# Parameter Estimation of Superimposed Signals Using the EM Algorithm

MEIR FEDER AND EHUD WEINSTEIN

*Abstract*—**We develop a computationally efficient algorithm for parameter estimation of superimposed signals based on the EM algorithm. The idea is to decompose the observed data into their signal components and then to estimate the parameters of each signal component separately. The algorithm iterates back and forth, using the current parameter estimates to decompose the observed data better and thus increase the likelihood of the next parameter estimates. The application of the algorithm to the multipath time delay and to the multiple source location estimation problems is considered.**

## I. Introduction

THE general problem of interest here may be characterized using the following model:

$$y(t) = \sum_{k=1}^{K} s_k(t; \boldsymbol{\theta}_k) + n(t)$$

where $\boldsymbol{\theta}_k$ are the vector unknown parameters associated with the $k$th signal component and $n(t)$ stands for the additive noise. This model covers a wide range of problems involving superimposed signals. The specific problem we have in mind is multiple source location estimation; in that case, $s_k(t; \boldsymbol{\theta}_k)$ are the array signals observed from the $k$th source, and $\boldsymbol{\theta}_k$ are the unknown source location parameters.

In this paper, we develop a computationally efficient scheme for the joint estimation of $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_K$ based on the Estimate Maximize (EM) algorithm. The idea is to decompose the observed data $y(t)$ into its signal components and then estimate the parameters of each signal component separately. The algorithm iterates back and forth, using the current parameter estimates to decompose the observed data better and thus improve the next parameter estimates. Under the stated regularity conditions, the algorithm converges to a stationary point of the likelihood function where each iteration cycle increases the likelihood of the estimated parameters.

Manuscript received February 20, 1986; revised September 28, 1987. This work was supported in part by the MIT–WHOI Joint Program, in part by the Advanced Research Projects Agency monitored by ONR under Contract N00014-81-K-0742 and the National Science Foundation under Grant ECS-8407285 in MIT, and in part by the ONR under Contract N00014-85-K-0272 in WHOI. It is WHOI Contribution 6584.

M. Feder is with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, and the Department of Ocean Engineering, Woods Hole Oceanographic Institution, Woods Hole, MA 02543.

E. Weinstein is with the Department of Electronic Systems, Faculty of Engineering, Tel-Aviv University, Tel-Aviv 69978, Israel, and the Department of Ocean Engineering, Woods Hole Oceanographic Institution, Woods Hole, MA 02543.

IEEE Log Number 8719013.

The results developed in this paper can be viewed as a generalization of the results presented by the authors in [1] and [2]. We note that the idea of iteratively decomposing the observed signal and estimating the parameters of each signal component separately has been previously proposed in several applications (e.g., [3]–[5]). However, in most cases, the approach is ad hoc, and there is no proof of convergence of the algorithm. As will be shown, the EM method suggests a very specific way of decomposing the observed signals, leading to a numerically as well as statistically stable algorithm.

The paper is organized as follows. In Section II the EM method is represented following the derivation in [6]. In Section III the EM method is applied to the parameter estimation of superimposed signals, and the basic algorithm is developed for deterministic (known) signals and for stationary Gaussian signals. In Section IV the algorithm is applied to the multipath time-delay estimation, in Section V the algorithm is applied to the multiple source angle of arrival estimation, and in Section VI we summarize the results.

## II. Maximum Likelihood Estimation Via the EM Algorithm

The EM algorithm, developed in [6], is a general method for solving maximum likelihood (ML) estimation problems given incomplete data. The considerations leading to the EM algorithm are given below.

Let $\boldsymbol{Y}$ denote the observed (incomplete) data, possessing the probability density $f_Y(y; \boldsymbol{\theta})$ indexed by the parameter vector $\boldsymbol{\theta} \in \Theta \subseteq R^K$, and let $\boldsymbol{X}$ denote the "complete" data, related to $\boldsymbol{Y}$ by

$$H(\boldsymbol{X}) = \boldsymbol{Y} \tag{1}$$

where $H(\cdot)$ is a noninvertible (many-to-one) transformation. Express densities

$$f_X(x; \boldsymbol{\theta}) = f_{X/Y=y}(x; \boldsymbol{\theta}) \cdot f_Y(y; \boldsymbol{\theta}) \qquad \forall H(x) = y \tag{2}$$

where $f_X(x; \boldsymbol{\theta})$ is the probability density associated with $\boldsymbol{X}$ and $f_{X/Y=y}(x; \boldsymbol{\theta})$ is the conditional probability density of $\boldsymbol{X}$ given $\boldsymbol{Y} = y$. Taking the logarithm on both sides of (2),

$$\log f_Y(y; \boldsymbol{\theta}) = \log f_X(x; \boldsymbol{\theta}) - \log f_{X/Y=y}(x; \boldsymbol{\theta}). \tag{3}$$

Taking the conditional expectation given $\boldsymbol{Y} = y$ at a parameter value $\theta'$,

$$\log f_Y(y; \theta) = E\{\log f_X(x; \theta)/Y = y; \theta'\} - E\{\log f_{X/Y=y}(x; \theta)/Y = y; \theta'\}. \quad (4)$$

Define, for convenience,

$$L(\theta) = \log f_Y(y; \theta) \quad (5)$$

$$U(\theta, \theta') = E\{\log f_X(x; \theta)/Y = y; \theta'\} \quad (6)$$

and

$$V(\theta, \theta') = E\{\log f_{X/y=y}(x; \theta)/Y = y; \theta'\}. \quad (7)$$

With these definitions, (4) reads

$$L(\theta) = U(\theta, \theta') - V(\theta, \theta'). \quad (8)$$

We identify $L(\theta)$, the log-likelihood of the observed data, as the function we want to maximize. Invoking the Jensen's inequality,

$$V(\theta, \theta') \leqq V(\theta', \theta'). \quad (9)$$

Hence, if

$$U(\theta, \theta') > U(\theta', \theta')$$

then

$$L(\theta) > L(\theta'). \quad (10)$$

The relation in (10) forms the basis for the EM algorithm. The algorithm starts with an arbitrary initial guess $\hat{\theta}^{(0)}$, and denote by $\hat{\theta}^{(n)}$ the current estimate of $\theta$ after $n$ iterations of the algorithm. Then, the next iteration cycle can be described in two steps, as follows:

*E step:* Compute

$$U(\theta, \hat{\theta}^{(n)}). \quad (11)$$

*M step:*

$$\operatorname*{Max}_{\theta} U(\theta, \hat{\theta}^{(n)}) \rightarrow \hat{\theta}^{(n+1)}. \quad (12)$$

If $U(\theta, \theta')$ is continuous in both $\theta$ and $\theta'$, the algorithm converges to a stationary point of the log-likelihood function [7] where the maximization in (12) ensures that each iteration cycle increases the likelihood. Of course, as in the case of all "hill climbing" algorithms, the convergence point may not be the global maximum of the likelihood function, and thus several starting points may be needed. The rate of convergence of the algorithm is exponential, depending on the fraction of the covariance of the "complete" data that can be predicted using the observed (incomplete) data [6], [8]. If that fraction is small, the rate of convergence tends to be slow, in which case one could use standard numerical methods to accelerate the algorithm.

In the Appendix we derive a closed-form analytical expression for $U(\theta, \theta')$ for the case where $X$ and $Y$ are jointly Gaussian, related by a linear transformation. We note that in general a closed-form analytical expression cannot be found, and thus the computation of $U(\theta, \theta')$ at each iteration cycle generally requires multiple integration. Appendix A has importance of its own since it covers a wide range of problems.

We note that the EM algorithm is not uniquely specified. The transformation $H(\cdot)$ relating $X$ to $Y$ can be *any* noninvertible transformation. Obviously, there are many possible "complete" data specifications $X$ that will generate the observed data $Y$. However, the choice of "complete" data may critically affect the complexity and the rate of convergence of the algorithm, and the unfortunate choice of "complete" data may yield a completely useless algorithm.

In the next section, it will be shown that for the class of problems involving superimposed signals, there is a natural choice of "complete" data, leading to a surprisingly simple algorithm to extract the ML parameter estimates.

## III. Parameter Estimation of Superimposed Signals in Noise

The general problem of interest here may be characterized using the following model:

$$y(t) = \sum_{k=1}^{K} s_k(t; \theta_k) + n(t) \quad (13)$$

where $\theta_k$ are the vector unknown parameters associated with the $k$th signal component and $n(t)$ denotes the additive noise. This model covers a wide range of problems arising in array and signal processing, including multitone frequency estimation, multipath time-delay estimation, and multiple source location estimation.

Given observations of $y(t)$, we want to find the ML estimate of $\theta_1, \theta_2, \cdots, \theta_K$ jointly. We shall now consider the cases of deterministic (known) signals and stochastic Gaussian signals separately.

### A. Deterministic Signals

Consider the model of (13) under the following assumptions.

1) The $s_k(t; \theta_k)$, $k = 1, 2, \cdots, K$, are conditionally known up to the vector parameters $\theta_k$.

2) The $n(t)$ are vector zero-mean white Gaussian processes whose covariance matrix is $E\{n(t)\, n^*(\sigma)\} = Q \cdot \delta(t - \sigma)$.

Under these assumptions, the log-likelihood function is given by (e.g., see [9])

$$L(\theta) = c - \frac{\lambda}{2} \int_T \left[ y(t) - \sum_{k=1}^{K} s_k(t; \theta_k) \right]^* \cdot Q^{-1} \left[ y(t) - \sum_{k=1}^{K} s_k(t; \theta_k) \right] dt \quad (14)$$

where $\lambda = 1$ if $n(t)$ is real-valued, $\lambda = 2$ if $n(t)$ is complex-valued, and $c$ is a normalizing constant. In the case of discrete-time observations, $L(\theta)$ is still given by (14) where the integral is replaced by the sum over all discrete points $t \in T$. Thus, to obtain the ML estimate of the various $\theta_k$'s, one therefore must solve

$$\min_{\theta_1, \theta_2, \cdots, \theta_K} \left\{ \int_T \left[ y(t) - \sum_{k=1}^{K} s_k(t; \theta_k) \right]^* \cdot Q^{-1} \left[ y(t) - \sum_{k=1}^{K} s_k(t; \theta_k) \right] dt \right\}. \quad (15)$$

This is a complicated multiparameter optimization problem. Of course, brute force can always be used to solve the problem, evaluating the objective function on a coarse grid to locate roughly the global minimum and then applying the Gauss method, the Newton–Raphson, or some other gradient-search iterative algorithm. However, when applied to the problem at hand, these methods tend to be computationally complex and time consuming.

Having the EM algorithm in mind, we want to simplify the computations involved. To apply the algorithm to the problem at hand, the first step is to specify the "complete" data. A natural choice for the "complete" data $x(t)$ is obtained by decomposing the observed data $y(t)$ into its signal components; that is,

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_k(t) \end{bmatrix} \quad (16)$$

where

$$x_k(t) = s_k(t; \theta_k) + n_k(t) \quad (17)$$

where the $n_k(t)$ are obtained by arbitrarily decomposing the total noise $n(t)$ into the $K$ components, so that

$$\sum_{k=1}^{K} n_k(t) = n(t). \quad (18)$$

From (13), (17), and (18), the relation between the "complete" data $x(t)$ and the incomplete data $y(t)$ is given by

$$y(t) = \sum_{k=1}^{K} x_k(t) = H \cdot x(t) \quad (19)$$

where

$$H = \underbrace{[I \ \ I \ \cdots \ I]}_{K \text{ terms}}; \quad (20)$$

We shall find it most convenient to let the $n_k(t)$ be statistically independent, zero-mean, and Gaussian with the covariance matrix $E\{n_k(t)\, n_k^*(\sigma)\} = Q_k \delta(t - \sigma)$ where $Q_k = \beta_k Q$ and the $\beta_k$'s are arbitrary real-valued scalars satisfying

$$\sum_{k=1}^{K} \beta_k = 1, \qquad \beta_k \geqq 0. \quad (21)$$

In that case, the log-likelihood of the complete data $x(t)$ is

$$\begin{aligned} \log f_X(x; \theta) &= c - \frac{\lambda}{2} \int_T [x(t) - s(t; \theta)]^* \cdot \Lambda^{-1}[x(t) - s(t; \theta)]\, dt \\ &= d + \frac{\lambda}{2} \int_T s^*(t; \theta)\, \Lambda^{-1} x(t)\, dt + \frac{\lambda}{2} \int_T x^*(t)\, \Lambda^{-1} s(t; \theta)\, dt \\ &\quad - \frac{\lambda}{2} \int_T s^*(t; \theta)\, \Lambda^{-1} s(t; \theta)\, dt \end{aligned} \quad (22)$$

where $d$ contains all the terms that are independent of $\theta$. $s(t; \theta)$ and $\Lambda$ are the mean and the covariance matrix of $x(t)$ given, respectively, by

$$s(t; \theta) = \begin{bmatrix} s_1(t; \theta_1) \\ s_2(t; \theta_2) \\ \vdots \\ s_K(t; \theta_K) \end{bmatrix} \quad (23)$$

and

$$\Lambda = \begin{bmatrix} Q_1 & & (0) \\ & Q_2 & \\ (0) & & Q_K \end{bmatrix} \quad (24)$$

where the notation in (24) indicates that $\Lambda$ is a block-diagonal matrix. Taking the conditional expectation of (22) given $y(t)$ at a parameter value $\theta'$,

$$\begin{aligned} U(\theta, \theta') &= d + \frac{\lambda}{2} \int_T s^*(t; \theta)\, \Lambda^{-1} \hat{x}(t)\, dt + \frac{\lambda}{2} \int_T \hat{x}^*(t)\, \Lambda^{-1} s(t; \theta) \\ &\quad - \frac{\lambda}{2} \int_T s^*(t; \theta)\, \Lambda^{-1} s(t; \theta)\, dt \\ &= e - \frac{\lambda}{2} \int_T [\hat{x}(t) - s(t; \theta)]^* \cdot \Lambda^{-1}[\hat{x}(t) - s(t; \theta)]\, dt \end{aligned} \quad (25)$$

where

$$\hat{x}(t) = E\{x(t)/y(t); \theta'\} \quad (26)$$

and $e$ is a constant independent of $\theta$. Since $x(t)$ and $y(t)$ are jointly Gaussian, related by the linear transformation $y(t) = Hx(t)$, then using (A7) of the Appendix,

$$\hat{x}(t) = s(t; \theta') + \Lambda H^*[H \Lambda H^*]^{-1}[y(t) - Hs(t; \theta')]. \quad (27)$$

Substituting (23) and (24) into (25) and following straightforward matrix manipulations, we obtain

$$U(\theta, \theta') = e - \frac{\lambda}{2} \sum_{k=1}^{K} \int_T [\hat{x}_k(t) - s_k(t; \theta_k)]^* \cdot Q_k^{-1}[\hat{x}_k(t) - s_k(t; \theta_k)]\, dt \tag{28}$$

where the $\hat{x}_k(t)$ are the components of $\hat{x}(t)$ given by

$$\hat{x}_k(t) = s_k(t; \theta_k') + \beta_k \left[ y(t) - \sum_{k=1}^{K} s_k(t; \theta_k') \right]. \tag{29}$$

Observing that the maximization of $U(\theta, \theta')$ with respect to $\theta$ is equivalent to the minimization of each of the terms in the $k$ sum of (28) *separately*, the EM algorithm assumes the following form.

*E step:* For $k = 1, 2, \cdots, K$, compute

$$\hat{x}_k^{(n)}(t) = s_k(t; \hat{\theta}_k^{(n)}) + \beta_k \left[ y(t) - \sum_{l=1}^{K} s_l(t; \hat{\theta}_l^{(n)}) \right]. \tag{30}$$

*M step:* For $k = 1, 2, \cdots, K$,

$$\min_{\theta_k} \int_T [\hat{x}_k^{(n)}(t) - s_k(t; \theta_k)]^* \cdot Q^{-1}[\hat{x}_k^{(n)}(t) - s_k(t; \theta_k)]\, dt \rightarrow \hat{\theta}_k^{(n+1)}. \tag{31}$$

We observe that $\hat{\theta}_k^{(n+1)}$ is, in fact, the ML estimate of $\theta_k$ based on $\hat{x}_k^{(n)}(t)$. The algorithm is illustrated in Fig. 1. We note that in the case of discrete observations, the integral in (31) is substituted by the sum over all points $t \in T$.

The most striking feature of the algorithm is that it decouples the complicated multiparameter optimization into $K$ separate ML optimizations. Hence, the complexity of the algorithm is essentially unaffected by the assumed number of signal components. As $K$ increases, we have to increase the number of ML processors in parallel; however, each ML processor is maximized separately. Since the algorithm is based on the EM method, each iteration cycle increases the likelihood until convergence is accomplished.

We note that the $\beta_k$'s must satisfy the constraint stated in (21), but otherwise they are arbitrary free variables in the algorithm. The $\beta_k$'s can be used to control the rate of convergence of the algorithm and possibly to avoid the convergence to an unwanted stationary point of the algorithm. These considerations are currently under investigation.

### B. Gaussian Signals

Consider the model of (13) under the following assumptions.

1) $n(t)$ and $s_k(t; \theta_k)$, $k = 1, 2, \cdots, K$, are mutually uncorrelated, wide-sense stationary (WSS), zero-mean Gaussian vector stochastic processes whose spectral density matrices are $N(\omega)$ and $S_k(\omega; \theta_k)$, $k = 1, 2, \cdots, K$, respectively.

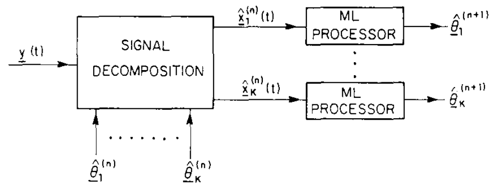



Fig. 1. The EM algorithm for deterministic (known) signals.

2) The observation interval $T$ is long compared to the correlation time (inverse bandwidth) of the signals and the noise, i.e., $WT/2\pi >> 1$.

Fourier analyzing $y(t)$, we obtain

$$Y(\omega_l) = \frac{1}{\sqrt{T}} \int_T y(t)\, e^{-j\omega_l t}\, dt. \tag{32}$$

Under the above assumptions, $y(t)$ is WSS, zero-mean, and Gaussian. Thus, for $WT/2\pi >> 1$, the $Y(\omega_l)$'s are asymptotically uncorrelated, zero-mean, and Gaussian with covariance matrix $P(\omega_l; \theta)$ where $P(\omega; \theta)$ is the spectral density matrix of $y(t)$ given by

$$P(\omega; \theta) = \sum_{k=1}^{K} S_k(\omega; \theta_k) + N(\omega). \tag{33}$$

The log-likelihood function given the $Y(\omega_l)$'s is therefore given by

$$L(\theta) = -\sum_l [\operatorname{logdet} \pi P(\omega_l; \theta) + Y^*(\omega_l)\, P^{-1}(\omega_l; \theta)\, Y(\omega_l)] \tag{34}$$

where the summation in (34) is carried over all $\omega_l$ in the signal frequency band. In the case of discrete observations $y_i = y(i\Delta t)$, the log-likelihood is still given by (34) where the $Y(\omega_l)$'s are the discrete Fourier transform (DFT) of the $y_i$'s. To obtain the ML estimate of the various $\theta_k$'s, we therefore must solve

$$\min_{\theta_1, \theta_2, \cdots, \theta_K} \left\{ \sum_l [\operatorname{logdet} P(\omega_l; \theta) + Y^*(\omega_l)\, P^{-1}(\omega_l; \theta)\, Y(\omega_l)] \right\}. \tag{35}$$

We want to use the EM method to bypass this complicated multiparameter optimization. Following the same considerations as in the deterministic signal case, let the "complete" data $x(t)$ be given by

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_K(t) \end{bmatrix} \tag{36}$$

where

$$x_k(t) = s_k(t; \theta_k) + n_k(t) \tag{37}$$

and the $\boldsymbol{n}_k(t)$ are chosen to be mutually uncorrelated, zero-mean, and Gaussian with respective spectral density matrices $N_k(\omega) = \beta_k \cdot N(\omega)$ where the $\beta_k$'s are arbitrary real-valued constants subject to (21).

Following the same considerations leading to (34), the log-likelihood of the "complete" data is given by

$$\log f_X(\boldsymbol{x}; \boldsymbol{\theta}) = -\sum_l \left[\log\det \pi\Lambda(\omega_l; \boldsymbol{\theta}) + \boldsymbol{X}^*(\omega_l)\, \Lambda^{-1}(\omega_l; \boldsymbol{\theta})\, \boldsymbol{X}(\omega_l)\right]$$
$$= -\sum_l \left[\log\det \pi\Lambda(\omega_l; \boldsymbol{\theta}) + \operatorname{tr}\left(\Lambda^{-1}(\omega_l; \boldsymbol{\theta})\, \boldsymbol{X}(\omega_l)\, \boldsymbol{X}^*(\omega_l)\right)\right] \quad (38)$$

where

$$\boldsymbol{X}(\omega_l) = \begin{bmatrix} \boldsymbol{X}_1(\omega_l) \\ \boldsymbol{X}_2(\omega_l) \\ \vdots \\ \boldsymbol{X}_K(\omega_l) \end{bmatrix} \quad (39)$$

$$\boldsymbol{X}_k(\omega_l) = \frac{1}{\sqrt{T}} \int_T \boldsymbol{x}_k(t)\, e^{-j\omega_l t}\, dt \quad (40)$$

and

$$\Lambda(\omega; \boldsymbol{\theta}) = \begin{bmatrix} \Lambda_1(\omega; \boldsymbol{\theta}_1) & & & \\ & \Lambda_2(\omega; \boldsymbol{\theta}_2) & & (0) \\ (0) & & \ddots & \\ & & & \Lambda_K(\omega; \boldsymbol{\theta}_K) \end{bmatrix} \quad (41)$$

$$\Lambda_k(\omega; \boldsymbol{\theta}_k) = S_k(\omega; \boldsymbol{\theta}_k) + \beta_k \cdot N(\omega). \quad (42)$$

Taking the conditional expectation of (38) given $\boldsymbol{Y}(\omega_l)$ at a parameter value $\boldsymbol{\theta}'$,

$$U(\boldsymbol{\theta}, \boldsymbol{\theta}') = -\sum_l \left[\log\det \pi\Lambda(\omega_l; \boldsymbol{\theta}) + \operatorname{tr}\left(\Lambda^{-1}(\omega_l; \boldsymbol{\theta})\, \widehat{\boldsymbol{X}(\omega_l)\, \boldsymbol{X}^*(\omega_l)}\right)\right] \quad (43)$$

where

$$\widehat{\boldsymbol{X}(\omega_l)\, \boldsymbol{X}^*(\omega_l)} = E\left\{\boldsymbol{X}(\omega_l)\, \boldsymbol{X}^*(\omega_l)/\boldsymbol{Y}(\omega_l); \boldsymbol{\theta}'\right\}. \quad (44)$$

Since $\boldsymbol{X}(\omega_l)$ and $\boldsymbol{Y}(\omega_l)$ are jointly Gaussian, related by the linear transformation $\boldsymbol{Y}(\omega_l) = H \cdot \boldsymbol{X}(\omega_l)$, then using (A8) of the Appendix,

$$\widehat{\boldsymbol{X}(\omega_l)\, \boldsymbol{X}^*(\omega_l)} = \Lambda(\omega_l; \boldsymbol{\theta}') - \Lambda(\omega_l; \boldsymbol{\theta}') \cdot H^*\left[H\Lambda(\omega_l; \boldsymbol{\theta}')H^*\right]^{-1} H\Lambda(\omega_l; \boldsymbol{\theta}') + \Lambda(\omega_l; \boldsymbol{\theta}')\, H^*\left[H\Lambda(\omega_l; \boldsymbol{\theta}')H^*\right]^{-1} \cdot \boldsymbol{Y}(\omega_l)\, \boldsymbol{Y}^*(\omega_l) \cdot \left[H\Lambda(\omega_l; \boldsymbol{\theta}')H^*\right]^{-1} \cdot H\Lambda(\omega_l; \boldsymbol{\theta}'). \quad (45)$$

Exploiting the block-diagonal form of $\Lambda(\omega; \boldsymbol{\theta})$, (43) becomes

$$U(\boldsymbol{\theta}, \boldsymbol{\theta}') = -\sum_{k=1}^{K} \sum_l \left[\log\det \pi\Lambda_k(\omega_l; \boldsymbol{\theta}_k) + \operatorname{tr}\left(\Lambda_k^{-1}(\omega_l; \boldsymbol{\theta}_k)\, \widehat{\boldsymbol{X}_k(\omega_l)\, \boldsymbol{X}_k^*(\omega_l)}\right)\right] \quad (46)$$

where $\widehat{\boldsymbol{X}_k(\omega_l)\, \boldsymbol{X}_k^*(\omega_l)}$ is the $(k, k)$ block of $\widehat{\boldsymbol{X}(\omega_l)\, \boldsymbol{X}^*(\omega_l)}$ given by

$$\widehat{\boldsymbol{X}_k(\omega_l)\, \boldsymbol{X}_k^*(\omega_l)} = \Lambda_k(\omega_l; \boldsymbol{\theta}_k') - \Lambda_k(\omega_l; \boldsymbol{\theta}_k') \cdot P^{-1}(\omega_l; \boldsymbol{\theta}')\, \Lambda_k(\omega_l; \boldsymbol{\theta}_k') + \Lambda_k(\omega_l; \boldsymbol{\theta}_k')\, P^{-1}(\omega_l; \boldsymbol{\theta}') \cdot \boldsymbol{Y}(\omega_l)\, \boldsymbol{Y}^*(\omega_l) \cdot P^{-1}(\omega_l; \boldsymbol{\theta}')\, \Lambda_k(\omega_l; \boldsymbol{\theta}_k') \quad (47)$$

where $P(\omega_l; \boldsymbol{\theta})$ is defined in (33). Observing that the maximization of $U(\boldsymbol{\theta}, \boldsymbol{\theta}')$ with respect to $\boldsymbol{\theta}$ is equivalent to the minimization of each of the terms in the $k$ sum of (46) *separately*, the EM algorithm assumes the following form.

*E step:* For $k = 1, 2, \cdots, K$, compute

$$\widehat{\boldsymbol{X}_k(\omega_l)\, \boldsymbol{X}_k^*(\omega_l)}^{(n)} = \Lambda_k(\omega_l; \hat{\boldsymbol{\theta}}_k^{(n)}) - \Lambda_k(\omega_l; \hat{\boldsymbol{\theta}}_k^{(n)}) \cdot P^{-1}(\omega_l; \hat{\boldsymbol{\theta}}^{(n)})\, \Lambda_k(\omega_l; \hat{\boldsymbol{\theta}}_k^{(n)}) + \Lambda_k(\omega_l; \hat{\boldsymbol{\theta}}_k^{(n)})\, P^{-1}(\omega_l; \hat{\boldsymbol{\theta}}^{(n)}) \cdot \boldsymbol{Y}(\omega_l)\, \boldsymbol{Y}^*(\omega_l) \cdot P^{-1}(\omega_l; \hat{\boldsymbol{\theta}}^{(n)})\, \Lambda_k(\omega_l; \hat{\boldsymbol{\theta}}_k^{(n)}). \quad (48)$$

*M step:* For $k = 1, 2, \cdots, K$,

$$\min_{\boldsymbol{\theta}_k} \sum_l \left[\log\det \Lambda_k(\omega_l; \boldsymbol{\theta}_k) + \operatorname{tr}\left(\Lambda_k^{-1}(\omega_l; \boldsymbol{\theta}_k)\, \widehat{\boldsymbol{X}_k(\omega_l)\, \boldsymbol{X}_k^*(\omega_l)}^{(n)}\right)\right] \rightarrow \hat{\boldsymbol{\theta}}_k^{(n+1)}. \quad (49)$$

We observe that $\hat{\boldsymbol{\theta}}_k^{(n+1)}$ is the ML estimate of $\boldsymbol{\theta}_k$ where the sufficient statistic $\boldsymbol{X}_k(\omega_l)\, \boldsymbol{X}_k^*(\omega_l)$ is substituted by its current estimate $\widehat{\boldsymbol{X}_k(\omega_l)\, \boldsymbol{X}_k^*(\omega_l)}^{(n)}$. The algorithm is illustrated in Fig. 2. The most attractive feature of the algorithm is that it decouples the full multidimensional optimization (35) into optimizations in the smaller-dimensional parameter subspaces. Thus, as in the deterministic signal case, the complexity of the algorithm is essentially unaffected by the assumed number of signal components. As $K$ increases, we have to increase the number of ML processors in parallel; however, an ML processor is maximized separately. Since the algorithm is based on the EM method, each iteration cycle increases the likelihood until convergence is accomplished.

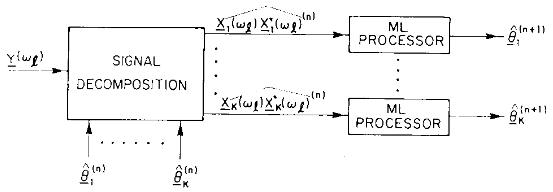



Fig. 2. The EM algorithm for stochastic Gaussian signals.

## IV. Application to Multipath Time-Delay Estimation

Let the observed signal $y(t)$ be modeled by

$$y(t) = \sum_{k=1}^{K} \alpha_k s(t - \tau_k) + n(t). \quad (50)$$

This model characterizes multipath effects where the transmitted signal $s(t)$ is observed at the receiver through more than one path. We shall concentrate here on the case where $s(t)$ is a deterministic known signal. The extension to the case where $s(t)$ is deterministic but unknown, and to the case where $s(t)$ is a sample function from a Gaussian random process, can be found in [11] and [1], respectively. We suppose that the additive noise $n(t)$ is spectrally flat over the receiver frequency band. The problem is to estimate the pairs $(\alpha_k, \tau_k)$, $k = 1, 2, \cdots, K$.

The direct ML approach requires the solution to [see (15)]

$$\min_{\substack{\tau_1, \tau_2, \cdots, \tau_K \\ \alpha_1, \alpha_2, \cdots, \alpha_K}} \left\{ \int_T \left| y(t) - \sum_{k=1}^{K} \alpha_k s(t - \tau_k) \right|^2 dt \right\}. \quad (51)$$

This optimization problem is addressed in [10], where it is shown that, at the optimum, the $\alpha_k$'s can be expressed in terms of the $\tau_k$'s. Thus, the $2K$-dimensional search can be reduced to a $K$-dimensional search. However, as pointed out in [10], for $K \geq 3$ the required computations become too involved. Consequently, ad hoc approaches and suboptimal solutions have been proposed. The most common solution consists of correlating $y(t)$ with a replica of $s(t)$ and searching for the $K$ highest peaks of the correlation function. If the various paths are resolvable, i.e., the difference between $\tau_k$ and $\tau_l$ is long compared to the temporal correlation of the signal for *all* combinations of $k$ and $l$, this approach yields near-optimal estimates. However, in situations where the signal paths are unresolvable, this approach is only distinctly suboptimal.

We identify the model in (50) as a special case of (13). Therefore, in correspondence with (30) and (31), we obtain the following algorithm.

*E step:* For $k = 1, 2, \cdots, K$, compute

$$\hat{x}_k^{(n)}(t) = \hat{\alpha}_k^{(n)} s(t - \hat{\tau}_k^{(n)}) + \beta_k \left[ y(t) - \sum_{l=1}^{K} \hat{\alpha}_l^{(n)} s(t - \hat{\tau}_l^{(n)}) \right]. \quad (52)$$

*M step:* For $k = 1, 2, \cdots, K$,

$$\min_{\alpha, \tau} \int_T \left| \hat{x}_k^{(n)}(t) - \alpha s(t - \tau) \right|^2 dt \rightarrow \hat{\alpha}_k^{(n+1)}, \hat{\tau}_k^{(n+1)}. \quad (53)$$

Assuming that the observation interval $T$ is long compared to the duration of the signal and the maximum expected delay, the two-parameter maximization required in (53) can be carried out in two steps as follows:

$$\max_{\tau} \left| g_k^{(n)}(\tau) \right| \rightarrow \hat{\tau}_k^{(n+1)} \quad (54)$$

$$\hat{\alpha}_k^{(n+1)} = \frac{g_k^{(n)}(\hat{\tau}_k^{(n+1)})}{E} \quad (55)$$

where $E = \int_T |s(t)|^2 dt$ is the signal energy and

$$g_k^{(n)}(\tau) = \int_T \hat{x}_k^{(n)}(t) s^*(t - \tau) dt. \quad (56)$$

We note that $g_k^{(n)}(\tau)$ can be generated by passing $\hat{x}_k^{(n)}(t)$ through a filter matched to $s(t)$. The algorithm is illustrated in Fig. 3. This computationally attractive algorithm decreases iteratively the objective function in (51) without ever going through the indicated multiparameter optimization. The complexity of the algorithm is essentially unaffected by the assumed number of signal paths. As $K$ increases, we have to increase the number of matched filters in parallel; however, each matched filter output is maximized separately.

If the number $K$ of signal paths is unknown, several criteria for its determination are developed in [12]. These criteria are based on the ML parameter estimates and can therefore be easily incorporated into the algorithm.

To demonstrate the performance of the algorithm, we have considered the following example. The observed signal $y(t)$ consists of three signal paths in additive noise

$$y(t) = \sum_{k=1}^{3} \alpha_k s(t - \tau_k) + n(t)$$

where $s(t)$ is the trapezoidal pulse

$$s(t) = \begin{cases} t/20 & 0 \leq t < 5 \\ 1/4 & 5 \leq t \leq 15 \\ (t - 10)/20 & 15 < t \leq 20. \end{cases}$$

The actual delays are

$$\tau_1 = 0 \quad \tau_2 = 5 \quad \tau_3 = 10,$$

and the amplitude scales are

$$\alpha_k = 1 \qquad \kappa = 1, 2, 3.$$

The additive noise is spectrally flat with a spectral level of $N = 0.025$ (so that the postintegration signal-to-noise ratio (SNR) per pulse is approximately 16 dB). The observed data consist of 100 time samples, as illustrated in

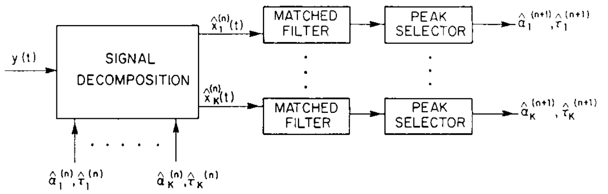



Fig. 3. The EM algorithm for multipath time-delay estimation.

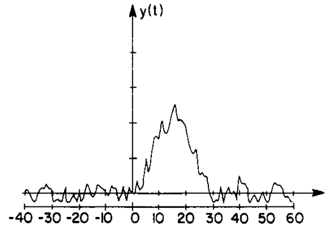



Fig. 4. The observed data.

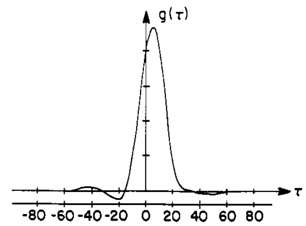



Fig. 5. The conventional matched filter response.

Fig. 4. In Fig. 5 we have plotted the matched filter output as a function of delay. As we can see, the conventional method cannot resolve the various signal paths and estimate their parameters.

First, we have computed the ML estimates by direct minimization of the objective function in (51) using exhaustive search. We obtain

$$\hat{\tau}_1 = 0.0117 \quad \hat{\tau}_2 = 5.0031 \quad \hat{\tau}_3 = 9.9884$$
$$\hat{\alpha}_1 = 1.1511 \quad \hat{\alpha}_2 = 0.7799 \quad \hat{\alpha}_3 = 0.9471,$$

and the value of the objective function at the minimum (corresponds to the value of the log-likelihood function at the maximum) is

$$J = 0.45879.$$

We have also computed lower bounds on the root mean square (rms) error of each parameter estimate using the Cramer–Rao inequality. We obtain

$$\sigma(\hat{\tau}_1) = 0.028 \quad \sigma(\hat{\tau}_2) = 0.030 \quad \sigma(\hat{\tau}_3) = 0.028$$
$$\sigma(\hat{\alpha}_1) = 0.076 \quad \sigma(\hat{\alpha}_2) = 0.079 \quad \sigma(\hat{\alpha}_3) = 0.076.$$

$\sigma(\hat{\tau}_k)$ denotes the minimum attainable rms error in the $\tau_k$ estimate, and $\sigma(\hat{\alpha}_k)$ denotes the minimum attainable rms error in the $\alpha_k$ estimate.

We now apply our algorithm. In Fig. 6 we have plotted the matched filter response to the various signal paths, as they are evolved with the iterations. In Fig. 7 we have tabulated the results using several arbitrarily selected starting points as indicated by the first line of each table. We see that after 10–15 iterations, the algorithm converges within the Cramer–Rao lower bound to ML estimates of all the unknown parameters, regardless of the initial guess. We note that the small differences in the final estimates result from the finite grid used for the optimization.

Using the asymptotic efficiency and lack of bias of the ML estimates, we can claim with some confidence that the rms error performance of the algorithm is the minimum attainable characterized by the Cramer–Rao lower bound.

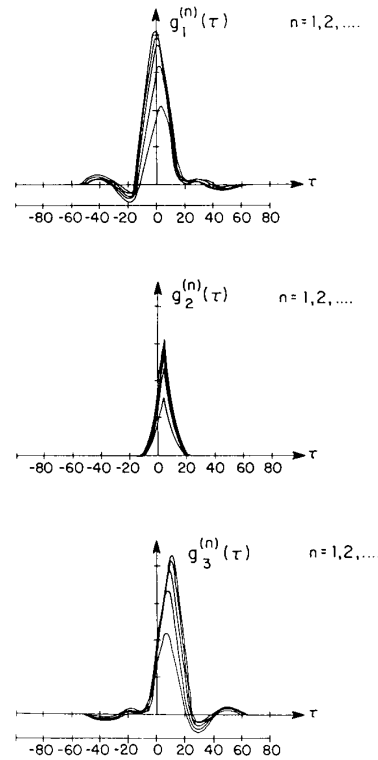



Fig. 6. The matched filter response to each signal path.

## V. Passive Multiple Source Location Estimation

The basic system of interest here consists of $K$ spatially distributed sources radiating noise-like signals toward an array of $M$ spatially distributed sensors. Assuming perfect propagation conditions in the medium, and ignoring amplitude attenuations of the signal wavefronts across the array, the actual waveform observed at the $m$th sensor output is

$$y_m(t) = \sum_{k=1}^{K} s_k(t - \tau_{km}) + n_m(t) \qquad m = 1, 2, \cdots, M \tag{57}$$

| n | $\hat{\tau}_1^{(n)}$ | $\hat{\tau}_2^{(n)}$ | $\hat{\tau}_3^{(m)}$ | $\hat{\alpha}_1^{(n)}$ | $\hat{\alpha}_2^{(n)}$ | $\hat{\alpha}_3^{(n)}$ | J |
|---|---|---|---|---|---|---|---|
| 0 | -5.000 | 7.000 | 12.000 | 0.5000 | 0.5000 | 0.5000 | 1.1846 |
| 1 | -3.471 | 6.664 | 11.136 | 0.6356 | 0.5356 | 0.6354 | 0.9475 |
| 2 | -2.638 | 5.892 | 10.694 | 0.7502 | 0.5842 | 0.6902 | 0.8223 |
| 3 | -1.305 | 5.557 | 10.373 | 0.8464 | 0.6142 | 0.7394 | 0.7199 |
| 4 | -0.821 | 5.380 | 10.258 | 0.8886 | 0.6699 | 0.7783 | 0.6733 |
| 5 | -0.516 | 5.268 | 10.112 | 0.9139 | 0.7082 | 0.8196 | 0.6205 |
| 6 | -0.339 | 5.142 | 10.059 | 0.9256 | 0.7205 | 0.8419 | 0.5834 |
| 7 | -0.162 | 5.084 | 10.034 | 0.9313 | 0.7524 | 0.8747 | 0.5582 |
| 8 | -0.078 | 5.053 | 10.022 | 0.9354 | 0.7703 | 0.8989 | 0.5265 |
| 9 | -0.033 | 5.037 | 10.016 | 0.9372 | 0.8018 | 0.9130 | 0.5012 |
| 10 | -0.014 | 5.024 | 10.012 | 0.9387 | 0.8360 | 0.9282 | 0.4826 |
| 11 | -0.007 | 5.018 | 10.009 | 0.9398 | 0.8591 | 0.9322 | 0.4707 |
| 12 | -0.003 | 5.015 | 10.006 | 0.9405 | 0.8673 | 0.9468 | 0.4681 |
| 13 | -0.001 | 5.012 | 10.004 | 0.9412 | 0.8757 | 0.9507 | 0.4639 |
| 14 | -0.001 | 5.010 | 10.002 | 0.9417 | 0.8743 | 0.9551 | 0.4612 |
| 15 | -0.000 | 5.009 | 10.001 | 0.9423 | 0.8768 | 0.9572 | 0.4597 |

| n | $\hat{\tau}_1^{(n)}$ | $\hat{\tau}_2^{(n)}$ | $\hat{\tau}_3^{(n)}$ | $\hat{\alpha}_1^{(n)}$ | $\hat{\alpha}_2^{(n)}$ | $\hat{\alpha}_3^{(n)}$ | J |
|---|---|---|---|---|---|---|---|
| 0 | -15.000 | 0.000 | 15.000 | 0.5000 | 0.5000 | [illegible] | [illegible] |
| 1 | -12.563 | 1.682 | 13.537 | 0.6773 | 0.5464 | [illegible] | [illegible] |
| 2 | -10.937 | 2.397 | 12.894 | 0.7242 | 0.5741 | [illegible] | [illegible] |
| 3 | -7.854 | 3.081 | 11.275 | 0.8009 | 0.6270 | [illegible] | 1.0828 |
| 4 | -4.709 | 3.990 | 10.861 | 0.8461 | 0.6638 | 0.7801 | [illegible] |
| 5 | -2.885 | 4.557 | 10.601 | 0.8725 | 0.7124 | 0.8240 | [illegible] |
| 6 | -1.518 | 4.832 | 10.518 | 0.8978 | 0.7513 | 0.8627 | [illegible] |
| 7 | -0.827 | 4.924 | 10.370 | 0.9023 | 0.7746 | 0.8895 | [illegible] |
| 8 | -0.395 | 4.961 | 10.126 | 0.9065 | 0.7971 | [illegible] | [illegible] |
| 9 | -0.187 | 4.983 | 10.088 | 0.9104 | 0.7956 | [illegible] | [illegible] |
| 10 | -0.094 | 4.992 | 10.056 | 0.9156 | 0.8044 | [illegible] | [illegible] |
| 11 | -0.048 | 4.996 | 10.039 | 0.9172 | 0.8091 | [illegible] | [illegible] |
| 12 | -0.025 | 4.998 | 10.024 | 0.9198 | 0.8116 | [illegible] | [illegible] |
| 13 | -0.011 | 4.999 | 10.011 | 0.9231 | 0.8178 | [illegible] | 0.4685 |
| 14 | -0.006 | 5.001 | 10.009 | 0.9265 | 0.8204 | [illegible] | [illegible] |
| 15 | -0.003 | 5.002 | 10.007 | 0.9287 | 0.8221 | 0.9476 | 0.4627 |

| n | $\hat{\tau}_1^{(n)}$ | $\hat{\tau}_2^{(n)}$ | $\hat{\tau}_3^{(n)}$ | $\hat{\alpha}_1^{(n)}$ | $\hat{\alpha}_2^{(n)}$ | $\hat{\alpha}_3^{(n)}$ | J |
|---|---|---|---|---|---|---|---|
| 0 | 30.000 | 40.000 | 50.000 | 0.5000 | 0.5000 | 0.5000 | 6.9146 |
| 1 | 23.822 | 35.372 | 44.625 | 0.6068 | 0.5632 | 0.5516 | 4.6387 |
| 2 | 17.432 | 29.241 | 38.781 | 0.6631 | 0.6321 | 0.6082 | 3.5261 |
| 3 | 12.321 | 23.542 | 32.232 | 0.7102 | 0.6918 | 0.6636 | 2.4471 |
| 4 | 8.746 | 18.352 | 25.724 | 0.7446 | 0.7467 | 0.7176 | 1.3299 |
| 5 | 5.022 | 14.606 | 19.621 | 0.7837 | 0.7845 | 0.7480 | 0.9465 |
| 6 | 3.319 | 10.902 | 15.807 | 0.8124 | 0.8268 | 0.7721 | 0.6827 |
| 7 | 2.582 | 8.456 | 13.466 | 0.8371 | 0.8550 | [illegible] | 0.6208 |
| 8 | 1.216 | 6.362 | 11.253 | 0.8509 | 0.8849 | 0.8302 | 0.5364 |
| 9 | 0.745 | 5.827 | 10.678 | 0.8627 | 0.9037 | 0.8565 | 0.5043 |
| 10 | 0.584 | 5.337 | [illegible] | 0.8761 | 0.9251 | 0.8692 | 0.4807 |
| 11 | 0.309 | 5.158 | 10.142 | 0.8845 | 0.9391 | 0.8810 | 0.4744 |
| 12 | 0.143 | 5.072 | 10.067 | 0.8912 | 0.9472 | 0.8927 | 0.4684 |
| 13 | 0.062 | 5.031 | 10.029 | 0.8963 | 0.9536 | 0.8989 | 0.4631 |
| 14 | 0.021 | 5.017 | 10.014 | [illegible] | 0.9603 | 0.9013 | 0.4611 |
| 15 | 0.007 | 5.008 | 10.005 | 0.9021 | 0.9652 | 0.9034 | 0.4597 |

| n | $\hat{\tau}_1^{(n)}$ | $\hat{\tau}_2^{(n)}$ | $\hat{\tau}_3^{(n)}$ | $\hat{\alpha}_1^{(n)}$ | $\hat{\alpha}_2^{(n)}$ | $\hat{\alpha}_3^{(n)}$ | J |
|---|---|---|---|---|---|---|---|
| 0 | 50.000 | 30.000 | -50.000 | 0.5000 | 0.5000 | 0.5000 | [illegible] |
| 1 | 37.927 | 24.347 | -36.489 | 0.7158 | 0.5324 | 0.6173 | [illegible] |
| 2 | 26.833 | 13.205 | -28.634 | 0.8364 | 0.5514 | 0.6812 | [illegible] |
| 3 | 19.496 | 9.881 | -17.156 | 0.8552 | 0.5929 | 0.7235 | 2.2951 |
| 4 | 12.155 | 7.437 | -8.207 | 0.8867 | 0.6331 | 0.7507 | 1.9131 |
| 5 | 7.928 | 6.248 | 0.323 | 0.9234 | 0.6874 | 0.7910 | [illegible] |
| 6 | 4.193 | 5.657 | 4.926 | 0.9605 | 0.7016 | 0.8272 | 1.1032 |
| 7 | 2.141 | 5.301 | 7.547 | 1.0131 | 0.7357 | 0.8534 | 0.7119 |
| 8 | 1.028 | 5.172 | 8.813 | 1.0552 | 0.7521 | 0.8875 | 0.5935 |
| 9 | 0.510 | 5.091 | 9.452 | 1.0976 | 0.7669 | 0.9174 | 0.5296 |
| 10 | 0.257 | 5.054 | 9.780 | 1.1254 | 0.7718 | 0.9217 | 0.4921 |
| 11 | 0.132 | 5.033 | 9.847 | 1.1395 | 0.7755 | [illegible] | [illegible] |
| 12 | 0.078 | 5.018 | 9.923 | 1.1464 | 0.7774 | [illegible] | [illegible] |
| 13 | 0.045 | 5.011 | 9.965 | [illegible] | 0.7783 | [illegible] | [illegible] |
| 14 | 0.029 | 5.007 | 9.971 | 1.1497 | 0.7791 | [illegible] | [illegible] |
| 15 | 0.021 | 5.005 | 9.979 | 1.1504 | 0.7795 | [illegible] | [illegible] |

Fig. 7. Tables of results for the multipath time-delay estimation.

where $s_k(t)$ is the $k$th source signal, $n_m(t)$ is the additive noise at the $m$th sensor output, and $\tau_{km}$ is the travel time of the signal wavefront from the $k$th source to the $m$th sensor.

Information concerning the various source location parameters can be extracted by measuring the various $\tau_{km}$. In the passive case, one can only measure the travel time differences, obtainable by selecting one sensor as a reference and comparing its output to that of every other sensor. If we let sensor $M$ be the reference and set $\tau_{km} = 0$, then $\tau_{km}$ measures the travel time difference of the $k$th signal wavefront to the $(m, M)$ sensor pair.

To simplify the exposition, suppose that the various signal sources are relatively far-field so that the observed signal wavefronts are essentially planar across the array. If we further suppose that the array sensors are colinear, then

$$\tau_{km} = \frac{d_m}{c} \sin \theta_k \tag{58}$$

where $d_m$ is the spacing between sensors $m$ and $M$, $c$ is the velocity of propagation in the medium, and $\theta_k$ is the angle of arrival of the $k$th signal wavefront relative to the boresight.

Substituting (58) into (57) and concatenating the various equations, we obtain

$$\boldsymbol{y}(t) = \sum_{k=1}^{K} \boldsymbol{s}_k(t; \boldsymbol{\theta}_k) + \boldsymbol{n}(t) \tag{59}$$

where

$$\boldsymbol{s}_k(t; \boldsymbol{\theta}_k) = \begin{bmatrix} s_k(t - \gamma_1 \sin \theta_k) \\ s_k(t - \gamma_2 \sin \theta_k) \\ \vdots \\ s_k(t - \gamma_{M-1} \sin \theta_k) \\ s_k(t) \end{bmatrix} \tag{60}$$

where $\gamma_m = d_m / c$.

Suppose that the various $s_k(t)$ and the various $n_m(t)$ are mutually independent, WSS, zero-mean Gaussian random processes with spectral densities $S_k(\omega)$ and $N_m(\omega)$, respectively (the cases where the $s_k(t)$ are deterministic known/unknown signals are presented in [2] and [11], respectively). We want to find the ML estimates of $\theta_1, \theta_2, \cdots, \theta_K$ given observations of $\boldsymbol{y}(t)$.

Assuming that the observation interval $T$ is long compared to the correlation time (inverse bandwidth) of the signals and the noises, the direct ML approach requires the solution to [see (35)]

$$\min_{\theta_1, \theta_2, \cdots, \theta_K} \sum_{l} \left[\operatorname{logdet} \boldsymbol{P}(\omega_l; \boldsymbol{\theta}) + \boldsymbol{Y}^*(\omega_l) \cdot \boldsymbol{P}^{-1}(\omega_l; \boldsymbol{\theta}) \boldsymbol{Y}(\omega_l)\right]. \tag{61}$$

where $\boldsymbol{Y}(\omega_l)$ are the Fourier transform coefficients [or the DFT in the discrete case of $\boldsymbol{y}(t)$] and

$$\boldsymbol{P}(\omega; \boldsymbol{\theta}) = \sum_{k=1}^{K} S_k(\omega) \boldsymbol{V}(\omega; \theta_k) \boldsymbol{V}^*(\omega; \theta_k) + N(\omega) \tag{62}$$

where

$$\boldsymbol{V}(\omega; \theta_k) = \begin{bmatrix} e^{-j\omega\gamma_1 \sin\theta_k} \\ e^{-j\omega\gamma_2 \sin\theta_k} \\ \vdots \\ e^{-j\omega\gamma_{M-1} \sin\theta_k} \\ 1 \end{bmatrix} \tag{63}$$

and

$$N(\omega) = \begin{bmatrix} N_1(\omega) & & (0) \\ & N_2(\omega) & \\ (0) & & N_M(\omega) \end{bmatrix}. \tag{64}$$

This is a complicated multiparameter optimization problem in several unknowns. Consequently, numerous ad hoc solutions and suboptimal approaches have been proposed in the recent literature (e.g., [13]-[18]). Still, the most common approach consists of beamforming and searching for the $K$ highest peaks. If the various signal sources are widely separated, this approach is nearly optimal. However, if the source signals are closely spaced, we are likely to obtain poor estimates of the various $\boldsymbol{\theta}_k$'s.

Identifying the model in (59) as a special case of (13), the algorithm specified by (48) and (49) is directly applicable, where

$$\Lambda_k(\omega; \theta_k) = S_k(\omega) \boldsymbol{V}(\omega; \theta_k) \boldsymbol{V}^*(\omega; \theta_k) + \beta_k N(\omega). \tag{65}$$

Now,

$$\det \Lambda_k(\omega; \theta_k) = \left[1 + \frac{1}{\beta_k} S_k(\omega) \boldsymbol{V}^*(\omega; \theta_k) \cdot N^{-1}(\omega) \boldsymbol{V}(\omega; \theta_k)\right] \det [\beta_k N(\omega)] \tag{66}$$

and

$$\Lambda^{-1}(\omega; \theta_k) = \frac{1}{\beta_k}\left[N^{-1} - \frac{S_k(\omega)}{\beta_k + S_k(\omega) \boldsymbol{V}^*(\omega; \theta_k) N^{-1}(\omega) \boldsymbol{V}(\omega; \theta_k)} \cdot N^{-1}(\omega) \boldsymbol{V}(\omega; \theta_k) \boldsymbol{V}^*(\omega; \theta_k) N^{-1}(\omega)\right]. \tag{67}$$

Substituting (66) and (67) into (49) and ignoring the terms that are independent of $\theta_k$, the $M$ step can be further simplified. The resulting algorithm is as follows.

*E step:* For $k = 1, 2, \cdots, K$, compute

$$\widehat{X_k(\omega_l)\, X_k^*(\omega_l)}^{(n)} = \Lambda_k(\omega_l; \hat{\theta}_k^{(n)}) - \Lambda_k(\omega_l; \hat{\theta}_k^{(n)}) \cdot P^{-1}(\omega_l; \hat{\boldsymbol{\theta}}^{(n)})\, \Lambda_k(\omega_l; \hat{\theta}_k^{(n)}) + \Lambda_k(\omega_l; \hat{\theta}_k^{(n)})\, P^{-1}(\omega_l; \hat{\boldsymbol{\theta}}^{(n)}) \cdot Y(\omega_l)\, Y^*(\omega_l) \cdot P^{-1}(\omega_l; \hat{\boldsymbol{\theta}}^{(n)})\, \Lambda_k(\omega_l; \hat{\theta}_k^{(n)}). \quad (68)$$

*M step:* For $k = 1, 2, \cdots, K$,

$$\max_{\theta} \sum_{l} V^*(\omega_l; \theta)\, N^{-1}(\omega_l)\, \widehat{X_k(\omega_l)\, X_k^*(\omega_l)}^{(n)} \cdot N^{-1}(\omega_l)\, V(\omega_l; \theta) \rightarrow \hat{\theta}_k^{(n+1)}.$$

We note that the objective function in (69) is the array beamformer, where the product $X_k(\omega_l)\, X_k^*(\omega_l)$ is substituted by its current estimate $\widehat{X_k(\omega_l)\, X_k^*(\omega_l)}^{(n)}$. The algorithm is illustrated in Fig. 8. This computationally attractive algorithm decreases iteratively the objective function in (61) without ever going through the indicated multiparameter optimization. The complexity of the algorithm is essentially unaffected by the assumed number of signal sources. As $K$ increases, we have to increase the number of beamformers in parallel; however, each beamformer output is maximized separately.

To demonstrate the performance of the algorithm, we have considered the following example. The array consists of five colinear and evenly spaced sensors. There are two far-field signal sources at bearings

$$\theta_1 = 0^\circ \quad \theta_2 = 10^\circ$$

relative to the boresight. The array–source geometry is shown in Fig. 9. The signals and the noises are spectrally flat with $S_k(\omega) = S$ and $N_m(\omega) = N$ over the frequency band $[-W/2, W/2]$. We assume that $S/N = 1$ and that $WT/2\pi = 20$ (so that the postintegration SNR per channel is approximately 23 dB). The array length is taken to be $L = 6\lambda$ where $\lambda$ is the wavelength associated with the highest frequency in the signal band.

In Fig. 10 we have plotted the array beamformer response as a function of the bearing angle. As we can see, the conventional beamformer cannot resolve between the signal sources and estimate their bearings.

The ML estimates, obtained by direct minimization of the objective function in (61), are

$$\hat{\theta}_1 = -0.0563 \quad \hat{\theta}_2 = 10.4556,$$

and the value of the objective function at the minimum (corresponds to the value of the log-likelihood function at the maximum) is

$$J = 159.0137.$$

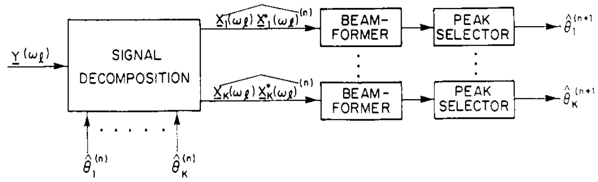



Fig. 8. The EM algorithm for multiple source angle-of-arrival estimation.

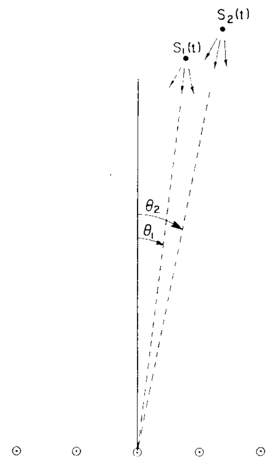


Fig. 9. Array–source geometry.

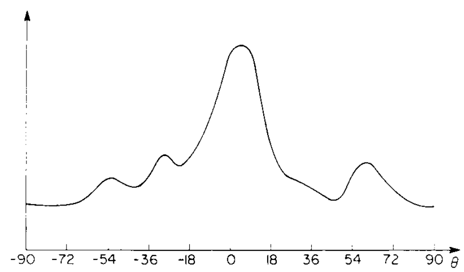



Fig. 10. The conventional beamformer.

We have also computed the Cramer–Rao lower bound on the rms error of each parameter estimate. We obtain

$$\sigma(\hat{\theta}_1) = 0.2680 \quad \sigma(\hat{\theta}_2) = 0.2722.$$

We now apply our algorithm. In Fig. 11 we have plotted the beamformer response to the various signal sources as they are evolved with iterations. In Fig. 12 we have tabulated the results using several arbitrarily selected initial guesses. We see that in all cases, after 5–10 iterations, the algorithm essentially converges, within the Cramer–Rao lower bound, to the ML estimates of all the unknown bearing parameters simultaneously, and the various signal sources are correctly resolved.

## VI. Discussion

We have presented a computationally efficient algorithm for ML parameter estimation of superimposed signals based on the EM method. The algorithm is developed

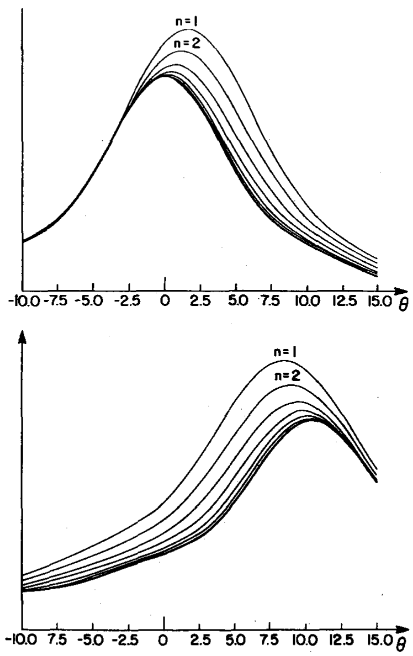



Fig. 11. The beamformer response to each signal source.

| $n$ | $\hat{\theta}_1^{(n)}$ | $\hat{\theta}_2^{(n)}$ | $J$ |
|---|---|---|---|
| 0 | 2.0000 | 8.0000 | 269.6614 |
| 1 | 1.5776 | 8.4652 | 241.1250 |
| 2 | 1.1220 | 8.9744 | 210.3016 |
| 3 | .6932 | 9.4568 | 184.1323 |
| 4 | .3716 | 9.8588 | 168.3571 |
| 5 | .1572 | 10.1268 | 161.7404 |
| 6 | .0500 | 10.2876 | 159.7169 |
| 7 | −.0036 | 10.3680 | 159.1958 |
| 8 | −.0304 | 10.4216 | 159.0440 |
| 9 | −.0304 | 10.4484 | 159.0218 |

| $n$ | $\hat{\theta}_1^{(n)}$ | $\hat{\theta}_2^{(n)}$ | $J$ |
|---|---|---|---|
| 0 | −5.0000 | 13.0000 | 520.2958 |
| 1 | −2.4692 | 11.9492 | 294.7128 |
| 2 | −1.2632 | 11.1452 | 196.5459 |
| 3 | −.6736 | 10.7164 | 167.7777 |
| 4 | −.3788 | 10.5288 | 161.0492 |
| 5 | −.2448 | 10.4484 | 159.6113 |
| 6 | −.1644 | 10.4216 | 159.2068 |
| 7 | −.1108 | 10.4216 | 159.0730 |
| 8 | −.0840 | 10.4216 | 159.0405 |

| $n$ | $\hat{\theta}_1^{(n)}$ | $\hat{\theta}_2^{(n)}$ | $J$ |
|---|---|---|---|
| 0 | 4.0000 | 7.0000 | 378.5662 |
| 1 | 3.6412 | 7.2592 | 351.2384 |
| 2 | 3.2928 | 7.5272 | 328.5127 |
| 3 | 2.9176 | 7.8220 | 306.1186 |
| 4 | 2.4888 | 8.1436 | 281.8299 |
| 5 | 2.0332 | 8.5188 | 254.9325 |
| 6 | 1.5240 | 8.9476 | 224.2842 |
| 7 | 1.0148 | 9.4032 | 194.4633 |
| 8 | .5860 | 9.8052 | 173.5086 |
| 9 | .2912 | 10.1000 | 163.5225 |
| 10 | .1304 | 10.2608 | 160.3573 |
| 11 | .0500 | 10.3680 | 159.3599 |
| 12 | −.0036 | 10.4216 | 159.0797 |
| 13 | −.0304 | 10.4484 | 159.0218 |

| $n$ | $\hat{\theta}_1^{(n)}$ | $\hat{\theta}_2^{(n)}$ | $J$ |
|---|---|---|---|
| 0 | 7.0000 | 13.0000 | 497.0527 |
| 1 | 6.0532 | 12.6996 | 448.0088 |
| 2 | 4.9544 | 12.3780 | 385.8327 |
| 3 | 3.8288 | 12.0832 | 318.2541 |
| 4 | 2.7836 | 11.8420 | 256.9945 |
| 5 | 1.9528 | 11.6276 | 214.1421 |
| 6 | 1.3632 | 11.4132 | 189.1838 |
| 7 | .9344 | 11.2256 | 175.1189 |
| 8 | .6396 | 11.0380 | 167.4013 |
| 9 | .4520 | 10.9040 | 163.6597 |
| 10 | .3180 | 10.7968 | 161.5851 |
| 11 | .2108 | 10.7164 | 160.3880 |
| 12 | .1304 | 10.6360 | 159.6668 |
| 13 | .0768 | 10.5824 | 159.3330 |
| 14 | .0500 | 10.5556 | 159.2103 |
| 15 | .0232 | 10.5288 | 159.1171 |
| 16 | −.0036 | 10.5020 | 159.0536 |

Fig. 12. Tables of results for the multiple source angle-of-arrival estimation.

for the case of deterministic (known) signals and for the case of stationary Gaussian signals. The most striking feature of the algorithm is that it decouples the full multidimensional search associated with the direct ML approach into searches in smaller-dimensional parameter subspaces, leading to a considerable simplification in the computations involved. The algorithm is applied to the multipath time-delay and multiple source location estimation problems; in both cases, we demonstrate the performance of the algorithm in a given example and show that the algorithm converges iteratively to the exact ML estimate of all the unknown parameters simultaneously where each iteration increases the likelihood.

We finally note that the derivation of the EM algorithm for the linear–Gaussian case (the Appendix) has importance of its own since it covers a wide range of applications.

## Appendix

## Development of the EM Algorithm for the Linear–Gaussian Case

Suppose the "complete" data $X$ and the observed (incomplete) data $Y$ are related by the linear transformation

$$Y = HX \tag{A1}$$

where $H$ is a noninvertible matrix and $X$ possesses the following multivariate Gaussian probability density:

$$f_X(x; \theta) = \frac{1}{\left[\det\left(\frac{2\pi}{\lambda}\Lambda(\theta)\right)\right]^{\lambda/2}} \cdot \exp\left[-\frac{\lambda}{2}(x - m(\theta))^* \cdot \Lambda^{-1}(\theta)(x - m(\theta))\right] \tag{A2}$$

where $\lambda = 1$ if $X$ is real-valued and $\lambda = 2$ if $X$ is complex-valued. Taking the logarithm of (A2),

$$\begin{aligned}\log f_X(x; \theta) &= c - \frac{\lambda}{2}\left[\text{logdet } \Lambda(\theta) + (x - m(\theta))^* \Lambda^{-1}(\theta)(x - m(\theta))\right] \\ &= c - \frac{\lambda}{2}\left[\text{logdet } \Lambda(\theta) + m^*(\theta)\Lambda^{-1}(\theta) \cdot m(\theta) - x^*\Lambda^{-1}(\theta)m(\theta) - m^*(\theta)\Lambda^{-1}(\theta)x + \text{tr}\left(\Lambda(\theta)xx^*\right)\right]\end{aligned} \tag{A3}$$

where $c$ is a constant independent of $\theta$. Taking the conditional expectation of (A3) given $Y = y$ at a parameter value $\theta'$,

$$U(\theta, \theta') = c - \frac{\lambda}{2}\left[\text{logdet } \Lambda(\theta) + m^*(\theta)\Lambda^{-1}(\theta)m(\theta) - \hat{x}^*\Lambda^{-1}(\theta)m(\theta) - m^*(\theta)\Lambda^{-1}(\theta)\hat{x} + \text{tr}\left(\Lambda^{-1}(\theta)\widehat{xx^*}\right)\right] \tag{A4}$$

where

$$\hat{x} = E\{x/Y = y; \theta'\} \tag{A5}$$

and

$$\widehat{xx^*} = E\{xx^*/Y = y; \theta'\}. \tag{A6}$$

Since $X$ and $Y$ are related by a linear transformation, they are jointly Gaussian, and the conditional expectations required in (A5) and (A6) can be computed by a straightforward modification of existing results. We obtain

$$\hat{x} = m(\theta') + \Lambda(\theta')H^*\left[H\Lambda(\theta')H^*\right]^{-1} \cdot \left[y - Hm(\theta')\right] \tag{A7}$$

$$\widehat{xx^*} = \Lambda(\theta') - \Lambda(\theta')H^*\left[H\Lambda(\theta')H^*\right]^{-1} \cdot H\Lambda(\theta') + \hat{x}\hat{x}^*. \tag{A8}$$

Note that if we set $\theta' = \theta$, (A7) and (A8) are the well-known formulas for the conditional expectations in the Gaussian case (e.g., [19]).

Substituting (A7) and (A8) into (A4), the function $U(\theta, \theta')$ required for the EM algorithm is given in a closed form. We observe that $U(\theta, \theta')$ and $\log f_X(x; \theta)$ have the same dependence on $\theta$. Maximizing $U(\theta, \theta')$ with respect to $\theta$ is therefore the same as maximizing $\log f_X(x; \theta)$ with respect to $\theta$. Hence, the EM algorithm essentially requires the ML solution in the $X$ model, which might be significantly simpler than the direct ML solution in the $Y$ model.

In correspondence with (11) and (12), the EM algorithm for the linear–Gaussian case is given by the following.

*E step:*

$$\hat{X}^{(n)} = m(\hat{\theta}^{(n)}) + \Lambda(\hat{\theta}^{(n)}) \cdot H^*\left[H\Lambda(\hat{\theta}^{(n)})H^*\right]^{-1}\left[y - Hm(\hat{\theta}^{(n)})\right] \tag{A9}$$

$$\widehat{xx^*}^{(n)} = \Lambda(\hat{\theta}^{(n)}) - \Lambda(\hat{\theta}^{(n)})H^*\left[H\Lambda(\hat{\theta}^{(n)})H^*\right]^{-1} \cdot H\Lambda(\hat{\theta}^{(n)}) + \hat{x}^{(n)}\hat{x}^{(n)*}. \tag{A10}$$

*M step:*

$$\min_{\theta}\left\{\text{logdet } \Lambda(\theta) + m^*(\theta) \cdot \Lambda^{-1}(\theta)m(\theta) - \hat{x}^{(n)*}\Lambda^{-1}(\theta)m(\theta) - m^*(\theta)\Lambda^{-1}(\theta)\hat{x}^{(n)} + \text{tr}\left(\Lambda^{-1}(\theta)\widehat{xx^*}^{(n)}\right)\right\} \to \hat{\theta}^{(n+1)}. \tag{A11}$$

## Acknowledgment

The authors wish to thank D. Kelter and Y. Eyal for their help in the simulations, and the referees for many helpful comments.

## References

[1] M. Feder and E. Weinstein, "Optimal multiple source location estimation via the EM algorithm," in *Proc. 1985 Int. Conf. Acoust., Speech, Signal Processing (ICASSP '85)*, Tampa Bay, FL, Mar. 1985, pp. 1762-1765.

[2] —, "Multipath and multiple source array processing via the EM algorithm," in *Proc. 1986 Int. Conf. Acoust., Speech, Signal Processing (ICASSP '86)*, Tokyo, Japan, Apr. 1986.

[3] J. R. Grindon, "Estimation theory for a class of nonlinear random vector channels," Ph.D. dissertation, Washington Univ., St. Louis, MO, 1970.

[4] J. S. Lim and A. V. Oppenheim, "All pole modeling of degraded speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, pp. 197-210, 1978.

[5] M. Segal, "Resolution and estimation techniques with application to vectorial miss-distance indicator," M.Sc. thesis, Technion–Israel Inst. Technol., Haifa, Israel, Aug. 1981.

[6] N. M. Laird, A. P. Dempster, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Ann. Roy. Stat. Soc.*, pp. 1-38, Dec. 1977.

[7] C. F. J. Wu, "On the convergence properties of the EM algorithm," *Ann. Stat.*, no. 11, pp. 95-103, 1983.

[8] I. Meilijson, "A fast improvement to the EM algorithm on its own terms," submitted for publication.

[9] H. L. Van Trees, *Detection Estimation and Modulation Theory, Part I.* New York: Wiley, 1968.

[10] J. E. Ehrenberg, T. E. Ewart, and R. D. Morris, "Signal processing techniques for resolving individual pulses in multipath signal," *J. Acoust. Soc. Amer.*, vol. 63, no. 6, pp. 1861-1865, 1978.

[11] M. Feder and E. Weinstein, "Optimal multiple source location estimation via the EM algorithm," Woods Hole Oceanographic Institution, Woods Hole, MA, Tech. Rep. WHOI-85-25, July 1985.

[12] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 387-392, 1985.

[13] S. S. Reddy, "Multiple source location—A digital approach," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-15, pp. 95-105, 1979.

[14] G. V. Borgiotti and L. J. Kaplan, "Superresolution of uncorrelated interference sources by using adaptive array techniques," *IEEE Trans. Antennas Propagat.*, vol. AP-27, pp. 842-845, 1979.

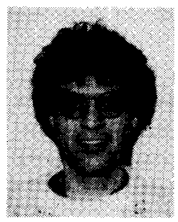
[15] R. O. Schmidt, "A signal subspace approach to multiple emitter location and spectral estimation," Ph.D. dissertation, Stanford Univ., Stanford, CA, 1981.

[16] G. Su and M. Morf, "The signal subspace approach for multiple wideband emitter location," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1503-1522, 1983.

[17] B. Porat and B. Friedlander, "Estimation of spatial and spectral parameters of multiple sources," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 412-425, 1983.

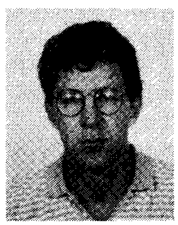
[18] A. Nehorai, G. Su, and M. Morf, "Estimation of time difference of arrival for multiple ARMA sources by pole decomposition," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-31, pp. 1478-1491, 1983.

[19] B. O. Anderson and J. B. Moore, *Optimal Filtering.* Englewood Cliffs, NJ: Prentice-Hall, 1979.



**Meir Feder** received the B.Sc. and M.Sc. degrees (summa cum laude) from Tel-Aviv University, Tel-Aviv, Israel, and the Sc.D. degree from the Massachusetts Institute of Technology, Cambridge, and the Woods Hole Oceanographic Institution, Woods Hole, MA, all in electrical engineering, in 1980, 1984, and 1987, respectively.

He is currently a Research Associate in the Department of Electrical Engineering and Computer Science at M.I.T. His research interests lie in the broad area of signal processing, stochastic estimation, and information theory. He is particularly interested in investigating new statistical estimation techniques and using them in signal processing applications, e.g., speech, image, and array processing problems.



**Ehud Weinstein** received the B.Sc. degree from the Technion-Israel Institute of Technology, Haifa, Israel, and the Ph.D. degree from Yale University, New Haven, CT, both in electrical engineering, in 1975 and 1978, respectively.

He is currently an Associate Professor in the Department of Electronic Systems, Faculty of Engineering, Tel-Aviv University, Tel-Aviv, Israel, and an Adjunct Scientist with the Department of Ocean Engineering, Woods Hole Oceanographic Institution, Woods Hole, MA. His research interests are in the general area of statistical estimation theory and its application to signal and array processing.

Dr. Weinstein is the co-recipient of the IEEE Acoustics, Speech, and Signal Processing Society's 1983 Senior Award.