Cooperative Tracking for Persistent Littoral Undersea Surveillance

by

Robert Derek Scott

Submitted to the Department of Electrical Engineering and Computer Science
and Department of Mechanical Engineering
in partial fulfillment of the requirements for the degrees of

Naval Engineer

and

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2007

Author ....
Department of Electrical Engineering and Computer Science
and Department of Mechanical Engineering
May 11, 2007

Certified by ....
John J. Leonard
Associate Professor of Mechanical and Ocean Engineering
Thesis Supervisor

Accepted by ....
Arthur C. Smith
Chairman, Department Committee on Graduate Students
Department of Electrical Engineering and Computer Science

Accepted by ....
Michael Triantafyllou
Chairman, Department Committee on Graduate Students
Center for Ocean Engineering

Accepted by ....
Lallit Anand
Chairman, Department Committee on Graduate Students
Department of Mechanical Engineering

Cooperative Tracking for Persistent Littoral Undersea Surveillance

by

Robert Derek Scott

Submitted to the Department of Electrical Engineering and Computer Science
and Department of Mechanical Engineering
on May 11, 2007, in partial fulfillment of the
requirements for the degrees of
Naval Engineer
and
Master of Science in Electrical Engineering and Computer Science

Abstract

The US Navy has identified a need for an autonomous, persistent, forward deployed system to Detect, Classify, and Locate submarines. In this context, we investigate a novel method for multiple sensor platforms acting cooperatively to locate an uncooperative target. Conventional tracking methods based on techniques such as Kalman filtering or particle filters have been used with great success for tracking targets from a single manned platform; the application of these methods can be difficult for a cooperative tracking scenario with multiple unmanned platforms that have considerable navigation error. This motivates investigation of an alternative, set-based tracking algorithm, first proposed by Detweiler *et al.* for sensor network localization, to the cooperative tracking problem. The Detweiler algorithm is appealing for its conceptual simplicity and minimal assumptions about the target motion. The key idea of this approach is to compute the temporal evolution of potential target positions in terms of bounded regions that grow between measurements as the target moves and shrink when measurements do occur based on an assumed worst-case bound for uncertainty.

In this thesis, we adapt the Detweiler algorithm to the scenario of cooperative tracking for persistent undersea surveillance, and explore its limitations when applied to this domain. The algorithm has been fully implemented and tested both in simulation and with postprocessing of autonomous surface craft (ASC) data from the PLUSNet Monterey Bay 2006 experiment. The results indicate that the method provides disappointing performance when applied to this domain, especially in situations where communication links between the autonomous tracking platforms are poor. We conclude that the method is more appropriate for a "large N" tracking scenario, with a large number of small, expendable tracking nodes, instead of our intended scenario with a smaller number of more sophisticated mobile trackers. The method may also be useful as an adjunct to a conventional Bayesian tracker, to reject implausible target tracks and focus computational resources on regions where the target is present.

Thesis Supervisor: John J. Leonard
Title: Associate Professor of Mechanical and Ocean Engineering

Acknowledgments

Given my history of foot in mouth disease and of being misread, I'll just list a few people and leave the how's and why's as an exercise for the reader.

- CAPT Pat Keenan
- CAPT Dave Herbein
- CDR Tim McCoy
- CDR Joe Harbour
- Professor John Leonard
- Robert Dale Scott
- Andy Hunt
- Alex Bahr
- Jip Mosman

Contents

List of Figures

Chapter 1

Introduction

This chapter describes the context for the work investigated in this thesis. We first review the US Naval need for cooperative tracking by autonomous mobile platforms, and subsequently describe the problem in the terms of the robotics research literature.

1.1 Naval Context

The world's population of submarines peaked at the height of the Cold War. At that time, most submarines were from the US or the USSR and were of two types: Ballistic Missile boats (SSBNs) and Attack boats (SSNs). These nuclear submarines carried out three main missions. SSBNs went on strategic deterrent patrols, hiding in the ocean and listening for commands to launch a retaliatory nuclear attack. SSNs tracked SSBNs and sought to kill them before any retaliatory attack could be launched once the hot war had begun. SSNs also performed Intelligence, Surveillance and Reconnaissance (ISR) missions near enemy coastlines, gathering intelligence and waiting to pick up enemy submarines to track. The end of the cold war induced a significant reduction in the numbers of US and Soviet submarines. The US aggressively downsized their submarine fleet as their traditional missions evaporated. Meanwhile the collapse of the Soviet Union and economic difficulties in Russia caused them to simply abandon some submarines[10].

As the US Navy was reducing their submarine population based on the loss of the traditional adversary and traditional mission, it became apparent that submarines were ideally suited to perform other emergent missions against new adversaries. These new missions included stealthy strike and answering an asymmetrical threat from quiet diesel electric submarines operating in the littoral.

The asymmetry of the diesel electric submarine (SSK) refers to costs and capabilities. An SSK with torpedoes can be obtained on the open market relatively inexpensively and then can threaten or deny access to much larger, more expensive, and more capable ships simply because the SSK cannot be detected. Many current SSKs are strike capable using cruise missiles. If these missiles are armed with chemical, biological, or nuclear warheads, the asymmetric leverage extends beyond threatening a capital ship to possibly threatening a capital city. Balancing the asymmetry and countering the threat is highly desired. Thus, SSKs must be detected and tracked.

Detection and tracking of submarines is a mission clearly tuned for submarines. But pairing an SSN to each SSK identifies another asymmetrical advantage for the SSK: since they are inexpensive, greater numbers can be purchased. Thus while the SSN is tracking one SSK, another SSK is attacking the major ship or land target undetected. To fully counter the threat requires the tables be turned by using an inexpensive network of many sensor platforms to track any SSKs and share track information with some combatant platform. Once the locations and tracks of the SSKs are known, they have no advantages but are instead hampered by being slow and less capable. This network of sensors is PLUSNet.

1.2 Cooperative Tracking Using Autonomous Marine Platforms

The Persistent Littoral Surveillance Network (PLUSNet) is a revolutionary new concept for Anti-Submarine Warfare (ASW). In contrast to existing systems, which are based on tracking by single manned platforms, PLUSNet envisions an autonomous network of static and mobile sensors that can be deployed for a long duration. The goal is for the network to provide persistent detection, classification, localization and tracking (DCLT) capabilities, with significant onboard autonomous decision-making. Figure 1-1 provides an overview of the concept, illustrating a network of moored sensors, autonomous undersea gliders, and autonomous underwater vehicles (AUVs) performing cooperative DCLT [11].

PLUSNet is a major current initiative funded by the Office of Naval Research, with over a dozen partners from across the U.S. investigating a broad range of technologies. These include [5]:

- Platform design (AUVs and gliders)
- Power
- Persistent autonomy
- Acoustic communications
- Mobility
- Navigation
- Signal processing
- Acoustic modeling
- Network level tracking and response
- Environmentally adaptive sensing and network control

This thesis primarily addresses the problem of network level tracking and response, however, the topics of communications and navigation are intimately related. Similarly, PLUSNet is a system of systems as are the component marine platforms and each system influences the final result.

The PLUSNet system design is an evolution of the concept of Autonomous Ocean Sampling Networks (AOSN) [3, 4]. AOSN is a radical new approach to obtaining measurements in the ocean through the use of networks of fixed and mobile nodes, combined with advances in undersea communications and navigation technology. Key to AOSN is the availability of robotic platforms that can operate robustly with minimal human direction in the ocean. Figures 1-2 through 1-5 provide examples of several AOSN platforms, including the XRAY glider, the Seaglider, a SCOUT autonomous surface craft, and a Bluefin AUV.

The main idea of AOSN is to overcome spatial and temporal aliasing of ocean measurements, by having a system that could be autonomous deployed for a long time in the ocean environment. AOSN served as a context to make advances in many underlying technologies of autonomous marine vehicles, encompassing power, communications, navigation, sensing, data processing, and autonomy. A recent special issue of the IEEE Journal of Ocean Engineering provides a comprehensive review of these technologies [4].

While AOSN was initially conceived for persistent oceanographic measurements, recently the concept has been adapted to the ASW mission described above. If autonomous platforms can be provided with passive acoustic sensing capabilities (e.g., via towed line arrays), then it is hypothesized that a network of AUVs and gliders can provide persistent surveillance of an area.

In the adaptation of the AOSN concept to the ASW mission, it is clear that autonomous sensor processing is one of the most difficult issue. Can we enable the glider and AUV platforms to autonomously process the sensor data that they receive? Can multiple robot platforms work cooperatively to track targets, with minimal communication links to each other and back to human operators? These questions provide the context for our investigation of a new style of cooperative tracking algorithm for AUVs.

1.3 Thesis Structure

The goal of the thesis research has been to investigate the application of set-based tracking methods for robot localization to the problem of undersea tracking with mobile sensor networks. Chapter 2 describes our new algorithm and presents simulations results to show its operation. Chapter 3 presents experimental results with this algorithm from the major PLUSNet experiment held in Summer, 2006. Finally, Chapter 4 reviews the contributions of the thesis and makes recommendations for future research.

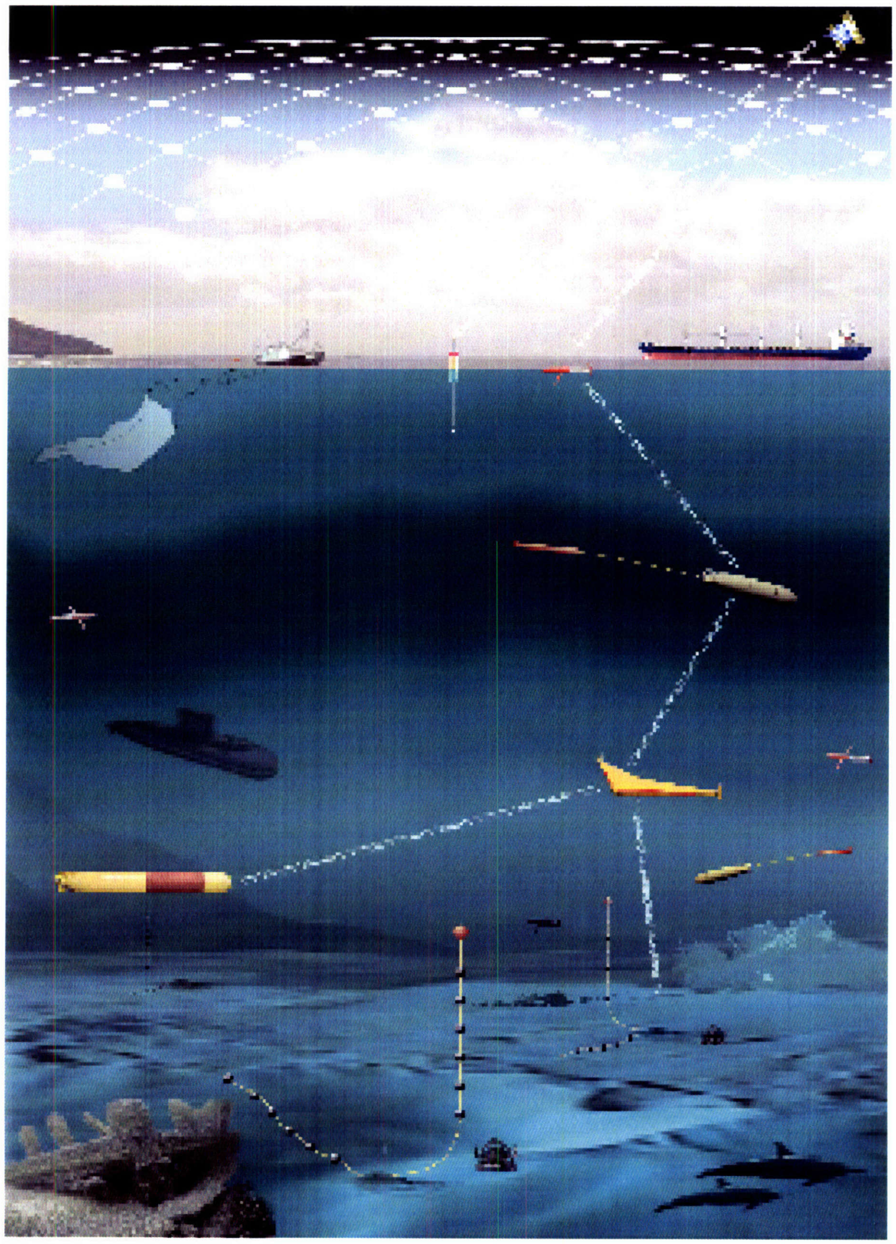

Figure 1-1: Concept overview for the Persistent Littoral Surveillance Network (PLUSNet). PLUSNet envisions a network of static and mobile sensors that can be deployed for a long duration over a wide area, providing persistent detection, classification, localization and tracking capabilities.

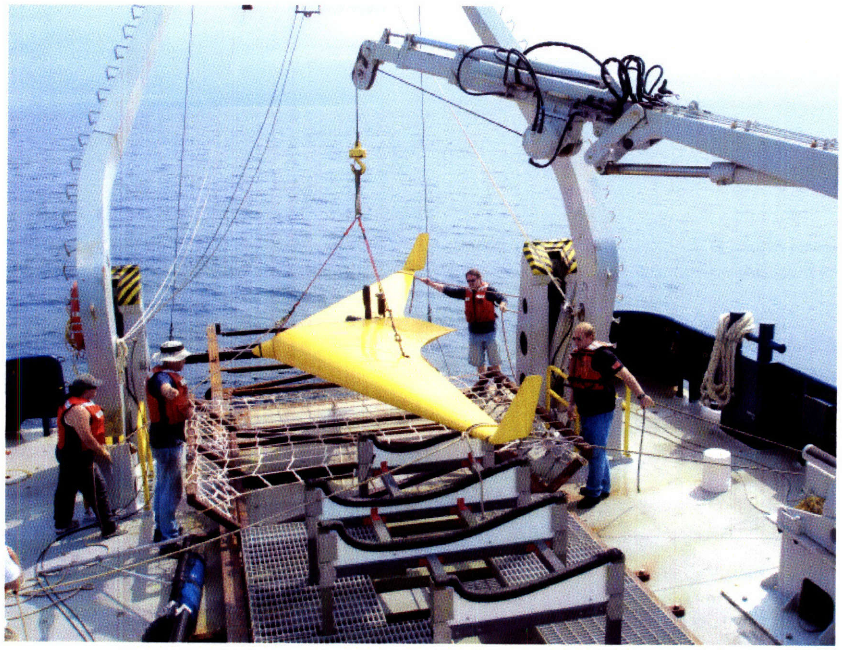

Figure 1-2: XRAY Glider [11]

Figure 1-3: Seaglider (image ©Alex Bahr, used with permission)

Figure 1-4: SCOUT ASC (image ©Alex Bahr, used with permission)

Figure 1-5: Bluefin AUV [11]

Chapter 2

Bounded Region Tracking Algorithm

This chapter describes the bounded region tracker. We first review Detweiler *et al.*'s [6] sensor network localization method and describe our adaptation of this approach to cooperative tracking at sea with autonomous platforms. We include a number of simulation results that illustrate the operation of the method.

2.1 Fixed Beacons and Mobile Sensors

Detweiler *et al.*[6] describe an intuitive geometric algorithm for localizing a mobile robot or agent within a field of fixed nodes using range-only or bearing-only sensors. The only information needed about the mobile robot is its maximum speed.

The fundamental concept of the algorithm is that the current location of the mobile node must be within the region it could reach (at maximum speed) from its previous location.

Consider a range-only, two dimensional example (all nodes are assumed to be on the same plane) as shown in figure 2-1. The very first sensor reading from beacon 'a' constrains the position of the mobile agent to the perimeter of a circle about the fixed node. When the next sensor data point, from beacon 'b', is collected, the time difference between the data points determines the reach of the mobile node from its last position. Thus the circle edge is grown by some distance ds which is the maximum speed times the time difference:

$$ds = v_{max} * dt$$

in every direction. (This is the Minkowski sum of the range circle perimeter and a filled circle of radius ds). This growth is an annulus and defines where the mobile node could be at this later time. It is illustrated in figure 2-1(b) in blue. Meanwhile, the second data point range information constrains the mobile node to another range circle (figure 2-1(c)). Both these constraints must be satisfied, so the intersection of the grown region and the new range circle, shown in red in figure 2-1(d), must contain the new location of the mobile robot.

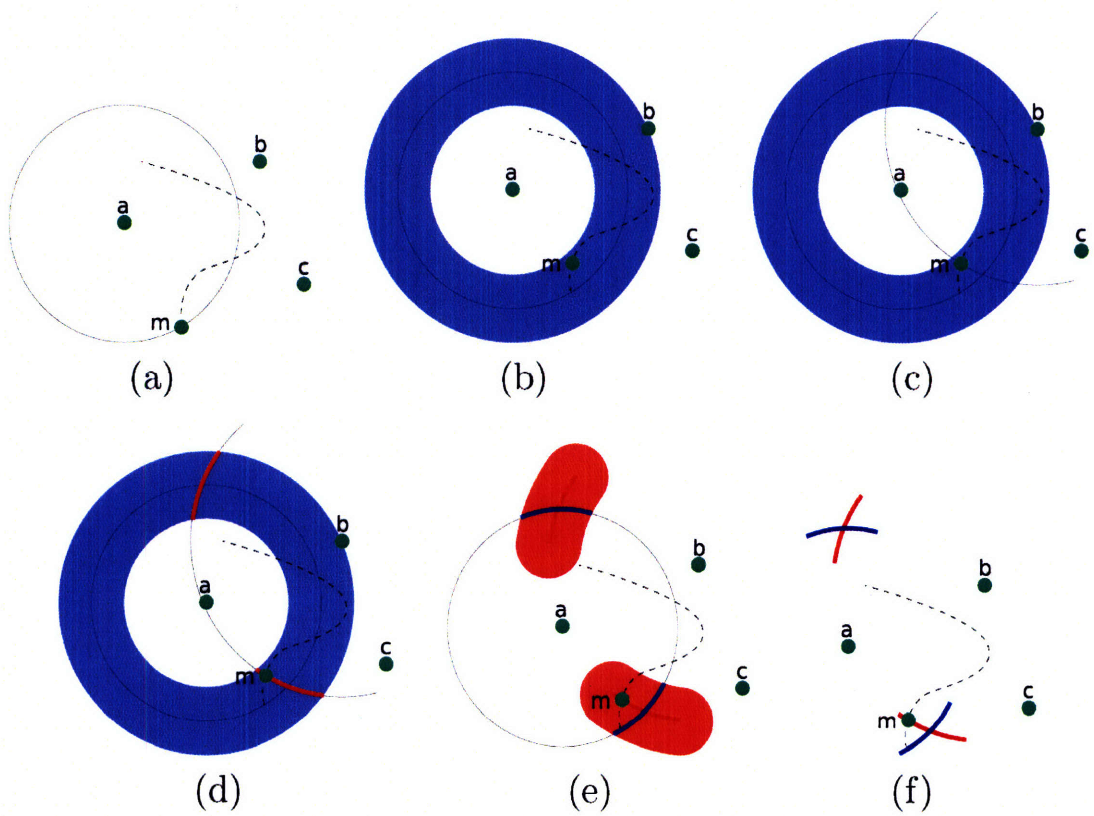

Figure 2-1: Range-only example (from [6])

This new constrained region can then be used looking backwards to constrain where the node must have been at the time of the previous reading. Just as for the forward looking case, to reach the final region, the mobile node must have started from a location within a distance *ds*. The final region can be grown by *ds* in all directions(figure 2-1(e)) and intersected with the original range circle to determine where on the original arc the node must have been(figure 2-1(f)).

As more data is received, the steps are repeated, identifying the region that must contain the mobile node. Similarly, each new localization region can be used to propagate constraints backwards in time, possibly improving past location information.

The perspective taken by [6] is that the mobile agent has passive sensors and uses the algorithm to locate itself relative to fixed beacon nodes. This perspective allows multiple mobile robots to locate themselves simultaneously and without disclosing any information to any other nodes. This perspective is not fundamental to the algorithm itself which only requires that some nodes know their positions, mobile or fixed.

Detweiler *et al.* prove that the algorithm is correct and optimal in [6]. Optimal here means that the localization regions found are the smallest, most constrained regions that contain the mobile node and consider all the data received in addition to the maximum speed of the mobile node.

Detweiler *et al.* also show that the algorithm including complete backwards propagation of constraints has run time complexity $O(n^3)$ where n is the number of input regions corresponding to the number of sensor data. The complexity can be reduced to $O(n^2)$ by limiting or removing the backwards propagation. Growing the localization region by *ds* and performing the intersection are each shown to be $O(n)$ but are then shown to have an expected run time complexity of $O(1)$. Thus the algorithm is expected to run in $O(n^2)$ time with full back propagation or $O(n)$ time with limited back propagation.[6]

Algorithm 1 Detweiler *et al.* Localization Algorithm [6]

procedure LOCALIZE($A_1 \cdots A_t$)
  $s \leftarrow$ max speed
  $I_1 = A_1$ ▷ Initialize the first intersection region
  for $k = 2$ to t **do**
    $\triangle t \leftarrow k - k - 1$
    $I_k =$ **Grow**$(I_{k-1}, s\triangle t) \cap A_k$ ▷ Create the new intersection region
    for $j = k - 1$ to 1 **do** ▷ Propagate measurements back
      $\triangle t \leftarrow j - j - 1$
      $I_j =$ **Grow**$(I_{j+1}, s\triangle t) \cap A_j$
    end for
  end for
end procedure

2.2 Mobile Sensor Nodes with Uncooperative Mobile Node

Figure 2-2 shows a bearing only example of the Detweiler *et al.* algorithm modified to reflect an uncooperative mobile node being tracked by two mobile sensor nodes. The blue circle and the green line represent the paths of two nodes cooperating to track a third uncooperative node whose path is shown in red. Initial detection is made at time 1 in (b). Based on the single bearing datum, the target is known to be within the fan of the bearing uncertainty and within range of the sensor thus the region is a triangle. At time 2, the green tracker detects the target. The time between detections and the upper bound on the speed of the target are used to calculate how far the target could be from its last boundary region. This area is shown in yellow in (d). At time 2, the target location must satisfy both the bearing and distance from previous location constraints, so it must be located at the intersection of the two regions, as shown in magenta in (e) and (f). This process is repeated for each new bearing datum: current region, possible distance, growth, bearing fan, and intersection. At any given time, it is known deterministically where the target can possibly be and where the target cannot be. Note that this modified algorithm does not propagate constraints backwards in time.

As stated previously, the algorithm of [6] is insensitive to motion of the localized

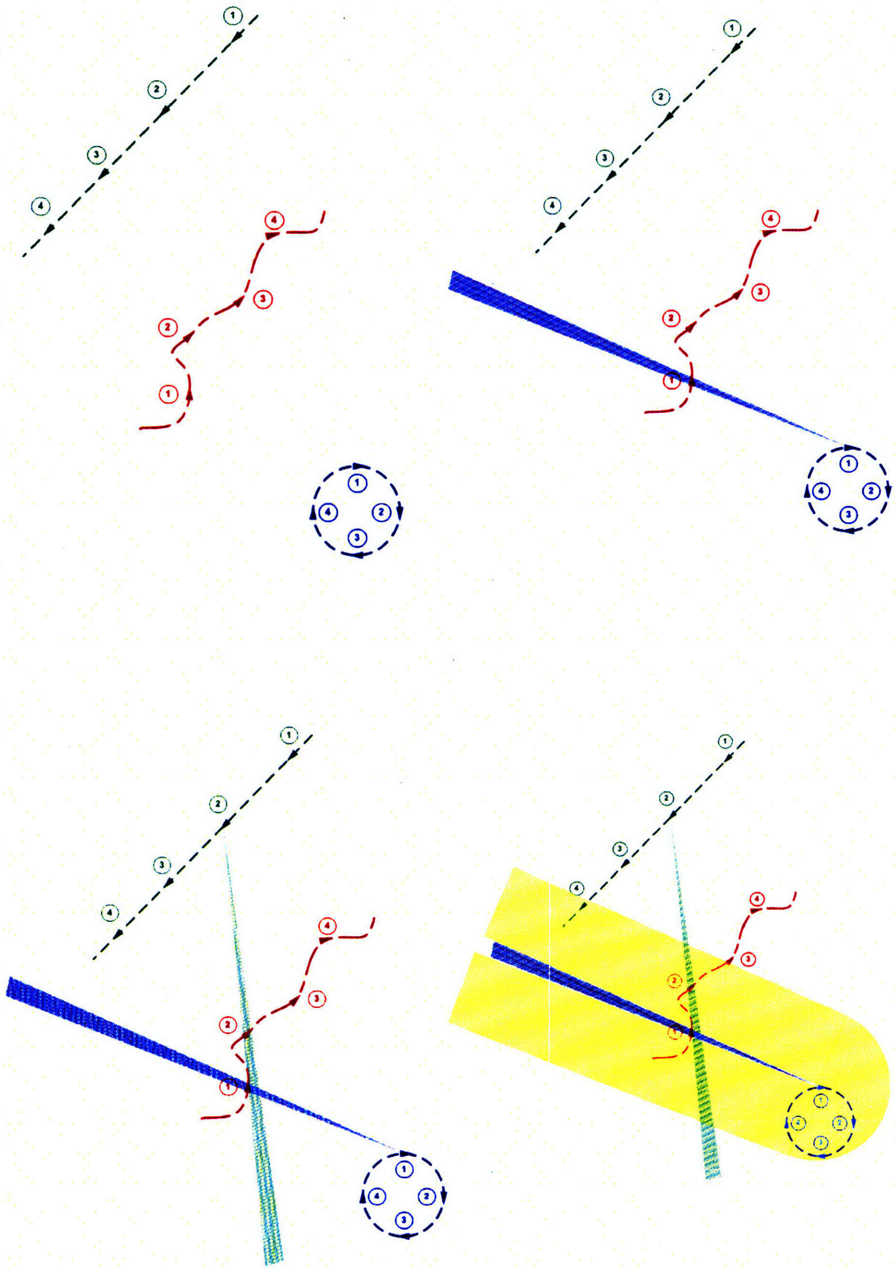

Figure 2-2: Example with bearing only data.

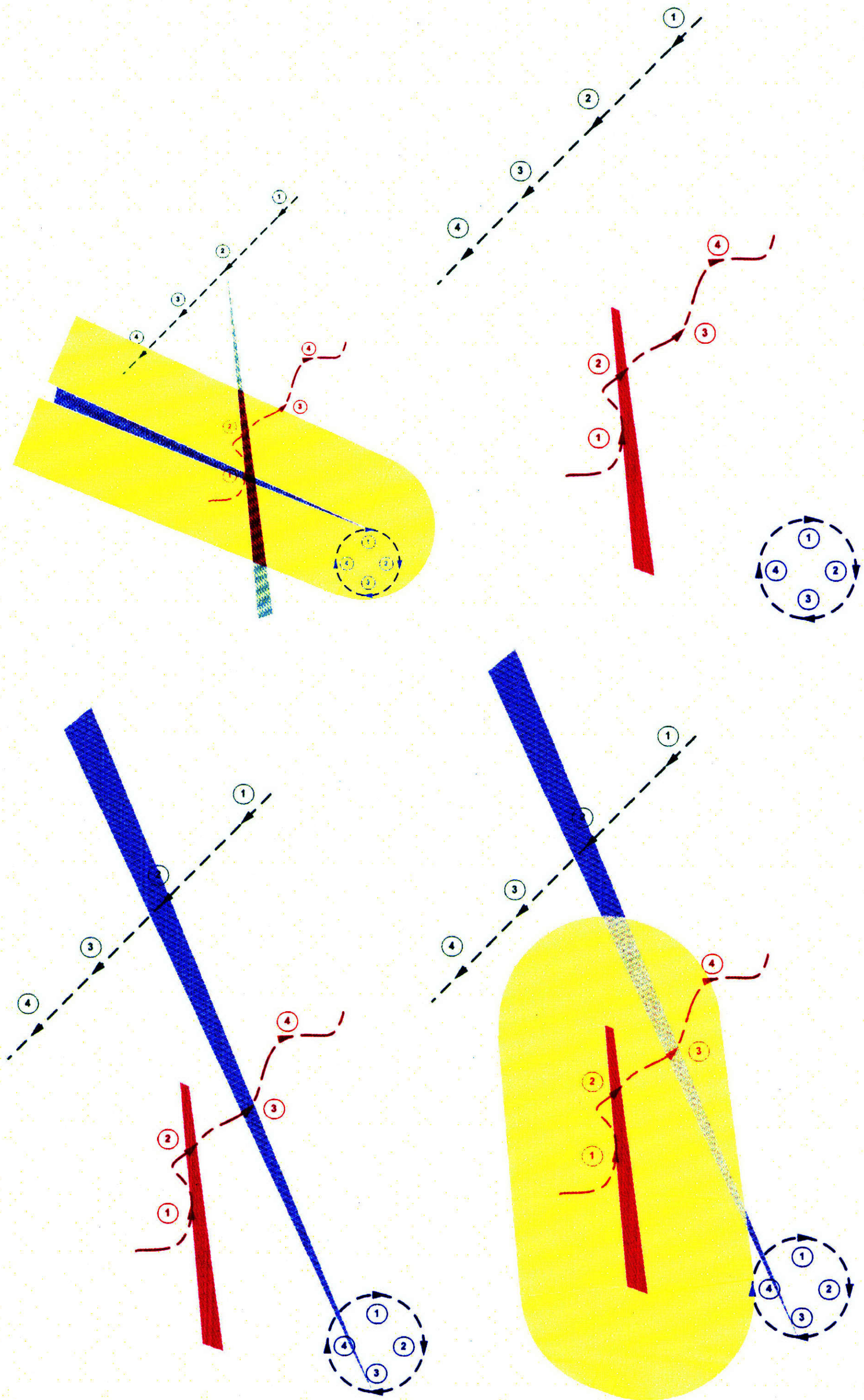

Figure 2-3: Example with bearing only data (continued).

nodes. That is, as long as a bearing or range from the node with unknown position to a node with known position can be determined, the algorithm works the same whether the known position nodes are moving or not. Thus this application of the Detweiler *et al.* algorithm is different only in eliminating the backward propagation altogether. The backward propagation was ignored in proving the correctness of the original algorithm therefore this modified algorithm is still correct. Similarly, the modified algorithm produces optimal localization regions moving forward considering all data received, the maximum speed of the uncooperative node, and the range and bearing error characteristics of the sensors.

Algorithm 2 Bounded Region Cooperative Tracking Algorithm

```
procedure LOCALIZE(A_1 ··· A_t)
    s ← max speed
    I_1 = A_1                              ▷ Initialize the first intersection region
    for k = 2 to t do
        Δt ← k − k − 1
        I_k = Grow(I_{k−1}, sΔt) ∩ A_k     ▷ Create the new intersection region
    end for
end procedure
```

2.3 Implementation

The bounded region tracker was implemented in C++ in a linux environment, making use of the Computational Geometry Algorithms Library (CGAL) and Qt graphics. Primitives from the CGAL Library were utilized for the low-level geometric operations incorporating computation of intersections and unions of polygon target localization regions. The most difficult part of the implementation was implementation of the Grow function that enlarged the localization region of the target forward through time as the target moved. The system was crafted in an application-agnostic manner with support utilities that transformed the input data into a suitable form for processing. For example, the core algorithm can be applied to data from any number of different sensors. The system was implemented in a text only version intended for embedded use and in a graphical user interface version with visualization to display the operation of the tracker in real-time.

2.4 Simulation Results

The algorithm's behavior was explored in simulation. Tracks for sensor nodes and an uncooperative target through time were constructed and used to generate bearing data from the sensors to the target. These bearing data were used by the algorithm which produced a list of localization regions through time. Plotting these regions and the position of the target node quickly identified the strong points and weaknesses of the method.

Single Tracker

Consider first a single tracking node given an unnatural advantage of beginning with a precise localization of the target.

In figure 2-4 a single stationary tracking node is located at (1000,0) identified by a red star. The target begins at (0,0) and this initial position is known to the tracker. The path of the target is shown in black as it proceeds on course 060. The localization regions are shown as blue polygons.

From the initial tight localization region surrounding the target at (0,0), the polygons expand rapidly in the absence of cross bearing data. This produces the distinctive stacked pyramid effect seen in the lower left of figure 2-4. Later, as the bearings change more rapidly, the new bearing fan intersects the grown region at an angle and exits the upper portion out the (angled) side rather than out the end. This is the source of the angled upper ends of the constrained regions.

Figure 2-5 is a partial screen capture from the Qt graphical implementation of the algorithm. This figure shows previous localization regions in black and blue, the grown area in yellow, the bearing fan in light blue, and the intersection region in green. Zooming in allows the shape of the grown region to be seen and shows the

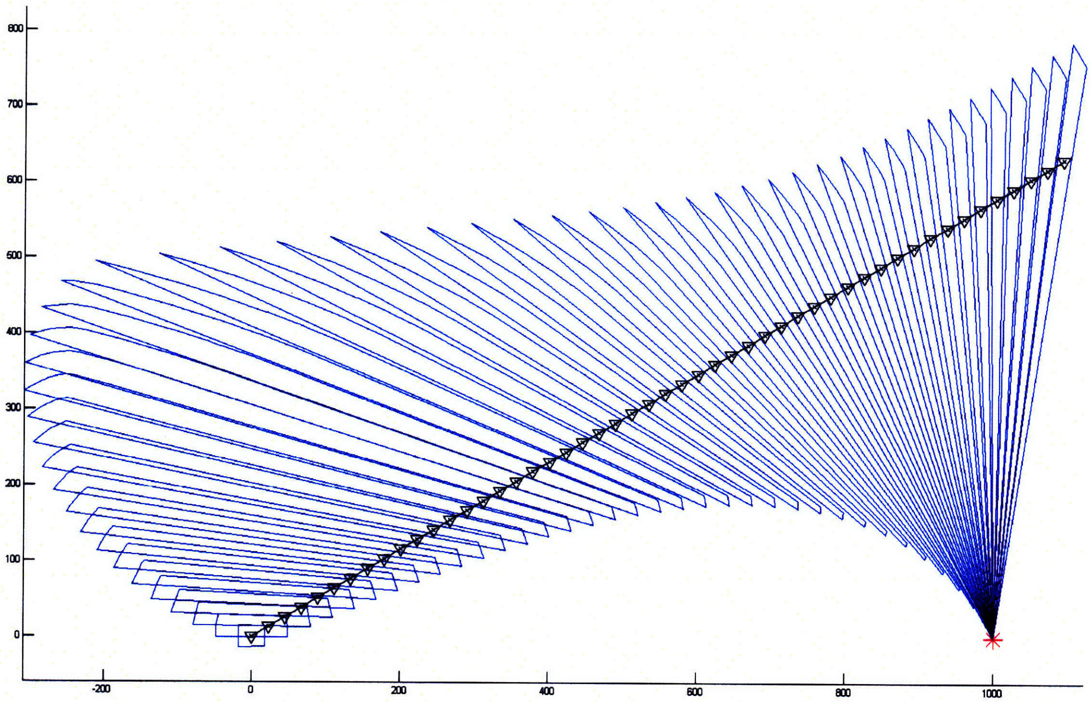

Figure 2-4: Single tracker. Tracker at (1000,0). Target position at start known to be (0,0).

new bearing fan leaving through the angled edge, thus preserving this feature into the next region. Comparison of the previous regions in black, on the left side of figure, shows the slow evolution reducing the angled edge.

This example shows the weakness of the algorithm for a single sensor. Without cross bearing data from another sensor node or a considerable speed advantage over the target node allowing a single node to collect bearings from very different angles, the algorithm tends toward merely noting the bearing to the target and offering no more insight into the target node's position. Note again that despite this example beginning with the tracker knowing a very precise location for the target, this information is quickly dissipated into large regions.

Two Fully Cooperating Trackers

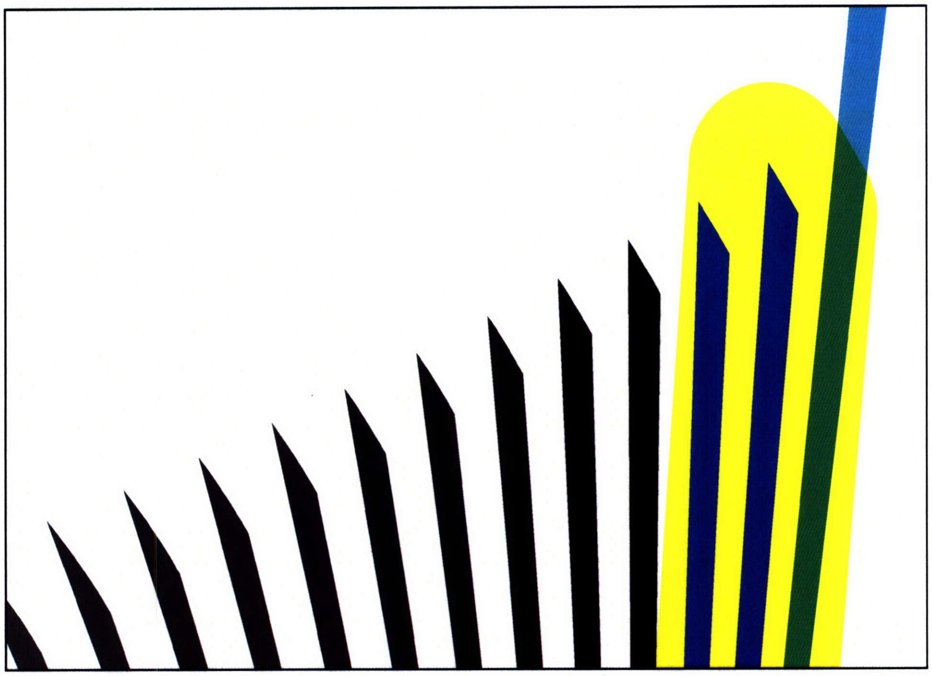

Figure 2-5: Zoomed view of simple, single tracker example.

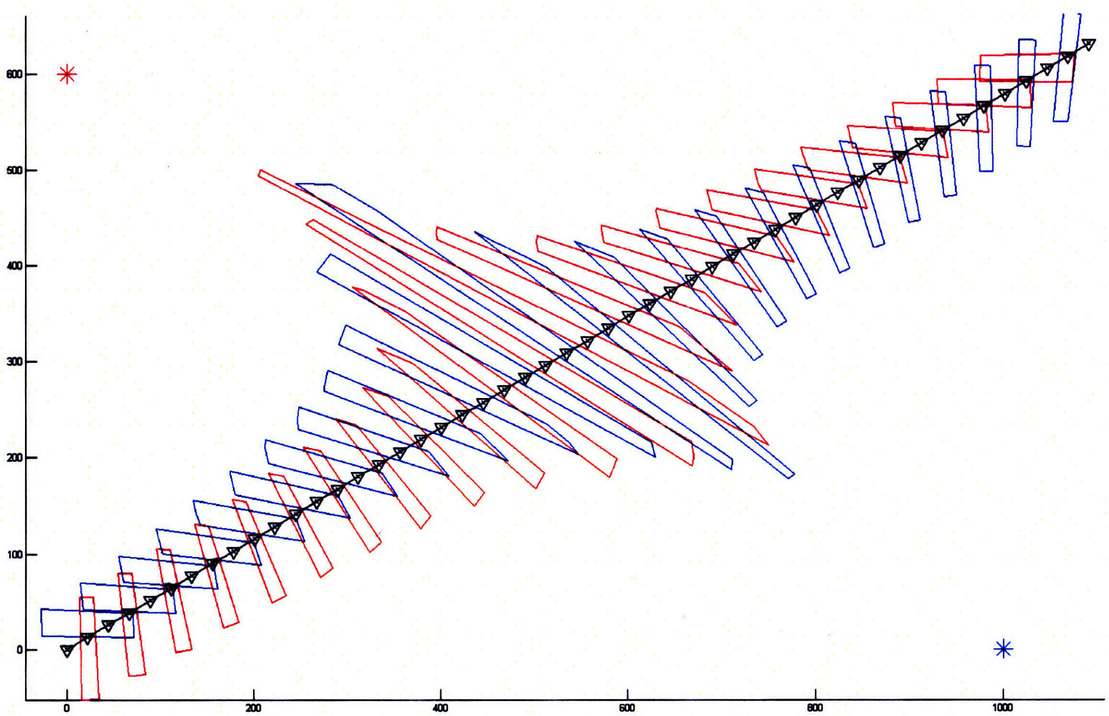

Figure 2-6: Two Fully Cooperating Trackers. Trackers at (1000,0) and (0,600). Target position known to be (0,0) at start. High Communication Bandwidth.

Figure 2-6 shows the algorithm's similarity to triangulation when sensor data rates and communication bandwidth are high. Stationary tracking nodes are located at (1000,0)(blue star) and (0,600)(red star). The target begins at (0,0) and this initial position is known to the trackers. The localization regions whose last bearing datum are from the upper left tracker are red, those of the lower right tracker are blue. Where the two trackers and the target are not collinear or close to it, the localization regions are small and tight. As the tracker crosses the middle of the plot, the nodes approach collinearity and localization regions grow larger, similar to the single tracker case. Once past the collinear area, the localization regions narrow down again.

This example shows the real strength of the method. Where two or more trackers can detect the target from different angles and communicate those detections, the localization regions are quickly closed in on the target and narrow the location quite well with little computational cost and no uncertainty.

Two Trackers, Limited Communication

Figure 2-7 shows the algorithm's distinctive behavior. Stationary tracking nodes are located at (1000,0) and (0,600). The target begins at (0,0) and this initial position is known to the trackers. In this case, the communication bandwidth is small, so the (1000,0) tracker only receives data from the (0,600) tracker intermittently. As the lower tracker is tracking using only it's own data, the localization region grows, constraining the target less and less. But once data is received from the upper tracker, it quickly constrains the target to a narrow localization region. This cycle repeats as the lower target tracks on its own and then receives outside data.

Figure 2-8 shows the effect of the semi co-operative concept of operations (CONOPS). In this CONOPS, one platform is tracking the target using it's own sensor data

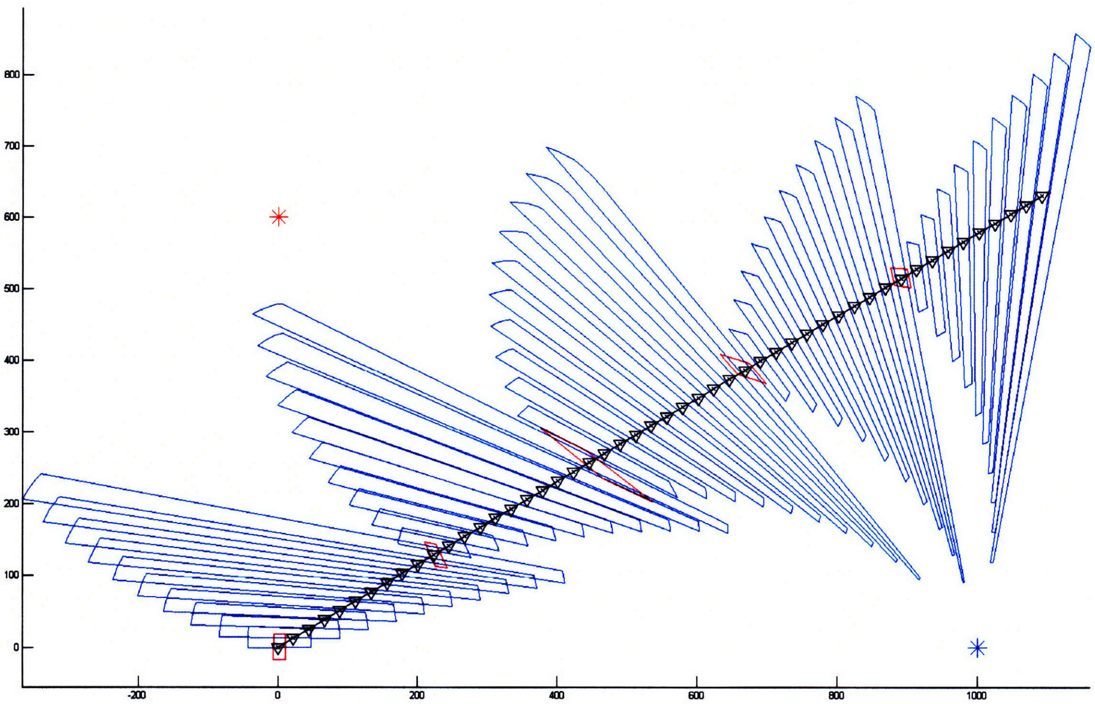

Figure 2-7: Two Trackers, Limited Communication. Trackers at (1000,0) and (0,600). Target position known to be (0,0) at start. Low Communication Bandwidth

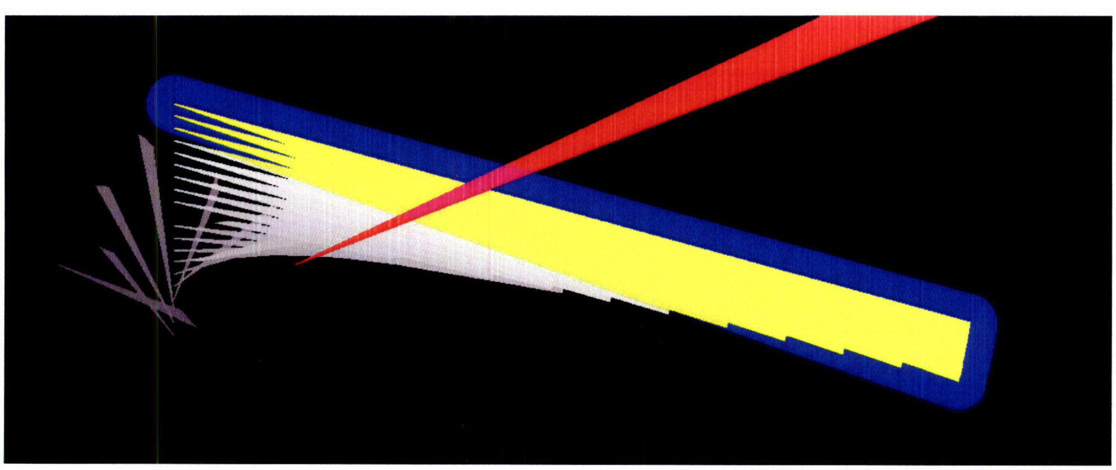

Figure 2-8: Semi Cooperative Screen Capture

primarily with occasional input from another platform. The light gray wedges that point to a line are all sensor data for one platform. Since this data is from one direction only, the bounding regions grow at each iteration, causing the inverted pyramid of gray wedges. Once cross bearing data is received, the region is shrunk down to essentially the bearing data tolerance.

2.5 Conclusion

This chapter has presented the bounded region tracker. The algorithm is effective in tracking targets when two or more trackers are sensing the target and sharing data and can achieve and maintain a favorable geometry with respect to each other and the target. If only one tracker is in contact, if data cannot be shared frequently, or if the node positions all approach a common line, the localization regions quickly degrade to simply bearing data for the target.

Chapter 3

Autonomous Surface Craft Experimental Results

This chapter describes an effort to apply the algorithm presented in the previous chapter to data from the MB06 PLUSNet experiment, held in August, 2006 in Monterey Bay. We first review the setup for the experiment, and then present post-processing analysis of a representative mission.

3.1 Setup

A large scale test of PLUSNet components, their interoperability, and communications and control was conducted in Monterrey Bay in August of 2006. The experiment was named MB06. The components of MB06 serve as a prototype for a future capability PLUSNet. Major components of the experiment included [13]:

- Vehicles
  - Bluefin-21/Odyssey AUVs,
  - Seahorse AUV,
  - Slocum glider AUV,
  - Seaglider AUVs,

- XRay glider AUV.

- Fixed
  - Hydrophone array - vertical (Kelp)
  - Vector sensor array - horizontal
  - Electric field sensors

Additionally, a number of SCOUT autonomous surface craft (ASC) [2] participated in MB06, operating off of R/V PT SUR.

The SCOUT (Surface Craft for Oceanographic and Undersea Testing) ASC's are inexpensive, flexible, highly capable and simple. They were designed to be surface only test beds for AUV algorithms [2] but have come to be appreciated as autonomous vehicles in their own right.

The design starts with a HDPE plastic recreational kayak hull with three watertight compartments and two watertight access ports. The after compartment is modified to hold the steering gear and mounting for a modified trolling motor that is the SCOUT's propulsion. The after compartment also houses the wet portion of the cooling system, including pumps and radiator tubes extending outside the hull into the sea.

The forward compartment holds a radio control (R/C) receiver and is otherwise available for use for payloads.

The middle compartment is not water tight, it is the area intended for the human user in the kayak's original design. This large volume houses a main vehicle computer (MVC) in a watertight enclosure, a deep cycle marine lead acid battery pack consisting of five batteries in its own watertight enclosure, sensor modules in separate watertight cases, and associated cabling.

There are two antenna masts upon which are mounted antennas for R/C, 802.11b WiFi, and a 2.4GHz or 900MHz radio (RF) modem. The aft antenna mast is about one meter high and also mounts strobe lights and an international orange 12in. square signal flag. The flag is important because the SCOUT's have such a small surface projection, they are very difficult to see when deployed, despite their bright yellow hulls.

The SCOUT can be operated out to limited range as a simple R/C vehicle, controlling thrust and steering. The R/C controller determines if the SCOUT acts under computer control or R/C control.

All the functions of the SCOUT and most payload functions, can be controlled by the MVC. The MVC can communicate over the WiFi or RF modem links to other computers, either on shore or deployed on other vehicles. Thus the SCOUT can be controlled autonomously by its own MVC, cooperatively with another vehicles MVC, or remotely by a shore based computer which can include interactive control by a human.[2]

During MB06, the SCOUT kayaks deployed with WHOI MicroModem acoustic communications units[7]. These units allowed the SCOUTs to communicate among themselves and with other platforms participating in MB06. One kayak mounted a MicroModem on a winch to control the deployment depth while another mounted two MicroModems about three meters apart to investigate short baseline navigation.

The SCOUT ASCs filled several different auxiliary roles during MB06 including acting as mobile navigation aids for AUV's and, for one extended period, acting as a fixed navigation beacon when a buoy designed for that purpose was inoperable. When not called upon to directly interact in the larger MB06 experiment, the SCOUT's were run extensively, conducting open ocean experiments in network control, autonomous navigation, cooperation, and environmentally adaptive behaviors.

To test the set-based tracker with field experimental data, albeit in a highly simplified setting, we consider the problem of using two SCOUTs to track the position of a third SCOUT using "simulated" bearing measurements derived from the kayak's broadcasted GPS positions.

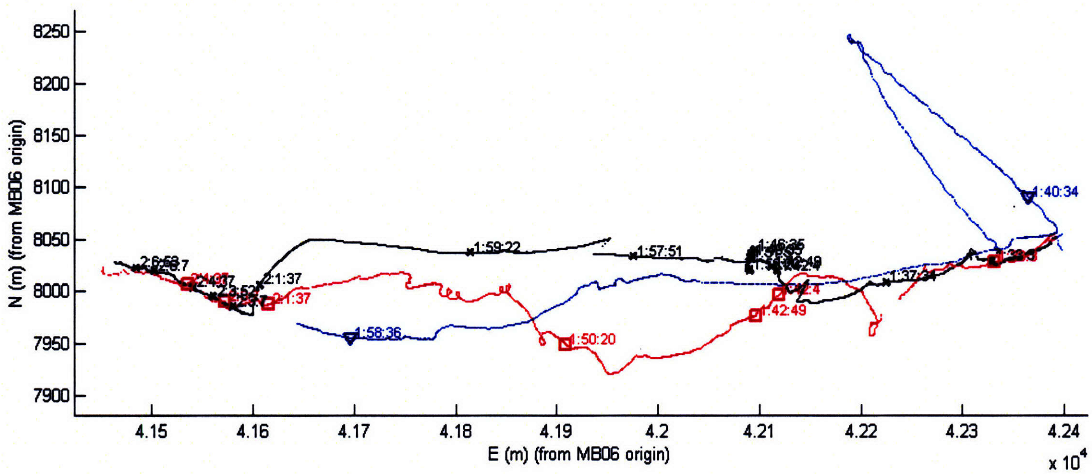

Figure 3-1: SCOUT GPS positions

3.2 Tracker Performance

Figure 3-1 shows a position plot for the GPS positions of three SCOUT ASC's for a typical MB06 experiment or "mission". The coordinates are meters North and East of an arbitrary origin selected for common use at MB06. The three kayaks begin on the right and move to the left over time. In practice, the kayaks are given individual names to distinguish them. Here Andy is the black line and will be the target node. Charlie is red and Elanor is blue, they will track Andy.

This mission was selected to show the effect of geometry on the performance of the algorithm. The best performance (small localization regions) is expected where the tracking nodes bearings to the target are near 90 degrees apart. Worst performance (large regions) is expected when the bearing difference is near zero or 180 degrees, when the trackers and target are all nearly collinear.

Figure 3-2, which plots this bearing difference over time, indicates we should obtain good performance around 01:35 and 02:00 and poor performance around 01:50.

Figure 3-3 confirms the prediction of 3-2: good localization for bearing differences around 90 degrees, poor localization for collinear and near collinear geometries. The localization region areas lag the bearing differences, it takes a little while for the localization to degrade or improve.

Figure 3-4 shows a broad view of the performance of the algorithm in a high sensor

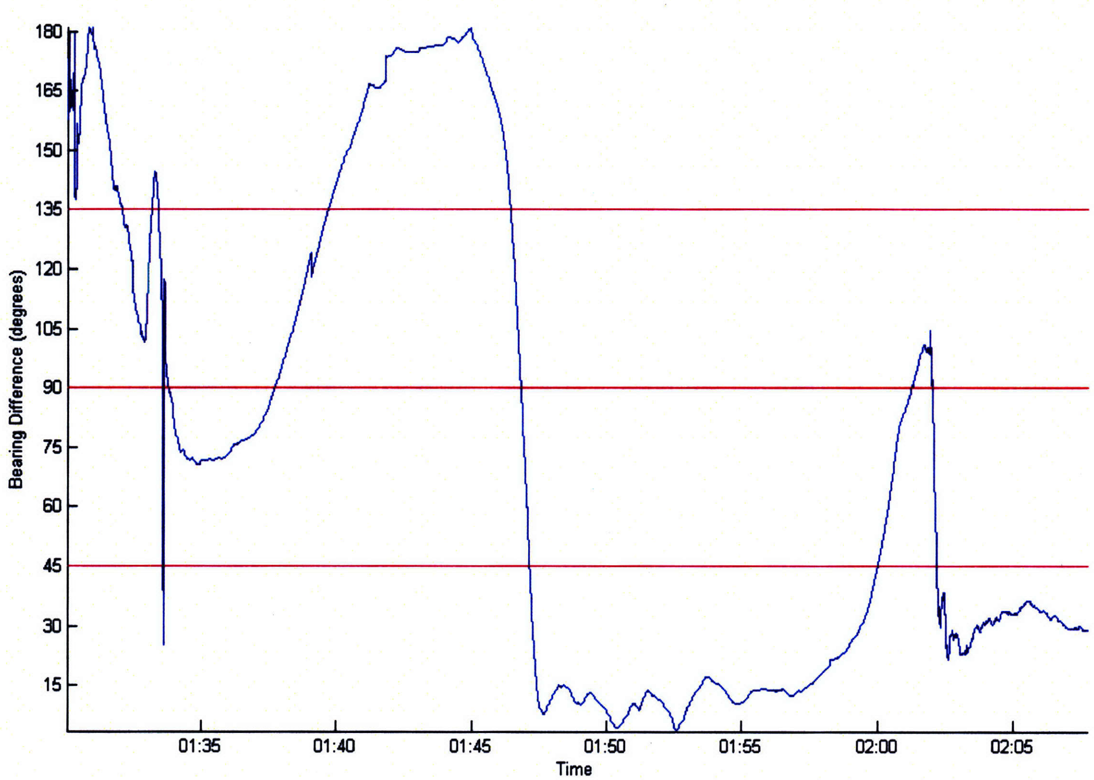

Figure 3-2: Tracker Bearing Differences

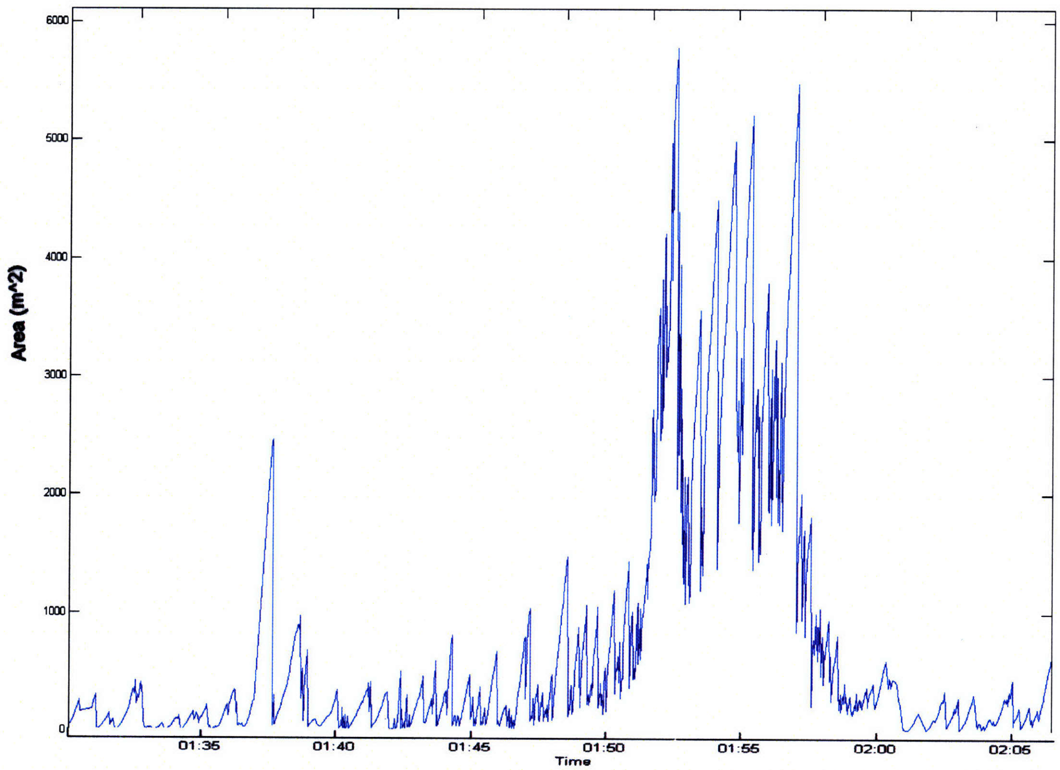

Figure 3-3: Tracker localization region areas vs. time.

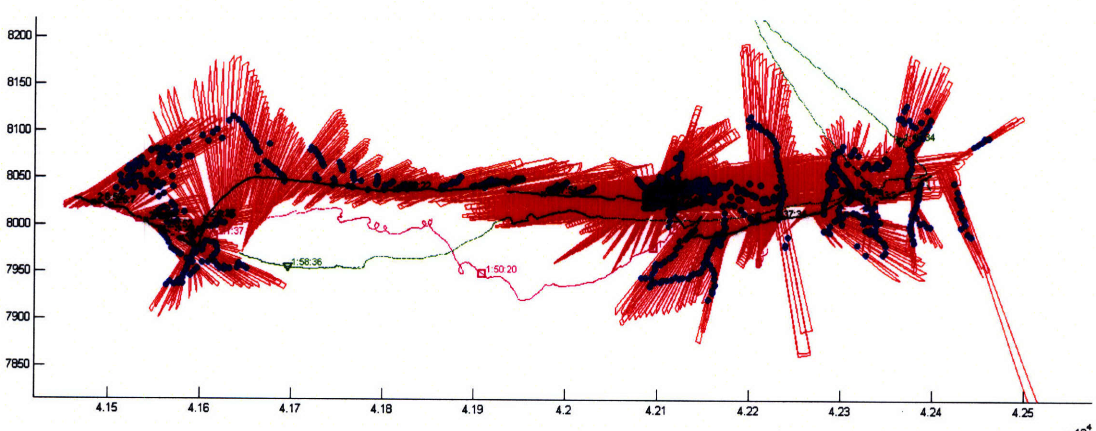

Figure 3-4: Localization regions with centroids marked

data rate, high communications bandwidth scenario. The localization regions are small due to the frequent updates and good cross bearing info (in selected locations). The centroids of the localization regions have been marked as a rough metric for comparison to the ground truth GPS position of the target node. Note that the algorithm itself gives no information of where in the localization region the target may be, only that it must be somewhere within the region.

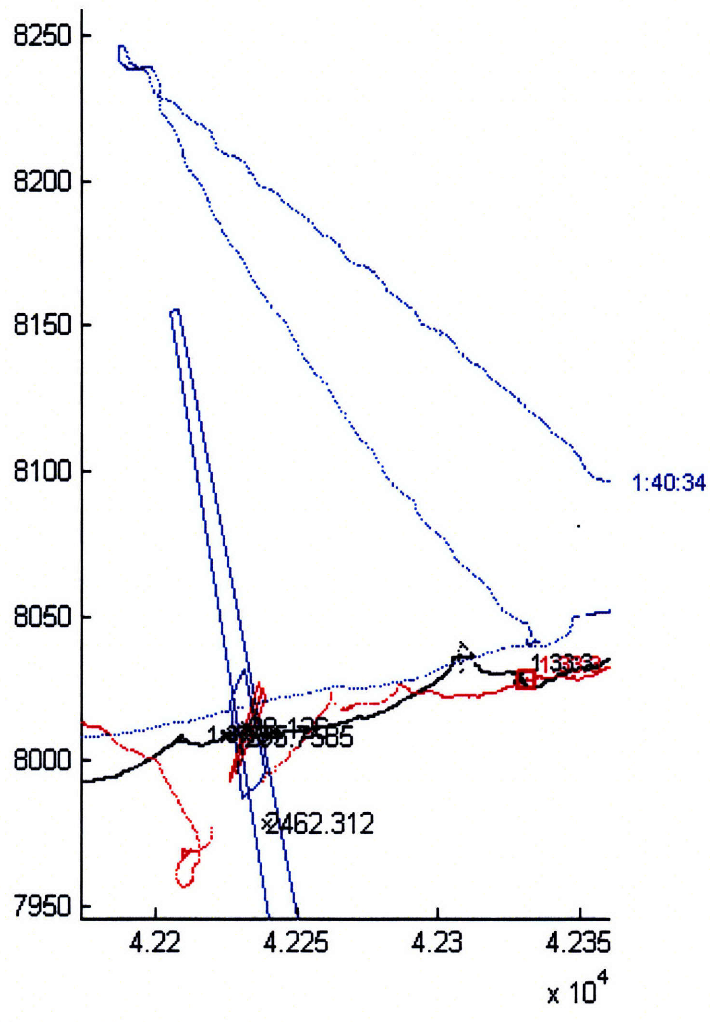

Figure 3-5: Good localization performance due to good cross bearings

Figure 3-5 depicts a period in the mission in which the geometry and communications cooperated to produce excellent localization. At this time, Elanor (blue) is well North of the target, Andy (black) and the other tracker, Charlie (red). Figure 3-6 is a zoomed in view of the target track for this period. Note that the tracks shown in figure 3-6 for Elanor and Charlie are for different times. These trackers were not within the area of figure 3-6 when the localization regions were valid.

In figure 3-6, the regions due to a final bearing from Elanor are shown in blue while those from Charlie are in red. Each region has it's calculated area in square meters labeled at the centroid of the area. We see then that Elanor's first region is $2462m^2$ whereas the first cross bearing from Charlie, shown in red, reduced the region area to only $38m^2$. The time until the next bearing for Elanor causes the region to grow again, back up to $296m^2$ which is again sharply reduced by the cross bearing data from Charlie (the last area is not shown for plot clarity).

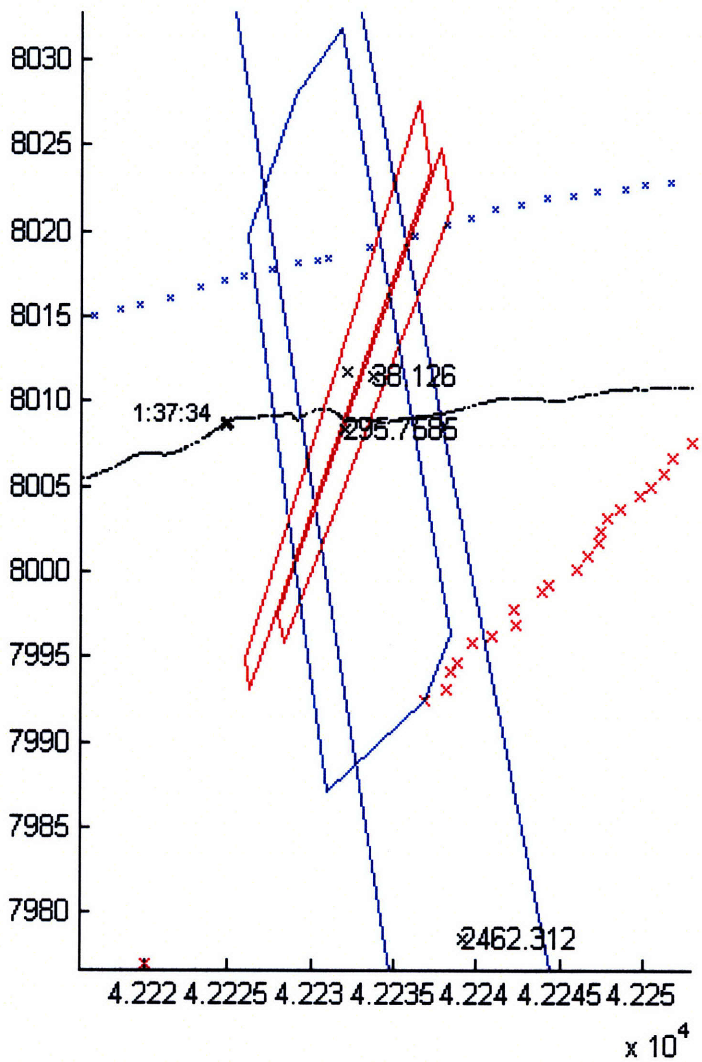

Figure 3-6: Good localization performance due to good cross bearings(detail)

3.3 Summary

This Chapter has described the concept of the Persistant Littoral Underwater Surveillance Network as realized during the large scale experiment in Monterey Bay in Summer of 2006. The SCOUT ASC kayaks which participated in the experiment were described in more detail. Data collected during open ocean experiments using the SCOUT ASCs was used to demonstrate the performance of the bounded region tracker algorithm in a real world setting.

When applied to real world data, the performance of the algorithm was worse than the strictly simulated situations, showing the fragile nature of the algorithm. In any situation where the geometry and communications are not at least very good, the tracker performance quickly degrades. For the real data sets, where gaps in data exist, this maps to a communications blackout and results in large localization regions.

Chapter 4

Conclusion

4.1 Thesis Summary

This thesis has developed a new tracking algorithm for persistent cooperative surveillance with unmanned platforms. Taking motivation on the application side from the PLUSNet project [13], and on the algorithm side from new set-based techniques for sensor network localization [6], we have developed a bounded region approach to cooperative tracking of submerged targets. The method has been tested in pure simulation and also via post processing of an oceanic data set with simulated bearing measurements derived from actual autonomous surface craft positions.

Set-based localization and tracking methods are appealing for their conceptual simplicity and minimal assumptions. The approach has roots in the robot motion planning community, with an underlying theoretical foundation coming from computational geometry. The forte of the approach is not to so much as to say "where the target is" but to say with confidence "where the target is *not*". In some applications, this capability can be very useful, e.g., to maintain a guarantee that a given surveillance region is free of targets, or to deploy network resources to further investigate a potential contact.

In our desired application of ASW, however, the approach has some serious shortcomings. The three shortcomings of the bounded region tracker are as follows:

- **Geometry**: the method is utterly reliant on the trackers achieving and maintaining a favorable geometry with respect to the target
- **Communications**: the method also depends critically on how often the platforms can establish connectivity and how much data can be exchanged.
- **Data Association**: the method is expected to perform poorly if there are many outlier data points and/or many potential targets.

If the geometry is bad, for example, all sensors lying on a line that passes close to the target, then the method degrades to a bearing only tracker. Similarly, poor communications between trackers leaves each tracker to track on it's own, again resulting in a bearing only tracker. Without a more sophisticated target motion model that includes more information than just a region the target must lie within, trackers operating alone and cooperating trackers in poor geometries yield minimally useful information.

4.2 Suggestions for Future Research

With the intended application of underwater covert surveillance, the current communication technology of acoustics falls short in providing bandwidth and stealth. This is a general concern for PLUSNet, not just a concern applicable to this algorithm. An additional concern is that of communications security which is typically addressed with methods that further increase communication bandwidth by adding overhead. Any advances in underwater communications technologies, whether they be acoustic or otherwise, would be of great benefit to this algorithm in particular and PLUSNet in general.

An application scenario where this algorithm could prove viable would be a "Large N" situation where the area of interest is saturated with sensor nodes. One configuration could consist of a bottomed sonar sensor to detect target submarines attached by tether to a floating buoy with a GPS sensor and a radio communications link, possibly similar to WiFi. The float could be camouflaged as a lobster pot float or something similar. To be effective, given the overtness of peer to peer radio communication, there would need to be many expendable nodes to maintain the network despite attrition.

Bibliography

[1] K. Briechle and U. D. Hanebeck. Localization of a mobile robot using relative bearing measurements. *Robotics and Automation, IEEE Transactions on*, 20(1):36–44, February 2004.

[2] J. Curcio, J. Leonard, and A. Patrikalakis. SCOUT — a low cost autonomous surface platform for research in cooperative autonomy. In *IEEE Oceans*, 2005.

[3] T. Curtin, J. G. Bellingham, J. Catipovic, and D. Webb. Autonomous ocean sampling networks. *Oceanography*, 6(3):86–94, 1993.

[4] T. B. Curtin and J. G. Bellingham (Editors). Special issue on autonomous ocean sampling networks. *IEEE J. Ocean Engineering*, 26(4), 2001.

[5] Thomas B. Curtin. Undersea persistent surveillance or connect the dots. PowerPoint from ONR Joint Review of Unmanned Systems Technology Development, Feb 2005.

[6] Carrick Detweiler, John Leonard, Daniela Rus, and Seth Teller. Passive mobile robot localization within a fixed beacon field. In *Proceedings of the International Workshop on the Algorithmic Foundations of Robotics*, New York, New York, August 2006.

[7] Lee Freitag, Matthew Grund, Sandipa Singh, James Partan, and Keenan Ball. The WHOI Micro-Modem: An Acoustic Communications and Navigation System for Multiple Platforms. In *IEEE OCEANS*, Washington DC, 2005.

[8] V. Isler and R. Bajcsy. The sensor selection problem for bounded uncertainty sensing models. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 151–158, april 2005.

[9] D. Meizel, O. Leveque, L. Jaulin, and E. Walter. Initial localization by set inversion. *IEEE Transactions on Robotics and Automation*, 18(3):966–971, December 2002.

[10] Michael Palin and Basil Pao. *Full Circle: A Pacific Journey with Michael Palin.* BBC Books, 1997.

[11] Henrik Schmidt. Nested autonomy a concept of operations for distributed acoustic sensing and surveillance. PowerPoint presentation at NATO Undersea Research Centre, March 2007.

[12] J. R. Spletzer and C. J. Taylor. A bounded uncertainty approach to multi-robot localization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, volume 2, pages 1258–1265, 2003.

[13] Marc S. Stewart and John Pavlos. A means to networked persistent undersea surveillance. In *Submarine Technology Symposium*, 2006.

[14] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics.* MIT Press, 2005.