Acoustic Node Calibration Using a Moving Source

Volkan Cevher, *Student Member, IEEE*, James H. McClellan, *Fellow, IEEE*

Abstract

Acoustic nodes, each containing an array of microphones, can track targets in x-y space from their received acoustic signals, if the node positions and orientations are known exactly. However, it is not always possible to deploy the nodes precisely, so a calibration phase is needed to estimate the position and the orientation of each node before doing any tracking or localization. An acoustic node can be calibrated from sources of opportunity such as beacons or a moving source. In this paper, we derive and compare several calibration methods for the case where the node can hear a moving source whose position can be reported back to the node. Since calibration from a moving source is, in effect, the dual of a tracking problem, methods derived for acoustic target trackers are used to obtain robust and high resolution acoustic calibration processes. For example, two direction-of-arrival-based calibration methods can be formulated based on combining angle estimates, geometry, and the motion dynamics of the moving source. In addition, a maximum-likelihood (ML) solution is presented using a narrow-band acoustic observation model, along with a Newton-based search algorithm that speeds up the calculation the likelihood surface. The ML estimate serves as a basis for comparison, so the Cramér-Rao lower bound on the node position estimates is also derived to show that the effect of position errors for the moving source on the estimated node position is much less severe than the variance in angle estimates from the microphone array. The performance of the calibration algorithms is demonstrated on synthetic and field data.

I. Introduction

Acoustic arrays with directionally sensitive or omnidirectional microphones can be used to localize and track targets using direction-of-arrival (DOA) estimates, derived from the measurements of their sounds [1]–[3]. If an acoustic node is defined to be an array of omnidirectional

V. Cevher and J. H. McClellan are with the Center for Signal and Image Processing, School of ECE, Georgia Institute of Technology, Atlanta GA 30332-0250.

Prepared through collaborative participation in the Advanced Sensors Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-02-0008.

microphones, whose relative positions are known with respect to each other, then the node calibration problem consists of determining the array center position (geometrical centroid) and the array orientation. Note that this problem differs from the problem of calibrating the individual microphone positions previously considered in the literature [4], [5]. In [4], accurate localization of individual microphone positions is done by considering the effects of the calibration on the array manifold matrix that affects the DOA estimate. On the other hand, the objective of this paper is to calibrate one or more nodes, in which the individual microphones have fixed relative positions. Multiple nodes would then be used to estimate target position via triangularization as shown in Fig. 1.

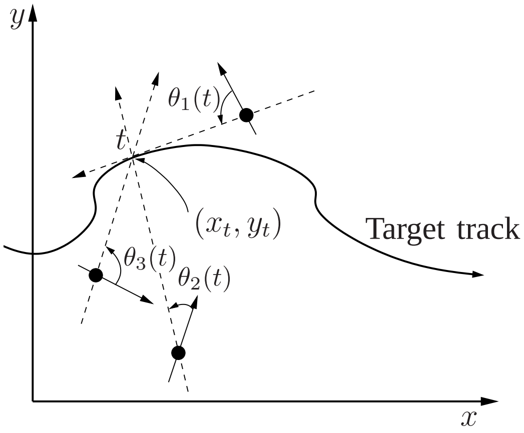


Fig. 1. Black dots represent the acoustic node array centers and the solid arrows through these nodes represent their reference orientations for the local DOA estimates. The DOAs are measured counterclockwise from the reference orientation. If the node positions and orientations are known, then it is possible to determine the target position (x_t, y_t) via, for example, triangulation.

There has been some previous work in calibrating acoustic nodes using beacons or moving sources [6]–[8]. The calibration scenario considered in [7], [8] employs a maximum-likelihood solution, using the time-of-arrival and direction-of-arrival of known point calibration sources deployed in the scene, to determine the array position and orientation. The scenario considered in this paper is similar to one considered in [6], where a moving calibration source is available. The multi-path effects are ignored, while deriving these calibration methods. Moreover, in this paper, it is assumed that the calibration source can report its position, e.g., acquired via the global positioning system (GPS). The acoustic nodes themselves are assumed to not have GPS capability due to battery limitations or jamming susceptibility (justifications can be found in [7].) The imperfect GPS position estimates of the moving source is modelled as noisy, and the effects of the GPS noise on the estimation performance of the node positions are treated in Section IV-A.

Several calibration methods are proposed in this paper, including a Maximum Likelihood (ML) solution that works directly with the acoustic microphone array output signals, and two DOA-based algorithms that first process the microphone outputs to estimate the angle to the moving source, and then calibrate from that angle information using simple geometry. The DOA-based algorithms include a Synthetic Aperture Calibration Method and a Metropolis-Hastings (MH) Calibration Method. We give the Cramér-Rao performance bounds for the calibration problem for these different estimation methods and also consider the sufficiency of the DOA estimates for the DOA-based algorithms.

The acoustic propagation delays significantly degrade the acoustic calibration accuracy, if not incorporated in the solution. A time-synchronization step is required to align the received acoustic data with the reported calibration target GPS points, a time-warp function $\tau(t)$ is introduced that is determined by using the time that the acoustic data, transmitted at time t, takes to arrive at the acoustic node. It is a function of the distance of the calibration target and the speed of sound c. For the tractability of the ML and the synthetic aperture calibration methods, we approximate the time-warp function and use an iterative technique to refine the position estimates. The Metropolis-Hastings method has a built-in time-warp strategy that avoids approximations. The relative sensitivity of the ML and the synthetic aperture algorithms to the approximations of the time-warp function is also discussed.

The maximum-likelihood estimator of the node-center location is derived by using a narrow-band acoustic array data model [9]. This algorithm is relatively insensitive to the approximations done on the time-warp function. To reduce the computational complexity, a Newton search algorithm is employed to avoid calculating the whole likelihood surface. To further speed up the algorithm, line search algorithms are used [10]. The line search algorithms can be derived based on at least one of the two Wolfe conditions: (i) the sufficient decrease condition and (ii) the curvature condition [10], [11]. For our problem, a *backtracking* line search algorithm is used to identify the largest acceptable step size, based on the first Wolfe condition.

The Synthetic Aperture calibration algorithm is a *sufficient statistics* algorithm that is based on the synthetic aperture idea used in the radar problems [12]. The algorithm creates virtual acoustic arrays along the calibration target track to triangulate back the node position. It is a sufficient statistics algorithm, because it is based on the DOAs calculated at these synthetic arrays, as opposed to the acoustic signals directly [13]. The synthetic aperture DOAs are used to determine the position in two ways: least squares estimation or maximum-likelihood estimation. The least squares solution assumes that the errors are Gaussian around the node position, whereas the maximum-likelihood solution assumes that the errors are normal in the temporal DOA track. Computer simulations demonstrate that the latter assumption is more accurate for calibration. Although computationally very attractive, the synthetic aperture calibration algorithm is very sensitive to the approximations on the time-warp function. In addition, it is found by simulations that the synthetic aperture method is biased estimator of the node position.

The Metropolis-Hastings (MH) calibration method is based on the mode-hungry accelerated version of the Metropolis-Hastings sampling algorithm [14]–[17]. It is also a sufficient statistics algorithm, because auxiliary DOA estimates are used to estimate the node position. The algorithm proposes a number of node positions in the 2D plane. Then, these node positions are perturbed until the discrete node positions are distributed according to the data-likelihood. To calculate the data-likelihood, the observed acoustic data is first used to calculate a set of motion compensated DOAs by using the time-warp function and the target doppler shift that are determined by the target GPS and the proposed node position. Then, the consistency of these motion compensated DOAs with the DOAs that are based on the geometry of the node position and the GPS, is used to determine the likelihood. The MH calibration method does not require any approximations of the time-warp function. Moreover, because of the unimodal nature of the node position density, the mode-hungry modification is used to decrease the number of Monté-Carlo iterations [14].

The organization of the paper is as follows. Section II formulates the problem and presents the ML solution along with performance bounds, and the Newton search algorithm. Section III describes the DOA-based calibration algorithms that use auxiliary DOA estimates and geometrical arguments. Computer simulations of typical scenarios are provided in Sect. IV.

II. Maximum-Likelihood Solution for the Calibration Problem

In this section, we present the ML solution for the node position and orientation, given a moving calibration source and the local measurements of its sound at the node microphones. It is assumed that the calibration source has a narrow-band time-frequency signature and its position estimates at each time are supplied by a GPS device. The GPS errors are modelled as *i.i.d.* Gaussian. Their effects on the calibration performance are considered in a later subsection.

An example calibration source is a helicopter, which also has the capability to deploy the nodes in the field. The speed of such a calibration source introduces non-stationarity problems not often treated in the standard array model for beamforming [1]. The ML solution, as well as the DOA-based calibration methods given in Sect. III, can handle this non-stationarity by assuming constant velocity target movement between GPS measurements. Finally, we note that our solution is also applicable, when using stationary calibration sources such as beacons.

A. ML Solution

We define $\boldsymbol{\xi}$ as the vector consisting of the unknown node-center position (x, y) and the unknown node orientation φ in the 2D plane:

$$\boldsymbol{\xi} = [x, y, \varphi]^T. \tag{1}$$

The known (noisy) track of the moving source supplied by the GPS is $\boldsymbol{\chi}_t = [x_T(t), y_T(t)]^T$. Using $\boldsymbol{\xi}$ and $\boldsymbol{\chi}_t$, the node-to-source bearing angle θ_t (measured counterclockwise with respect to the node orientation) and the node-to-target range R_t are given by

$$\begin{aligned} \theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t) &\triangleq -\varphi + \tan^{-1}\left(\frac{y - y_T(t)}{x - x_T(t)}\right), \\ R_t &\triangleq \|\boldsymbol{\xi}_{x,y} - \boldsymbol{\chi}_t\|. \end{aligned} \tag{2}$$

If the node estimate $\boldsymbol{\xi}$ is known, there is a one-to-one correspondence between $\boldsymbol{\chi}_t$ and (θ_t, R_t).

The acoustic signals at the node microphones are modelled according to the narrow-band acoustic observation model discussed in [4], [5], [9], [18], because the calibration source is assumed narrow-band. This is a reasonable assumption for many targets such as helicopters (see Fig. 10). For a node with P microphones, the narrow-band observation model relates the complex envelope of the node microphone outputs to that of the target signal, respectively called $\mathbf{y}(t) \in \mathbb{C}^{P\times 1}$ and $s(t) \in \mathbb{C}^1$. The observation model is

$$\mathbf{y}(t) = \mathbf{a}(\theta_t)s(t) + \mathbf{n}_{\mathbf{a}}(t), \tag{3}$$

where $\mathbf{a}(\theta_t) \in \mathbb{C}^{P\times 1}$ is the narrow-band steering vector for a target at DOA θ_t, and $\mathbf{n}_{\mathbf{a}}(t) \in \mathbb{C}^{P\times 1}$ is additive noise. For a narrow-band calibration source, whose center frequency is f_0, the p^{th}

element ($p = 1, 2, \ldots, P$) of the vector $\mathbf{a}(\theta_t)$ is given by

$$[\mathbf{a}(t)]_p \triangleq [\mathbf{a}(\theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t))]_p = \exp\left[j \frac{2\pi f_0 \rho_p}{c} \sin(\theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t) + \phi_p) \right], \tag{4}$$

where c is the speed of sound, and (ρ_p, ϕ_p) is the p^{th} microphone position given in polar coordinates.

When the target reports its GPS position $\boldsymbol{\chi}_t$ electronically, the GPS information arrives at the node with the speed of light. However, the acoustic data corresponding to GPS-reported position arrives at the node with the speed of sound, which is six orders of magnitude slower. Hence, time-synchronization is required to match the acoustic data to proper GPS points. Denote $\tau(t)$ as the arrival time at the node for a sound emitted from the "source" at time t. Then, the following time-warping needs to be applied to synchronize the acoustic data and the GPS estimates:

$$\tau(t) = t + ||\boldsymbol{\xi}_{x,y} - \boldsymbol{\chi}_t||/c. \tag{5}$$

Assuming *i.i.d.* Gaussian array noise $\mathbf{n}_\mathbf{a}(t)$, the probability density function (pdf) of the observed data with the appropriate time-warping is given by

$$p(\mathbf{Y}_K | \boldsymbol{\xi}, \boldsymbol{\chi}_0, \ldots, \boldsymbol{\chi}_t) = \prod_{t=0}^{K-1} \frac{1}{\pi^P \sigma^{2P}} \exp\left[-\frac{1}{\sigma^2} \|\mathbf{y}(\tau(t)) - \mathbf{a}(t)s(t)\|^2 \right], \tag{6}$$

where σ^2 is the array noise variance, K is the total number of observations at the sampling frequency F_s, $\mathbf{a}(t)$ is defined in (4), and $\mathbf{Y}_K$ is the aggregate data vector formed by stacking all the observed data:

$$\mathbf{Y}_K = \begin{bmatrix} \mathbf{y}(t) \\ \mathbf{y}(t+\kappa) \\ \vdots \\ \mathbf{y}(t+(K-1)\kappa) \end{bmatrix}, \qquad \text{where } \kappa = \frac{1}{F_s}. \tag{7}$$

To calculate (6), note that the source's GPS values should be interpolated up to the acoustic sampling frequency, because the GPS points are usually reported less frequently (e.g., every other second). We use a constant velocity assumption to perform the GPS interpolation.

The data-likelihood (6) is maximized w.r.t. $\boldsymbol{\xi}$ to find the ML solution. However, the likelihood function is very complex due to the time-warping defined in (5). To simplify the solution, if an initial guess for the parameter vector $\boldsymbol{\xi}_0$ is available, then one can assume that $\tau_0(t) \approx t + ||\boldsymbol{\xi}_0 - \boldsymbol{\chi}_t||/c$. It is the author's observation that as long as $\boldsymbol{\xi}_0$ is within $c/4$ meters of the true position $\boldsymbol{\xi}$, the ML solution is robust. However, the DOA-solutions presented next section are sensitive to the time-warp function and require that the initial guess $\boldsymbol{\xi}_0$ be within $c/20$ meters of the true position. Once an estimate of $\boldsymbol{\xi}$ is available using $\boldsymbol{\xi}_0$, it can be further used to refine the time-warping function. Hence, this procedure is iterated to determine the node position estimates. In the rest of the paper, we ignore the time-warping issue, except when discussing the Metropolis-Hastings method that can automatically handle it.

To calibrate the array, we first obtain the negative log-likelihood function:

$$L^{-} \doteq KP\log(\pi\sigma^2) + \frac{1}{\sigma^2}\sum_{t=0}^{K-1} \|\mathbf{y}(t) - \mathbf{a}_t(\boldsymbol{\xi})s(t)\|^2 \tag{8}$$

where $\doteq$ denotes equality up to a constant, and $\mathbf{a}(t)$ is written as $\mathbf{a}_t(\boldsymbol{\xi})$ to emphasize the dependence on the parameter vector. Because of this dependence, the optimal solution for $\boldsymbol{\xi}$ (node-center position and orientation) is the one that best aligns the microphone output signals $\mathbf{y}(t)$ with shifted versions of the common source signal $s(t)$, given the source GPS track and the relative microphone positions.

The ML estimate (maximizing the log-likelihood function) is equivalent to the minimum of L^{-}. First of all, we fix $\boldsymbol{\xi}$ and $s(t)$ and minimize L^{-} with respect to σ^2 to find the ML noise variance. Next, the signal estimate is found by taking the variation of (8) with respect to $s(t)$, and setting it equal to zero. The results are:

$$\begin{aligned} \sigma^2_{ML} &= \frac{1}{KP}\sum_{t=0}^{K-1} \|\mathbf{y}(t) - \mathbf{a}_t(\boldsymbol{\xi})s(t)\|^2, \\ s_{ML}(t) &= \tfrac{1}{P}\,\mathbf{a}_t^H(\boldsymbol{\xi})\mathbf{y}(t). \end{aligned} \tag{9}$$

Substituting these ML estimates into L^{-} and noting that the log operator is monotonic, we can rewrite the fist term in (8) as an ML cost function that depends on $\boldsymbol{\xi}$ alone:

$$J(\boldsymbol{\xi}) = \sum_{t=0}^{K-1} \text{trace}\left\{\mathbf{P}_t\hat{\mathbf{R}}_t\right\} = \sum_{t=0}^{K-1} \text{trace}\left\{(\mathbf{I} - \tfrac{1}{P}\,\mathbf{a}_t(\boldsymbol{\xi})\mathbf{a}_t^H(\boldsymbol{\xi}))\hat{\mathbf{R}}_t\right\} = \sum_{t=0}^{K-1} J_t(\boldsymbol{\xi}), \tag{10}$$

where $\mathbf{P}_t = \mathbf{I} - \frac{1}{P}\,\mathbf{a}_t(\boldsymbol{\xi})\mathbf{a}_t^H(\boldsymbol{\xi})$ is the projection onto the null space of $\mathbf{a}_t^H(\boldsymbol{\xi})$, and $\hat{\mathbf{R}}_t = \mathbf{y}(t)\mathbf{y}^H(t)$ is the one-sample autocorrelation estimate. Finally, the ML estimate would be obtained by minimizing the cost function

$$\boldsymbol{\xi}_{ML} = \arg\min_{\boldsymbol{\xi}} J(\boldsymbol{\xi}). \tag{11}$$

B. Newton Search Algorithm

The solution to (11) requires the evaluation of the cost function $J(\boldsymbol{\xi})$ over the entire domain of the parameter vector $\boldsymbol{\xi}$ to obtain the global minimum, because $J(\boldsymbol{\xi})$ may have multiple local minima. However, this would be computationally expensive and unnecessary in most cases. If an initial estimate can be found that is reasonably close to the global minimum, then the cost function can be approximated via the expansion:

$$J(\boldsymbol{\xi} + \delta\boldsymbol{\xi}) \approx J(\boldsymbol{\xi}) + \mathbf{g}^T(\boldsymbol{\xi})\delta\boldsymbol{\xi} + \tfrac{1}{2}\,\delta\boldsymbol{\xi}^T\mathbf{H}(\boldsymbol{\xi})\delta\boldsymbol{\xi}, \tag{12}$$

where $\mathbf{g}$ and $\mathbf{H}$ are the gradient and the Hessian of $J(\boldsymbol{\xi})$, respectively. This leads to Newton's method, which is known to exhibit quadratic convergence when starting sufficiently close to the optimum point [10].

The necessary gradient of the cost function (10) can be calculated via the chain rule:

$$\mathbf{g} \triangleq \frac{\partial J(t)}{\partial \boldsymbol{\xi}} = \frac{1}{M}\sum_{t=0}^{M-1} V(t)\mathrm{diag}(\nabla_\theta J_t), \tag{13}$$

where

$$V(t) \triangleq \frac{\partial \theta_t}{\partial \boldsymbol{\xi}}, \tag{14}$$

and

$$\nabla_\theta J_t = -2\Re\left\{\mathbf{a}_t^\dagger \hat{\mathbf{R}}_t \mathbf{P}_t \frac{\partial \mathbf{a}_t}{\partial \theta_t}\right\}. \tag{15}$$

The Hessian $\mathbf{H} \triangleq \dfrac{\partial^2 J(t)}{\partial \boldsymbol{\xi}\partial \boldsymbol{\xi}^H}$ is then given by

$$\mathbf{H} = \frac{1}{M}\sum_{t=0}^{M-1} \left\{[\nabla^2_{\theta\theta} J_t \otimes 1] \odot [V(t)^H V(t)] + [\mathrm{diag}(\nabla_\theta J_t) \otimes I] \odot \Upsilon(t)\right\}, \tag{16}$$

where $\Upsilon(t)$ is the Hessian of θ_t with respect to $\boldsymbol{\xi}$, and $\mathbf{1}$ denotes a 3×3 matrix of all ones, and I is the identity matrix of the same dimensions. Symbols $\otimes$ and $\odot$ denote the Kronecker and Schur products, respectively. The operator diag($\mathbf{x}$) denotes a matrix whose diagonal elements consists of the vector $\mathbf{x}$. To guarantee the positive definiteness of $\mathbf{H}$, the term containing $\Upsilon(t)$

in (16) should be ignored while calculating the Hessian. The following derivative also needs to be approximated while calculating $\nabla^2_{\theta\theta} J_t$:

$$\frac{\partial^2 J_t}{\partial\theta(t)\partial\theta(t)} \simeq 2\Re\{\gamma^H(t)\gamma(t)\}, \tag{17}$$

where we define $\gamma(t) = \dfrac{\partial \mathbf{P}_t \mathbf{y}(t)}{\partial \theta_t}$.

TABLE I

NEWTON ALGORITHM WITH BACKTRACKING STEP SIZE SELECTION

```
At the $k^{\mathrm{th}}$ iteration of Newton-Raphson algorithm:
  • Calculate the descent direction $p_k = -\mathbf{H}_k^{-1}\mathbf{g}_k$. Then,
      – Set $\mu^0 = 1$;
      – While $J(\boldsymbol{\xi}_k + \mu^m p_k) > J(\boldsymbol{\xi}_k) + 10^{-4}\mu^m \mathbf{g}_k^T p_k$ (sufficient decrease condition), do
          ∗ Choose the contraction factor $\rho \in [0.1, 0.5]$,
          ∗ $\mu^m = \rho\mu^{m-1}$.
      – Set $\mu_k = \mu^m$.
  • $\boldsymbol{\xi}_k = \boldsymbol{\xi}_{k-1} + \mu_k p_k$.
  • Continue until the stopping condition.
```

Netwon's method is used to find the minimum of (11). Table I describes the algorithm with the variable step size selection modification [10], [11]. Lastly, it is also crucial for the Newton algorithm to use a good stopping condition for terminating the search. Among various choices outlined in [11], we use the relative change in the cost function to stop the algorithm. At this point, it is important to recall that the ML solution above does not include the effects of errors in the GPS track ($\boldsymbol{\chi}_t$) on the estimation performance of finding $\boldsymbol{\xi}$. This is addressed in the next subsection, where it is shown that such errors are usually negligible.

C. Effects of the GPS Errors on the Estimation Performance

In the 2D problem under consideration, the GPS outputs $\boldsymbol{\chi}_t$ have components in the x and y directions. If we model errors in the GPS estimates as zero mean $i.i.d.$ Gaussian noise, $\mathbf{n}_\chi \sim \mathcal{N}(\mathbf{0}, \sigma_\chi^2 \mathbf{I})$, then we can compare the relative effects of GPS noise and microphone noise on the final calibrated node-position estimate. In many practical scenarios, the size of the GPS errors is generally on the order of 1–5 meters, when differential GPS data is jointly used with the GPS system [19]. Rather than finding the variance of the final calibrated node-position estimates directly, we develop a formula for the array output $\mathbf{y}(t)$ that includes a perturbation due to GPS noise. Then, the size of this perturbation term is compared to the array noise to derive the condition, under which GPS noise is negligible.

The array output $\mathbf{y}(t)$ depends on the angle to the source θ_t, which, in turn, depends on the source location. With GPS noise, there is uncertainty in the source location, which translates into a perturbation of the estimated DOA angle and finally into additive noise in the array output. The first step is to find the effect of GPS noise on the auxiliary variable θ_t defined in (2). Expanding in a Taylor series, the first-order perturbation can be modelled as follows:

$$\theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t + \mathbf{n}_\chi(t)) \approx \theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t) + n_\chi^x(t) \frac{\partial \theta_t}{\partial n_\chi^x(t)}\bigg|_{n_\chi^x(t)=0} + n_\chi^y(t) \frac{\partial \theta_t}{\partial n_\chi^y(t)}\bigg|_{n_\chi^y(t)=0}. \tag{18}$$

By taking the derivatives of (2), and noting that the noise $\mathbf{n}_\chi(t)$ is independent in the x and y directions, and the second-order terms are very small, we can approximate the DOA θ_t with a Gaussian density:

$$\theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t + \mathbf{n}_\chi(t)) \sim \mathcal{N}\left(\theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t), \frac{1}{R_t^2}\sigma_\chi^2\right) \tag{19}$$

with the correct mean $\theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}_t)$. This leads to an intuitive result: when the GPS errors are very small compared to the range, the position errors will translate into an approximate angle error of $\tan^{-1}(\sigma_\chi / R_t) \approx \sigma_\chi / R_t$, which is tiny.

The next step is to derive the effect of GPS noise on the steering vector. Once again, a first-order approximation can be used:

$$\mathbf{a}(\boldsymbol{\xi}, \boldsymbol{\chi}_t + \mathbf{n}_\chi(t)) \approx \mathbf{a}(\boldsymbol{\xi}, \boldsymbol{\chi}_t) + \frac{\partial \mathbf{a}}{\partial \theta} n_\theta, \tag{20}$$

where the steering vector is as defined in (4). The derivative of (4) becomes

$$\frac{\partial \mathbf{a}}{\partial \theta} = \frac{j2\pi f_0}{c} \mathbf{a}(\boldsymbol{\xi}, \boldsymbol{\chi}_t) \boldsymbol{\lambda}(\theta),$$

where $\boldsymbol{\lambda}(\theta) = \mathrm{diag}\{[\rho_1 \cos(\theta + \phi_1), \ldots, \rho_P \cos(\theta + \phi_P)]\}$. If we define $\boldsymbol{\Lambda}(\theta) = \boldsymbol{\lambda}(\theta) \odot \boldsymbol{\lambda}(\theta)$, then the array outputs for a source signal with constant envelope magnitude of one can be shown to obey the following Gaussian distribution

$$\mathbf{y}(t) \sim \mathcal{N}\left(\mathbf{a}(\boldsymbol{\xi}, \boldsymbol{\chi}_t), \boldsymbol{\Sigma}(\boldsymbol{\xi}, \boldsymbol{\chi}_t)\right), \tag{21}$$

where the autocorrelation matrix $\boldsymbol{\Sigma}$ is a function of the array noise as well as the GPS noise:

$$\boldsymbol{\Sigma}(\boldsymbol{\xi}, \boldsymbol{\chi}_t) = \sigma^2 \mathbf{I} + \left(\frac{2\pi f_0 \sigma_\chi}{c R_t}\right)^2 \mathbf{a}(\theta_t) \boldsymbol{\Lambda}(\theta_t) \mathbf{a}^H(\theta_t). \tag{22}$$

The second term in (22) is the perturbation due to GPS errors. If $\sigma \gg \dfrac{2\pi f_0 \sigma_\chi}{c R_t}$, then it can be argued that GPS errors have a very small impact on the estimation performance because the data likelihood (6) is not affected. For most cases of interest in our work, this is reasonable because the narrow-band frequencies of the source are usually less than 100 Hz, the GPS error standard deviation σ_χ is on the order of a few meters, while the range R_t is a kilometer or more.

D. Cramér-Rao Lower Bound for the Estimate of $\boldsymbol{\xi}$

The Cramér-Rao lower bound (CRLB) is an information theoretic inequality that provides a lower bound for the variances of the unbiased estimators. If an estimator achieves the CRLB, then it also maximizes the likelihood equation. However, it is not always true that the ML solution achieves the CRLB (at least, for finite sample sizes) or that it will be unbiased [13]. The CRLB is still a useful metric with which to compare the performance of the algorithm, and is derived for the calibration problem in this section.

First, we derive an expression for the Fisher information matrix (FIM). Assume that the noise variance σ^2 is known. Ignoring the time-warping issue, the log-likelihood function (6) for the parameter vector $\boldsymbol{\xi}$ simplifies to the following relation:

$$L(\boldsymbol{\xi}) \doteq -\frac{1}{\sigma^2} \sum_{t=0}^{K-1} \|\mathbf{y}(t) - \mathbf{a}_t s(t)\|^2 \tag{23}$$

where $\mathbf{a}_t = \mathbf{a}_t(\theta_t(\boldsymbol{\xi}, \boldsymbol{\chi}))$ is the steering vector from the node position to the calibration source. The $(i,j)^{\text{th}}$ element of the FIM is given by partial derivatives of (23) with respect to the i^{th} and j^{th} parameters of the vector $\boldsymbol{\xi}$

$$F_{i,j} = E_{\mathbf{y}}\left\{\frac{\partial^2 L_\chi(\boldsymbol{\xi})}{\partial \boldsymbol{\xi}_i \partial \boldsymbol{\xi}_j}\right\} = -\frac{2}{\sigma^2} \sum_t \Re\left\{\left(\frac{\partial \mathbf{a}_t}{\partial \boldsymbol{\xi}_i}\right)^H \frac{\partial \mathbf{a}_t}{\partial \boldsymbol{\xi}_j}\right\} \tag{24}$$

where $\boldsymbol{\xi}_i = [\boldsymbol{\xi}]_i$ and $E_{\mathbf{y}}\{\cdot\}$ denotes the expected value with respect to the data distribution (see [4] for a similar derivation). The Cramér-Rao lower bound is the inverse of this expression [13]. Interestingly, the CRLB depends on the source's narrow-band frequency: the higher the frequency, the lower the localization bound. In the simulations section, Fig. 6 shows this dependence for a specific scenario.

III. DOA-Based Calibration Algorithms

In the previous section, the calibration problem was introduced and the maximum-likelihood solution presented. The remainder of the paper treats another set of methods called *DOA-Based calibration algorithms* that exploit the geometry of the problem defined by the GPS (carried by the source) along with estimated DOAs (at the node). These methods rely on the fact that the DOA is a sufficient statistic, from which it is possible to determine the node-center position and orientation. We first explain the angle matching idea used by the DOA-based calibration methods. Then, two DOA-based calibration methods are studied: the first uses the synthetic aperture concept from radar, and the second, a Metropolis-Hastings type of Monté-Carlo algorithm. For both DOA calibration algorithms, performance bounds are derived and examples are included.

A. Angle Matching

The DOA-based calibration algorithms use a simple angle matching idea derived from the geometry of the problem, illustrated in Fig. 2. As the source moves, the node estimates a DOA track $\theta(t)$ with respect to its orientation by using the received acoustic data, independent of the node position. The node can also calculate a node-to-GPS angle track $\psi(t)$ as a function of the node position (x, y) and the GPS track $(x_T(t), y_T(t))$ transmitted by the source (Fig. 2):

$$\psi(x, y, t) = \tan^{-1}\left(\frac{y - y_T(t)}{x - x_T(t)}\right) = \theta(t) + \varphi. \tag{25}$$

Then, by assuming that the DOA estimation errors are zero mean Gaussian random variables, a maximum-likelihood solution can be found for $\boldsymbol{\xi}$ by minimizing the following cost function:

$$\boldsymbol{\xi}_{ML} = \arg\min_{\boldsymbol{\xi}} \sum_t \frac{(\psi(x, y, t) - \theta(t) - \varphi)^2}{\sigma^2_{\theta_t}}, \tag{26}$$

where $\sigma^2_{\theta_t}$ is the DOA estimation variance, which is discussed later in this section. Another Newton-type algorithm can be formulated for the solution of the above equation. This solution will not be discussed here, because the algorithm is similar to the ML solution considered in the previous section.

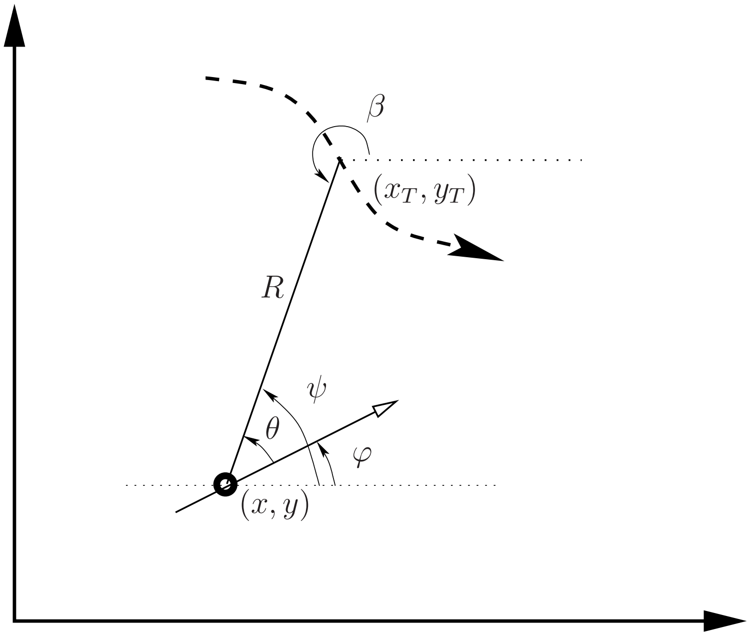


Fig. 2. The node position is shown with a circle at the position (x, y). As the calibration source moves (dashed line), the node can calculate two angles $\theta(t)$ and $\psi(t)$ that should be matched to determine the unknown parameter vector $\boldsymbol{\xi}$. The parameter β is called the synthetic aperture radar (SAR) angle to be explained in the next section.

In reality, the above solution calibrates the array only moderately well, because it does not consider the acoustic data propagation time from the calibration source to the node. The angle matching equation (25) as well its ML solution should be modified using the time-warp relation in (5):

$$\psi(x, y, \tau(t)) = \theta(t) + \varphi, \tag{27}$$

$$\boldsymbol{\xi}_{ML} = \arg\min_{\boldsymbol{\xi}} \sum_t \frac{(\psi(x, y, \tau(t)) - \theta(t) - \varphi)^2}{\sigma_{\theta_t}^2}. \tag{28}$$

The Metropolis-Hastings calibration method uses a data-likelihood function based on this cost function.

B. Synthetic Aperture Calibration Method

The synthetic aperture concept is the idea of creating a large aperture size from a small moving physical aperture to obtain better angular resolution. In the node calibration problem, if the problem is reformulated by time-reversing the events such that the fixed node is considered to be the sound source and the moving calibration source the receiver(s), then it is possible to apply the synthetic aperture concept. That is, we create a moving pseudo-receiver defined by the calibration source's GPS track and assume that the signals are coming from the acoustic node (Fig. 3). The moving pseudo-receiver can be grouped into synthetic subarrays, from which we can estimate the DOAs to the fixed node, and then calibrate the node position by doing triangulation.

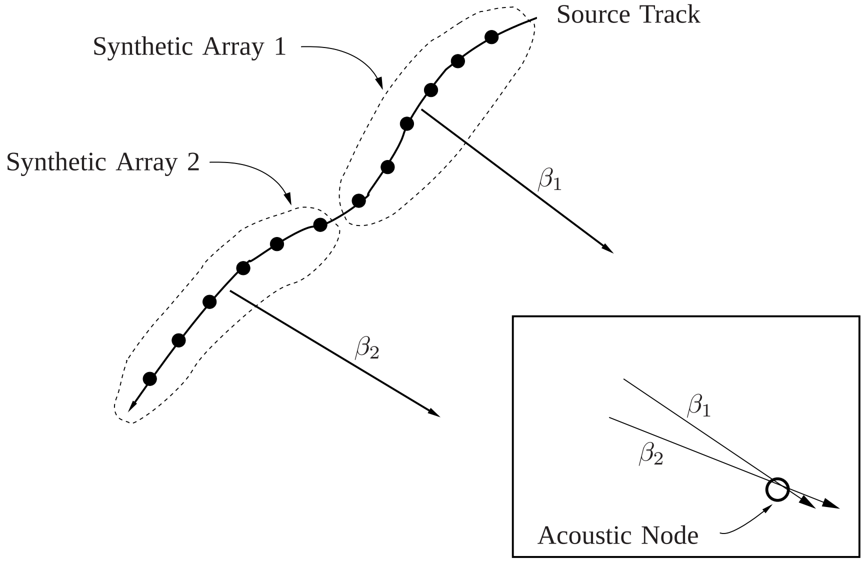


Fig. 3. The moving calibration source can be interpreted as a moving pseudo-receiver that creates a synthetic aperture. Pseudo-receiver positions can then be grouped into subarrays and used to estimate DOAs assuming the signals are coming from the fixed sensor node. This does not require additional transmission of the recorded sound data from the node, since this estimation can be done at the node.

A single *synthetic receiver* position consists of a fixed number of signal samples M and is assigned a fixed position, even though the *real receiver* is actually moving, while the M samples are acquired. The number of samples M determines the intersensor spacing for the synthetic array, so spatial aliasing of the acoustic signals must be taken into account. The distance traveled by the calibration source moving at a velocity v during M samples (sampled at F_s) is Mv/F_s. In conventional array processing, it is well-known that if the sensor spacing is less than half the wavelength of the signal of interest, then spatial aliasing can be avoided [9]. Hence, an upper bound on M is:

$$M < \frac{cF_s}{2f_0 v} \tag{29}$$

where f_0 is the narrow-band center frequency, and c is the speed of sound. For reasonable beamforming, at least one cycle of the signal must be observed; hence, $M > f_0$. As a rule of thumb, we recommend using the largest M that satisfies (29).

The total number of synthetic receiver positions used in forming a subarray (Q)is subject to the following trade-off: longer subarrays would give better DOAs, but provide fewer DOAs for triangulation. Surprisingly, it appears that the node position estimation accuracy is not affected much by this trade-off when tested on synthetic data.

We now describe the details of the synthetic aperture calibration method. The acoustic data recorded at each microphone of the acoustic node is temporally partitioned into M-sample data sets, and the midpoint time is used to define the position for each synthetic receiver along the path of the moving calibration source, $\boldsymbol{\chi}_t$. Two issues arise from this definition of the synthetic receiver. First, each sample in the block of M data samples comes from the neighborhood of the defined receiver position, through which the source was moving. We can model this as a non-stationary effect in the received data. Second, since the GPS positions are usually supplied at a much lower rate than the acoustic data sampling rate, the synthetic receiver positions must be estimated by using a straight-line interpolation between the given GPS data points. For example, if $F_s = 1024$ Hz, and $M = 64$, then the synthetic receiver positions must be determined 16 times per second.

A subarray is formed by grouping together Q synthetic receiver positions. The intrasensor spacing is Mv/F_s meters, assuming v is constant. Simple beamforming such as MUSIC (Multiple Signal Classification) or MVDR (Minimum Variance Distortionless Response) results in biased DOA estimates if the calibration source is moving fast (e.g., a helicopter). This bias is caused by the non-stationarity of the synthetic array data mentioned above, but previous work in [1] has addressed this same issue and shown how to calculate the bias values theoretically. Therefore, we propose the following ML cost function, derived similarly to (10), for obtaining unbiased synthetic DOA estimates β:

$$J(\beta) = \sum_{t=1}^{M} \operatorname{tr}\left\{ \left[\mathbf{I} - \frac{1}{Q}\mathbf{a}_t(\beta)\mathbf{a}_t^H(\beta) \right] \hat{\mathbf{R}}_z(t) \right\}, \tag{30}$$

where $\mathbf{a}_t(\beta)$ is the steering vector corresponding to aperture points and β is defined as the synthetic aperture radar angle (Fig. 2). The time dependence in $\mathbf{a}_t(\beta)$ is caused by the data samples at each synthetic sensor being from a neighborhood of the synthetic sensor position. The angle β is calculated with respect to the center of the synthetic aperture of size MQ samples, and $\hat{\mathbf{R}}_z(t)$ is the one-sample (outer product) autocorrelation estimate of the synthetic array. Equation (30) differs from (10) in the time dependence of the steering vectors. In (10), the microphone positions are fixed with respect to t; whereas, in (30), the SAR angle β is fixed, and the position where the data is collected is changing along the aperture.

Equation (30) can be used to determine the position of each individual microphone within the node. Since there are P microphones in the node, it is possible to obtain P independent estimates of the node position using the individual microphone outputs, assuming that the additive noise is spatially white at the acoustic node. Then, the P estimated microphone positions can be averaged to obtain an estimate of the node-center position. The node orientation would then be estimated from the acoustic node DOA estimates along with the node-center position estimate. Once again, it is important to recognize the array non-stationarity issue caused by rapid source movement. The ML cost function to be minimized at the acoustic node in this case is

$$J(\theta) = \sum_{t=1}^{M} \mathrm{tr}\left\{ \left[\mathbf{I} - \frac{1}{P}\mathbf{a}(\theta_t)\mathbf{a}^H(\theta_t) \right] \hat{\mathbf{R}}_y(t) \right\}, \tag{31}$$

where

$$\theta_t = \tan^{-1}\left\{ \frac{F_s \sin\theta + qt\cos\phi}{F_s\cos\theta + qt\sin\phi} \right\}, \tag{32}$$

where θ is the DOA at the beginning of the batch, $q = v/R$, and ϕ is the approximate source heading during the estimation batch. Equation (32) can be derived by a straight-line approximation for the calibration source during a batch period. Given the node-center position estimate, if the DOAs are estimated using (31), one can estimate the node orientation by simply calculating the bearings from the node position and GPS estimates, and then finding the difference between the mean values of these estimates.

When (30) is used for the SAR angles, the cost function will display two minima corresponding to two different candidate SAR angles. This is attributed to the cone of ambiguity problem for microphone arrays and is applicable to our problem due to the constant velocity interpolation of the GPS points. Even when there is no spatial aliasing, the estimator (30) results in two DOA estimates that are symmetric with respect to the array axis (calibration source orientation angle)[1]. This issue has been addressed for uniform linear arrays in [9]. The solution to this problem is to also track the DOA estimates and impose a constraint that the DOA estimates not change too much from one aperture to another. This fact also helps reduce the computational load while finding the minimum of the cost function in (31), since previous estimates are usually close to the sought minima. In turn, two node positions can be triangulated using the two DOA tracks, one corresponding to the node and the other corresponding to a shadow. The real node can easily be determined by picking the node position with the least minimum-mean squared error, since it actually corresponds to a physical position.

Once the SAR angles (β_i's $i = 1, \ldots, L$) are calculated for each synthetic array with respect to the array centers $(x_T^{(i)}, y_T^{(i)})$, determination of the node position requires one more step. Figure 3 suggests an intuitive solution based on determining the intersection of the lines created by the aperture positions and the SAR angles. This leads to the following overdetermined system of equations to determine the node position:

$$\begin{bmatrix} \sin(\pi - \beta_1) & -\cos(\pi - \beta_1) \\ \vdots & \vdots \\ \sin(\pi - \beta_L) & -\cos(\pi - \beta_L) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \sin(\pi - \beta_1) x_T^{(1)} - y_T^{(1)} \cos(\pi - \beta_1) \\ \vdots \\ \sin(\pi - \beta_L) x_T^{(L)} - y_T^{(L)} \cos(\pi - \beta_L) \end{bmatrix} \tag{33}$$

This equation is based on the following geometrical relationship between the node DOA θ_i, its orientation φ, and the SAR angle β_i, as illustrated by Fig. 2:

$$\beta_i = \pm\pi + \theta_i + \varphi = \pm\pi + \tan^{-1}\left(\frac{y - y_T^{(i)}}{x - x_T^{(i)}}\right) \tag{34}$$

Equation (34) defines an under-determined system for $\boldsymbol{\xi}$ for one index i, hence making it analytically intractable to derive a joint probability density function $p(x, y)$, since the required Jacobians for the coordinate transformation from β to (x, y) are not defined[2]. Equation (33) provides the least-squares solution for the node position. However, the estimated DOAs become approximately Gaussian distributed as the sampling frequency of the data increases. Based on this observation, similar to the ML solution using the node-to-GPS angles ψ shown in the previous

[1] The average of these two DOA candidates can be used to check the consistency of the source heading angle.

[2] That is because the coordinate transformation defined by (34) is not one-to-one.

section, we propose the following

$$\begin{aligned} \boldsymbol{\xi}_{x,y} &= \arg\min_{\boldsymbol{\xi}_{x,y}} J \\ J &= \sum_{i=1}^{L} \frac{\left(\beta_i - \tan^{-1}\left(\frac{y-y_T^{(i)}}{x-x_T^{(i)}}\right) \pm \pi\right)^2}{\sigma_{\beta_i}^2}, \end{aligned} \tag{35}$$

where the sign in front of π must be determined from the geometry and the SAR angle error variance $\sigma_{\beta_i}^2$ should be calculated from the data. A possible expression will be given in the later sections. This estimator can be refined by weighting the terms in the summation by the estimated noise variances per index. The following sections give analytical equations for these variance estimates. Equations (33) (SAR-LS solution) and (35) (SAR-ML solution) are biased estimators (shown by simulations); however, the bias is difficult to analyze.

A Newton based search can again be employed to solve (35). The Newton algorithm is useful for estimating the angles, since the SAR angles β_i are very close to each other. Hence, the previous estimate β_{i-1} can be refined using the Newton recursion to estimate β_i. This is conditioned on the fact that the calibration source is not maneuvering too much, because the Newton search can become trapped at a local minimum. We recommend calculating J in (35) first on a rough grid near to β_{i-1} (e.g., a grid of angles $\beta \in [\beta_{i-1} - 15°, \beta_{i-1} + 15°]$ with $1°$ spacing), then using the minima over that rough grid and the Newton algorithm to refine the estimate β_i.

C. Metropolis-Hastings Calibration Method

The objective of the Metropolis-Hastings (MH) algorithm is to distribute particles (discrete state samples $\boldsymbol{\xi}_i$) according to a target distribution $\pi(\cdot)$. Hence, at each iteration k, the algorithm recursively redistributes its states so that, asymptotically, the resulting Markov chain is distributed according to the target distribution. The MH scheme [15] is depicted in Fig.4. In the figure, the Markov chain at iteration k is represented by $\boldsymbol{\xi}^{(k)}$. The new chain candidates $\boldsymbol{\gamma}$ are generated by the proposal function $q(\boldsymbol{\xi}, \boldsymbol{\gamma})$, which is usually the spherically symmetric random walk:

$$q(\boldsymbol{\xi}, \boldsymbol{\gamma}) = q(|\boldsymbol{\xi} - \boldsymbol{\gamma}|) \propto \exp\left\{-\frac{(\boldsymbol{\xi} - \boldsymbol{\gamma})^2}{2\sigma_q^2}\right\} \tag{36}$$

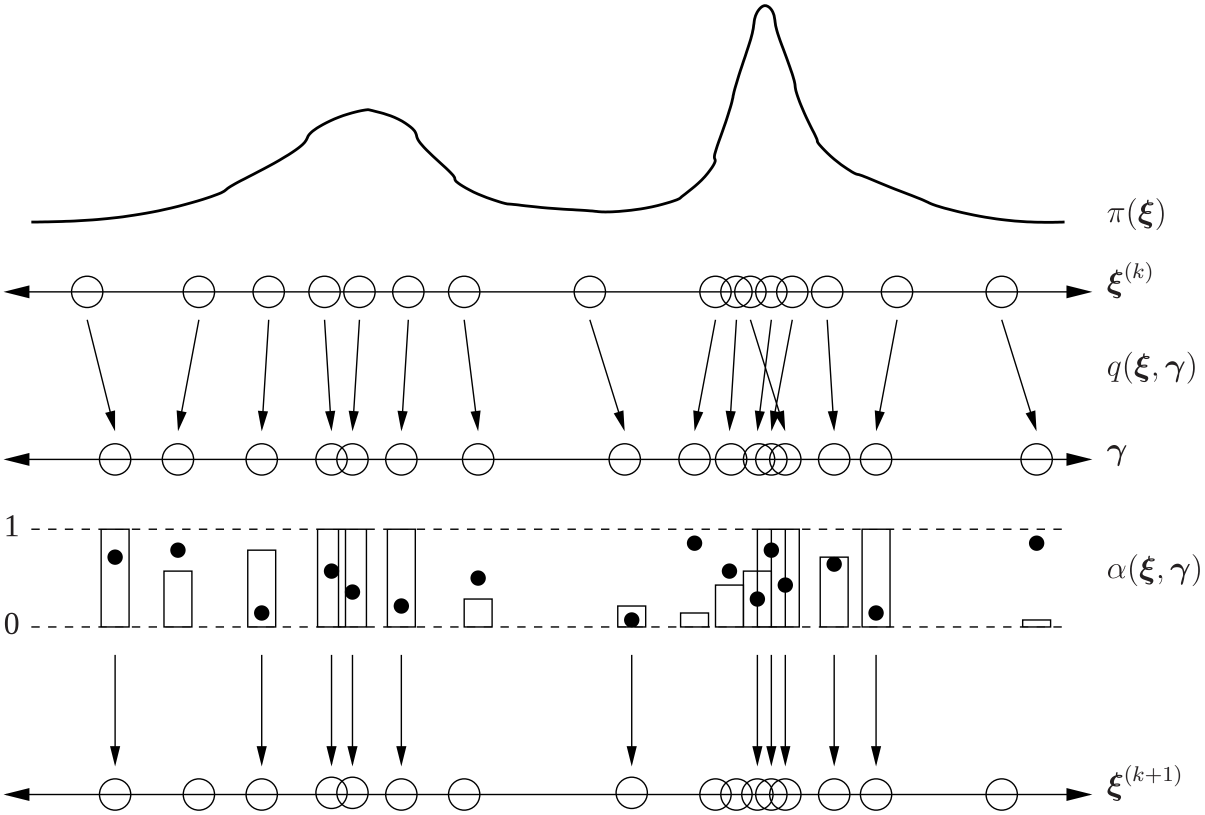

$\pi(\boldsymbol{\xi})$
$\boldsymbol{\xi}^{(k)}$
$q(\boldsymbol{\xi},\boldsymbol{\gamma})$
$\boldsymbol{\gamma}$
1
0
$\alpha(\boldsymbol{\xi},\boldsymbol{\gamma})$
$\boldsymbol{\xi}^{(k+1)}$

Fig. 4. The Metropolis-Hastings scheme is demonstrated. Each circle represents a sample from the chain in the respective state space. The algorithm uses its current state to generate new candidates for its next state using a candidate generating function q. The new candidates are accepted or rejected in a way that the Markov chain asymptotically converges to the target posterior π.

Once the new candidates are generated, the algorithm accepts the moves or keeps the current state according to the acceptance ratio $\alpha(\boldsymbol{\xi},\boldsymbol{\gamma})$ derived from the stochastic reversibility condition [15]:

$$\alpha(\boldsymbol{\xi},\boldsymbol{\gamma}) = \min\left\{\frac{\pi(\boldsymbol{\gamma})q(\boldsymbol{\gamma},\boldsymbol{\xi})}{\pi(\boldsymbol{\xi})q(\boldsymbol{\xi},\boldsymbol{\gamma})}, 1\right\} \tag{37}$$

In Fig. 4, the acceptance ratios are represented by the height of the boxes for each candidate. To accept or reject the new candidate, a random number generator is used to generate uniform random numbers in $(0,1)$, $u \sim \mathcal{U}(0,1)$, represented by the black dots in Fig. 4. If u is less than the acceptance ratio for the specific particle, the move is accepted, otherwise, it is rejected. Also, the *acceptance rate* at iteration k is defined to be the number of accepted moves divided by the chain size. Visually, it is the number of arrows in the last stage in Fig. 4 divided by the number of particles. Finally, the chain moves to $\boldsymbol{\xi}^{(k+1)}$ and the scheme is repeated.

For the calibration problem, the target function is actually the exponential of the negative ML cost function defined earlier in (27):

$$\pi(\boldsymbol{\xi}) \propto \exp\left\{-\sum_t \frac{(\psi(x,y,\tau(t))-\theta(t)-\varphi)^2}{2\sigma_{\theta_t}^2}\right\}, \tag{38}$$

where ψ is the node-to-GPS angle, θ_t is the DOA estimate from the acoustic data using (31), and φ is the node orientation angle. The time-synchronization is easily incorporated into this solution because it is possible to calculate $\tau(t)$ given a proposed node position and the GPS of the calibration source.

Moreover, it is easy to see that (38) peaks at the correct node position when there is no noise. Intuitively, this is the complement of the synthetic aperture solution, where the angles are calculated on the node side, however it performs more robustly than the estimator proposed in (35). As for the candidate generating function q, we employ the symmetric random walk, where the walk variances should be picked subjectively. For example, a few meters works well for the walk variances on $\boldsymbol{\xi}_{x,y}$, and we use a few degrees for the orientation φ.

The MH scheme, as presented here, takes a notoriously long time to converge. Hence, it is necessary to speed it up for a real-time application. Pseudo code is given in Table II for the classical MH algorithm for the calibration problem. In Table III, we outline the generic Mode-Hungry MH algorithm for accelerated convergence of the chain $\boldsymbol{\xi}_i$. The initialization is usually the uninformed uniform density in the calibration space. A grid based initialization of this algorithm is also possible [20], when prior information is available.

D. Performance of the DOA Calibration Algorithms

The DOA calibration algorithms use a sequence of estimated DOAs corresponding to a batch size of M samples, together with geometrical arguments, to estimate $\boldsymbol{\xi}$. In other words, the DOA at index i is estimated using M array samples corresponding to the i^{th} synthetic receiver, sampled at F_s. To evaluate the performance of the DOA calibration algorithms, the DOA estimation performance should first be related to the node array signal-to-noise (*SNR*) ratio. Note that the DOAs need to be estimated using (31) if the calibration source is moving relatively fast. Classical DOA estimators such as MUSIC and MVDR result in biased estimates due to the non-stationarity of the data caused by rapid motion of the calibration source [1], [3]. Using the likelihood function in (23), we write the Fisher information for the DOA at the beginning of the

TABLE II

PSEUDO CODE FOR THE MH CALIBRATION

```
• At time $k$, for each particle $i$ ($i = 1, \dots, N$) , $\boldsymbol{\xi}_i^{(k)}$:
    i. generate a candidate $\boldsymbol{\gamma}_i$ using a candidate generating function $q(\boldsymbol{\xi}_i, \boldsymbol{\gamma}_i)$, which is usually a random walk
   ii. estimate the time-reference frame for data synchronization using the proposed position, the source GPS
       track, and the speed of sound $c$
  iii. calculate the DOAs, $\theta(t)$, using the motion compensated beamformer (31)
   iv. calculate the acceptance ratio, where the target distribution $\pi(\cdot)$ is as given in (38)

           $\alpha(\boldsymbol{\xi}_i, \boldsymbol{\gamma}_i) = \min\left(\frac{\pi(\gamma_i)}{\pi(\boldsymbol{\xi}_i)}, 1\right)$

    v. sample $u \sim \mathcal{U}(0,1)$
   vi. if $u \le \alpha(\boldsymbol{\xi}_i, \boldsymbol{\gamma}_i)$, set $\boldsymbol{\xi}_i^{(k+1)} = \boldsymbol{\gamma}_i$, else, $\boldsymbol{\xi}_i^{(k+1)} = \boldsymbol{\xi}_i^{(k)}$.
• Repeat until convergence is detected.
```

batch as

$$\begin{aligned} F_{\theta_1} &= E\left\{\left(\frac{\partial L(\theta_1)}{\partial \theta_1}\right)^2\right\} \\ &= -\frac{2}{\sigma_a^2}\sum_{t=1}^{M}\Re\left[\left(\frac{\partial \mathbf{a}_t}{\partial \theta_t}\frac{\partial \theta_t}{\partial \theta_1}\right)^H\left(\frac{\partial \mathbf{a}_t}{\partial \theta_t}\frac{\partial \theta_t}{\partial \theta_1}\right)\right], \end{aligned} \tag{39}$$

where $\frac{\partial \theta_t}{\partial \theta_1}$ is calculated from (32). The inverse of F_θ bounds the best achievable performance by an unbiased estimator, but in most cases, DOA estimation performance will be close to this bound for large M. Hence, it is reasonable to approximate the noise variance on the estimated DOAs as

$$\sigma_\theta^2 \approx -\frac{\sigma_a^2}{2}\left\{\sum_{t=1}^{M}\Re\left[\left(\frac{\partial \mathbf{a}_t}{\partial \theta_t}\frac{\partial \theta_t}{\partial \theta_1}\right)^H\left(\frac{\partial \mathbf{a}_t}{\partial \theta_t}\frac{\partial \theta_t}{\partial \theta_1}\right)\right]\right\}^{-1}. \tag{40}$$

This estimated noise variance (40) can be also used in the solution (35) to weight the SAR angles. Similarly, most beamformers can calculate their estimation accuracy by checking to see how well the observations conform to the underlying data model. The curvature of the beamformer output at the peak location can be converted into an angle variance estimate. In this case, the array noise can be calculated in various ways, for example, using eigenvalue analysis on the array auto-covariance matrix [9], [18]. Another measurement noise model for narrow-band target signals

TABLE III
MHMH SAMPLING ALGORITHM

- For each particle $\boldsymbol{\xi}_i$, $i = 1, 2, \dots, N_{\mathrm{MHMH}}$, use the Mode-Hungry scheme every $L_{jump} = 2$ iterations:
  - determine a subpartition of size $M_l < N_{\mathrm{MHMH}}$, (e.g.,$M_l = N_{\mathrm{MHMH}}/2$),
  - order the current particles according to their probabilities in descending order: $\boldsymbol{\xi}_i \rightarrow \boldsymbol{\xi}_j^*$, where $\boldsymbol{\xi}^*$ is the ordered particle set,
  - generate candidates $y^*(1)$ for $\boldsymbol{\xi}^*(1) = \{\boldsymbol{\xi}_j^* | j : j = 1, 2, \dots, N_{\mathrm{MHMH}} - M_l\}$, using $q(\cdot, \cdot)$,
  - calculate the acceptance ratio $\alpha(\boldsymbol{\xi}^*(1), y^*(1))$, and set $\boldsymbol{\xi}_j^{(l+1)}$ to $\boldsymbol{\xi}_j^*(1)$ or $y_j^*(1)$, accordingly for $j = 1, 2, \dots, N_{\mathrm{MHMH}} - M_l$,
  - distribute M_l candidates $y^*(2)$ from $\boldsymbol{\xi}^*(1)$ uniformly,
  - set $\boldsymbol{\xi}_j^{(l+1)}$ to $y^*(2)$ for $j = N_{\mathrm{MHMH}} - M_l + 1, \dots, N_{\mathrm{MHMH}}$.

can be determined from the following relation [21]:

$$\sigma_\theta = \frac{\lambda_0}{AC\sqrt{2\mathit{SNR}}}, \tag{41}$$

where λ_0 is the monotone signal wavelength, *SNR* is the signal-to-noise ratio, A is the aperture size of the array, and C is a constant depending on circular or plane-wave illumination of the aperture. As a rule of thumb, one can use $\sigma_{\theta_t}^2 \propto R_t \sigma_\theta^2 \Big/ \sum_t R_t$ for the field examples, where σ_θ^2 is proportional to the largest expected DOA error.

Now, if we assume the DOA noise is independent from batch to batch and is Gaussian with zero mean and with variance of (40), then the DOA likelihood from (38) becomes

$$L(\theta|\boldsymbol{\xi}) \doteq -\frac{1}{2}\sum_{i=1}^{L}\frac{1}{\sigma_{\theta_i}^2}\left\{\theta_i + \varphi - \tan^{-1}\left(\frac{y - y_T^{(i)}}{x - x_T^{(i)}}\right)\right\}^2, \tag{42}$$

with $\theta = [\ \theta_1\ ,\ \dots\ ,\ \theta_i\ ,\ \dots\ ,\ \theta_L\ ]$, where i is the estimation batch index and $\sigma_{\theta_i}^2$ is estimated using (40). Then, the estimation bound on the parameter $\boldsymbol{\xi}$ becomes

$$\mathit{CRB}_{\boldsymbol{\xi}} = \left(\sum_{i=1}^{L}\boldsymbol{\gamma}_i\boldsymbol{\gamma}_i^T\right)^{-1} \tag{43}$$

where

$$\boldsymbol{\gamma}_i = -\frac{1}{\sigma_{\theta_i}^2}\left[\ 1,\ \ \frac{y - y_T^{(i)}}{R_i^2},\ \ -\frac{x - x_T^{(i)}}{R_i^2}\ \right]^T \tag{44}$$

In the simulations section, we show that the above CRLB (43) is quite close to the bound that is calculated directly from the data likelihood. This, in turn, raises the question of the sufficiency of the DOA estimates in the calibration problem, which is covered in the next subsection.

E. Sufficiency of the Auxiliary DOA Estimates in the Calibration Problem

The parameter vector $\boldsymbol{\xi}$ affects the distribution of the observations $\mathbf{Y}_M$ defined by (7) through the distribution $p(\mathbf{Y}_M|\boldsymbol{\xi}, \boldsymbol{\chi}_0, \ldots, \boldsymbol{\chi}_t)$ given in (6). Therefore, in the calibration problem, the statistical behavior of the acoustic data constitutes the only information about the parameter $\boldsymbol{\xi}$ when there is a lack of a prior distribution on $\boldsymbol{\xi}$. Hence, if knowing β_i removes any dependence on $\boldsymbol{\xi}$ from the data distribution (6), then it can be said that β_i contains all the relevant information in the data needed to estimate the parameter $\boldsymbol{\xi}$ [13]. Hence, the objective is to prove the following equality:

$$p(\mathbf{Y}_M|\boldsymbol{\xi}, \beta_0, \ldots, \beta_L, \boldsymbol{\chi}_0, \ldots, \boldsymbol{\chi}_t) = p(\mathbf{Y}_M|\beta_0, \ldots, \beta_L, \boldsymbol{\chi}_0, \ldots, \boldsymbol{\chi}_t) \tag{45}$$

At this point, some assumptions should be reiterated. The SAR angles β_i are only given once per block of MQ data samples (a total of L blocks), whereas the data likelihood depends on the parameter at every sample. This is not a problem if the assumption of straight-line motion (32) is true between GPS data points. Thus, the data likelihood can be written as

$$\begin{aligned} &p(\mathbf{Y}_M|\boldsymbol{\xi}, \beta_0, \ldots, \beta_L, \boldsymbol{\chi}_0, \ldots, \boldsymbol{\chi}_t) = \\ &\quad \prod_{t=0}^{K-1} \frac{1}{\pi^P \sigma^{2P}} \exp\left[-\frac{1}{\sigma^2}\left\|\mathbf{y}(t) - \mathbf{a}(\pi + \beta^{(t)} - \varphi)s(t)\right\|^2\right], \end{aligned} \tag{46}$$

where

$$\begin{aligned} &\beta^{(t)} + \pi - \varphi = \tan^{-1}\left\{\frac{F_s \sin\theta_{t-1} + q^*(t-1)\cos\phi^*}{F_s \cos\theta_{t-1} + q^*(t-1)\sin\phi^*}\right\}, \\ &\text{with} \quad \theta_{t-1} = \beta^{(t-1)} + \pi - \varphi. \end{aligned} \tag{47}$$

Parameter ϕ^* is the approximate heading direction of the calibration source at time t and is calculated from the GPS data. Parameter q^* is the only parameter that depends on the node position (x, y) and can be approximated by

$$q^* \approx \left|\frac{F_s \sin(\beta_i - \beta_{i-1})}{(MQ-1)\sin(\beta_t - \phi^*)}\right|. \tag{48}$$

Hence, given the synthetic DOAs β_i, it is possible to remove the dependence of the data on the node position, but not on the node orientation. That is,

$$p(\mathbf{Y}_M|\boldsymbol{\xi}, \beta_0, \ldots, \beta_L, \boldsymbol{\chi}_0, \ldots, \boldsymbol{\chi}_t) = p(\mathbf{Y}_M|\varphi, \beta_0, \ldots, \beta_L, \boldsymbol{\chi}_0, \ldots, \boldsymbol{\chi}_t) \tag{49}$$

Equation (49) implies that the estimated synthetic DOAs approximately form sufficient statistics for the acoustic node position. This is quite intuitive because two anchor points and two DOAs is sufficient to triangulate the node position. In addition, to remove the dependence on the orientation angle φ, at least one local estimate of θ_t is required from the node.

IV. Simulations

Using computer simulations, our objectives are to (i) show the effects of the GPS errors on the estimation performance of the various algorithms, and (ii) demonstrate the ML and DOA calibration algorithms and compare their performance. The synthetic simulation examples use two calibration source tracks. The first track has a calibration source circling the origin at a range of 600m with a constant speed of 75 mph. The second calibration source starts at $(2000, 500)$m and moves in the negative x-direction with the same speed. The acoustic node is a uniform circular array with $P = 6$ omnidirectional microphones with a radius of 1.22m. The total estimation time for the calibration is 120s and the GPS error standard deviation is $\sigma_\chi = 1$m, sampled at 1Hz.

A. Effects of the GPS Errors on the Estimation Performance

Figure 5 demonstrates that the effects of the GPS errors on the likelihood surface is negligible. Moreover, the estimated positions, with and without GPS errors, differ by less than a centimeter. The smooth convex shape of the surface justifies the argument that a Newton or gradient-descent type of search algorithm can be used to estimate the node parameters, instead of calculating the whole surface. In Fig. 5, we try to determine the position of the acoustic node placed at $[\,100,\,50\,]$m. The target narrow-band frequency used for the calibration is $f_0 = 20$ Hz, the node sampling rate is $F_s = 128$ Hz, and the acoustic node *SNR* is 7 dB. The *SNR* is defined as

$$SNR = 10 \log \left(\frac{1}{\sqrt{2}\sigma_a^2} \right). \tag{50}$$

The ellipse in Fig. 5 represents the CRLB. The ellipse axes are determined by the square root of the diagonal elements of the CRLB. The ellipse orientation is determined by the correlation

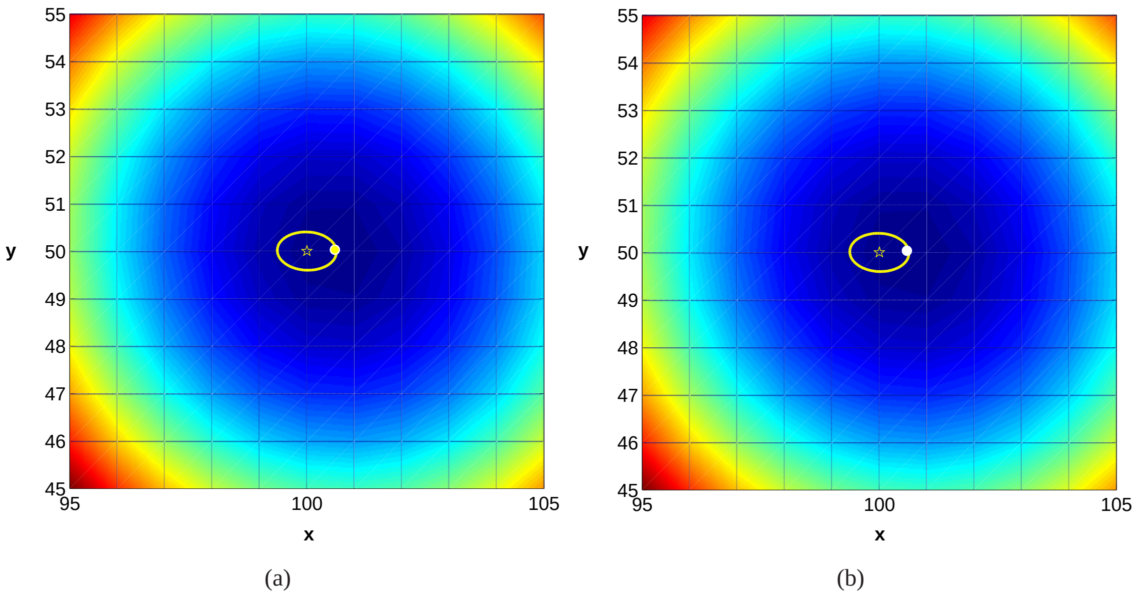


Fig. 5. (a). The likelihood surface J with GPS errors. The GPS error standard deviation is $\sigma_\chi = 1$m and it is circularly symmetric in the x and y directions. The star indicates the true node position at $[\,100,\ 50\,]$m , whereas the dot is the estimated position $[\,100.5899,\ 50.0289\,]$m using the Newton algorithm. The ellipse is the Cramér-Rao bound on the position estimates. (b). The likelihood surface J without any GPS errors. The Newton method estimates the node position as $[\,100.5811,\ 50.0360\,]$m . The solutions are almost identical.

terms in the CRLB.

B. Performance Comparisons

In this section, we first compare the two CRLBs, one derived directly from the array model (inverse of the FIM (24)) and the other through DOA estimation methods (43). Figure 6 shows that both bounds are identical even at different frequencies. Intuitively, this may suggest similar performance for the ML calibration algorithm and the DOA calibration algorithms. However, Figs. 7 and 8 show that this is not the case.

Figure 7 shows the results of a Monté-Carlo run for the calibration algorithms. The ML and SAR calibration methods are given the correct time-warping function. When approximate time-warping is used, the author's observed that the ML calibration algorithm is much less sensitive than the SAR calibration algorithm. In Fig. 7, the SAR method is first run with the least squares solution (SAR-LS solid line with circle markers in the figure) that has higher estimation errors. The output of this algorithm is then used to initialize the SAR-ML solution as well as the ML calibration method. The MH calibration method performs similar to the ML and SAR-ML methods even though it is uninformed of the correct time-warping function.

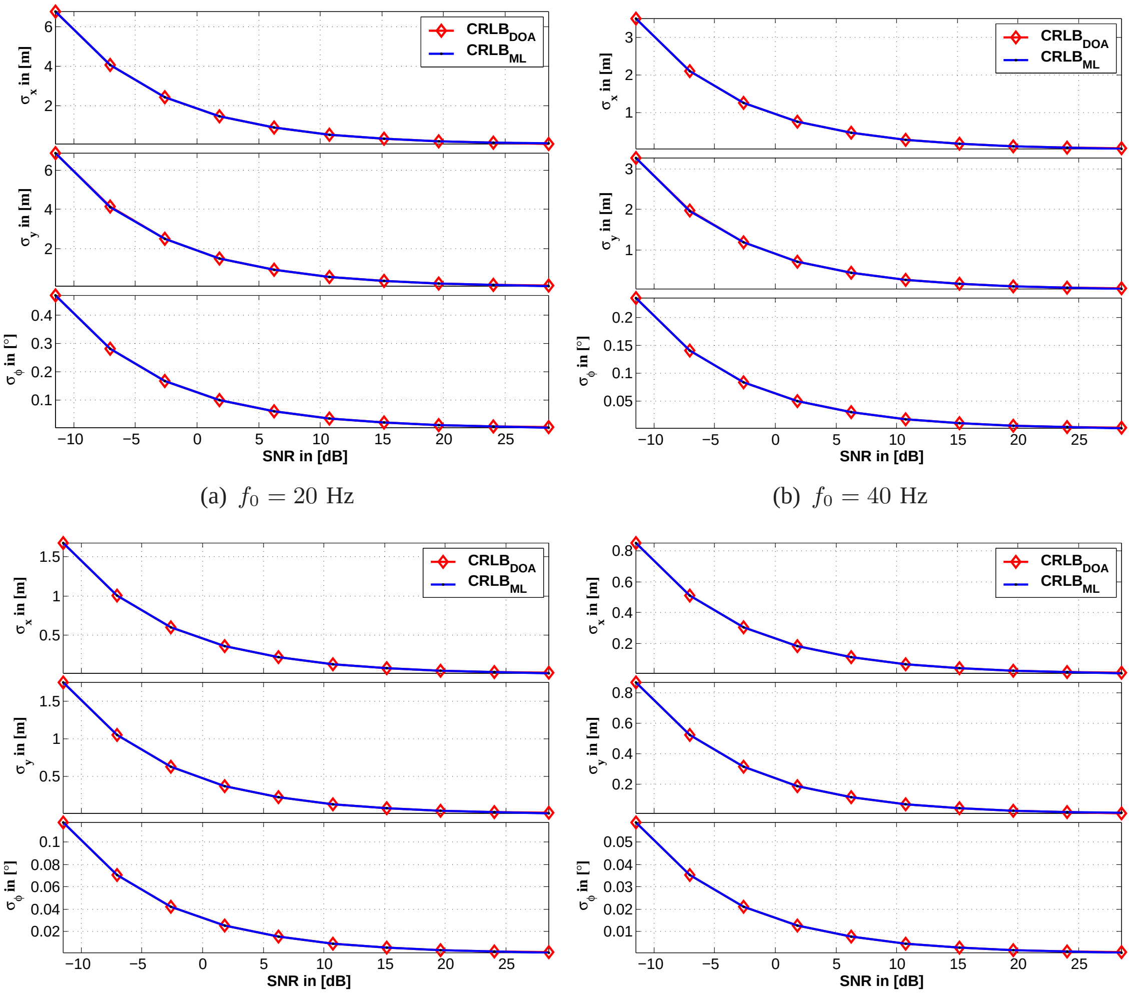
Figure 8 compares the performance of each algorithm with the CRLB. Figure 8(a) uses the circular track, whereas Fig. 8(b) is simulated with the straight line track. The position estimation bounds for the circular track is much lower, because the node observes a much larger angular span. The ML and the MH calibration methods follow the CRLB closely. The performance of the algorithms flattens at high *SNR*'s, because of the GPS errors.

In Fig. 8(a) and (b), the SAR-ML and LS methods have a constant variance over the entire *SNR* range with increased estimation bias. At some *SNR* ranges, the SAR-ML method beats the CRLB. Such estimators do exist but are hard to find [13]. Note that an efficient unbiased estimator has a performance bound defined by the CRLB. It is also known that if an estimator achieves the CRLB, then it is the ML estimator. In our problem, the ML estimation algorithm achieves the CRLB as the number of data points are increased.

V. Field Data Results

We applied the MH calibration scheme on field data from a small acoustic array (also appeared in [22]). A helicopter flew sorties around the acoustic node for the calibration purposes (Fig. 9). For the experiment, the array was hand-emplaced so that the true location and orientation would be known. The acoustic node has six omnidirectional microphones placed uniformly on a circle with a radius of 1.219m, which also corresponds to the inter-microphone distance. The spatial aliasing frequency is approximately 135Hz corresponding to the case where the radius of the array is equal to a half-wavelength. For the MVDR beamforming results, we tracked the ten highest peaks using the short-time Fourier transform (Fig. 10), and averaged the estimated DOA's accordingly. For the time-synchronization, third-order b-splines were used to interpolate the irregular time-grid for the algorithm's proposed positions for each particle.

The GPS track of the helicopter used for the calibration is shown in Fig. 9. Figure 11 demonstrates the results of the Metropolis-Hastings scheme with the Mode-Hungry modification. Figure 11 also demonstrates that the MH calibration results are significantly worse without the time-synchronization step. Interestingly, for this test run, the "true position" of the acoustic node was determined with a low-cost GPS system that is only accurate to within 7 meters, implying that the MH scheme performed within this variance.The ML calibration method with the approximate time-warping ($\xi_0 = [\ 0,\ 0\ ]$m) results in an estimate of $\xi = [\ -5.76,\ 6.98\ ]$m at



(a) $f_0 = 20$ Hz

(b) $f_0 = 40$ Hz

(c) $f_0 = 80$ Hz

(d) $f_0 = 160$ Hz

Fig. 6. The Cramér Rao lower bound dependency on the calibration source's center frequency. The calculated CRLB from the ML formulation is shown with the solid line; the CRLB from the DOA calibration formulations is shown with the diamonds. The bounds are exceptionally close, even though they are derived from different formulations, as indicated earlier.

$f_0 = 60Hz$. When $\xi_0 = [\ 10,\ 10\ ]$m, the ML solution results in a position of $[\ 22.14,\ 24.58\ ]$m. To establish a baseline, we also determine the CRLB to approximate the lower bounds for the variances for the parameter vector as $[\sigma_x, \sigma_y, \sigma_\theta] = [1.3931\text{m}, 1.3597\text{m}, 0.13296°]$.

Note that the CRLB assumes a single sinusoid signal model for the observations, but this assumption is not satisfied for the helicopter signal. In Fig. 11, the field data between $t = 50$s and $t = 120$s also violate the multi-path assumption. Hence, the DOAs in this region are not normally distributed around their true mean. Without post-processing of the data, it is difficult to detect this incorrect mean, since the orientation estimates can also bias the DOA distribution. The

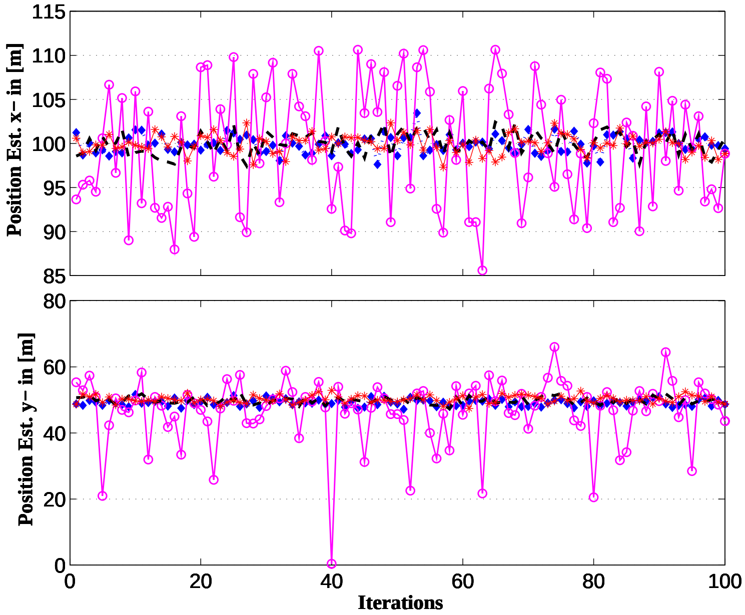


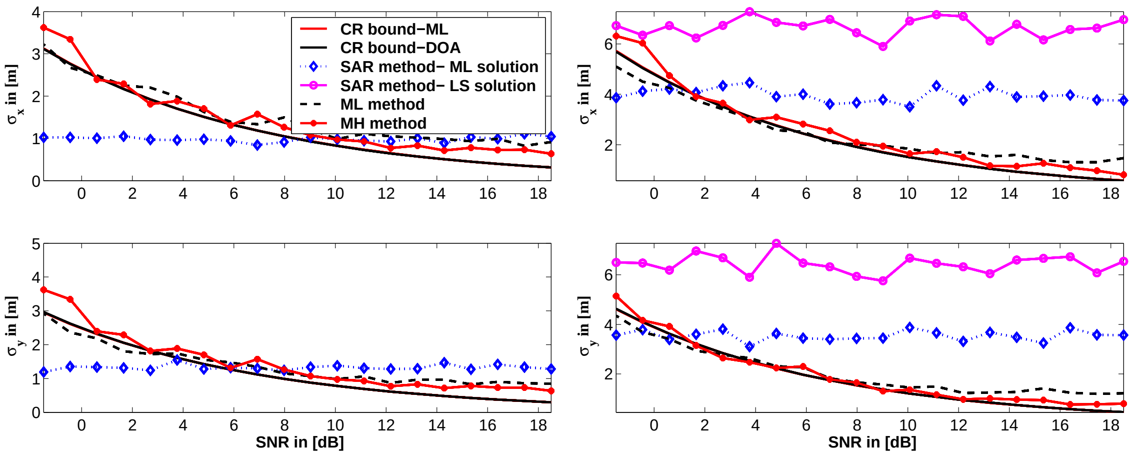
Fig. 7. The node is situated at $[\,100,\ 50\,]$m. The calibration is repeated 100 times with independent noise realizations at $SNR = 8$ dB, using the circular calibration track. The performance of the synthetic aperture method is shown with diamonds, where (35) is used to solve for the position. The solid line marked with circles is also the synthetic aperture method, except the least squares (33) approach is used for the solution. Full ML estimates using the acoustic data are shown with the dashed line, whereas the Metropolis-Hastings calibration method is shown with stars.

algorithm still managed to do well because the source distribution weights the calculated DOAs according to their estimated range. When the source is far away from the node, the estimates are expected to get worse. Hence, the MH calibration algorithm puts less importance on the DOA estimates corresponding to large ranges.

Unfortunately, the SAR calibration methods diverged for the field data example. The authors believe that this is because they are very sensitive to errors in the time-warp function, as observed in the synthetic simulations. Another likely reason is the relatively low GPS sampling period: 2 seconds might introduce interpolation errors. In addition, the field data and the GPS observations are block synchronized (every minute) and each block has a different global time clock shift. This poses additional difficulties for automatically handling the synchronization problem through the time-warp function.

VI. Conclusions

Various node calibration algorithms were demonstrated given a calibration target that carries a GPS system. The maximum-likelihood solution was demonstrated and compared to the biased



(a)

(b)

Fig. 8. (a) Calibration comparison using the circular track. The synthetic aperture method with the least squares solution has approximately 15m estimation error (out of the graph area). (b) Calibration comparison using the straight line track. Note that all the methods except the MH method are simulated with the correct time-warp function.

synthetic aperture method, which has a lower minimum mean squared error. The DOA calibration algorithms were formulated to compensate for array non-stationarity problems using array models for moving sources. Among the calibration methods, the Metropolis-Hastings (MH) calibration algorithm is the most flexible and suitable for field data processing because it can incorporate time synchronization as well as motion compensation directly into the calibration. In addition to processing some field data, the performance of the proposed algorithms was compared with simulation examples.

References

[1] Y. Zhou, P.C. Yip, and H. Leung, “Tracking the direction-of-arrival of multiple moving targets by passive arrays: Algorithm,” *IEEE Trans. on Signal Processing*, vol. 47, no. 10, pp. 2655–2666, October 1999.

[2] M. Orton and W. Fitzgerald, “A Bayesian approach to tracking multiple targets using sensor arrays and particle filters,” *IEEE Trans. on Signal Processing*, vol. 50, no. 2, pp. 216–223, February 2002.

[3] V. Cevher and J. H. McClellan, “General direction-of-arrival tracking with acoustic nodes,” *IEEE Trans. on Signal Processing*, vol. 53, no. 1, pp. 1–12, January 2005.

[4] B.C. Ng and C.M.S. See, “Sensor array calibration using a maximum-likelihood approach,” *IEEE Trans. on Antennas and Propagation*, vol. 44, pp. 827–835, June 1996.

[5] B.C. Ng and A. Nehorai, “Active array sensor localization,” in *ICASSP 1993*, 27-30 April 1993, vol. 4, pp. 21–24.

[6] V. Cevher and J. H. McClellan, “Sensor array calibration via tracking with the extended Kalman filter,” in *Proc. of the the Fifth Ann. Fed. Lab. Symp. on Adv. Sensors*, College Park, MD, 20-22 March 2001, pp. 51–56.

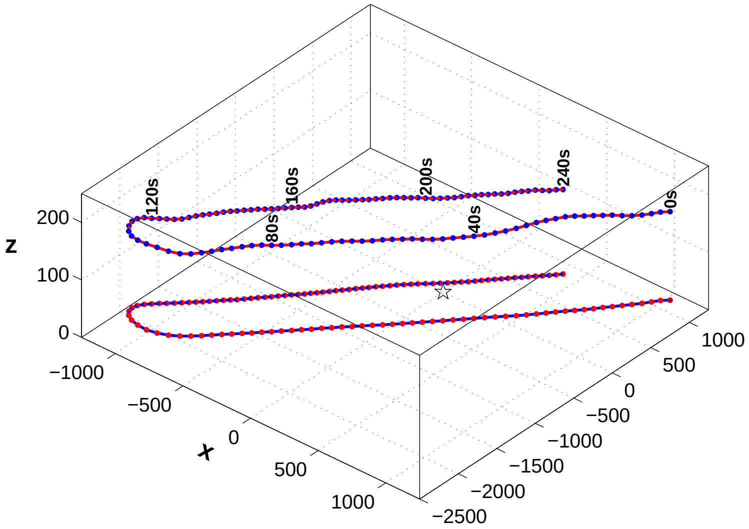


Fig. 9. The acoustic node (star) is situated at the origin. The calibration helicopter completes two sorties around the node for the calibration, corresponding to a four-minute run. The actual helicopter track as well as its projection on the x-y plane are shown.

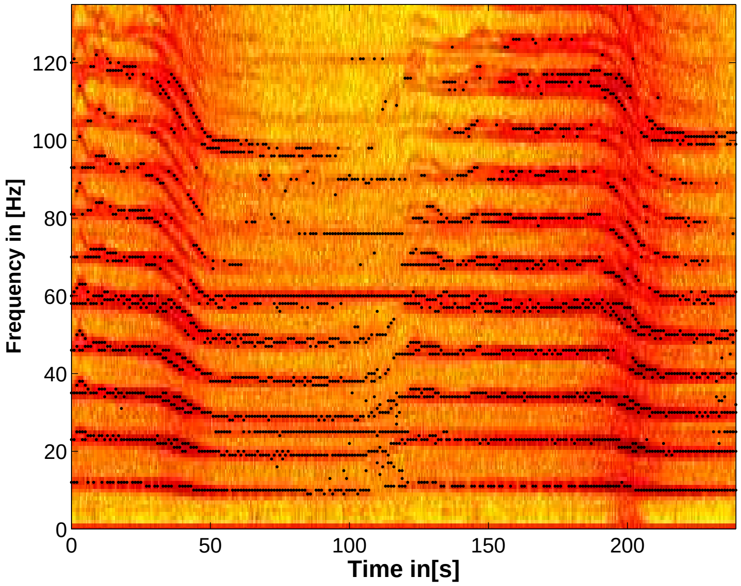


Fig. 10. The helicopter spectrum displays strong harmonic lines. The ten highest peaks in the time-frequency plane are picked using the magnitude of the Fourier transform once per second. These frequencies as well as their time-frequency amplitudes are used in determining the DOA estimates.

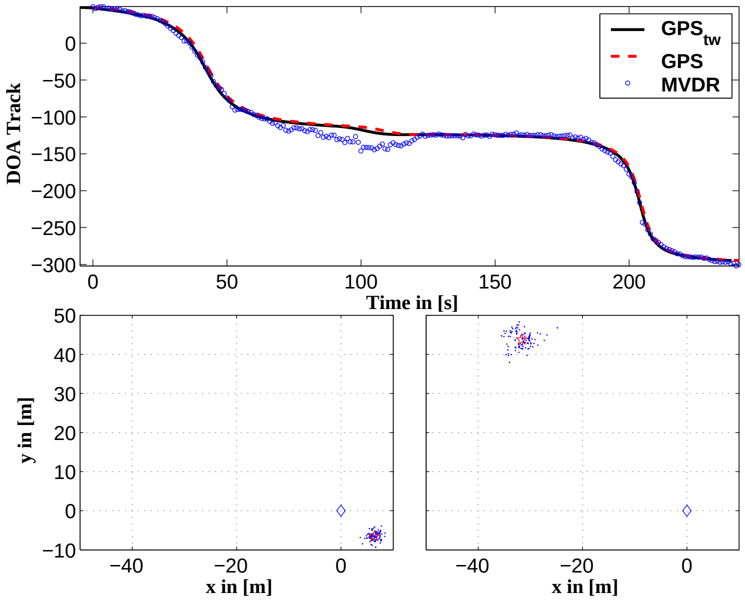


Fig. 11. Top figure shows the GPS track coming from the helicopter (dashed line), the time-warped GPS track using (5), and the MVDR beamformer estimates of the field data. The bottom left plot is the resulting MHMH algorithm distribution with estimate $\xi = [\ 6.43,\ -6.52\ ]$m using time synchronization, whereas, at the bottom right, the distribution with estimate $\xi = [\ -31.66,\ 43.77\ ]$m is the MHMH result without time synchronization. The estimated orientation for both cases is 1°. The true node location is shown with the diamond.

[7] R.L. Moses, D. Krishnamurthy, and R. Patterson, "An auto-calibration method for unattended ground sensors," in *ICASSP 2002*, Orlando, FL, May 2002, vol. 3, pp. 2941–2944.

[8] R.L. Moses, D. Krishnamurthy, and R. Patterson, "An auto-calibration method for unattended ground sensors," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 348–358, 2003.

[9] D.H. Johnson and D.E. Dudgeon, *Array Signal Processing: Concepts and Techniques*, Prentice Hall, 1993.

[10] J. Nocedal and S.J. Wright, *Numercial Optimization*, Springer-Verlag, 1999.

[11] J.E.Dennis Jr. and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Society for Industrial and Applied Mathematcis, Philadelphia, PA, 1996.

[12] M. Soumekh, *Synthetic Aperture Radar Signal Processing With Matlab Algorithms*, Wiley, 1999.

[13] H.V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, 1994.

[14] V. Cevher and J. H. McClellan, "Fast initialization of particle filters using a modified Metropolis-Hastings algorithm: Mode-Hungry approach," in *ICASSP 2004*, Montreal, CA, 17–22 May 2004.

[15] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings algorithm," *The American Statistician*, vol. 49, no. 4, pp. 327–335, 1995.

[16] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller, "Equations of state calculations by fast computing machines," *Journal of Chemical Physics*, vol. 21, pp. 1087–1092, 1953.

[17] W.K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, pp. 97–109, 1970.

[18] P. Stoica and A. Nehorai, "Music, maximum likelihood, and Cramér-Rao bound," *IEEE Trans. on ASSP*, vol. 37, no. 5,

pp. 720–741, May 1989.

[19] M.G. Soares, B. Malheiro, and F.J. Restivo, “An internet DPGS service for precise outdoor navigation,” in *IEEE Conf. of Emerging Techs. and Factory Aut.*, 16–19 Sept. 2003.

[20] A. Gelman, G.O. Roberts, and W.R. Gilks, “Efficient Metropolis jumping rules,” *Bayesian Statistics*, vol. 5, 1996.

[21] M. Skolnik, *Introduction to Radar Systems*, McGraw-Hill, 2001.

[22] V. Cevher and J. H. McClellan, “Acoustic node calibration using helicopter sounds and monte carlo markov chain methods,” in *IEEE DSP Workshop*, Taos Ski Valley, NM, 1-4 August 2004.