Tracking Aquatic Invaders: Autonomous Robots for Invasive Fish

Pratap Tokekar, Elliot Branson, Joshua Vander Hook, Volkan Isler

***Abstract*—Carp is a highly invasive, bottom-feeding fish which pollutes and dominates lakes by releasing harmful nutrients. Recently, biologists started studying carp behavior by tagging the fish with radio-emitters. The biologists search for and localize the radio-tagged fish manually using a GPS and a directional antenna. We are developing a novel robotic sensor system in which the human effort is replaced by autonomous robots capable of finding and tracking tagged carp.**

In this paper, we report the current state of our system. We present a new coverage algorithm for finding tagged fish and active localization algorithms for localizing them precisely. In addition to theoretical analysis and simulation results, we report results from field experiments.

I. Introduction

Invasive fish, such as the common carp, pose a major threat to the ecological integrity of freshwater ecosystems around the world. Presently, these fish are controlled using non-specific toxins which are expensive, ecologically damaging, and impractical in large rivers and lakes. Recent studies in small lakes have established that common carp aggregate densely at certain times and regions within the lakes [1]. Their population can be controlled by targeting these aggregations using netting. To predict the locations of aggregations within a lake, biologists surgically implant radio tags on some fish, and use radio antennas to track them periodically (Figure 1). Manually locating tagged fish in large, turbid bodies of water is difficult and labor-intensive. Our goal is to build a robotic system to automate this tedious manual task.

In our previous work [2], we demonstrated the feasibility of finding radio-tagged carp using a custom-made, catamaran-style autonomous robot. Our current system features a new boat (Figure 1) and improved system architecture, presented in Section III.

We perform the task of locating tagged fish in two phases: *search* and *localization*. The goal in the search phase is to find a location of the robot from which the tag can be sensed. In the localization phase, the goal is to accurately estimate the location of the fish. We developed a new search algorithm suitable for finding tagged carp under the assumption that they loiter in their home ranges for long periods of time [3]. In Section IV we present this algorithm, its analysis, and results from a field trial.

In Section V, we first present a method to robustly obtain a bearing measurement towards a tag using a directional radio antenna. We then propose three strategies to compute locations from which to obtain bearing measurements from, so as to localize stationary tagged fish precisely. These strategies were previously presented in [4] with experiments performed for a ground robot. We report additional experiments using our boat. Finally in Section VI, we report results from field experiments where the robot executed both search and localization phases for locating reference tags and radio-tagged fish.

The authors are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA. {tokekar,ebranson,jvander,isler}@cs.umn.edu

II. Related Work

Recent years have witnessed the development of many environmental monitoring systems which use aquatic robots. Applications include tracking dynamic phytoplankton [5] and collecting biological and environmental data from stationary sensors [6], [7]. The recent survey by Dunbabin and Marques [8] provides an excellent overview of such systems.

The search problem is closely related to the robotic coverage problem which has received significant attention (c.f. [9]). We introduce a new version of the problem in which regions scattered in the lake must be covered, as opposed to the whole lake. We present a constant-factor polynomial-time approximation algorithm for this problem (see e.g. [10] for more on approximation algorithms).

In the localization phase, we perform active, bearing-only target localization. In one of the earlier works on this problem, Hammel et al. [11] used the determinant of the Fisher Information Matrix (FIM) as the objective function to be maximized, and numerically computed an optimal open-loop trajectory for a robot in the case where measurements are obtained continuously. The resulting trajectory follows a spiral shape, but is an open-loop trajectory which does not depend on the actual measurements the robot obtains. Frew [12] presented a feedback strategy for tracking targets with bearing measurements obtained using monocular vision. The strategy is based on a state-exploration tree, and a trajectory is obtained using breadth-first search for the minimum uncertainty. Recently, Zhou and Roumeliotis [13] considered the active localization problem for a team of robots capable of taking range and/or bearing measurements towards a moving target. They consider maximum speed and minimum sensing range constraints, and plan for the next best sensing location using the trace of the posterior covariance matrix as the uncertainty measure.

What differentiates our problem from these works is that each measurement in our system takes a long time, and the uncertainty in measurements is considerably larger. We address these factors in our strategy by using the best worst-case behavior as our objective function, and limit the number of measurements as part of our planning process.

We start with an overview of the system and present our search and active localization strategies in Sections IV and V respectively.

Fig. 1. **Left:** Fish biologists manually tracking carp. **Center:** Targeted removal of the carp aggregation in Lake Lucy, MN, USA, using a large under-ice seine. Over 95% of all carp in this lake were captured in 4 hours. Photos courtesy of Peter Sorensen. **Right:** Robotic boat during coverage experiments at Lake Keller, MN, USA. A directional loop antenna can be seen mounted on a pan-tilt unit.

III. System Description

Our system consists of an autonomous robotic boat with on-board sensors, radio tags and receiver, and a directional antenna. We discuss each of these in turn.

A. Robotic Boat

The hull of our robotic platform (Figure 1) is the *QBoat* designed by Oceanscience[1]. The QBoat has dimensions of 182cm $\times$ 71cm, and can carry a payload of approximately 40kg. The boat is capable of a maximum speed of 1.65m/s. An on-board 12V, 30Ah NiMH battery allows approximately two hours of continuous operation.

The electronics on-board the boat (Figure 2) consist of a laptop running high-level software, an Atmel micro-controller board for low level interface, radio receiver equipment (described in the following section), a digital compass and a Garmin 18x Global Positioning System (GPS) unit. We also have a remote override radio control system that can directly control the boat, if desired.

We have a modular software architecture based on the Robot Operating System (ROS), comprising of packages for navigation, localization, simulation, and reading sensor data. ROS allows remote monitoring of data from another computer on the shore via an ad-hoc network formed with the on-board laptop. At the core of the navigation package is the implementation of a waypoint navigation algorithm reported in our previous work [2], with an EKF-based localization routine similar to the catamaran solution presented in [14]. The electronics and software can be easily transferred to other platforms. In fact, in winter we move the system to a ground robot used on frozen lakes [4].

B. Radio Tag and Receivers

For sensing the fish, we use radio tags manufactured by Advanced Telemetry Systems (ATS)[2]. A complete fish sensing system by ATS consists of radio tags, a loop antenna connected to a radio receiver, and a data logger which provides the computer interface for the receiver. Each radio tag emits a short pulse roughly once per second. The radio antenna (shown on top of the boat in Figure 1) is used to detect these pulses.

The radio receiver reports the received signal strength of the pulse. However, the signal strength is not directly useful in determining the distance to the tag, as it also depends on the unknown depth of the fish, conductivity of the water, and the remaining battery-life of the tag. Therefore we rely only on the directional nature of the antenna, and obtain a bearing measurement towards the tag. Our method for estimating the bearing is presented in Section V.

[1] http://www.oceanscience.com/

[2] http://atstrack.com

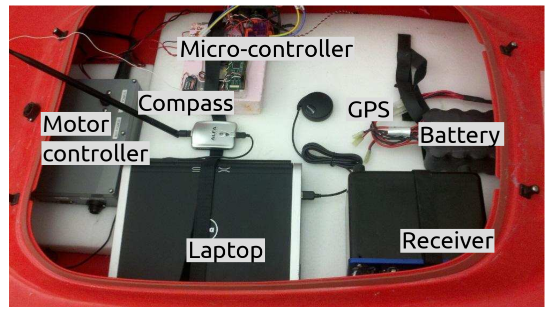


Fig. 2. Top view of the electronics compartment: On-board electronics comprises of a laptop, GPS, micro-controller board, batteries, digital compass, motor controller, and radio receiver.

The tag on each fish is assigned a unique frequency. The receiver can be programmed to tune in on one or more frequencies. In the search phase, we program the receiver to loop through a list of frequencies of tagged fish present in the lake. To reliably detect a pulse from a tag, the receiver needs to stay tuned on the corresponding frequency for more than one second since the tags emit pulses at about 1Hz. After detecting a radio tagged-fish in the search phase, we program the receiver to tune only to the corresponding frequency and switch to the localization phase. In the search phase, described in the next section, we do not rotate the antenna or obtain bearing measurements, since the goal of this phase is only to detect the presence of a tag.

IV. Search and Coverage

The first phase of our monitoring task is to search for all tagged fish in the lake. One of the current models for carp mobility suggests that each fish moves to its preferred region in the lake during day time, and remains in that region for long periods of time [3]. To increase the efficiency of our system, we restrict the search to only those regions of the lake which are likely to contain the fish (Figure 3). We assume that these regions are connected in the sense that there is a path between any two points. We also assume the fish remain stationary

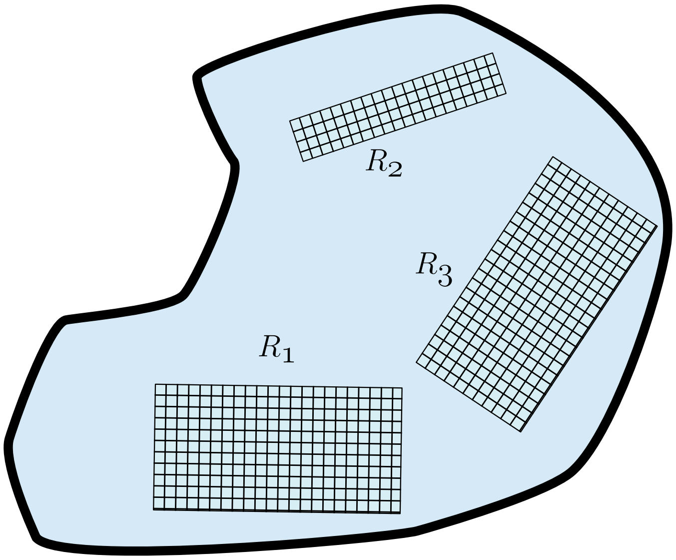


Fig. 3. We incorporate domain knowledge to restrict the search region for the fish to a given set of regions, e.g. $R = \{R_1, R_2, R_3\}$. A simple approach of discretizing these regions and finding a TSP tour becomes infeasible when the regions are large.

for the duration of covering a region and reduce the search problem to a coverage problem.

The radio antenna has limited sensing range. A given robot path is said to cover a point if the point lies within the sensing range of the robot at some instance along the path. The coverage problem can be defined as follows: *Given a set of connected regions $R = \{R_1, R_2, \ldots, R_n\}$, find a minimum length tour which covers every point in each region $R_i \in R$.*

A possible approach for solving this problem is based on the Traveling Salesperson Problem (TSP). We can discretize each region with a resolution dependent on the sensing range (Figure 3), and compute a TSP tour of this set of points. However, approximation algorithms for TSP usually require a metric graph, which is typically represented as a complete graph whose vertex set is the point set to be covered. In such a representation the number of edges is quadratic in the number of points. As the lake size or the sampling granularity increases, maintaining and operating on this large graph can become infeasible.

The coverage problem defined above is a generalization of Euclidean TSP and consequently NP-Hard. Next, we present a general approach for solving the coverage problem and show that the length of the path using this approach does not deviate significantly from the length of an optimal tour.

A. Algorithm Description

Our general approach is composed of two steps. First, we compute a tour τ_R that visits all the regions in R exactly once. We say that region R_i is *visited* if *any* point in R_i is visited by the tour. The tour τ_R imposes an ordering on the regions, and defines (possibly same) entry and exit points for each region. The entry and exit points are where τ_R intersects a region for the first and the last time. Such a tour is not necessarily a solution to the original problem, since it is not guaranteed to cover all points in each region. We compute a coverage tour C_{R_i} for each region $R_i \in R$ independently, starting and finishing at the entry and exit points for each R_i. The final tour τ is constructed by augmenting the coverage tours of each region to the visiting tour τ_R.

We now analyze the performance of this algorithm. Let OPT be an optimal tour which visits *and* covers all the regions in R in minimum time. Let τ_R^* be the optimal tour which visits all the regions in R. Since OPT also visits all the regions in R, we have $|OPT| \geq |\tau_R^*|$, where $|\tau|$ denotes the length of tour τ. Let $C_{R_i}^*$ be the optimal coverage tour for a region R_i. OPT covers every region in R therefore, we have $|OPT| \geq \sum_{R_i \in R} |C_{R_i}^*|$.

Suppose we use an α-approximation algorithm for computing τ_R, and a β-approximation algorithm for finding the coverage tour of each region. Then the tour τ obtained by visiting the regions according to the order given by τ_R, and covering each region independently when it is visited, has a cost of at most $\alpha|\tau_R^*| + \sum_{R_i \in R} \beta |C_{R_i}^*|$. Equivalently,

$$|\tau| \leq \alpha|\tau_R^*| + \sum_{R_i \in R} \beta |C_{R_i}^*| \leq \alpha|OPT| + \beta|OPT|$$
$$\therefore \quad |\tau| \leq (\alpha + \beta)|OPT|$$

Therefore, this approach costs at most a factor $(\alpha + \beta)$ of an optimal algorithm.

We now present algorithms for the two components of the strategy: computing a tour that visits the regions and covering the regions with specified entry and exit points.

B. Visiting the Regions: TSPN and the Zookeeper Problems

Computing a tour τ_R that visits all the regions depends on the geometric properties of the regions. This problem is commonly known as TSP with neighborhoods (TSPN). Most geometric instances of the TSPN problem are NP-Hard. In general, we can use constant-factor approximation algorithms for TSPN such as [15] to find τ_R and α.

In our application, it is reasonable to model the lake as a simply-connected region, i.e., without any holes. Further, regions of interest where the fish may lie are usually close to the shore. If the regions are convex polygons touching the boundary of a simply-connected lake then the tour can be computed using the so-called zookeeper's route [16]. This special case of the TSPN can be solved optimally ($\alpha = 1$) in polynomial time due to the following lemma.

Lemma 1 ([16]): Let $R = \{R_1, R_2, \ldots, R_i, \ldots, R_n\}$ be a set of convex regions located along the perimeter of a simply connected polygon P. There exists an optimal solution for visiting the regions in R which visits them in the order they appear along the boundary of P.

Once the ordering of the regions is known, the shortest tour visiting all regions can be calculated using dynamic programing. The exact solution is given in [16]. We use a simpler solution by discretizing the boundary of the regions for determining the entry and exit locations for each region. We build a table $C(i, s_i)$ which stores the length of a tour that

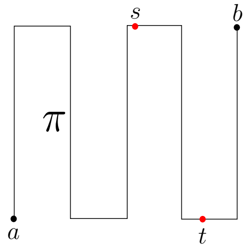


Fig. 4. Covering a rectangle with given entry and exit points (s and t) with a 2-approximation: Follow π from s to a, complete π, follow π from b to t. In the worst-case the optimal path π is covered twice.

enters the region R_i at location s_i for the first time. The entries of the table are computed using the following recurrence:

$$C(i, s_i) = \min_{t_{i-1}} \left[\min_{s_{i-1}} (C(i-1, s_{i-1})) + d(t_{i-1}, s_i) \right], \quad (1)$$

where s_{i-1} and t_{i-1} lie on the boundary of R_{i-1} and $d(x, y)$ is the Euclidean distance between points x and y. The cost of entering the region R_i at point s_i is equal to the minimum cost of reaching the previous region, R_{i-1}, entering at location s_{i-1}, plus the shortest distance from R_{i-1} to R_i, $d(t_{i-1}, s_i)$. By Lemma 1, the ordering of the regions is optimal. Since we cover all possible values of t and s, the algorithm computes an optimal solution up to the discretization error.

To turn these tours into coverage paths, we need a way to cover a region with specified entry and exit points. We next present such a technique when the regions are arbitrarily oriented rectangles. Rectangles are both easy to specify and general enough for practical purposes.

C. Covering Regions with Given Entry and Exit Points

The algorithm presented in Section IV-B generates an entry and exit point for each region. These points impose a constraint on our algorithm for finding a path that covers the rectangle. The following lemma shows that we can cover a rectangle satisfying this constraint, and be only a constant factor away from an optimal coverage path without such constraints. We assume that the rectangle has an $x \times y$ grid imposed on it, such that visiting all grid cell covers the rectangle.

Lemma 2: Let R be a rectangle with a grid imposed on it. Let s and t be two grid points on the boundary specified as entry and exit points. There exists a tour T which starts at s, visits every grid point and exits at t such that the length of T is at most twice that of an optimal tour which visits every grid point but can start and end at any points on the boundary of R, not necessarily s and t.

Proof: Let π be the optimal path to cover R without any restrictions on the starting and ending points. When R is a rectangle, π is a boustrophedon path which visits every point exactly once.

Suppose that π starts at a and ends at b. Note that $s, t \in \pi$. Without loss of generality, we assume that t is between s and b along π (See Figure 4). We form a coverage path from s to t using π as follows: From s, go to a along π and come back to s by retracing these steps. Then go to b from s along π (passing through t). Finally, arrive at t from b along π. This path visits every point on π and has length at most twice that of π. ■

The result is tight; when the input is a rectangle with one side length equal to r, and $s = t$, each point is covered twice.

To summarize, we showed that the following algorithm for covering rectangles along the boundary of a lake is an $(\alpha+\beta)$-approximate algorithm with $\alpha = 1$ and $\beta = 2$.

(i) Compute the shortest tour τ_R which visits each region in R using the dynamic programming solution presented in Section IV-B. This algorithm returns an entry and exit point for each rectangle, along with their ordering.

(ii) Follow τ_R as follows: Starting from the entry point of an arbitrary region, whenever a region is visited, cover it using the strategy given in Lemma 2 which ends at the exit point. Move to the entry point of the next region given by τ_R and repeat till all regions are visited.

D. Experiments

We implemented this search algorithm on our robotic boat and evaluated it through field trials at Lake Phalen, MN, USA. The input regions for one such trial are shown in Figure 5(a) (chosen arbitrarily for testing our algorithm). The series of offline waypoints generated by our algorithm are shown in Figure 5(b). We used an empirically determined sensing range of 50m to generate the boustrophedon paths. The actual trajectory executed by the robot is shown in Figure 5(c). The robot traveled a total distance of 5.6km in about 87min while executing this trajectory.

While moving along the search path at certain locations, the robot detected signals from five radio-tagged fish and a reference tag present in the lake. Such robot locations are marked with the corresponding tag frequency in Figure 5(c). The actual position of the fish can be anywhere within a distance equal to the sensing range from these locations. To better localize the fish, we switch to the *localization* phase whenever we detect a signal on one of the tuned frequencies. We describe our algorithms for the localization phase next.

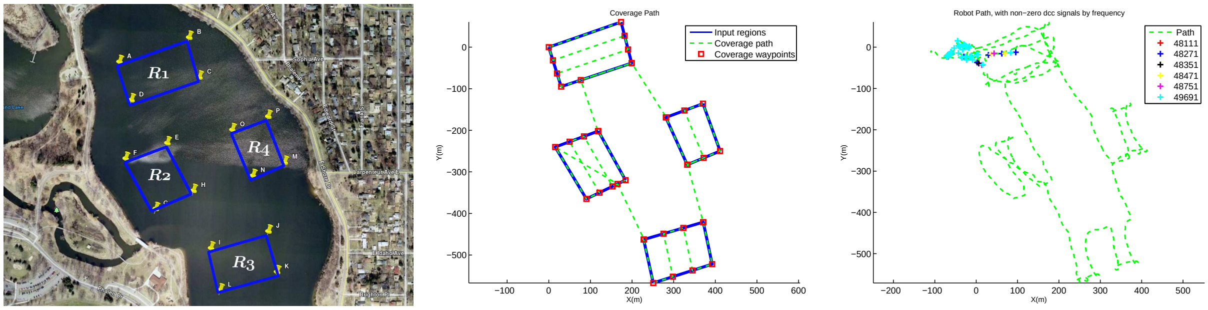
V. LOCALIZATION

The objective of the localization phase is to use bearing measurements from the radio antenna to localize a tagged fish accurately, once it is found during the search phase. The boat must choose sensing positions which provide the most information about the location of the tag. We use an EKF to estimate the position of the tag and represent the uncertainty in the position of the tag with its covariance. We seek sensing locations which minimize the determinant of the covariance matrix.

We begin by describing how we use the signal strength output of the radio receiver to obtain bearing towards the tag.

A. Measurement Model

The received signal strength varies with the relative angle of the plane of the loop antenna with the tag. If the tag is directly aligned with this plane, the signal strength is highest. Since the antenna is mounted on a pan-tilt unit we can rotate



(a) Coverage regions (b) Coverage path found using our algorithm. (c) Actual path executed by the robot.

Fig. 5. Coverage experiment conducted at Lake Phalen, MN, USA. **(a)** the four input regions, **(b)** the path found by the algorithm described in Section IV, **(c)** the actual path followed by the robot during coverage. The robot traveled a total distance of 5.6km in about 87min. Locations where signals from radio-tags were detected are also marked, along with their frequencies.

it and sample the signal strength as a function of the relative angle from the boat. We fit a smooth function to the samples obtained and use the point of maximum value of this function as the bearing measurement. Based on a number of trials, we concluded that least squares fitting of a cubic polynomial works best for computing the bearing in our system.

Note that the bearing obtained is an infinite line (as opposed to a directed ray), and hence there is ambiguity in the obtained bearing. For example, if α is the direction with maximum signal strength, then α and $\alpha + \pi$ are both valid bearing measurements. We disambiguate by moving along either α or $\alpha+\pi$ and checking if the signal strength increases or decreases. A detailed description of other methods for disambiguating the measurements is given in [17].

B. Optimization of Robot Motion

Since measuring the bearing takes time (about 1min), the estimation must be performed using a small number, say k, of measurements. Further, these locations must be chosen in an online fashion as the measurements become available. We present three strategies to compute k sensing locations, and compare their performance in simulations and real-world experiments.

All three active localization strategies require an initial estimate of the tag. In [4], we presented a scheme to initialize the target based on two bearing measurements taken from different sensing locations. Using this initial estimate, we propose the following three strategies to determine the next k sensing locations of the robot.

1) FIM: The Cramer-Rao Lower Bound (CRLB) for an unbiased estimator is a lower bound on the estimation error covariance. This lower bound is equal to the inverse of the FIM (denoted by I) for the k measurements. The determinant of I is inversely proportional to the square of the area of the 1-σ uncertainty ellipse, and is commonly used as the objective function to be maximized. For k bearing measurements with zero-mean Gaussian noise, the determinant of I (denoted by $|I|$) is given as,

$$|I| = \frac{1}{\sigma^4} \sum_{i=1}^{k} \sum_{j=1}^{k} \left[\frac{\sin(\theta_i - \theta_j)}{d_i d_j} \right]^2 . \tag{2}$$

where θ_i and d_i is the angle and distance from the i^{th} sensing location to the true target location.

We impose a grid of size $n \times n$ centered at the current position of the robot. To compute the k sensing locations, we exhaustively consider each of the $\binom{n^2}{k}$ combinations as a candidate trajectory, and compute $|I|$. An optimal trajectory can then be chosen as one with the minimum value of $|I|$.

2) Greedy: Instead of computing a fixed path for the k measurements, we can use an online greedy strategy which picks the next sensing location based on the current estimate and uncertainty of the position of the tag. Given the current robot and tag estimates, Greedy considers all neighboring locations of the robot as candidate sensing locations, and computes the posterior covariance by simulating an EKF update at each sensing location with a discretized set of possible bearing measurements. Greedy then picks the candidate location where the determinant of the posterior is minimum.

3) Enumeration Tree: We extend the objective function of Greedy to look ahead k measurements, in the Enumeration tree strategy. We build a min-max tree that explores the set of all sensing locations and all possible measurements that can be obtained, since the uncertainty depends on both. The tree consists of two types of nodes at alternate levels (Figure 6): MAX nodes (u_i) represent neighboring robot locations to the current, and MIN nodes (z_i) represent the discretized set of possible measurements. Each node stores an estimate of the target's state and covariance by simulating EKF updates based on the sensing locations and bearing measurements stored along the path in the tree. Details are presented in [4].

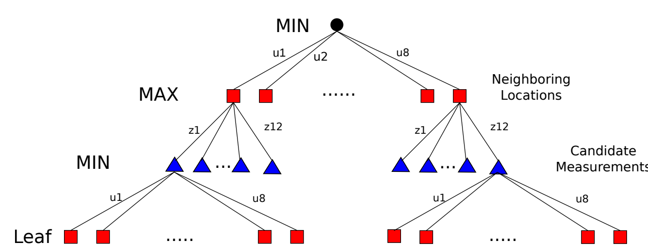


Fig. 6. Min-max tree: $u_1, \ldots, u_8$ are the neighboring locations for the robot, and $z_1, \ldots, z_{12}$ are candidate bearing measurements.

Once the tree is built, the min-max values for each node are propagated bottom-up starting with the leaf. The min-max value for the leaf nodes is defined as the determinant of the simulated posterior covariance matrix stored at that node. The min-max value for all MAX nodes is the $\max$ of min-max values of its children, and that for non-leaf MIN nodes is $\min$ of min-max values of its children.

During execution, the robot chooses the MAX node with the minimum min-max value as next sensing location at each iteration. The MIN node is chosen as per the actual bearing obtained. Since we use discrete measurement samples, there might not be a node with bearing exactly equal to the actual measurement. In addition, there is uncertainty associated with the position of the robot itself. Hence, we use the Bhattacharya Distance [18] to find a MIN node with posterior covariance closest to the covariance after the measurement update.

C. Simulations and Experiments

We first compared the performance of the three active localization strategies in simulation. We ran 100 random trials with the true tag 25m away from the starting position of the robot in each trial. A grid with side length 3m was used to generate sensing locations for 3 measurements. We generated noisy bearing measurements by corrupting the true bearing with Gaussian noise ($\sigma = 15°$).

The mean errors for FIM, Greedy, and Enumeration tree were 6.30m, 5.98m, and 5.73m, and the mean determinant of the final covariances were 54.81, 40.59, and 48.36 units respectively. The poor performance for the FIM strategy can be attributed to the fact that it is an open-loop strategy which depends on the initial estimate. Further, it computes locations which minimize the lower bound on final uncertainty of an "efficient estimator" (i.e. estimator whose variance is equal to the CRLB). Since EKF is not an efficient filter, there is no guarantee that it would achieve this lower bound. On the other hand, the Enumeration tree and the Greedy compute the actual covariance of the EKF estimator and pick the location which would minimize its determinant.

Although the Enumeration Tree performs better than the Greedy strategy, the performance gains are not significant to warrant the extra computational time. Hence, we decided to use the Greedy strategy on our system in field experiments. To test the system we conducted field trials with a reference tag submerged in a lake at a known position. The results from one such trial are shown in Figure 7. Sensing locations r_4, r_5, r_6 were obtained by running the Greedy strategy. The resulting 1-σ uncertainty ellipses are shown (in blue) along with the tag estimates (as red crosses). The true location of the tag is marked by a black star. The final error of this triangulation was 1.21m, with 1-σ bounds of 3.3m and 2.7m in the x and y directions, respectively.

The field experiments demonstrate that our system is capable of localizing stationary reference tags reliably. Next, we present complete field experiments where the robot executed both the *search* and *localization* phases.

VI. Field Experiments

In this section, we report results from two field tests conducted at Lake Gervais in MN, USA. In the first test, a

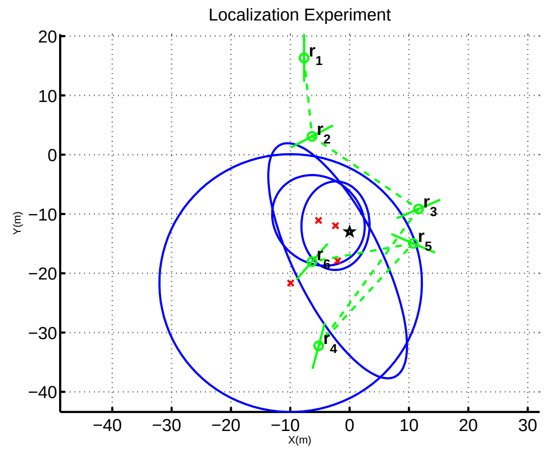


Fig. 7. Localization experiments with the Greedy strategy. Ellipses shown encompass the 1-σ uncertainty after each measurement. We use the second measurement to disambiguate which side the tag lies from bearing obtained at r_1. Bearing measurements are shown as solid green lines.

reference tag was deployed at a known location. In the second test, we searched for tagged fish present in the lake.

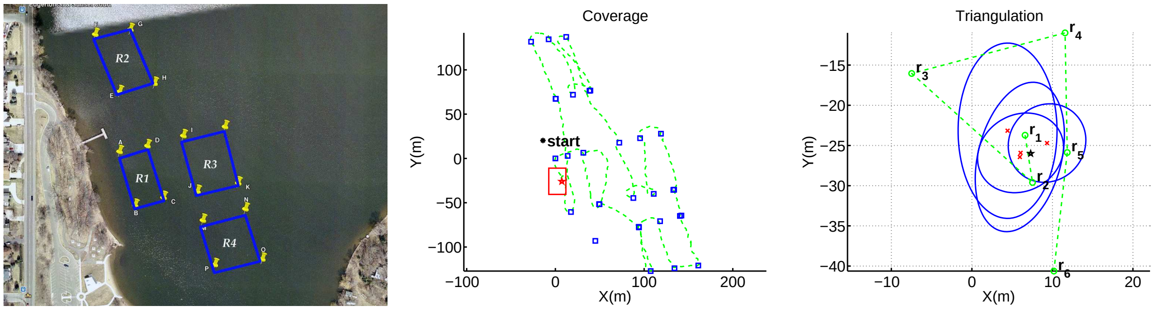
Figure 8 shows results from the first experiment with a reference tag deployed at the location marked in Figure 8(b). The robot executed the coverage pattern while continuously monitoring the radio antenna on the frequency of this reference tag. After detecting signal from this radio tag at r_1 (Figure 8(c)), the robot switched from the search phase to the localization phase.

During the localization phase, the robot executed the Greedy strategy with $k = 2$ measurements, in addition to the two initialization measurements taken at r_1 and r_3. The measurement at r_2 is used to distinguish which side of the bearing at r_1 the fish is located by comparing their signal strengths. The robot then moved to r_4 and r_5 and obtained bearing measurements as shown. The 1-σ uncertainty ellipse after each step is also shown. After completing the localization, the robot continued to cover the rest of the regions. The robot covered a total path of approximately 2km in 49min.

The second field trial (Figure 9) was conducted in the same lake without a reference tag. We programmed the robot to search for frequencies corresponding to actual radio-tagged fish present in this lake. While searching the first region, the robot detected one of the frequencies in the list, executed the localization strategy, and returned to the search plan. Figure 9(b) shows the coverage path followed by the robot. The red box marks the area where the robot followed the localization strategy to obtain additional bearing measurements and localize the unknown tag. Figure 9(c) shows a closeup of the localization phase. Since this was an actual radio-tagged fish, the ground truth is unknown.

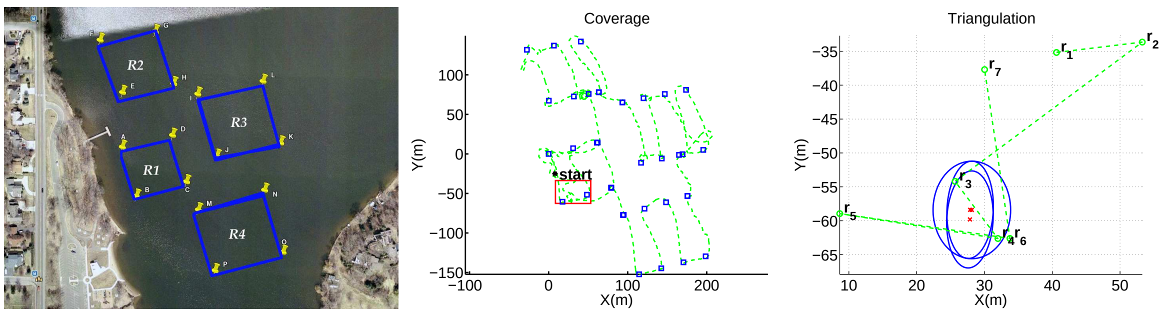
VII. Conclusion

We presented a novel system for monitoring radio-tagged invasive fish. We divided the monitoring task into two subtasks of finding the tagged fish and localizing them accurately. For the first task, we presented an algorithm for finding a tour whose length is at most a constant factor away from an optimal



(a) Coverage regions (b) Coverage track of the robot. (c) Localization with reference tag.

Fig. 8. A field trial on Lake Gervais, MN. The robot covered the regions via the path shown in the middle figure. On the right, a closeup of the localization of a reference tag is shown. The true location of the reference tag is marked with a black star. The red box in the middle figure corresponds to the triangulation area on the right.



(a) Coverage regions (b) The track of the robot during coverage. (c) Close up of the measurements.

Fig. 9. Field experiments at Lake Gervais, MN. Figure 9(a) shows the areas we wish to cover. The computed search path and the trajectory executed by the robot are shown in Figure 9(b). Upon detecting a tag, the robot executed a localization strategy as shown in Figure 9(c). Because the robot attempted to triangulate a tagged fish, we do not know the true location of the tag.

tour. For localizing the tagged fish, we first showed how the bearing of the tag can be estimated by using measurements obtained by rotating a directional antenna. Next we addressed the problem of actively choosing sensing locations to reduce localization uncertainty. We compared three algorithms in simulations and field experiments, and incorporated the most effective one into our system. We concluded the paper with additional field trials.

While the initial results are encouraging, there is significant future work including developing fish mobility models and corresponding search and tracking algorithms. We plan to use multiple boats to improve the coverage time and localization uncertainty. Additional issues faced when building such a system (e.g. communication and coordination) must be addressed. Finally, we plan to use our system in larger lakes and help biologists to study carp behavior.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant Nos. 1111638, 0916209, 0917676, 0936710. We thank members of the Sorensen lab at the Department of Fisheries at the University of Minnesota for many useful discussions and sharing equipment. We also thank Deepak Bhadauria for his contributions to an earlier version of this system.

References

[1] P. G. Bajer, C. J. Chizinski, and P. W. Sorensen, "Using the Judas technique to locate and remove wintertime aggregations of invasive common carp," *Fisheries Management and Ecology*, 2011.

[2] P. Tokekar, D. Bhadauria, A. Studenski, and V. Isler, "A Robotic System for Monitoring Carp in Minnesota Lakes," *Journal of Field Robotics*, vol. 27, no. 6, pp. 779–789, 2010.

[3] P. Bajer, H. Lim, M. Travaline, B. Miller, and P. Sorensen, "Cognitive aspects of food searching behavior in free-ranging wild Common Carp," *Environmental Biology of Fishes*, vol. 88, no. 3, pp. 295–300, 2010.

[4] P. Tokekar, J. Vander Hook, and V. Isler, "Active Target Localization for Bearing-Based Robotic Telemetry," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 488–493.

[5] J. Das, F. Py, T. Maughan, T. OReilly, M. Messié, J. Ryan, K. Rajan, and G. Sukhatme, "Simultaneous Tracking and Sampling of Dynamic Oceanographic Features with Autonomous Underwater Vehicles and Lagrangian Drifters," in *International Symposium on Experimental Robotics*, 2010.

[6] R. N. Smith, J. Das, H. Heidarsson, A. A. Pereira, F. Arrichiello, I. Cetinic, L. Darjany, M.-E. Garneau, M. D. Howard, C. Oberg, M. Ragan, E. Seubert, E. C. Smith, B. Stauffer, A. Schnetzer, G. Toro-Farmer, D. A. Caron, B. H. Jones, and G. S. Sukhatme, "USC CINAPS Builds Bridges: Observing and Monitoring the Southern California Bight," *IEEE Robotics and Automation Magazine*, vol. 17, no. 1, pp. 20–30, Mar 2010.

[7] M. Dunbabin, P. Corke, I. Vasilescu, and D. Rus, "Experiments with Cooperative Control of Underwater Robots," *The International Journal of Robotics Research*, vol. 28, no. 6, p. 815, 2009.

[8] M. Dunbabin and L. Marques, "Robots for environmental monitoring: Significant advancements and applications," *IEEE Robotics and Automation Magazine*, vol. 19, no. 1, pp. 24 –39, Mar 2012.

[9] H. Choset, "Coverage for robotics: A survey of recent results," *Annals of Mathematics and Artificial Intelligence*, vol. 31, no. 1–4, pp. 113–126, 2001.

[10] V. Vazirani, *Approximation algorithms*. Springer Publishing Company, Incorporated, 2001.

[11] S. E. Hammel, P. T. Liu, E. J. Hilliard, and K. F. Gong, "Optimal observer motion for localization with bearing measurements," *Computers and Mathematics with Applications*, vol. 18, no. 1-3, pp. 171 – 180, 1989.

[12] E. Frew, "Observer trajectory generation for target-motion estimation using monocular vision," Ph.D. dissertation, Stanford University, 2003.

[13] K. Zhou and S. Roumeliotis, "Multi-robot active target tracking with combinations of relative observations," *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 678 –695, Aug. 2011.

[14] M. Caccia, M. Bibuli, R. Bono, and G. Bruzzone, "Basic navigation, guidance and control of an Unmanned Surface Vehicle," *Autonomous Robots*, vol. 25, no. 4, pp. 349–365, Aug 2008.

[15] J. S. Mitchell, "A constant-factor approximation algorithm for TSP with pairwise-disjoint connected neighborhoods in the plane," in *Proceedings of the 2010 Annual Symposium on Computational Geometry*. New York, NY, USA: ACM, 2010, pp. 183–191.

[16] W. Chin and S. Ntafos, "The zookeeper route problem," *Information Sciences: an International Journal*, vol. 63, no. 3, pp. 245–259, 1992.

[17] J. Vander Hook, P. Tokekar, E. Branson, P. Bajer, P. Sorensen, and V. Isler, "Local-Search Strategy for Active Localization of Multiple Invasive Fish," in *International Symposium on Experimental Robotics*, 2012.

[18] A. Bhattacharyya, "On a measure of divergence between two multinomial populations," *Sankhyā: The Indian Journal of Statistics (1933-1960)*, vol. 7, no. 4, pp. 401–406, 1946.