellow with the Center for Advanced Studies. He has consulted for several industrial organizations in the general areas of signal and image processing and digital communications. His research interests include robust signal processing and its applications, communication theory, image processing, communications, and electronic imaging. He holds four U.S. patents.

Dr. Arce has served as Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING, as Guest Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, and Guest Editor for the Optical Society of America's *Optics Express*. He is a member of the Digital Signal Processing Technical Committee of the Circuits and Systems Society and of the Board of Nonlinear Signal and Image Processing.