MAP-PF POSITION TRACKING WITH A NETWORK OF SENSOR ARRAYS

Kristine L. Bell

Dept. of Applied & Engineering Statistics
George Mason University
Fairfax, VA 22030-4444, USA
kbell@gmu.edu

ABSTRACT

The maximum a posteriori penalty function (MAP-PF) approach is applied to target position tracking with a network of sensor arrays. The track estimation problem is formulated directly from the array data as using the MAP estimation criterion. The penalty function method of nonlinear programming is used to obtain a tractable solution. A sequential update procedure is developed in which penalized maximum likelihood estimates of target bearing and power are computed for each array, and then used as synthetic measurements in an extended Kalman filter. The two steps are coupled via the penalty function. The current target state is used to guide the bearing estimation, and estimated signal powers control the influence of the bearing estimates from each array on the final track estimate. The algorithm can be implemented in a decentralized manner where bearing estimation is performed at the arrays, and track estimation is performed at a central processing site.

1. INTRODUCTION

Traditional position tracking techniques partition the track estimation operation into two isolated processes: bearing-only estimation at each array, followed by position track estimation from the bearing estimates. In this paper, we consider estimation of the target state directly from the data at all of the arrays using the maximum a posteriori (MAP) criterion. This approach is based on the MAP penalty function (MAP-PF) tracking technique developed for tracking the bearings of multiple targets at a single array in [1]. A key feature of the approach is the use of the penalty function method of nonlinear programming to obtain a tractable solution. A sequential track state update procedure similar to the extended Kalman filter (EKF) is developed which updates the state first from the motion model, and then from the current array data. Penalized maximum likelihood estimates of the source bearing and power are computed for each array, which then act as a synthetic "measurements" in an EKF update of the state. The two-step estimation process is similar to traditional methods, except the processes are coupled via the penalty function. In the bearing estimation step, the current target state is used to guide the estimation process and help eliminate ambiguous or spurious estimates. In the track estimation step, estimated signal powers control the influence of the bearing estimates from each array on the final track estimate. The algorithm can be implemented in a decentralized manner where bearing estimation is performed at the arrays, and track estimation is performed at a central processing site.

This research was sponsored by the U.S. Army Research Laboratory and was accomplished under Cooperative Agreement Number DAAD19-02-2-0004. The views and conclusions contained in this document are those of the author and should not be interpreted as presenting the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

2. STATISTICAL MODEL AND ASSUMPTIONS

The model consists of a moving target radiating a narrowband signal that is received by a network of $\mathcal{Q}$ sensor arrays, each with N_q elements. The target and the arrays are assumed to lie in the xy-plane. Let (x_k, y_k) denote the target's position at time k and $(\dot{x}_k, \dot{y}_k)$ denote its velocity. The four-dimensional target state at time k is $\mathbf{x}_k = [x_k\ \dot{x}_k\ y_k\ \dot{y}_k]^T$. We assume the motion of the objects is described by a first order Gauss-Markov process, i.e.,

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{w}_k, \tag{1}$$

where $\mathbf{F}$ is the state transition matrix and $\mathbf{w}_k$ is a zero mean white Gaussian noise process with covariance matrix $\mathbf{Q}$ which is assumed known and fixed over the observation period. Under these assumptions, the probability density function (pdf) of $\mathbf{x}_k$ given $\mathbf{x}_{k-1}$ is

$$p(\mathbf{x}_k|\mathbf{x}_{k-1}) = \frac{\exp\left\{-\frac{1}{2}(\mathbf{x}_k - \mathbf{F}\mathbf{x}_{k-1})^T\mathbf{Q}^{-1}(\mathbf{x}_k - \mathbf{F}\mathbf{x}_{k-1})\right\}}{|2\pi\mathbf{Q}|^{\frac{1}{2}}}, \tag{2}$$

where $|\mathbf{Q}|$ denotes the determinant of $\mathbf{Q}$. There are $\mathcal{K}$ snapshots in an observation batch. No data is available at $k = 0$ so we assume the prior distribution on initial object states, $p(\mathbf{x}_0)$, is Gaussian with mean $\bar{\mathbf{x}}_0$ and covariance $\mathbf{\Omega}_0$.

Let (d^x_{qn}, d^y_{qn}) denote the position of the nth element of the qth array; $n = 1, \ldots, N_q$ and $q = 1, \ldots\ \mathcal{Q}$. The element positions can also be described relative to a reference position for each array. Let (d^x_q, d^y_q) denote the reference position of the qth array and $(\Delta^x_{qn}, \Delta^y_{qn})$ denote the location of the nth element with respect to the reference position. Then $(d^x_{qn}, d^y_{qn}) = (d^x_q + \Delta^x_{qn}, d^y_q + \Delta^y_{qn})$. The target range r_{qk} is the distance between the target and the reference position of the array,

$$r_{qk} = \sqrt{\left(x_k - d^x_q\right)^2 + (y_k - d^y_q)^2}, \tag{3}$$

and the target bearing ϕ_{qk} is the angle between the target and the reference position of the array,

$$\phi_{qk} = \arctan\left(\frac{y_k - d^y_q}{x_k - d^x_q}\right). \tag{4}$$

We define the nonlinear function

$$h_q(\mathbf{x}_k) \equiv \phi_{qk} = \arctan\left(\frac{y_k - d_q^y}{x_k - d_q^x}\right) \quad (5)$$

to express the bearing as a function of the target state, and the $\mathcal{Q}\times 1$ vector

$$\mathbf{h}(\mathbf{x}_k) = [h_1(\mathbf{x}_k)\ h_2(\mathbf{x}_k)\ \cdots\ h_{\mathcal{Q}}(\mathbf{x}_k)]^T \quad (6)$$

to represent the collection of target bearings across all arrays.

The signals received at the elements of a particular array are modelled as coherent, time-shifted versions of the same signal, but the signals received at different arrays are modelled as different, uncorrelated signals. The signals and noise are assumed to be sample functions of stationary zero-mean Gaussian random processes. At the qth array, the $N_q \times 1$ observed data vector during the kth observation snapshot has the form

$$\mathbf{z}_{qk} = s_{qk}\mathbf{v}_q(h_q(\mathbf{x}_k)) + \mathbf{n}_{qk}, \quad (7)$$

where s_{qk} is a random signal sample from the target at the kth snapshot with $E[s_{qk}s_{qk}^*] = \gamma_{qk}$. The vector $\mathbf{v}_q(h_q(\mathbf{x}_k))$ is the $N_q \times 1$ array response vector to a target whose state is $\mathbf{x}_k$. We use this notation to emphasize that when the array element spacings are small relative to the separation between the source and the array, the $N_q \times 1$ array response vector depends on the target position only through the bearing $h_q(\mathbf{x}_k)$. The vector $\mathbf{n}_{qk}$ is an $N_q \times 1$ vector of uncorrelated sensor noise samples at the kth snapshot. It is assumed that the snapshots are sufficiently spaced that the observations are independent from snapshot to snapshot. The signal powers, γ_{qk}, are assumed to be time-varying and unknown. The noise covariance matrix is assumed to be constant and known with $E[\mathbf{n}_{qk}\mathbf{n}_{qk}^H] = \sigma_q^2\mathbf{I}$.

At a particular array and snapshot, the array data $\mathbf{z}_{qk}$ is then jointly complex Gaussian with zero mean and covariance matrix

$$\mathbf{K}_{qk}(\mathbf{x}_k, \gamma_{qkl}) = \gamma_{qk}\mathbf{v}_q(h_q(\mathbf{x}_k))\mathbf{v}_q(h_q(\mathbf{x}_k))^H + \sigma_q^2\mathbf{I}, \quad (8)$$

and the pdf of the data vector conditioned on the target state is given by

$$p(\mathbf{z}_{qk}|\mathbf{x}_k;\gamma_{qk}) = \frac{\exp\left\{-\mathbf{z}_{qk}^H\mathbf{K}_{qk}^{-1}(\mathbf{x}_k,\gamma_{qk})\mathbf{z}_{qk}\right\}}{|\pi\mathbf{K}_{qk}(\mathbf{x}_k,\gamma_{qk})|}. \quad (9)$$

We have used the notation $p(\mathbf{z}_{qk}|\mathbf{x}_k;\gamma_{qk})$ to indicate that the pdf is conditioned on the random vector $\mathbf{x}_k$ but is also a function of the non-random but unknown parameter γ_{qk}.

We denote the collection of target powers over all arrays at time k as $\boldsymbol{\gamma}_k \equiv \{\gamma_{1k}, \gamma_{2k}, \ldots, \gamma_{\mathcal{Q}k}\}$ and the collection of data vectors across arrays as $\mathbf{z}_k \equiv \{\mathbf{z}_{1k}, \mathbf{z}_{2k}, \ldots, \mathbf{z}_{\mathcal{Q}k}\}$. At each snapshot, the joint pdf of the data conditioned on the target state is the product of the pdfs for each array and is given by:

$$p(\mathbf{z}_k|\mathbf{x}_k;\boldsymbol{\gamma}_k) = \prod_{q=1}^{Q} p(\mathbf{z}_{qk}|\mathbf{x}_k;\gamma_{qk}). \quad (10)$$

The single snapshot joint pdf of the observations and target state conditioned on the previous target state is then

$$p(\mathbf{z}_k, \mathbf{x}_k|\mathbf{x}_{k-1};\boldsymbol{\gamma}_k) = p(\mathbf{z}_k|\mathbf{x}_k;\boldsymbol{\gamma}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}). \quad (11)$$

Let $\mathbf{X}$, $\boldsymbol{\Gamma}$, and $\mathbf{Z}$ denote the collections $\mathbf{X} \equiv \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{\mathcal{K}}\}$, $\boldsymbol{\Gamma} \equiv \{\boldsymbol{\gamma}_1, \boldsymbol{\gamma}_2, \ldots, \boldsymbol{\gamma}_{\mathcal{K}}\}$, and $\mathbf{Z} \equiv \{\mathbf{z}_1, \ldots, \mathbf{z}_{\mathcal{K}}\}$. The joint pdf of the array snapshot data and target states over the batch is given by

$$p(\mathbf{Z}, \mathbf{X};\boldsymbol{\Gamma}) = p(\mathbf{x}_0)\prod_{k=1}^{\mathcal{K}} p(\mathbf{z}_k|\mathbf{x}_k;\boldsymbol{\gamma}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1}). \quad (12)$$

3. MAP TRACKING ALGORITHM

We wish to estimate the random variable $\mathbf{X}$ (the target track) from the observations $\mathbf{Z}$. Two optimization criteria commonly used in Bayesian estimation problems are minimum mean square error (MMSE) and maximum a posteriori probability (MAP). The MMSE estimate is the mean of the a posteriori pdf $p(\mathbf{X}|\mathbf{Z})$ while the MAP estimate is found at the peak of $p(\mathbf{X}|\mathbf{Z})$, or equivalently at the peak of the joint pdf $p(\mathbf{Z}, \mathbf{X})$. In many cases, these two estimates coincide [2]. In the classical single target tracking problem where the observations are a linear function of the target states and the observations and states are Gaussian, the discrete time Kalman filter provides both the MMSE and MAP estimates of the target states given the observations [3]. In the problem considered here, the observations depend on the target states in a nonlinear manner, therefore the MMSE and MAP estimates will not coincide. We also have the added complication of the unknown nuisance parameter vector $\boldsymbol{\Gamma}$. The MAP methodology provides a tractable framework for solving this problem. We can jointly find the MAP estimate of $\mathbf{X}$ and the maximum likelihood (ML) estimate of $\boldsymbol{\Gamma}$ by maximizing the joint pdf $p(\mathbf{Z}, \mathbf{X};\boldsymbol{\Gamma})$, or equivalently its logarithm, with respect to both $\mathbf{X}$ and $\boldsymbol{\Gamma}$.

The MAP/ML estimates of $\mathbf{X}$ and $\boldsymbol{\Gamma}$ are the solutions to the optimization problem:

$$\max_{\mathbf{X},\boldsymbol{\Gamma}} \ \ln\left[p(\mathbf{x}_0)\prod_{k=1}^{\mathcal{K}} p(\mathbf{z}_k|\mathbf{x}_k;\boldsymbol{\gamma}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})\right]. \quad (13)$$

We note that the pdf $p(\mathbf{z}_k|\mathbf{x}_k;\boldsymbol{\gamma}_k)$ depends on the target state $\mathbf{x}_k$ through the vector of bearings $\mathbf{h}(\mathbf{x}_k)$. Following [1], we introduce a set of $\mathcal{KQ}$ auxiliary variables μ_{qk} for $q = 1, \ldots, \mathcal{Q}$ and $k = 1, \ldots, \mathcal{K}$ to assist in the solution. Let the $\mathcal{Q} \times 1$ vector $\boldsymbol{\mu}_k = [\mu_{1k}\ \mu_{2k}\ \cdots\ \mu_{\mathcal{Q}k}]^T$ represent the collection of these variables across all arrays and define the collection of these vectors over snapshots as $\mathbf{M} \equiv \{\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \ldots, \boldsymbol{\mu}_{\mathcal{K}}\}$. We replace the bearing vector $\mathbf{h}(\mathbf{x}_k)$ with the new auxiliary variable vector $\boldsymbol{\mu}_k$ in the $p(\mathbf{z}_k|\mathbf{x}_k;\boldsymbol{\gamma}_k)$ term. In order to retain the original optimization problem, we then require the new variables to be equal to the old variables, i.e. $\boldsymbol{\mu}_k = \mathbf{h}(\mathbf{x}_k)$. The unconstrained optimization problem in (13) can be written as an equivalent constrained optimization problem as follows:

$$\max_{\mathbf{X},\mathbf{M},\boldsymbol{\Gamma}} \ \ln\left[p(\mathbf{x}_0)\prod_{k=1}^{\mathcal{K}} p(\mathbf{z}_k;\boldsymbol{\mu}_k,\boldsymbol{\gamma}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})\right] \quad (14)$$
$$s.t. \quad \boldsymbol{\mu}_k = \mathbf{h}(\mathbf{x}_k), \ \ k = 1, \ldots, \mathcal{K},$$

where $p(\mathbf{z}_k;\boldsymbol{\mu}_k,\boldsymbol{\gamma}_k) \equiv p(\mathbf{z}_k|\mathbf{x}_k;\boldsymbol{\gamma}_k)|_{\mathbf{h}(\mathbf{x}_k)=\boldsymbol{\mu}_k}$.

This formulation allows us to use the penalty method of nonlinear programming [4] for constrained optimization problems. It is an iterative procedure in which a sequence of easier unconstrained optimization problems is solved. The easier problems are related to the original constrained problem by a continuous, differentiable penalty function which is equal to zero in the feasible region where the constraints are satisfied, and which is negative in the infeasible region. The penalty function relaxes the equality constraint resulting in a problem which is an approximation to the original problem. With each iteration, a stronger penalty is imposed for infeasibility, and the solution to the unconstrained problem converges to the solution to the original constrained problem. An overview of the method and the convergence properties is provided in [1]. As in [1], we use the quadratic penalty function,

$$
\begin{aligned}
P(\mathbf{X},\mathbf{M}) &= -\sum_{k=1}^{\mathcal{K}}\sum_{q=1}^{\mathcal{Q}} \frac{(\mu_{qk}-h_q(\mathbf{x}_k))^2}{2\xi_{qk}} \qquad (15)\\
&= -\frac{1}{2}\sum_{k=1}^{\mathcal{K}}(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k))^T\mathbf{R}_k^{-1}(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k)),
\end{aligned}
$$

where $\mathbf{R}_k = \text{diag}\left([\xi_{1k}\ \xi_{2k}\ \cdots\ \xi_{\mathcal{Q}k}]\right)$, and ξ_{qk} are parameters that affect the strength of the penalty.

To enforce a more costly penalty at each iteration, a term $(r_i)^{-1}$ scales the penalty function, where i is the iteration index and $r_i, i = 1, 2, \ldots$ is a positive, decreasing sequence converging to zero. The penalized unconstrained maximization problem at the ith iteration is given by

$$
\begin{aligned}
\max_{\mathbf{X},\mathbf{M},\boldsymbol{\Gamma}}\quad & \ln\left[p(\mathbf{x}_0)\prod_{k=1}^{\mathcal{K}} p(\mathbf{z}_k;\boldsymbol{\mu}_k,\boldsymbol{\gamma}_k)p(\mathbf{x}_k|\mathbf{x}_{k-1})\right]\\
&-\frac{1}{2r_i}\sum_{k=1}^{\mathcal{K}}(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k))^T\mathbf{R}_k^{-1}(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k)). \qquad (16)
\end{aligned}
$$

Expanding and rearranging the terms in (16), we have

$$
\begin{aligned}
\max_{\mathbf{X},\mathbf{M},\boldsymbol{\Gamma}}\quad & \sum_{k=1}^{\mathcal{K}}\ln p(\mathbf{z}_k;\boldsymbol{\mu}_k,\boldsymbol{\gamma}_k)\\
&+\left(\ln p(\mathbf{x}_0)+\sum_{k=1}^{\mathcal{K}}\ln p(\mathbf{x}_k|\mathbf{x}_{k-1})\right) \qquad (17)\\
&-\frac{1}{2}\sum_{k=1}^{\mathcal{K}}(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k))^T\left(r_i\mathbf{R}_k\right)^{-1}\left(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k)\right).
\end{aligned}
$$

Note that the first term is only a function of $\mathbf{Z}$, $\mathbf{M}$ and $\boldsymbol{\Gamma}$, the second term is only a function of $\mathbf{X}$, and the third term provides the coupling between the parameter sets $\mathbf{M}$ and $\mathbf{X}$.

First consider that for a fixed $\mathbf{M}$ and $\boldsymbol{\Gamma}$, we can find an estimate of $\mathbf{X}$ by maximizing over the second and third terms in (17). Expanding the pdfs, the problem becomes

$$
\begin{aligned}
\max_{\mathbf{X}}\quad & -(\mathbf{x}_0-\bar{\mathbf{x}}_0)^T\boldsymbol{\Sigma}_0^{-1}(\mathbf{x}_0-\bar{\mathbf{x}}_0)\\
&-\sum_{k=1}^{\mathcal{K}}(\mathbf{x}_k-\mathbf{F}\mathbf{x}_{k-1})^T\mathbf{Q}^{-1}(\mathbf{x}_k-\mathbf{F}\mathbf{x}_{k-1}) \qquad (18)\\
&-\sum_{k=1}^{\mathcal{K}}(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k))^T\left(r_i\mathbf{R}_k\right)^{-1}\left(\boldsymbol{\mu}_k-\mathbf{h}(\mathbf{x}_k)\right).
\end{aligned}
$$

This problem has the form of the classical single source tracking problem with $\boldsymbol{\mu}_k$ acting as the noisy measurement vector, which has a Gaussian distribution with mean $\mathbf{h}(\mathbf{x}_k)$ and covariance matrix $r_i\mathbf{R}_k$. The extended Kalman filter (EKF) provides a sequential update algorithm similar to the standard Kalman filer by using a linear approximation to $\mathbf{h}(\mathbf{x}_k)$ [3].

Next consider that for a fixed $\mathbf{X}$, we can solve for both $\mathbf{M}$ and $\boldsymbol{\Gamma}$ by maximizing over the first and third terms in (17). These terms decouple with respect to the snapshots and arrays, therefore this problem reduces to solving $\mathcal{KQ}$ separate single source bearing estimation problems of the form

$$
\max_{\mu_{qk},\boldsymbol{\gamma}_{qk}}\quad \ln p(\mathbf{z}_{qk};\mu_{qk},\boldsymbol{\gamma}_{qk})-\frac{(\mu_{qk}-h_q(\mathbf{x}_k))^2}{2r_i\xi_{qk}}. \qquad (19)
$$

This is a penalized maximum likelihood (PML) estimation problem in which the bearing estimate is found from a one-dimensional search over bearing space and the power estimate has a closed form expression as a function of bearing.

We then alternate between finding the bearing and power estimates using the PML algorithm, and the track estimate via the EKF. As presented above, there are two levels of iteration: the penalty method iteration in which the penalty parameter r_i forces the solution into the feasible region and the alternating maximization iteration between bearing/power estimation and track estimation. Each of the iterations may be performed until a convergence criterion is satisfied or for a fixed number of cycles. The trade-off is algorithm complexity versus estimation accuracy.

The MAP-PF algorithm described above is a batch algorithm that provides an elegant solution to the problem of target tracking with a network of sensor arrays. However, in real applications we often cannot wait while a batch of data is collected and processed, and we would like a sequential solution in which state estimates are computed each time a data snapshot is received. A slight modification to the batch algorithm can be made which allows for a fully sequential implementation. First, the alternating maximization iteration between bearing/power estimation and track state estimation as well as the penalty method iteration are both reduced to one step. The penalty parameter r_1 is set to one and the terms ξ_{qk} must be chosen carefully to strongly enforce the penalty as well as to act as the measurement error variance in the tracking stage. We set ξ_{qk} to be inversely proportional to the target's estimated power, i.e. $\xi_{qk} = \beta/\hat{\gamma}_{qk}$, where β is a tuning parameter that must specified. Next, in (19), we replace ξ_{qk} with $\xi_{q,k-1}$ and $\mathbf{x}_k$ with the predicted state $\hat{\mathbf{x}}_{k|k-1}$ from the extended Kalman filter. An explicit pseudo-code description of the sequential algorithm is provided in Table 1.

Table 1. Sequential MAP-PF tracking algorithm pseudo code.

Initialize $\hat{\mathbf{x}}_0 \equiv \bar{\mathbf{x}}_0, \mathbf{P}_{0|0} \equiv \boldsymbol{\Omega}_0, \mathbf{R}_0, \beta$
for $k = 1, \ldots, \mathcal{K}$
Predict current state
$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}\hat{\mathbf{x}}_{k-1}$
$\mathbf{H}_k = \left.\dfrac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}^T}\right|_{\mathbf{x}=\hat{\mathbf{x}}_{k|k-1}}$
Bearing and Power estimates
for $q = 1, \ldots, \mathcal{Q}$
$\hat{\mu}_{qk} =$
$\underset{\mu}{\text{argmax}}\quad \eta_{qk}(\mu)+\ln\eta_{qk}(\mu)-\dfrac{(\mu-h_q(\hat{\mathbf{x}}_{k|k-1}))^2}{2\xi_{q,k-1}}$
where $\eta_{qk}(\mu) = \max\left[1, \left|\mathbf{z}_{qk}^H\mathbf{v}_q(\mu)\right|^2/\left|\mathbf{v}_q(\mu)\right|^2\sigma_q^2\right]$
$\hat{\gamma}_{qk} = \left(\eta_{qk}(\hat{\mu}_{qk})-1\right)\sigma_q^2/\left|\mathbf{v}_{ql}(\hat{\mu}_{qk})\right|^2$
$\xi_{qk} = \beta/\hat{\gamma}_{qk}$
end $\{q\}$
$\boldsymbol{\mu}_k = [\mu_{1k}\ \cdots\ \mu_{\mathcal{Q}k}]^T$
$\mathbf{R}_k = \text{diag}\left([\xi_{1k}\ \xi_{2k}\ \cdots\ \xi_{\mathcal{Q}k}]\right)$
EKF State Update
$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^T+\mathbf{Q}$
$\mathbf{G}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k^T\left\{\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k{}^T+\mathbf{R}_k\right\}^{-1}$
$\mathbf{P}_{k|k} = \mathbf{P}_{k|k-1}-\mathbf{G}_k\mathbf{H}_k\mathbf{P}_{k|k-1}$
$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1}+\mathbf{G}_k\left\{\hat{\boldsymbol{\mu}}_k-\mathbf{h}(\hat{\mathbf{x}}_{k|k-1})\right\}$
end $\{k\}$

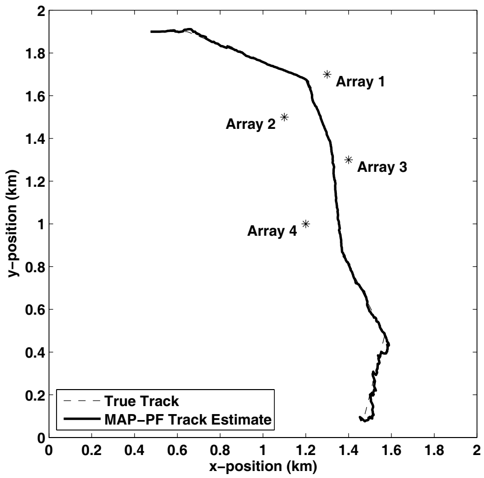

Fig. 1. MAP-PF track estimate.

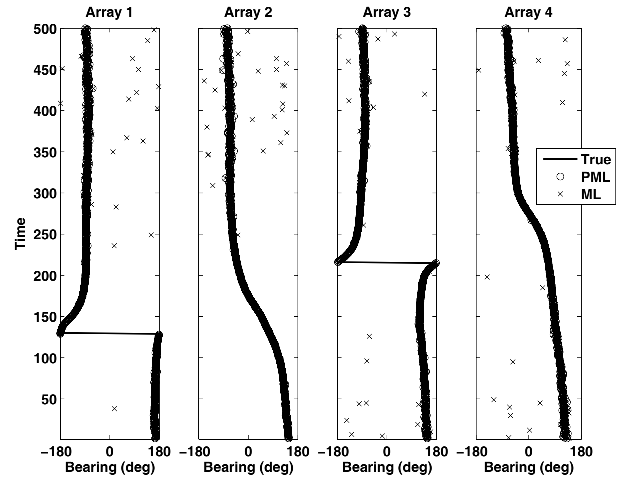

Fig. 2. Penalized ML estimates from MAP-PF tracker.

4. SIMULATION RESULTS

We consider a scenario which is typical of unattended ground sensor networks used to localize and track ground targets such as tanks, trucks, and other military vehicles [5]-[7]. The scenario is shown in Figure 1, where four acoustic sensor arrays are placed at distributed sites and a vehicle travels along a track between the arrays. Acoustic emissions of the target are collected by the arrays and each site has the ability to perform computations to process the data locally as well as to transmit raw and/or processed data to a central processing site. The arrays are close to the track travelled by the vehicle, therefore the range and bearing of the target seen at a particular array changes rapidly as the target passes by. Plots of the target bearing vs. time as the vehicle moves past the arrays from northwest to southeast are shown in Figure 2.

The four acoustic sensor arrays are each six element circular arrays with half wavelength spacing. The vehicle travels at a relatively constant speed which varies between 15-20 km/hr due to maneuvers. The target power is 45 dB at 15 m. and varies inversely with range squared. The tracker was initialized with the correct target state at the beginning of the track.

Figure 1 shows bearing estimates obtained at each array using ML with and without the penalty function. Without the penalty function, good estimates are obtained at each array when the target is close by, but many erroneous estimates are produced when the target is far away. A tracker using these estimates as measurements is not able to maintain a track, even during the period when the target is passing the arrays. The penalized ML bearing estimates are significantly better than the ML estimates, and exhibit no outliers. Figure 2 shows the track estimate obtained by the MAP-PF tracking algorithm. The track estimate is quite close to the actual track, even at the end of the run when the vehicle is a long distance from any of the arrays.

The penalty function is critical to the success of the MAP-PF tracking algorithm. During bearing estimation, it prevents erroneous estimates which can cause the tracker to lose track. During the track update step, it allows the tracker to combine the bearing estimates from the different arrays in the most useful way.

5. EXTENSIONS

This technique can be generalized to handle multiple wideband sources in a straightforward manner. This generalization has been applied to real aeroacoustic data from field tests conducted by the Army Research Laboratory in [6]-[7].

6. REFERENCES

[1] R. E. Zarnich, K. L. Bell, and H. L. Van Trees, “A Unified Method for Measurement and Tracking of Multiple Contacts from Sensor Array Data,” *IEEE Trans. Signal Processing*, vol. 49, no. 12, pp. 2950-2961, Dec. 2001.

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

[3] Y. Bar-Shalom and X. R. Li, *Estimation and Tracking: Principles, Techniques and Software*. Storrs, CT: YBS Publishing, 1998.

[4] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*. New York, NY: McGraw-Hill, 1996.

[5] R. J. Kozick and B. M. Sadler, “Source Localization with Distributed Sensor Arrays and Partial Spatial Coherence,” *IEEE Trans. Signal Processing*, vol. 52, no. 3, pp. 601-616, Mar. 2004.

[6] K. L. Bell,“Maximum a Posteriori (MAP) Target Tracking Using a Network of Sensor Arrays,” *Proc. 2003 Meeting MSS Battlefield Acoustics*, Laurel, MD, Oct. 2003. *Restricted distribution*.

[7] K. L. Bell,“Maximum a Posteriori (MAP) Multi-Target Tracking Using a Network of Sensor Arrays,” *Proc. 2004 Meeting MSS Battlefield Acoustics*, Laurel, MD, Aug. 2004. *Restricted distribution*.