Research Article

Underwater Localization and Tracking of Physical Systems

Patrick Carroll,¹ Shengli Zhou,¹ Hao Zhou,¹ Xiaoka Xu,¹ Jun-Hong Cui,² and Peter Willett¹

¹ Department of Electrical and Computer Engineering, University of Connecticut, Storrs, CT 06269, USA

² Department of Computer Science and Engineering, University of Connecticut, Storrs, CT 06269, USA

Correspondence should be addressed to Patrick Carroll, pec05001@engr.uconn.edu

Received 15 December 2011; Accepted 16 May 2012

Academic Editor: Charalampos C. Tsimenidis

Copyright © 2012 Patrick Carroll et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

We investigate the problem of localizing an underwater sensor node based on message broadcasting from multiple surface nodes. With the time-of-arrival measurements from a DSP-based multicarrier modem, each sensor node localizes itself based on the travel time differences among multiple senders to the receiver. Using one-way message passing, such a solution can scale to accommodate a large number of nodes in a network. We consider the issue from not only the physical layer, but also at the node processing layer by incorporating a tracking solution. We present simulation results, testing results in a swimming pool featuring both stationary and moving receivers, and results from a lake test with a mobile receiver.

1. Introduction

Underwater localization is a topic of great interest and study, with increasingly hungry applications driving the need for better and better solutions [1, 2]. Several current systems feature augmented inertial navigation methods, which use filtering and tracking methods to provide corrections and improvements upon traditional onboard navigational equipments [3, 4]. Aside from these methods, there are several localization techniques based on acoustic signaling. The first is the long base line (LBL) system, where transponders are installed at the sea floor, and an underwater vehicle interrogates the transponders for round-trip delay estimation followed by triangulation [5]. LBL has good localization accuracy, but it requires long-time calibration. The second is the short base line (SBL) system, where a mothership moves above the underwater vehicle. The ship locates the vehicle using high-frequency directional emitters. The third approach is based on floating buoys [6, 7]. This system acts like a long base line system except that the reference points are surface buoys. There are commercial products—the GPS intelligent buoys (GIB)—that route signals from an underwater node to surface buoys [7], and using radio links, the surface buoys forward all information to a mothership, wherein the localization is performed. The floating buoys are easier to deploy and calibrate than LBL systems.

Several systems have been proposed which consider a network of underwater nodes instead of a single underwater vehicle or small group of nodes. These approaches typically use an anchor-client-based approach, where nodes systematically position themselves and disseminate this information with other nearby nodes in an effort to systematically localize the entire network [8–19]. However, few of these methods have been fully implemented in physical networking systems.

In this paper, we propose a new localization approach based on message broadcasts from multiple surface nodes, coupled with tracking algorithms and implemented on a physical system to provide a complete analysis. With the time-of-arrival measurements, the receiver computes its own localization based on the differences of the travel time among multiple senders to the receiver. We present one solution based on exhaustive search and the other based on the least-squares formulation [20]. By implementing the localization algorithms in the OFDM modem prototypes developed in [21], we have carried out tests in swimming pool and lake environments. With these point measurements, tracking analysis was also carried out on the collected data to further refine the position estimate.

Thus, we consider the problem not only at the modem physical layer with regards to timing and detection, but further analyze it in a single-point estimate, and ultimately combine the point estimates for a tracking implementation.

FIGURE 1: An underwater sensor network with multiple surface buoys.

In particular, we consider two tracking scenarios: a largely static scenario in which the nodes are assumed to be tethered or freely floating with no self-propulsion and a mobile scenario in which the object being tracked is assumed to make deliberate maneuvers and have some control of its motion, such as an autonomous underwater vehicle (AUV).

The advantage of the proposed localization method is that the broadcast messages can serve an arbitrary number of underwater nodes once they are in range, in contrast to many existing solutions which can only serve a small number of users.

The rest of the paper is organized as follows. We present the system overview in Section 2. Section 3 discusses the timing concerns of the physical layer. In Section 4, we consider the self-localization of a node for a single-point estimate. In Section 5, we present the series of tracking algorithms that are implemented. Sections 6, 7, and 8 contain simulation results and testing results from a swimming pool and lake, respectively. Conclusions are in Section 9.

2. System Overview

Figure 1 depicts the considered system setup, with several surface nodes and multiple underwater nodes. The surface nodes are equipped with satellite-based GPS receivers. Relying on the internal pulse provided by the GPS device that is accurate to within 1 microsecond GPS time, the surface nodes are assumed to be well synchronized. At predetermined intervals, the surface nodes sequentially broadcast their current location and time.

The underwater nodes within the broadcast range will detect a series of transmissions and decode those messages. By comparing the reception time with the transmission time encoded in the message, each underwater node can obtain estimates of the time of arrivals (or time of flights) of messages from different surface nodes, based on which it tries to compute its own position. Note that the broadcast from the surface to underwater nodes is a one-way transmission, that localization quality is independent of the number of underwater nodes in the network, and that there is no additional interference involved among different underwater nodes.

FIGURE 2: One round of surface buoy broadcasting leads to one point estimate, while tracking algorithms improve the localization accuracy when multiple-point estimates are available.

The overall scheme of localization refinement is presented in Figure 2. For one round of transmissions, the receiver obtains the travel times from all the surface buoys, based on which a point estimate is available. Once several of these broadcast periods have occurred, individual point estimates may be combined via tracking algorithms to form a more accurate understanding of the current node position. The algorithms at different stages are described in the following sections.

3. One-Way Travel Time Estimation

Let us focus on one receiver at position (x_r, y_r, z_r) . Suppose that there are N surface nodes, at positions (x_n, y_n, z_n) , $n = 1, \dots, N$. Let d_n denote the distance between the receiver node and the n th surface node

$$d_n = \sqrt{(x_r - x_n)^2 + (y_r - y_n)^2 + (z_r - z_n)^2}. \quad (1)$$

Without loss of generality, we set the first surface node at the origin, that is, $x_1 = y_1 = z_1 = 0$, such that

$$d_1^2 = x_r^2 + y_r^2 + z_r^2. \quad (2)$$

The actual time of arrival is $t_n = d_n/c$, where c is the sound propagation speed.

The transmission time is encoded at each broadcast message. The receiver needs to estimate the arrival time of each message to provide an estimate on the time of arrival t_n . In this paper, we use real-time DSP-based OFDM modem prototypes [21], as shown in Figure 3. First, the communication channel is monitored to detect signal arrivals, based on a background noise level monitoring performed by the modem at initialization. When a signal is detected, the correlation of the signal with a sliding window of itself is compared to determine when the level of peak correlation in the preamble of the message occurs, indicated by a plateau in the correlation. Once this plateau is selected, the time of arrival is coarsely estimated as having been approximately

FIGURE 3: The OFDM modem prototype with an attached GPS unit.

halfway during this plateau period [22]. Once coarse channel estimation has occurred, the preamble, which is entirely known to the receiver, is used to estimate the instantaneous underwater channel conditions, and from there, a more refined estimation of the time of arrival is performed via the modified page test as in [23].

Let $\hat{t}_n$ denote the estimate of t_n from the OFDM modem. It can be expressed as the sum of the real transmission propagation, the delay in signal processing at both transmitter and receiver b_n , and the estimation noise w_n :

$$\hat{t}_n = t_n + b_n + w_n. \quad (3)$$

Multiple tests of the OFDM modem reveal that the noise w_n has standard deviation on the order of 5–10 ms. (The bandwidth used by the OFDM modem is 6 kHz, leading to a baseband sampling interval of 1/6 ms.) On the other hand, the processing delay (bias) b_n has large magnitude, which might be on the order of 500 ms. However, tests have also shown that b_n is nearly identical across modems with similar hardware and operating software with the assumed GPS synchronization. Thus, we will treat b_n as a constant $b_n = b$ and present the localization algorithms based on

$$\hat{t}_n = t_n + b + w_n, \quad n = 1, \dots, N. \quad (4)$$

4. Localization

In each round of broadcasting from all surface nodes, a node collects N travel time measurements and can form a single-point estimate of its current position. This is accomplished by way of localization algorithms based on the intersection of spherical surfaces.

Since the bias b is unknown and usually large, time-of-arrival- (TOA-) based methods are not suitable. Instead, we use the time-difference-of-arrival (TDOA) method to cancel the common bias term b by forming

$$\Delta \hat{t}_{n1} = \hat{t}_n - \hat{t}_1, \quad n = 2, \dots, N. \quad (5)$$

The distance difference $d_{n1} = d_n - d_1$ is then estimated by

$$\hat{d}_{n1} = c \Delta \hat{t}_{n1}. \quad (6)$$

The TDOA method also corrects for clock skew alongside this bias term, due to the nature of the shared GPS clock. Each receiving node will have its own internal clock, which at some update period k will have drifted by an unknown, nonlinear skew factor $\phi(k)$. Each of the surface transmitters, however, will have the same clock skew, and due to the periodic corrections by the GPS clock, this value can be assumed approximately 0 for any period k . Thus, each transmission time can be represented as

$$\hat{t}_n = t_n + b + \phi(k) + w_n, \quad n = 1, \dots, N, \quad (7)$$

and again, by taking the difference of the time-of-arrival estimates, this common clock skew is eliminated from the timing estimate.

Next, we present the localization methods based on the exhaustive search and least-squares formulations.

4.1. Exhaustive Search. The individual time estimates $\hat{t}_n$ generally have correlated noise in the underwater channel. For simplicity, we assume instead that they are independent and identically distributed (i.i.d.) and pursue a maximum likelihood function

$$\min_{x_r, y_r, z_r} f(x_r, y_r, z_r) = \sum_{n=2}^N \left[c \Delta \hat{t}_{n1} - (d_n - d_1) \right]^2. \quad (8)$$

The solution to (8) is found by exhaustive search.

In subtracting a common random variable, $\hat{t}_1$, from all subsequent TOA estimates, we are correlating each measurement by a factor of approximately 1/2. As such, assuming again i.i.d. measurements, a differencing measurement bias modification can be made as follows:

$$\min_{x_r, y_r, z_r} f(x_r, y_r, z_r) = \left(c \hat{\mathbf{t}}_{\Delta} - \mathbf{d}_{\Delta} \right)^T \Sigma^{-1} \left(c \hat{\mathbf{t}}_{\Delta} - \mathbf{d}_{\Delta} \right), \quad (9)$$

where

$$\hat{\mathbf{t}}_{\Delta} = \begin{bmatrix} \Delta \hat{t}_{21} \\ \Delta \hat{t}_{31} \\ \vdots \\ \Delta \hat{t}_{n1} \end{bmatrix}, \quad \mathbf{d}_{\Delta} = \begin{bmatrix} d_2 - d_1 \\ d_3 - d_1 \\ \vdots \\ d_n - d_1 \end{bmatrix}, \quad (10)$$

and Σ is an $(N-1) \times (N-1)$ normalized covariance matrix

$$\Sigma = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} & \cdots & \frac{1}{2} \\ \frac{1}{2} & 1 & \frac{1}{2} & \cdots & \frac{1}{2} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \cdots & 1 \end{bmatrix}. \quad (11)$$

4.2. Least-Squares Solution. We use the least-squares solution from [20]. Since $d_n = d_{n1} + d_1$, we have

$$(d_{n1} + d_1)^2 = x_n^2 + y_n^2 + z_n^2 - 2x_n x_r - 2y_n y_r - 2z_n z_r + d_1^2, \quad (12)$$

which can be simplified as

$$x_n x_r + y_n y_r + z_n z_r = \frac{1}{2} ([x_n^2 + y_n^2 + z_n^2 - d_{n1}^2]) - d_{n1} d_1. \quad (13)$$

Define the following matrix and vectors:

$$\mathbf{H} = \begin{bmatrix} x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ \vdots & \vdots & \vdots \\ x_N & y_N & z_N \end{bmatrix}, \quad \mathbf{v} = \begin{bmatrix} -\hat{d}_{21} \\ -\hat{d}_{31} \\ \vdots \\ -\hat{d}_{N1} \end{bmatrix}, \quad (14)$$

$$\mathbf{u} = \frac{1}{2} \begin{bmatrix} x_2^2 + y_2^2 + z_2^2 - \hat{d}_{21}^2 \\ x_3^2 + y_3^2 + z_3^2 - \hat{d}_{31}^2 \\ \vdots \\ x_N^2 + y_N^2 + z_N^2 - \hat{d}_{N1}^2 \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} x_r \\ y_r \\ z_r \end{bmatrix}.$$

The least-squares solution can be obtained as

$$\hat{\mathbf{a}} = d_1 \mathbf{H}^\dagger \mathbf{v} + \mathbf{H}^\dagger \mathbf{u}, \quad (15)$$

where † stands for pseudoinverse. Substituting the entries of into (2) yields a quadratic equation for d_1 [20]. Solving for d_1 and substituting the positive root back into (15) provides the final solution for the receiver position $\mathbf{a}$. We compare this LS solution with our exhaustive search in Figure 4, where timing error was assumed Gaussian distributed with zero mean and a varied standard deviation.

5. Tracking Algorithms

To further reduce the localization error from a single-point measurement, tracking algorithms can be implemented to combine the knowledge of multiple measurements into a more accurate position estimate.

In deciding which tracking approach would be best, we first consider the scenarios in which the node is being localized. There are two distinct modes in which underwater nodes typically move: either passively, with the water currents as a free-floating node, or actively as an underwater vehicle such as an AUV. Both are characterized primarily by long periods of relatively straight motion at a fairly constant speed. Typically, AUV motion differs in that at certain random intervals, it will change direction according to operator or preprogrammed instruction. Most search patterns for AUVs are defined by spiral paths, or by rectangular search grids. In either case, the vehicle is likely to alter its direction by way of a continuous turn; that is, to make a turn at a fixed angular velocity until the desired heading is achieved (or in the case of a spiral, until the search area is exhausted).

5.1. Kalman Filter. For simple motion models, characterized by long periods of motion free from maneuvers, a Kalman filter is the best linear state estimator for a Gaussian initial state assumption [24].

FIGURE 4: Root-mean-squared (RMS) localization error as a function of the standard deviation of the distance measurements. (Two curves overlap and are not easily distinguishable).

In the KF, we choose to model the movement of the node as set of discrete white noise acceleration (“kinematic”) models [24], with a separate model for each possible direction, that is, x , y , and z . As such, the state equation for the Kalman filter at time index $k + 1$ based on information from time step k becomes

$$\mathbf{L}(k+1) = \mathbf{F}(k)\mathbf{L}(k) + \mathbf{v}(k), \quad (16)$$

with measurement

$$z(k+1) = \mathbf{H}(k+1)\mathbf{L}(k+1) + w(k+1), \quad (17)$$

where

$$\mathbf{F}(k) = \begin{bmatrix} 1 & \tau & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & \tau & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (18)$$

$$\mathbf{H}(k+1) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \quad (19)$$

$\mathbf{v}(k)$ is process noise, $w(k)$ is measurement noise, and τ is the sampling interval of the discrete model in seconds.

The state covariance is modeled as

$$\mathbf{P}(k+1 | k) = \mathbf{F}(k)\mathbf{P}(k | k)\mathbf{F}(k)^T + \mathbf{Q}(k). \quad (20)$$

The corresponding process noise has a covariance given as

$$\mathbf{Q} = \begin{bmatrix} \frac{1}{4}\tau^4 & \frac{1}{2}\tau^3 & 0 & 0 & 0 & 0 \\ \frac{1}{2}\tau^3 & \tau^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{4}\tau^4 & \frac{1}{2}\tau^3 & 0 & 0 \\ 0 & 0 & \frac{1}{2}\tau^3 & \tau^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{4}\tau^4 & \frac{1}{2}\tau^3 \\ 0 & 0 & 0 & 0 & \frac{1}{2}\tau^3 & \tau^2 \end{bmatrix} \sigma_v^2. \quad (21)$$

Here, σ_v is a design parameter that is chosen to match the most likely level of process noise to be experienced by the object in question, which is to say it controls how much the model anticipates the object to maneuver. Given that the object in question is likely to be either stationary or altering its velocity at a slow, steady rate, a process noise level of $\sigma_v = 0.5 \text{ m/s}^2$ was selected to best emulate this behavior. The filter was initialized with two-point differencing [24].

5.2. Probabilistic Data Association Filter. During analysis of the performance of the KF, it became apparent that within a tracking window, there were point estimates which would appear as outliers by a considerable margin. On closer inspection, it was found that these were likely the result of one or more transmissions during which the direct-path signal propagation had been blocked, and a bounce was instead detected and treated as the direct path. Due to the assumption implicit to the KF approach that all of our messages are the direct-path propagation, this resulted in a drastic alteration of the point estimate, to the point where it could be classified as a false measurement. In that context, the probabilistic data association filter (PDAF) offers an improved performance over the standard KF, by allowing outlier estimates such as these to be ignored as false-alarm or clutter detections [25]. The PDAF is very similar to the KF in terms of state equations, presented here for measurement $k + 1$:

$$\mathbf{L}(k + 1) = \mathbf{F}(k)\mathbf{L}(k) + \mathbf{v}(k), \quad (22)$$

with measurement

$$z(k + 1) = \mathbf{H}(k + 1)\mathbf{L}(k + 1) + w(k + 1), \quad (23)$$

where $\mathbf{F}(k)$ and $\mathbf{H}(k + 1)$ are given in (18) and (19), respectively.

The state covariance is modeled similarly as

$$\mathbf{P}(k + 1 | k) = \mathbf{F}(k)\mathbf{P}(k | k)\mathbf{F}(k)^T + \mathbf{Q}(k). \quad (24)$$

The difference is on how to compute $\mathbf{P}(k | k)$. First, let P_D denote the probability of detection, which is a design parameter. Operating under the assumption that the measurement is always gated, define

$$b = 2 \left(\frac{1 - P_D}{P_D} \right). \quad (25)$$

Define the following variables and matrices:

$$\begin{aligned} \mathbf{v}(k) &= z(k) - \mathbf{H}(k)\mathbf{L}(k), \\ e(k) &= e^{-(1/2)\mathbf{v}(k)^T \mathbf{S}(k)^{-1} \mathbf{v}(k)}, \\ \mathbf{S}(k) &= \mathbf{H}(k)\mathbf{P}(k | k - 1)\mathbf{H}(k)^T + \mathbf{R}, \\ \mathbf{W}(k) &= \mathbf{P}(k | k - 1)\mathbf{H}(k)^T \mathbf{S}(k)^{-1}, \end{aligned} \quad (26)$$

where $\mathbf{R}$ is the observation noise covariance. The probability of no correct measurement available is

$$\beta_0(k) = \frac{b}{b + e(k)}. \quad (27)$$

The probability of a correct measurement is

$$\beta_1(k) = \frac{e(k)}{b + e(k)}. \quad (28)$$

Further defining

$$\begin{aligned} \mathbf{P}^c(k | k) &= \mathbf{P}(k | k - 1) - \mathbf{W}(k)\mathbf{S}(k)\mathbf{W}(k)^T, \\ \tilde{\mathbf{P}}(k) &= \mathbf{W}(k) \left(\beta_1(k)\mathbf{v}(k)\mathbf{v}(k)^T - \mathbf{v}(k)\mathbf{v}(k)^T \right) \mathbf{W}(k)^T, \end{aligned} \quad (29)$$

the covariance matrix update is as follows:

$$\mathbf{P}(k | k) = \beta_0(k)\mathbf{P}(k | k - 1) + \beta_1(k)\mathbf{P}^c(k | k) + \tilde{\mathbf{P}}(k). \quad (30)$$

We are assuming that the measurement is always gated, and that there is only a single target and a single measurement at each time step. Accordingly, the P_D is then the probability that the current measurement is a valid estimate of the node being tracked. Based on experimental data, the number of “false detection” measurements was around 5% of the total samples, and so a value of 0.95 was selected for P_D .

5.3. Interacting Multiple Model (IMM) Filter. For the more complex motion of an active underwater node, an interacting multiple model filter (IMM) was implemented, as the expected maneuvering index of underwater vehicles can easily exceed the threshold for which a single linear filter is likely to have any benefit. To this end, the IMM was a simple two-model filter, with a single, linear, low process noise ($\sigma_v = 0.05 \text{ m/s}^2$) KF to account for the straight motion travel, and an extended Kalman filter (EKF), configured in a coordinated-turn mode [26]. This validity of the coordinated turn assumption is dependent on the scenario in question,

though given the previously described search patterns, it should be sufficiently accurate [27].

The linear KF uses similar system equations as given previously, augmented with an additional column and row of zeros in order to accommodate the use of the EKF's additional state in the IMM. The EKF in this problem uses one of two sets of state equations: the first set is an approximation used when the predicted coordinated turn rate is near 0 ($\hat{\Omega}(k) \approx 0$), and the second set is used when the predicted coordinated turn rate is greater than some detection threshold ($|\hat{\Omega}(k)| > 0$) [24].

The first set of EKF state equation modifications ($\hat{\Omega}(k) \approx 0$) is as follows:

$$L(k+1) = \mathbf{F}_L(k)L(k) + v(k), \quad (31)$$

where

$$\mathbf{F}_L(k) = \begin{bmatrix} 1 & \tau & 0 & 0 & 0 & 0 & -\frac{1}{2}\tau^2\hat{\eta}(k) \\ 0 & 1 & 0 & 0 & 0 & 0 & -\tau\hat{\eta}(k) \\ 0 & 0 & 1 & \tau & 0 & 0 & \frac{1}{2}\tau^2\hat{\xi}(k) \\ 0 & 0 & 0 & 1 & 0 & 0 & \tau\hat{\xi}(k) \\ 0 & 0 & 0 & 0 & 1 & \tau & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (32)$$

where η and ξ represent the x and y directions, respectively, and we denote $\hat{\eta}$ as the velocity component in the η direction. When $|\hat{\Omega}(k)| > 0$,

$$\mathbf{F}_L(k) = \begin{bmatrix} 1 & \frac{\sin \hat{\Omega}(k)}{\hat{\Omega}(k)}\tau & 0 & -\frac{1-\cos \hat{\Omega}(k)}{\hat{\Omega}(k)}\tau & 0 & 0 & f_{\Omega,1(k)} \\ 0 & \cos \hat{\Omega}(k)\tau & 0 & -\sin \hat{\Omega}(k)\tau & 0 & 0 & f_{\Omega,2(k)} \\ 0 & \frac{1-\cos \hat{\Omega}(k)}{\hat{\Omega}(k)}\tau & 1 & \frac{\sin \hat{\Omega}(k)}{\hat{\Omega}(k)}\tau & 0 & 0 & f_{\Omega,3(k)} \\ 0 & \sin \hat{\Omega}(k)\tau & 0 & \cos \hat{\Omega}(k)\tau & 0 & 0 & f_{\Omega,4(k)} \\ 0 & 0 & 0 & 0 & 1 & \tau & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (33)$$

where the partial derivatives $f_{\Omega,1(k)}, \dots, f_{\Omega,4(k)}$ are found as:

$$\begin{aligned} f_{\Omega,1(k)} &= \frac{(\cos \hat{\Omega}(k)\tau)\tau\hat{\xi}(k)}{\hat{\Omega}(k)} - \frac{(\sin \hat{\Omega}(k)\tau)\hat{\xi}(k)}{\hat{\Omega}(k)^2} \\ &\quad - \frac{(\sin \hat{\Omega}(k)\tau)\tau\hat{\eta}(k)}{\hat{\Omega}(k)} - \frac{(-1 + \cos \hat{\Omega}(k)\tau)\hat{\eta}(k)}{\hat{\Omega}(k)^2}, \\ f_{\Omega,2(k)} &= -(\sin \hat{\Omega}(k)\tau)\tau\hat{\xi}(k) - (\cos \hat{\Omega}(k)\tau)\tau\hat{\eta}(k), \\ f_{\Omega,3(k)} &= \frac{(\sin \hat{\Omega}(k)\tau)\tau\hat{\xi}(k)}{\hat{\Omega}(k)} - \frac{(1 - \cos \hat{\Omega}(k)\tau)\hat{\xi}(k)}{\hat{\Omega}(k)^2} \\ &\quad + \frac{(\cos \hat{\Omega}(k)\tau)\tau\hat{\eta}(k)}{\hat{\Omega}(k)} - \frac{(\sin \hat{\Omega}(k)\tau)\hat{\eta}(k)}{\hat{\Omega}(k)^2}, \\ f_{\Omega,4(k)} &= (\cos \hat{\Omega}(k)\tau)\tau\hat{\xi}(k) - (\sin \hat{\Omega}(k)\tau)\tau\hat{\eta}(k). \end{aligned} \quad (34)$$

In both cases, the process noise covariance is determined in the following state equations:

$$\mathbf{P}(k+1 | k) = \mathbf{F}_L(k)\mathbf{P}(k | k)\mathbf{F}_L(k)' + \Gamma_{\text{EKF}}\mathbf{Q}(k)\Gamma_{\text{EKF}}', \quad (35)$$

where

$$\Gamma_{\text{EKF}} = \begin{bmatrix} \frac{1}{2}\tau^2 & 0 & 0 & 0 \\ \tau & 0 & 0 & 0 \\ 0 & \frac{1}{2}\tau^2 & 0 & 0 \\ 0 & \tau & 0 & 0 \\ 0 & 0 & \frac{1}{2}\tau^2 & 0 \\ 0 & 0 & \tau & 0 \\ 0 & 0 & 0 & \tau \end{bmatrix}. \quad (36)$$

From our assumptions of AUV motion, the value of $\mathbf{Q}(k)$ was selected as:

$$\mathbf{Q}(k) = \begin{bmatrix} r_s & 0 & 0 & 0 \\ 0 & r_s & 0 & 0 \\ 0 & 0 & r_s & 0 \\ 0 & 0 & 0 & r_d \end{bmatrix}, \quad (37)$$

where $r_s = (1.25 \text{ m/s}^2)^2$ and $r_d = (0.3\pi/180 \text{ rad})^2$.

The IMM-CT is outlined in Figure 5. It combines a set of filters (in this case a KF and EKF) and mixes the weighted previous state estimates to determine the current hypothesis of each of the filters. The linear KF is designed as described previously, whereas the nonlinear EKF has a different set of model selection parameters which define how it interprets large differences in the measurements. In particular, its covariance matrix describes how much variation is expected during a coordinated maneuver in terms of the angular velocity, represented as two directional speed components and a rate of angular change component.

5.4. Computational Complexity. For a Kalman filter, the computational complexity is approximately of $O(n^3)$, where

FIGURE 5: Outline of an interacting multiple-model filter.

$n = \max(n_x, n_z)$, n_x is the number of state variables, and n_z is the number of measurements [24]. It follows that the PDAF, having a similar number of calculations, is also nearly approximated by $O(n^3)$. The IMM-CT, which is dominated by the computation of multiple filters in parallel, is approximately $O(pn^3)$, where p is the number of filters used in the IMM which are not being processed in a parallel fashion.

6. Simulation Results

We first carry out simulations using a simple noise model to generate the TOA measurements and evaluate the localization accuracy. For simplicity, z is assumed to be known, and we only solve for x and y coordinates. Four transmitters are placed on a square grid with coordinates $(0, 0)$, $(100, 0)$, $(0, 100)$, and $(100, 100)$. One receiver is placed at the $(0, 50)$ point and moves at a constant rate of 0.125 m/s parallel to the x -axis.

The TOAs are generated according to (3) where b is a fixed large bias, and w_n is i.i.d. zero-mean white Gaussian noise with standard deviation of 7.5 m. Position updates were taken every 16 seconds.

The localization position error is shown in Figure 6 as a function of total number of measurements acquired. We see that the Kalman filter clearly outperforms the point estimation based on exhaustive search (the LS solution has similar performance as the exhaustive search). We have also tested with a varied clock bias (not depicted), which had no effect upon the position error, as expected.

In addition to the Kalman filter, simulations for the proposed IMM-CT were also run, using the relatively

challenging scenario presented in Figure 7, with the corresponding RMS position error given by Figure 8. The dashed lines depicted in the figure indicate the beginning or end of one of the maneuvers from Figure 7. As can be seen in Figure 8, the point estimates are drastically improved by all three of the trackers, with the KF and PDAF slightly outperforming the IMM-CT on the straight-path portion after exiting a maneuver. There was no noticeable difference in the performance of the KF and PDAF, which is to be expected, as the scenario did not feature any indirect path propagations.

7. Pool Tests

We carried out tests in a standard competitive athletic swimming pool at The University of Connecticut, Storrs, whose dimensions are perfectly known. These tests did not use the GPS capabilities of the nodes, due to the limitation of the GPS receivers indoors. The nodes were fixed to the corner locations of the pool, such that their locations are measured accurately. The receiver was positioned approximately in the center of the pool, as outlined in Figure 9. All the transducers are placed about 1 m below the surface. The pool has a depth about 2 m. Stationary test was conducted during March 2010, while the mobile test was conducted during December 2010.

7.1. Test Case 1 (Stationary Test, March 2010). During the test, not all the messages from the transmitters were decoded correctly. For this reason, we use the data set with at least three measurements within one cycle of broadcasting from

FIGURE 6: Root-mean-squared (RMS) localization error as a function of the number of measurements acquired as the receiver moved in a straight line.

FIGURE 7: Simulation path for the IMM-CT, with distances in meters.

the four surface nodes. The favorable geometry and the known value of z allow an estimate based on only three surface nodes. The location estimates by the exhaustive search method are shown in Figure 9, and those by the LS method are shown in Figure 10. We see that the LS estimates from these data sets are biased.

Although advanced algorithms could be applied to fuse the data from multiple data sets, here we simply average the location estimates from multiple data sets. As more data sets are available, the localization accuracy improves, as shown in Figure 11. A localization error of about 5 m is achieved with about 10 data sets.

FIGURE 8: Root-mean-squared (RMS) localization error as a function of the number of measurements acquired as the receiver moved in the scenario.

FIGURE 9: Node deployment during March 2010 pool test. The transmitters are denoted by squares, and the receiver is denoted by the diamond. The scattered plus signs are the estimates by the exhaustive search method.

7.2. Test Case 2 (Mobile Test, December 2010). For the moving test in the pool, a simple straight-line maneuver was carried out. All of the previous conditions apply from the stationary pool test, including the use of only three nodes for localizing purposes. The scatter plot and approximate trajectory of the mobile node along with the four stationary transmitter locations are depicted in Figure 12. There was a significant upgrade in the hardware and software used for the second test, which resulted in a large reduction in the overall error, as can be seen in Figure 13. The high initial error of the Kalman filter can be attributed to the very low process noise of the filter, combined with a very bad initialization

FIGURE 10: Node deployment during March 2010 pool test. The transmitters are denoted by squares, and the receiver is denoted by the diamond. The scattered plus signs are the estimates by the least-squares method.

FIGURE 11: Localization error during March 2010 pool test.

which lead the filter to believe the node was traveling almost horizontally. Additionally, the measurements overall had a very low level of error, and the tracking system therefore could offer little improvement.

8. Field Test in a Local Lake

The lake testing was performed in Mansfield Hollow lake, located in Mansfield, CT during August 2011. The average depth of the area in which the testing occurred is approximately 2.5 m, with minor variations of approximately 0.5 m. During the test, wind speed was minimal, typically less than 5 mph. The nodes were positioned in a roughly square formation, with an average separation of 110 m, as depicted in Figure 14. The receiver was attached to a boat which would

FIGURE 12: Localization error in the swimming pool.

FIGURE 13: Localization error in the swimming pool.

freely float inside of this node formation for the duration of the testing.

For this test, the ground truth was determined via an onboard GPS device which would record the position of the receiver whenever a message was received. The data for a single run is presented, during which the boat moved at approximately 1 knot while moving along a slightly curved trajectory (approximately 10°). Note, however, that at a certain point during the test, the ground-truth GPS stopped updating its position, while the boat continued to move. In order to correct for this, the remaining ground truth was extrapolated from the initial GPS measurements. This introduces a nonnegligible amount of uncertainty, but still enables conclusions to be drawn regarding the behavior of the tracking algorithms. The approximate trajectory and measurements of the boat along with relative node positions are depicted in Figure 15.

FIGURE 14: Approximate node deployment of August 2011 test via Google Maps.

FIGURE 15: Plot of measurements and ground truth of lake test.

As can be seen in Figure 16, the tracking algorithms smooth the error out over the course of the maneuver and eventually reduce the overall error by a slight amount. Over the whole period, approximated error never exceeded a combined 5 m. While the raw measurements do exceed the trackers at one particular stretch, this is a combination of a rather small process noise coupled with a set of increasingly accurate measurements and is considered an acceptable trade-off due to the nature of node motion.

On inspection, the PDAF does not perform much better than the Kalman filter in most cases. There was one measurement where it clearly offered an improvement, but overall they did not differentiate much in performance. In almost all cases, the IMM-CT was superior to both filters.

9. Conclusion

In this paper, we presented an underwater localization solution based on one-way message broadcasting from multiple surface nodes. In addition to simulation results, we provided

FIGURE 16: Localization error during second lake test.

testing results in a swimming pool and in a local lake. Future work would involve large-scale field tests featuring multiple nodes, as well as sea deployments over larger distances. Those will be pursued when there are suitable opportunities.

Acknowledgments

This work was supported by the ONR Grant N00014-09-1-0704 (PECASE) and the NSF Grant CNS-0821597 (MRI). P. Carroll is supported by a GANNN Fellowship from the Department of Education.

References

- [1] V. Chandrasekhar, W. K. G. Seah, Y. S. Choo, and H. V. Ee, "Localization in underwater sensor networks—survey and challenges," in *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom '06)*, pp. 33–40, New York, NY, USA, September 2006.
- [2] H. Tan, R. Diamant, W. Seah, and M. Waldmeyer, "A survey of techniques and challenges in underwater localization," *Ocean Engineering*, vol. 38, no. 14–15, pp. 1663–1676, 2011.
- [3] C. M. Lee, P. M. Lee, S. W. Hong, S. M. Kim, and W. Seong, "Underwater navigation system based on inertial sensor and doppler velocity log using indirect feedback Kalman filter," *International Journal of Offshore and Polar Engineering*, vol. 15, no. 2, pp. 88–95, 2005.
- [4] R. Hartman, W. Hawkinson, and K. Sweeney, "Tactical Underwater Navigation System (TUNS)," in *Proceedings of the IEEE/ION Position, Location and Navigation Symposium, PLANS*, pp. 898–911, Fairfax, Va, USA, May 2008.
- [5] R. Stuart, "Acoustic digital spread spectrum: an enabling technology," *Sea Technology*, vol. 46, no. 10, pp. 15–20, 2005.
- [6] A. Caiti, A. Garulli, F. Livide, and D. Prattichizzo, "Localization of autonomous underwater vehicles by floating acoustic buoys: a set-membership approach," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 1, pp. 140–152, 2005.
- [7] C. Bechaz and H. Thomas, "GIB system: the underwater GPS solution," in *Proceedings of the ECUE*, May 2001.

- [8] Z. Zhou, J. H. Cui, and S. Zhou, "Efficient localization for large-scale underwater sensor networks," *Ad Hoc Networks*, vol. 8, no. 3, pp. 267–279, 2010.
- [9] M. Erol-Kantarci, H. T. Mouftah, and S. Oktug, "Localization techniques for underwater acoustic sensor networks," *IEEE Communications Magazine*, vol. 48, no. 12, pp. 152–158, 2010.
- [10] D. Mirza and C. Schurgers, "Motion-aware self-localization for underwater networks," in *Proceedings of the ACM International Conference on Mobile Computing and Networking (MobiCom '08)*, pp. 51–58, San Francisco, Calif, USA, September 2008.
- [11] M. T. Isik and O. B. Akan, "A three dimensional localization algorithm for underwater acoustic sensor networks," *IEEE Transactions on Wireless Communications*, vol. 8, no. 9, pp. 4457–4463, 2009.
- [12] R. M. Eustice, L. L. Whitcomb, H. Singh, and M. Grund, "Experimental results in synchronous-clock one-way-travel-time acoustic navigation for autonomous underwater vehicles," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pp. 4257–4264, April 2007.
- [13] S. E. Webster, R. M. Eustice, H. Singh, and L. L. Whitcomb, "Preliminary deep water results in single-beacon one-way-travel-time acoustic navigation for underwater vehicles," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '09)*, pp. 2053–2060, October 2009.
- [14] R. Diamant and L. Lampe, "Underwater localization with time-synchronization and propagation speed uncertainties," in *Proceedings of the 8th Workshop on Positioning Navigation and Communication (WPNC '11)*, April 2011.
- [15] X. Tan and J. Li, "Cooperative positioning in underwater sensor networks," *IEEE Transactions on Signal Processing*, vol. 58, no. 11, pp. 5860–5871, 2010.
- [16] W. Cheng, A. Y. Teymorian, L. Ma, X. Cheng, X. Lu, and Z. Lu, "Underwater localization in sparse 3D acoustic sensor networks," in *Proceedings of the 27th IEEE Communications Society Conference on Computer Communications (INFOCOM '08)*, pp. 798–806, April 2008.
- [17] X. Cheng, H. Shu, Q. Liang, and D. H. C. Du, "Silent positioning in underwater acoustic sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 3, pp. 1756–1766, 2008.
- [18] L. F. M. Vieira, U. Lee, and M. Gerla, "Phero-trail: a bio-inspired location service for mobile underwater sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 4, pp. 553–563, 2010.
- [19] M. Erol-Kantarci, S. Oktug, L. Vieira, and M. Gerla, "Performance evaluation of distributed localization techniques for mobile underwater acoustic sensor networks," *Ad Hoc Networks*, vol. 9, no. 1, pp. 61–72, 2011.
- [20] G. Mellen, M. Pachter, and J. Raquet, "Closed-form solution for determining emitter location using time difference of arrival measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 3, pp. 1056–1058, 2003.
- [21] H. Yan, L. Wan, S. Zhou et al., "DSP based receiver implementation for OFDM acoustic modems," *Elsevier Journal on Physical Communication*, vol. 5, no. 1, pp. 22–32, 2012.
- [22] S. F. Mason, C. R. Berger, S. Zhou, and P. Willett, "Detection, synchronization, and doppler scale estimation with multicarrier waveforms in underwater acoustic communication," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 9, pp. 1638–1649, 2008.
- [23] D. A. Abraham and P. K. Willett, "Active sonar detection in shallow water using the page test," *IEEE Journal of Oceanic Engineering*, vol. 27, no. 1, pp. 35–46, 2002.
- [24] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, John Wiley & Sons, New York, NY, USA, 2001.
- [25] Y. Bar-Shalom, P. Willett, and X. Tian, *Tracking and Data Fusion: A Handbook of Algorithms*, YBS Publishing, 2011.
- [26] H. Wang, T. Kirubarajan, and Y. Bar-Shalom, "Precision large scale air traffic surveillance using IMM/assignment estimators," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 1, pp. 255–266, 1999.
- [27] B. Ferreira, A. Matos, and N. Cruz, "Estimation approach for AUV navigation using a single acoustic beacon," *Sea Technology*, vol. 51, no. 12, pp. 54–58, 2010.