2D Location Estimation of Angle-Only Sensor Arrays Using Targets of Opportunity

David F. Crouse, Richard W. Osborne III, Krishna Pattipati, Peter Willett, and Yaakov Bar-Shalom

Dept. of Electr. and Comp. Engineering, University of Connecticut, Storrs, CT 06269, USA *

Abstract – *Passive acoustic sensor arrays for tracking ground targets are becoming increasingly popular due to their low cost and ease of deployment. In this paper we present an algorithm for locating sensor arrays in two-dimensions in an acoustic network (or in any network where angle-only measurements are used) when external references, such as GPS or known-location targets, are unavailable. We consider sensor localization when angular measurements are taken from the sensor arrays to targets of opportunity when all sensors take measurements with respect to a common axis of unknown orientation and where the sensors can not "see" each other.*

The solutions provided consist of low-complexity (generally closed-form) methods of getting initial estimates with no prior information, followed by maximum likelihood (ML) optimization to refine the estimates. Simulation shows that the accuracy approaches the Cramér Rao Lower Bound (CRLB), something that similar algorithms from previous research have been unable to achieve.

1 Introduction

DUE to their low cost and ease of deployment, the use of passive acoustic sensors for target tracking, be they underwater or land-based, has been increasing in popularity. One method of building such a tracking system is to deploy arrays of sensors, whereby each array produces measurements consisting of arrival angles and classification information useful in data association (used, for example, in [10]). Another approach is to deploy individual microphones that self-assemble into arrays [15]. In this paper, we consider estimating the locations of arrays of passive acoustic sensors, that best matches the first scenario. The terms "sensor array" and "node" shall be used interchangeably.

When considering land-based systems, one can not assume that the network of satellites forming the Global Positioning System (GPS) will be available, and GPS signals can not penetrate far underwater. Many non-GPS-based location estimation algorithms designed primarily for use in underwater and wireless networks may be used. A number of families of methods as applied to sonar channels are described in [3]. These approaches typically utilize the communication characteristics between sensors and are divided into two categories: Range-based and Range-free. Range-based methods utilize range and/or bearing information of inter-sensor signals, such as the angle of arrival, the time difference of arrival, or the received signal strength. This may include the use of mobile beacons (anchors) that broadcast their locations to nearby sensors. Range-free schemes do not use this information, but may utilize moving anchor nodes that broadcast their position [6], [5], [7], [13].

We consider algorithms for node localization based upon the angle-only observations of the nodes. Such estimates are particularly useful in networks involving data MULEs (Mobile Ubiquitous LAN[1] Extensions) [12]. In such situations, individual sensors generally have a limited broadcast range. Rather than having a network between sensors, data is periodically queried from the sensors using a data mule, some type of device that physically approaches the sensors and reads from them. In such a network, traditional methods of sensor localization, which rely on communication channels between sensors, are not applicable. Additionally, when considering wireless land-based acoustic sensors, estimates of the sensor locations obtained using acoustic data would be expected to be uncorrelated with those obtained using more traditional means. Since our method approaches the CRLB, as demonstrated in Section 4, the CRLB may be used as the measurement covariance for merging the estimates.

A network where nodes observe common targets of opportunity is equivalent to a network where nodes observe certain sensors that produce no observations of their own. Using a number of anchor nodes (the minimum number to achieve observability is two), we consider sensor array registration where stationary sensors take measurements of common targets of opportunity with respect to a common, unknown coordinate axis. The targets could be one target over time, multiple targets simultaneously or any combination of the two choices. Tracking is not performed. Associa-

*This work was partially supported by the Office of Naval research under contracts N00014-09-10613 and N00014-10-10412.

[1] Local Area Network

tion of measurement across sensors might be accomplished, for example, by utilizing acoustic patterns for classification, as has previously been done to aid acoustic tracking [10].

A maximum likelihood (ML) solution to this problem with a known axis of orientation was outlined in [8]. However, their solution does not take into account the fact that angles outside of $(-\pi, \pi)$ are equivalent to those in the aforementioned range. Without proper compensation, even with good initialization estimates will be significantly worse than the CRLB. We provide a corrected form in Section 3. In [8] the authors also did not consider the initialization of the likelihood maximization algorithm. Using a random initialization, we found that the estimates, when computed using the corrected likelihood, converged to local maxima having errors that are far worse than the CRLB, and generally are far worse than the initialization itself. Thus, in Section 2 we discuss the generation of initial estimates.

Other than [8], work considering location and orientation determination of angle-only sensor arrays has been sparse. A significant amount of work has been done regarding localizing users within cellular networks [14], with very little attention to the angle-only measurement case. In what has been done, a single user must always be in range of at least two base stations (anchor nodes). Other work has considered similar issues for cellular networks [9], whereby all users are in range of a number of anchor nodes.

Many papers dealing with sensor registration only correct for residual bias after an initial estimate has somehow been obtained. Most require full range and angle estimates (see [4] for an extensive bibliography), though some are adaptable to the range-only case [11].

Section 2 discusses methods for generating estimates when given *no prior information* beyond the location of at least two anchor nodes. These estimates may be used in the maximum likelihood algorithm of Section 3, which generally won't converge to a meaningful value without a good initial estimate. All the methods are exact in the noiseless case. Section 4 shows simulation results for three noisy scenarios.

2 Algorithms for Generating Initial Estimates

In all of the following subsections, we shall assume that the sensors reside in the (x, y)-plane. The sensor arrays and the targets are assumed to be individual points in space. The measurements between sensor arrays are assumed to be synchronized with respect to the targets. That is, in each scan, measurements from disparate sensor arrays represent the angle from the arrays to the targets at the same target position[2].

Let the location of the ith sensor array in the plane be designated by the pair $\left(s_x^i, s_y^i\right)$. The target-angle measured by the ith sensor array at time-step k shall be designated $\theta_i(k)$. It should be noted that individual observations may occur simultaneously or at different times. If multiple targets are present at the same time, then classification information may be used to associate measurements between sensors. We will not address the problem where measurements can not be associated between sensors, in which case there may be multiple possible solutions for the target location based upon a particular set of observations.

[2]This is equivalent to saying that the sensor arrays are assumed to be synchronized and the acoustic propagation delay between target and sensor is assumed to be zero

2.1 A Method of Producing Initial Sensor and Target Location Estimates

We shall begin by assuming that all measurements are taken with respect to a common, known axis, and then show in Section 2.2 how that assumption may be relaxed.

2.1.1 Jointly Estimating Sensor Array and Target Locations

The angular measurement of sensor i at time k, $\theta_i(k)$, taken with respect to a known, common axis may be expressed as follows

$$\tan\left[\theta_i(k)\right] = \frac{t_y(k) - s_y^i}{t_x(k) - s_x^i} \tag{1a}$$

$$\cot\left[\theta_i(k)\right] = \frac{t_x(k) - s_x^i}{t_y(k) - s_y^i} \tag{1b}$$

These equations may be rearranged to get

$$t_x(k)\tan\left[\theta_i(k)\right] - s_x^i\tan\left[\theta_i(k)\right] - t_y(k) + s_y^i = 0 \tag{2a}$$

$$t_y(k)\cot\left[\theta_i(k)\right] - s_y^i\cot\left[\theta_i(k)\right] - t_x(k) + s_x^i = 0 \tag{2b}$$

Thus, using (2a) and (2b) one can easily generate a linear system of equations that, in the absence of measurement noise, may be solved exactly for the sensor and target locations, given that enough sensors have known locations a priori. Sensors with known locations are necessary for uniqueness of a nontrivial solution and are simply put on the right-hand side of the equation.

A necessary condition for the observability of the system is that the locations of at least two sensors are known. Note that the two sensor arrays with known locations do not necessarily have to observe the same target at the same time; they need simply be linked by a number of measurements seen by connected subsets of sensor arrays.

For example, consider the presence of three sensor arrays observing a common target at times k_1 and k_2. We shall assume that the location of sensor arrays one and three are known *a priori*. We would like to determine the location of the other sensor array. Using (8a) we may write the set of equations in (3).

Thus, in this case, the location of the second sensor array and the target at both times is the solution to

$$\mathbf{A}\mathbf{s} = \mathbf{b} \tag{4}$$

Note that due to the use of the tangent in Equation (2a), serious estimation inaccuracies can occur if the targets used for estimation are close to $\pm 90^\circ$ with respect to any of the

$$\underbrace{\begin{bmatrix} 0 & 0 & \tan[\theta_1(k_1)] & -1 & 0 & 0 \\ -\tan[\theta_2(k_1)] & 1 & \tan[\theta_2(k_1)] & -1 & 0 & 0 \\ 0 & 0 & \tan[\theta_3(k_1)] & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \tan[\theta_1(k_2)] & -1 \\ -\tan[\theta_2(k_2)] & 1 & 0 & 0 & \tan[\theta_2(k_2)] & -1 \\ 0 & 0 & 0 & 0 & \tan[\theta_2(k_2)] & -1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} s_x^2 \\ s_y^2 \\ t_x(k_1) \\ t_y(k_1) \\ t_x(k_2) \\ t_y(k_2) \end{bmatrix}}_{\mathbf{s}} = \underbrace{\begin{bmatrix} \tan[\theta_1(k_1)]\, s_x^1 - s_y^1 \\ 0 \\ \tan[\theta_3(k_1)]\, s_x^3 - s_y^3 \\ \tan[\theta_1(k_2)]\, s_x^1 - s_y^1 \\ 0 \\ \tan[\theta_3(k_2)]\, s_x^3 - s_y^3 \end{bmatrix}}_{\mathbf{b}} \tag{3}$$

observing sensors since measurement error can push the measured angle above or below $\pm 90^\circ$, changing a large positive entry in the $\mathbf{A}$ matrix to a very large negative value or vice versa. This problem may be minimized by using Equation (2b), which uses the cotangent, when the observation is between 45° and 135° or between -45° and -135°.

2.1.2 Estimating the Sensor Array Locations Alone

If the target locations are not needed, they may be eliminated from the estimation. We shall once again assume that all angles are taken with respect to a reference direction common for all sensor arrays. In this subsection, we shall also assume that each target is observed simultaneously by at least three sensor arrays. We shall define the measurements as being taken with respect to the x-axis in our 2-D coordinate system. For simplicity of notation, let us define the following function

$$\Delta_{a,b}^T(k) \triangleq \tan[\theta_a(k)] - \tan[\theta_b(k)] \tag{5}$$

$$\Delta_{a,b}^C(k) \triangleq \cot[\theta_a(k)] - \cot[\theta_b(k)] \tag{6}$$

$$\Psi_{a,b}(k) \triangleq 1 - \cot[\theta_a(k)] \tan[\theta_b(k)] \tag{7}$$

As proven in the appendix, given any three sensor arrays simultaneously observing the target, we may combine (1a) and (5) using (2a) for each sensor array to get an expression relating the sensor locations independent of the Cartesian location of the target. For sensors 1 through 3 this gives us equations (8a), (8b), (8c) and (8d).

As was the case in the previous section, the equations derived in this section may be used with multiple observations of the targets over time to reduce the solution of the sensor array locations to that of solving $\mathbf{As} = \mathbf{b}$, where in this case $\mathbf{s}$ is only the sensor array locations.

It should be noted that as the number of sensor arrays increases, the number of possible equations that may be written increases quickly. However, the equations are not all independent. For example, for N sensors observing a common target, there are $\binom{N}{3}$ possible variants of (8a) that may be written depending upon which three targets are put into the equation. However, for $N > 3$ only N of these equations can be linearly independent and do not provide any new information, because they are not based on new observations. Linearly dependent equations may be removed by using the Modified Gram-Schmidt Orthonormalization Algorithm or other, similar methods, though, as shown in Section 4, this can hurt the performance of the algorithm.

2.2 Dealing with Measurements with Respect to an Unknown, Common Axis

We would like to consider the case where all sensors have the same unknown bias in their measurements. This might occur, for example, if all measurements are taken with respect to magnetic north, but the anchor node locations are given in terms of geographic north.

Consider Figure 1. In the noiseless case, if we were to remove the second anchor node and not to compensate for the bias, then Figures 1 (b), (c), and (d) are three possible solutions for the system described by $\mathbf{As} = \mathbf{b}$ using the equations from Section 2.1. All of the biased solutions are rotated by the bias angle. As shown in (b) and (c), the figure may be scaled about the single anchor node without changing any of the measured angles (in the case where angles are measured to targets, the apparent locations of the targets are scaled as well). Figure 1 (d) comes about due to our use of the tangent and cotangent in Section 2.1, whereby the equations do not change if all angles are flipped 180°.

In the case of only two anchor nodes, a method of estimating the sensor array locations while correcting for the unknown global rotation is as follows:

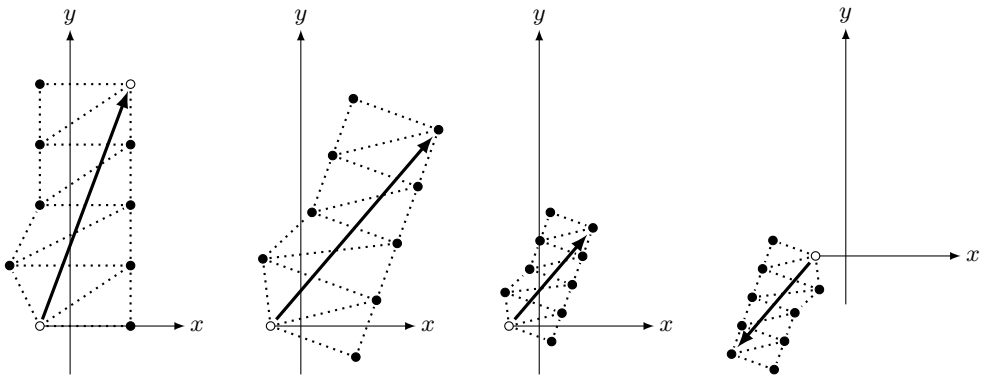
1. Find an observation from the first anchor node. Assume that it is a known fixed distance from the first anchor node (e.g. 10m). Find its location using the bias measurement under this assumption. This shall be a pseudo-anchor node.
2. Perform the sensor array location estimation as described in Section 2.1 using the biased measurements, the first anchor node and the previously determined pseudo-anchor node as an anchor node assuming that the location of the second anchor node is unknown.
3. Find the vector between the first anchor node and the true location of the second anchor node (e.g. for the scenario in 1 (a), it has been drawn). We shall call this $\mathbf{v}_1$. Also find the vector between the first anchor node and the apparent position of the second anchor node as given by the previous estimation (e.g. the vectors in 1 (b) or 1 (c)). The choice of the pseudo-anchor node rules out the geometry of 1 (d)); we shall call this $\mathbf{v}_2$.

$$0 = s_y^1 \Delta_{2,3}^T(k) + s_x^1 \tan\left[\theta_1(k)\right] \Delta_{3,2}^T(k) + s_y^2 \Delta_{3,1}^T(k) + s_x^2 \tan\left[\theta_2(k)\right] \Delta_{1,3}^T(k) + s_y^3 \Delta_{1,2}^T(k) + s_x^3 \tan\left[\theta_3(k)\right] \Delta_{2,1}^T(k) \tag{8a}$$

$$0 = s_y^1 \cot\left[\theta_1(k)\right] \Delta_{2,3}^T(k) + s_x^1 \Delta_{3,2}^T(k) - s_y^2 \Psi_{1,3}(k) + s_x^2 \tan\left[\theta_2(k)\right] \Psi_{1,3}(k) + s_y^3 \Psi_{1,2}(k) - s_x^3 \tan\left[\theta_3(k)\right] \Psi_{1,2}(k) \tag{8b}$$

$$0 = s_y^1 \cot\left[\theta_1(k)\right] \Psi_{2,3}(k) - s_x^1 \Psi_{2,3}(k) - s_y^2 \cot\left[\theta_2(k)\right] \Psi_{1,3}(k) + s_x^2 \Psi_{1,3}(k) + s_y^3 \Delta_{2,1}^C(k) + s_x^3 \tan\left[\theta_3(k)\right] \Delta_{1,2}^C(k) \tag{8c}$$

$$0 = s_y^1 \cot\left[\theta_1(k)\right] \Delta_{3,2}^C(k) + s_x^1 \Delta_{2,3}^C(k) + s_y^2 \cot\left[\theta_2(k)\right] \Delta_{1,3}^C(k) + s_x^2 \Delta_{3,1}^C(k) + s_y^3 \cot\left[\theta_3(k)\right] \Delta_{2,1}^C(k) + s_x^3 \Delta_{1,2}^C(k) \tag{8d}$$



(a) The True Sensor Locations (b) One Biased Solution (c) A Second Biased Solution (d) A Third Biased Solution

Figure 1: The open circles represent the anchor sensors, whose locations are known *a priori*. The array is the vector between the first and last nodes. Subfigure (a) shows the true setup of the problem. Subfigures (b), (c) and (d) show possible solutions when the second anchor node is removed and a common bias is left uncompensated.

$$\frac{\partial \lambda}{\partial s_x^i} = \frac{1}{\sigma^2} \sum_j f\left(\theta_{i,j} - \tan^{-1}\left[\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right]\right) \frac{-(t_y^j - s_y^i)}{(t_x^j - s_x^i)^2 + (t_y^j - s_y^i)^2} \tag{9}$$

$$\frac{\partial \lambda}{\partial s_y^i} = \frac{1}{\sigma^2} \sum_j f\left(\theta_{i,j} - \tan^{-1}\left[\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right]\right) \frac{(t_x^j - s_x^i)}{(t_x^j - s_x^i)^2 + (t_y^j - s_y^i)^2} \tag{10}$$

$$\frac{\partial \lambda}{\partial t_x^i} = \frac{1}{\sigma^2} \sum_i f\left(\theta_{i,j} - \tan^{-1}\left[\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right]\right) \frac{(t_y^j - s_y^i)}{(t_x^j - s_x^i)^2 + (t_y^j - s_y^i)^2} \tag{11}$$

$$\frac{\partial \lambda}{\partial t_y^i} = \frac{1}{\sigma^2} \sum_i f\left(\theta_{i,j} - \tan^{-1}\left[\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right]\right) \frac{-(t_x^j - s_x^i)}{(t_x^j - s_x^i)^2 + (t_y^j - s_y^i)^2} \tag{12}$$

4. Evaluate $\theta = \angle \mathbf{v}_2 - \angle \mathbf{v}_1$.
5. Perform the sensor array location estimation again using the adjusted angles and both of the true anchor nodes to get a final estimate of the sensor array locations.

The algorithm finds a solution for the biased system and then compares how that solution is rotated with respect to the true system. The first step of the aforementioned method creates a pseudo anchor point to set a reference for the scaling of the solution. This is important to make sure that we do not get a solution that is inverted by $180°$, as in Figure 1(d). Moreover, it is necessary for setting the scaling of the figure. We would like to find a solution, but one possible solution places the nodes infinitesimally close to the first anchor point. The use of such a solution would be subject to precision problems on any computer.

When more sensors are present, one may break the observations into subsets according to the connectivity between anchor nodes, and calculate separate biases before averaging them. We shall not consider that case here.

3 ML Estimation

In order to further refine the estimates calculated using the methods outlined in Section 2, the estimates may be passed to a Maximum Likelihood (ML) estimator [1]. Assuming that the angular measurement from sensor i to target location j is

$$\theta_{i,j} = \tan^{-1}\left[\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right] + w_{i,j} \tag{13}$$

where $w_{i,j}$ is zero mean Gaussian measurement noise, the likelihood function is

$$\Lambda(\mathbf{s},\mathbf{t}) = \prod_{i,j} p(\theta_{i,j}|\mathbf{s},\mathbf{t}) \tag{14}$$

where

$$p(\theta_{i,j}|\mathbf{s},\mathbf{t}) \sim \mathcal{N}\left[\tan^{-1}\left[\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right], \sigma^2\right] \tag{15}$$

Determining the maximum of (14) is equivalent to finding the minimum of the negative log-likelihood function $\lambda(\mathbf{s},\mathbf{t})$, where (assuming the standard deviation of each sensor array's measurement noise is identical)

$$\lambda(\mathbf{s},\mathbf{t}) = \frac{1}{2\sigma^2}\sum_{i,j}\left[f\left(\theta_{i,j} - \tan^{-1}\left[\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right]\right)\right]^2 \tag{16}$$

The sum in (16) is over all pairs (i,j) where sensor i observes target j. The minimization of (16) is carried out using the Quasi-Newton optimization algorithm [2]. The function f is defined as follows

$$f(\theta) = \begin{cases} \theta - 2\pi & \text{If } \theta > \pi \\ \theta + 2\pi & \text{If } \theta < -\pi \\ \theta & \text{Otherwise} \end{cases} \tag{17}$$

The function f ensures that the magnitude of the difference term in (16) is never larger than π in magnitude. If this function were not added, a 2π difference, which actually means that there is zero error in the angle, would impose a large penalty[3]. This is not taken into account in [8] and without proper compensation, the CRLB can not be approached in simulation.

The gradient information required by the Quasi-Newton algorithm utilizes the terms in equations (9), (10), (11) and (12). Again, care should be taken to utilize (17) for the difference terms.

3.1 The Cramér-Rao Lower Bound

In order to evaluate the efficiency of the estimator, the CRLB for the particular scenarios must be calculated [1]. The CRLB provides a lower bound on the covariance matrix of an unbiased estimator as

$$E\left\{\left[\hat{\mathbf{x}} - \mathbf{x}_0\right]\left[\hat{\mathbf{x}} - \mathbf{x}_0\right]^T\right\} \geq J^{-1} \tag{18}$$

where $\mathbf{x}$ is a vector parameter, $\hat{\mathbf{x}}$ is the parameter estimate, $\mathbf{x}_0$ is the true parameter value, and J is the Fisher Information Matrix (FIM).

The FIM is defined as

$$J \triangleq E\left\{\left[\nabla_{\mathbf{x}}\lambda(\mathbf{x})\right]\left[\nabla_{\mathbf{x}}\lambda(\mathbf{x})\right]^T\right\}\Big|_{\mathbf{x}=\mathbf{x}_0} \tag{19}$$

When $\nabla_{\mathbf{s},\mathbf{t}}\lambda(\mathbf{s},\mathbf{t})$ is plugged into (19), the FIM is

$$\begin{aligned} J =& \frac{1}{\sigma^2}\sum_{i,j}\left[\nabla_{\mathbf{s},\mathbf{t}}\tan^{-1}\left(\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right)\right] \\ &\times\left[\nabla_{\mathbf{s},\mathbf{t}}\tan^{-1}\left(\frac{t_y^j - s_y^i}{t_x^j - s_x^i}\right)\right]^T\Bigg|_{\mathbf{s}=\mathbf{s}_0,\mathbf{t}=\mathbf{t}_0} \end{aligned} \tag{20}$$

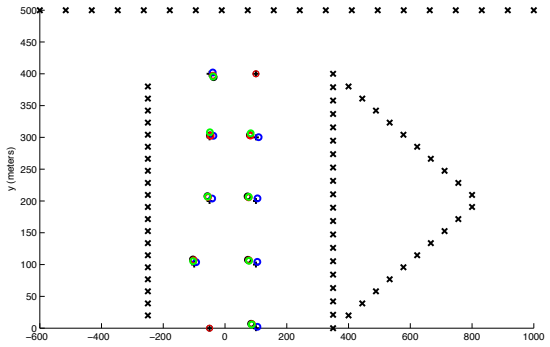
The appropriate diagonal entries of J^{-1} provide a lower bound for the mean squared error (MSE) of each estimated parameter, assuming that the estimator is unbiased.

4 Simulations

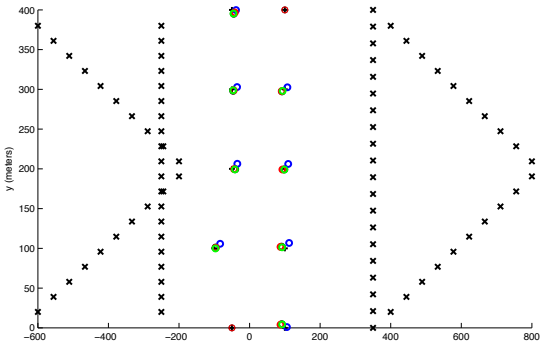
4.1 The Scenario

To test the accuracy of this sensor array location estimation algorithms, we used scenarios involving ten sensors and four targets over 20 time-steps. The sensor arrays were placed in x locations in the set of $\{-50, 100\}$ meters and y locations in the set of $\{0, 100, 200, 300, 400\}$ meters, with the exception of the one that would have been at $(-50, 100)$, which was instead set to $(-100, 100)$ in order to break the symmetry of the arrangement so that it would be clear if poor estimates flipped anything. It is the same configuration as shown in 1(a). The locations of the sensor arrays at

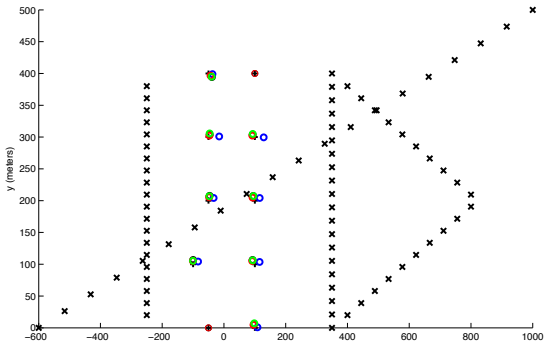
[3] This is accounts for the fact that the Gaussian distribution is defined across $(-\infty, \infty)$ whereas the measured angles and the inverse tangent function will be between $(0, 2\pi)$. It should be noted that this is an approximation, as the true PDF has an infinite number of terms. That said, for variances on the order of a few degrees it is an *excellent* approximation.



(a) Scenario 1

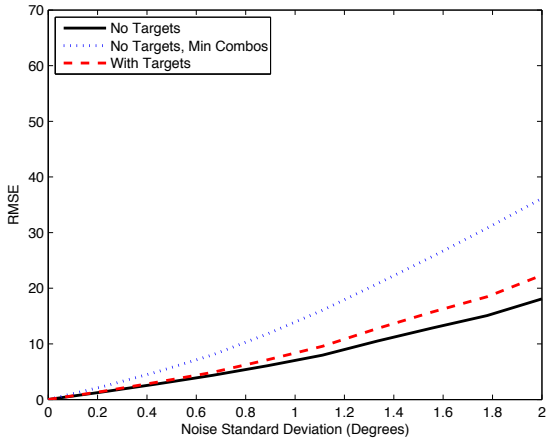


(b) Scenario 2

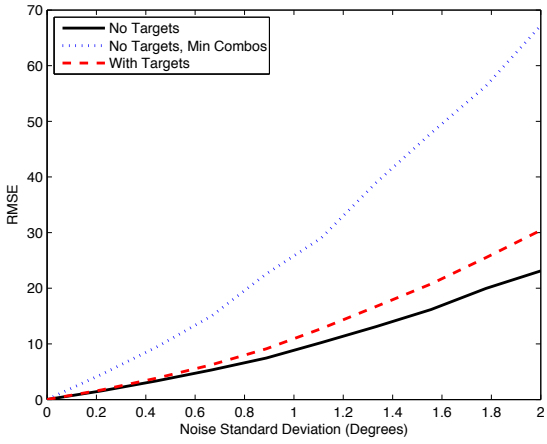


(c) Scenario 3

Figure 2: The target observations for the three scenarios. The sensor location estimates for various runs are shown as circles of various colors. The anchor nodes are in the upper-right and lower-left corners.



(a) Known Measurement Axis



(b) Unknown Measurement Axis

Figure 3: The RMSE of the estimated sensor locations of the initialization algorithms for Scenario 1. In (a) the measurements are taken with respect to a common, known axis. In (b) they are taken with respect to a common, unknown axis. 1000 Monte Carlo runs were performed.

$(-50, 0)$ and $(100, 400)$ were assumed known *a priori*, and were thus used as anchor arrays.

To assess the effect of geometry on the accuracy of the estimates, we considered three different scenarios for target motion. In all scenarios, the first target was located at an x location of -250 meters and traveled at a constant speed from 20 to 380 meters in y. The second target was placed at an x location of 350 meters and traveled at a constant speed from 0 to 400 meters in y. The third target started at 400 meters in x, traveled at a constant speed to 800 meters by step 10 and came back to 400 meters in the x direction by step 20. In the y direction, it traveled at a constant speed from 20 to 380. The fourth target traveled one of three paths, depending on the particular simulation. In Target Scenario 1, the fourth target was placed at a y location of 500 meters and traveled at a constant speed from -600 to 1000 meters in x. In Target Scenario 2, the fourth target takes an identical path to the third target, but shifted -1000 meters in the x direction. In Target Scenario 3, the fourth target traveled at a constant speed from $(-600, 0)$ to $(1000, 500)$. The scenarios are shown in Figure 2.

The targets were only visible to a subset of the sensor arrays at each time. From steps 1 to 4, only the sensor arrays at y locations of 0 and 100 meters could see the targets. From steps 5 through 8, the sensor arrays between 0 and 200 meters could see the targets. From steps 9 through 12 the sensor arrays between 100 and 300 meters could see the targets. From steps 13 through 16 the sensor arrays between 200 and 400 meters could see the targets and from steps 17 through 20 the sensor arrays from 300 to 400 meters in y could see the target. This means that the two sensor arrays with known locations never both saw any of the targets at the same time.

4.2 The Initialization Algorithms

We used target Scenario 1 to assess the performance of the three initialization algorithms as a function of the measurement noise covariance at the sensors from Section 2. These were the algorithm jointly estimating sensor and target locations (Section 2.1), the algorithm estimating only the target locations (Section 2.1.2) and the same algorithm whereby redundant equations have been eliminated using the Gram-Schmidt algorithm. In all cases the solution to $\mathbf{As} = \mathbf{b}$ was found using least squares. The noise generated was zero-mean Gaussian and was independent between sensors. 1000 Monte Carlo runs were performed. We considered both the case where the measurements are taken with respect to a common, known axis and when they are taken with respect to a common, unknown axis. When taken with respect to an unknown axis, the axis was chosen randomly for each Monte Carlo run. The performance is shown in Figure 3.

4.3 ML Maximization

Using the three scenarios from (1), we compared the performance of the uncorrected ML algorithm of [8] with our corrected version and the CRLB. The least-squares algorithm of Section 2.1 estimating both sensor and target locations was used to provide initial estimates. All measurements were taken with respect to a common, known axis. 1000 Monte Carlo runs were performed. The results are shown in Figure 4.

5 Conclusions

We presented a number of methods of location estimation for angle-only sensors observing targets of opportunity. The best linear (least squares) solution was obtained by eliminating the target locations from the estimation and using the maximum number of possible equations. The addition of a common, unknown bias that had to be corrected worsened the overall performance slightly. When performing ML estimation using a least squares initial estimate, in all geometries considered, estimates closely approached the CRLB. Our results use a model similar to that in [8]. Different from [8], we provide the means of initialization that iterative ML estimation requires, we address the (vital) issue of angle wrap-around, and we compare out results to the CRLB.

References

[1] Y. Bar-Shalom, X. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation: Theory, Algorithms and Software*. J. Wiley and Sons, 2001.

[2] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

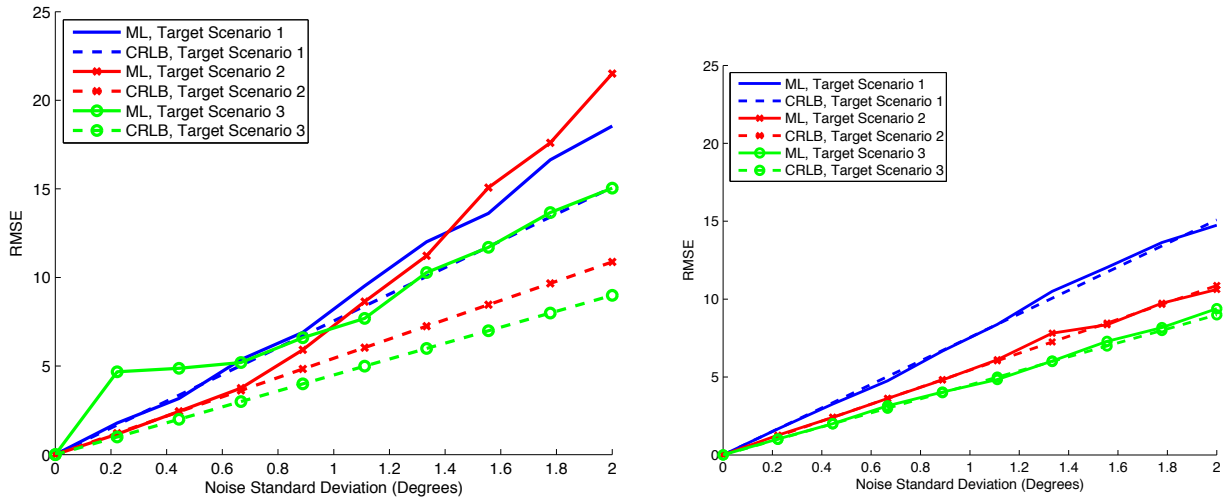
[3] V. Chandrasekhar, W. Seah, Y. S. Choo, and H. V. Ee, "Localization in underwater sensor networks - survey and challenges," in *Proceedings of the First ACM International Workshop on Underwater Networks*, 2006, pp. 33–40. [Online]. Available: http://wuwnet.engr.uconn.edu/papers/p033-chandrasekhar.pdf

[4] D. F. Crouse, Y. Bar-Shalom, and P. Willett, "Sensor bias estimation in the presence of data association uncertainty," in *Proceedings of SPIE: Signal and Data Processing of Small Targets Conference*, vol. 7445, 2009, pp. 74 450P–74 450P–22.

[5] M. Erol, L. Vieira, and M. Gerla, "AUV-aided localization for underwater sensor networks," in *Proceedings of the International Conference on Wireless Algorithms, Systems and Applications*, Aug. 2007, pp. 44–54.

[6] A. Galstyan, B. Krishnamachari, K. Lerman, and S. Pattem, "Distributed online localization in sensor networks using a moving target," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks*, 2004, pp. 61–70.

[7] W.-L. Huang, S.-J. Li, and L. Duo, "Designing reduced beacon trajectory for sensor localization," *Journal of Zheijang University*, vol. 8, no. 12, pp. 1971–1982, 2007.



(a) ML Not Corrected

(b) ML Corrected

Figure 4: The RMSE of the estimated sensor locations of the three scenarios compared to the CRLB. In (a) we use the method of [8] that does not use the correction of (17). In (b) we use the aforementioned correction.

[8] R. L. Moses, D. Krishnamurthy, and R. Patterson, "An auto-calibration method for unattended ground sensors," in *Proceedings of the IEEE International Conference on Acoustics Speech and Signal Processing*, vol. 3, 2002, pp. 2941–2944.

[9] A. Nasipuri and K. Li, "A directionality based location discovery scheme for wireless sensor networks," in *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, 2002, pp. 105–111.

[10] V. C. Ravindra, Y. Bar-Shalom, and T. Damarla, "Feature-aided localization of ground vehicles using passive acoustic sensor arrays," in *Proceedings of the 12th International Conference on Information Fusion*, Seattle, Jul. 2009, pp. 70–77.

[11] B. Ristic and N. Okello, "Sensor registration in ECEF coordinates using the MLR algorithm," in *Proceedings of the Sixth International Conference of Information Fusion*, Cairns, Queensland, Australia, 2003, pp. 135–142.

[12] R. C. Shah, S. Roy, S. Jain, and W. Brunette, "Data MULEs modeling a three-tier architecture for sparse sensor networks," in *Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications*, Seattle, May 2003, pp. 30–41.

[13] K.-F. Ssu, C.-H. Ou, and H. C. Jiau, "Localization with mobile anchor points in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 54, no. 3, pp. 1187–1197, May 2005.

[14] G. Sun, J. Chen, W. Guo, and K. Liu, "Signal processing techniques in network-aided positioning: A survey of state-of-the-art positioning designs," *IEEE Signal Processing Magazine*, vol. 22, no. 4, pp. 12–23, Jul. 2005.

[15] J. Zhang, M. Walpola, D. Roelant, H. Zhu, and K. Yen, "Self-organization of unattended wireless acoustic sensor networks for ground target tracking," *Pervasive and Mobile Computing*, vol. 5, no. 2, pp. 148–164, Apr. 2009.

6 Appendix: Derivation of (8a)

Here we derive (8a), which underlies much of the algorithm. The derivations of (8b), (8c) and (8d) are performed similarly. Equation (1a) applied to the first sensor array gives us

$$\tan[\theta_1] = \frac{y - s_y^1}{x - s_x^1} \tag{21}$$

$$y = s_y^1 + \left(x - s_x^1\right)\tan[\theta_1] \tag{22}$$

Substituting (22)into (1a) applied to the second and third sensor arrays gives us

$$\tan[\theta_2] = \frac{s_y^1 - s_y^2 + \left(x - s_x^1\right)\tan[\theta_1]}{x - s_x^2} \tag{23}$$

$$\tan[\theta_3] = \frac{s_y^1 - s_y^3 + \left(x - s_x^1\right)\tan[\theta_1]}{x - s_x^3} \tag{24}$$

Solving (23) for the x location of the targets gives us

$$x = \frac{s_y^2 - s_y^1 + s_x^1\tan[\theta_1] - s_x^2\tan[\theta_2]}{\tan[\theta_1] - \tan[\theta_2]} \tag{25}$$

Substituting (25) back into (24) and simplifying gives us the form of (8a).