



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**UNDERWATER ACOUSTIC NETWORK
AS A DEPLOYABLE POSITIONING SYSTEM**

by

Rebecca E. King

June 2012

Thesis Advisor:
Second Reader:

Joseph A. Rice
Grace A. Clark

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE June 2012	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE Underwater Acoustic Network as a Deployable Positioning System			5. FUNDING NUMBERS	
6. AUTHOR(S) Rebecca E. King				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ____N/A____.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) Underwater acoustic communications networks serve to pass data between submerged nodes and a command and control center ashore. Range data recorded incident to regular transmissions between nodes afford the acoustic network an additional use as a navigational aide or tracking system for a vehicle operating in the network domain. Previous studies developed algorithms to solve for the position of the mobile node using node-to-node range data. An alternative localization algorithm is proposed. This study implements the algorithms for experimentation in simulation and an actual maritime environment. Reference nodes are deployed to establish a Seaweb network in Del Monte Lake on the Naval Postgraduate School campus. Experimental range data are recorded and used to plot the path of a surface vehicle towing a network sensor node. GPS fixes are simultaneously recorded for comparison of the calculated track to the true track. This comparison throughout the experiment provides a more definitive measure of the localization algorithms' performance than was possible in previous studies.				
14. SUBJECT TERMS Seaweb, sensor network, acoustic communication, ranging, localization, navigation, acoustic modem			15. NUMBER OF PAGES 125	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**UNDERWATER ACOUSTIC NETWORK
AS A DEPLOYABLE POSITIONING SYSTEM**

Rebecca E. King
Ensign, United States Navy
B.S., United States Naval Academy, 2011

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN APPLIED PHYSICS

from the

**NAVAL POSTGRADUATE SCHOOL
June 2012**

Author: Rebecca E. King

Approved by: Joseph A. Rice
Thesis Advisor

Grace A. Clark
Second Reader

Andres Larraza
Chair, Department of Physics

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Underwater acoustic communications networks serve to pass data between submerged nodes and a command and control center ashore. Range data recorded incident to regular transmissions between nodes afford the acoustic network an additional use as a navigational aide or tracking system for a vehicle operating in the network domain. Previous studies developed algorithms to solve for the position of the mobile node using node-to-node range data. An alternative localization algorithm is proposed. This study implements the algorithms for experimentation in simulation and an actual maritime environment. Reference nodes are deployed to establish a Seaweb network in Del Monte Lake on the Naval Postgraduate School campus. Experimental range data are recorded and used to plot the path of a surface vehicle towing a network sensor node. GPS fixes are simultaneously recorded for comparison of the calculated track to the true track. This comparison throughout the experiment provides a more definitive measure of the localization algorithms' performance than was possible in previous studies.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	OVERVIEW.....	1
B.	OBJECTIVE	2
C.	APPROACH.....	2
II.	ACOUSTIC RANGING	3
A.	UNDERWATER SOUND PROPAGATION	3
1.	Sound Speed	3
2.	Geometric Spreading.....	3
3.	Attenuation	3
4.	Noise	4
5.	Refraction	4
6.	Multipath	4
7.	Doppler Spreading.....	5
B.	FURTHER ENVIRONMENTAL CONSIDERATIONS	5
1.	Location of Deployed Nodes.....	5
2.	Node Mobility	6
C.	SEAWEB RANGE DATA.....	7
1.	Format.....	7
2.	Time Synchronization.....	8
3.	Basic Range Calculation.....	8
III.	LOCALIZATION TECHNIQUES	11
A.	RANGE-BASED APPROACHES.....	11
1.	Received Signal Strength Indicator (RSSI).....	11
2.	Time Difference of Arrival (TDOA).....	11
3.	Time of Arrival (TOA)	12
B.	TOA APPROACHES TO LOCALIZATION	12
1.	Intersecting Circles and Spheres	12
2.	Intersecting Hyperbolas and Hyperboloids.....	12
3.	Unambiguous Solutions	13
IV.	VEHICLE TRILATERATION WITH RANGE DATA	15
A.	GEOMETRY.....	15
B.	INHERENT AND SYSTEMATIC ERRORS	18
1.	Geometric Dilution of Precision	18
2.	Two-dimensional Model	18
3.	Vehicle Motion	19
V.	PREVIOUS WORK ON VEHICLE LOCALIZATION ALGORITHMS.....	23
A.	WEIGHTING METHOD.....	23
B.	CENTER OF MASS METHOD	23
C.	DIFFERENCE LINEARIZATION METHOD	23
1.	Eigenvalues	24

2.	Condition Number of Matrix.....	25
3.	Number of Available Ranges	26
VI.	THE FILTERED PAIRWISE ALGORITHM.....	27
A.	ALGORITHM DESCRIPTION	27
B.	ALGORITHM STRUCTURE	27
1.	Computing All Solutions	27
2.	Comparing All Solutions	28
3.	Averaging All Filtered Solutions	28
VII.	ALGORITHM IMPLEMENTATION FOR SYNTHETIC DATA	29
A.	SETUP AND ASSUMPTIONS	29
B.	CASE STUDIES.....	29
1.	Data Sets with Zero Range Error.....	29
2.	Data Sets with Random Range Error	30
3.	Data Sets with Dwell Time	32
4.	Data Sets with Nodes in a Straight Line	33
5.	Data Sets Varying the Number of Nodes	34
VIII.	DESCRIPTION OF EXPERIMENT	37
A.	EXPERIMENTAL PLAN OVERVIEW	37
B.	SHALLOW LAKE ENVIRONMENT.....	37
C.	IN-AIR TESTING OF EQUIPMENT	40
IX.	PRACTICAL RANGING APPROACHES.....	41
A.	BROADCAST PING	41
B.	SEQUENTIAL PINGING.....	41
C.	LONG BASELINE MODE	42
X.	FURTHER EXPERIMENTAL CONSIDERATIONS.....	45
A.	SPEED OF SOUND CALCULATION	45
B.	GPS ERROR	46
XI.	ALGORITHM IMPLEMENTATION FOR EXPERIMENTAL DATA	49
A.	PROGRAM DESCRIPTION.....	49
1.	Initial Data Processing.....	49
2.	Sorting Data.....	51
3.	Determining Average Localization Error.....	52
B.	EMPIRICAL CALIBRATIONS	53
C.	RESULTS DISCUSSION.....	54
D.	CONDITION NUMBERS REVISITED.....	56
XII.	PRACTICAL IMPLEMENTATION OF LOCALIZATION ALGORITHMS ..	59
A.	DEPLOYMENT CONSIDERATIONS	59
B.	REAL-TIME IMPLEMENTATION CAPABILITY	59
C.	CRITICAL ASSUMPTIONS.....	61
D.	AVAILABILITY OF ADDITIONAL INFORMATION	62
1.	Vehicle Speed and Heading.....	62
2.	Previous Positions	63

XIII. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK.....	65
A. THESIS CONCLUSIONS.....	65
B. FUTURE WORK.....	66
1. Kalman Filters.....	66
2. Experimental Controls	67
C. APPLICATIONS	68
APPENDIX A: DEL MONTE LAKE AND MODEM LOCATIONS	69
APPENDIX B: PLOTTED EXPERIMENTAL RESULTS.....	71
APPENDIX C: CONDITION NUMBERS OF MATRICES	79
APPENDIX D: RESULTS FOR CONDITION NUMBER MINIMUMS	81
APPENDIX E: CORRECTING EIGENVALUE ERROR.....	83
APPENDIX F: STRAIGHT LINE NODE DEPLOYMENT	85
APPENDIX G: PRELIMINARY EXPERIMENTAL PROPOSAL.....	87
LIST OF REFERENCES.....	97
INITIAL DISTRIBUTION LIST	99

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

Figure 1.	Pictured is a model trajectory of a deployed sensor. The graph shows the average displacement of sensors from computer simulations as the velocity of the current is increased. The trajectories of two types of sensors, labeled “Sensor” and “Uw gateway,” are modeled at three bottom depths. From [9].	6
Figure 2.	Tides and currents will affect the location and orientation of an anchored modem. The modem can lie anywhere within a watch circle, whose radius depends on the length of tether and the water depth.	7
Figure 3.	Procomm terminal display of information from Seaweb utility packet. The <i>atep</i> command sends a ping to Node 20 from Node 26. The “Delay” is the round-trip propagation time for the transmission.	8
Figure 4.	Seaweb handshake operation and broadcast ping with dwell time. After [3] and [5].	9
Figure 5.	Overlapping range circles can yield two ambiguous solutions, labeled here as P_2 . The positions of Node i and Node j are known as well as the radii. After [16].	15
Figure 6.	Range circles may not intersect. In the case shown, the point between the two range circles is chosen as the solution.	17
Figure 7.	Range circles can lie within one another as shown. The point at which the circles are closest is chosen as the solution.	17
Figure 8.	An example of GDOP. A small error in one range measurement will lead to a large error in the calculated solutions for certain geometries.	18
Figure 9.	Projection of three-dimensional range onto two-dimensional coordinate system.	19
Figure 10.	Schematic of the effect of vehicle motion on ranging.	20
Figure 11.	Schematic showing the effect of built-in dwell times for the remote nodes. The addition of a dwell time yields a range error equivalent to that found when the vehicle speed is increased.	20
Figure 12.	Comparison of localization error for the three algorithms using synthetic data with introduced error.	31
Figure 13.	A measure of the robustness of each algorithm is found by plotting the average localization error as it increases with the amount of error introduced to each individual range measurement.	31
Figure 14.	The performance of each algorithm with dwell times at the remote nodes is plotted. The synthetic data are created for a vehicle moving 0.5 m/s.	32
Figure 15.	Nodes in a straight line will yield ambiguous solutions using the FP method. Incorporating information about the previous vehicle position, speed and orientation can resolve the ambiguity.	34
Figure 16.	Average localization error for the three methods, varying the number of remote reference nodes.	35

Figure 17.	Diagram of fixed node and mobile node towed by a surface vehicle. Sound speeds, densities and the lake dimensions are estimated for modeling sound propagation in Del Monte Lake.....	37
Figure 18.	Bellhop sound propagation modeling software is used to produce eigenray and impulse response plots given a sound speed profile, medium characteristics, source and receiver locations, and dimensions of the operating space. From [21].	39
Figure 19.	Plots of the two simultaneously logged GPS tracks shows that significant error can exist in the reference track. Tracks with good agreement were chosen for data analysis.	48
Figure 20.	The FP algorithm output is shown for data taken on 16 April 2012. Sections of sparse data occurred when transiting around the fountain or when sequential pings were sent too quickly.....	50
Figure 21.	The first image shows range circles from six sequential pings, the vehicle track, and the average position calculated with the filtered pairwise algorithm with the six returns. The second image shows the same at a higher resolution.	51
Figure 22.	The schematic shows successful echo returns from nodes 1–5. The return from node 6 was unsuccessful. Two echoes from node 3 were recorded in a row. These errors limit the division of data for analysis.....	52
Figure 23.	Calibration for the time offset between Seaweb clocks and GPS logger clocks.	53
Figure 24.	Determination of the sound speed that minimizes localization error.	54
Figure 25.	Example of discontinuities over a synthetic track due to sequential pinging.	55
Figure 26.	Average localization error as it increases with the amount of error introduced to each individual range measurement. The FP method is compared to the DLC method that uses the minimum matrix condition number.	58
Figure 27.	A vehicle’s speed can be used to solve for its future positions.	62
Figure 28.	Schematic of a Kalman filter procedure. After [25].	67

LIST OF TABLES

Table 1.	Comparison of localization error for the three algorithms using error-free synthetic data.	30
Table 2.	Average localization error for the three methods when a dwell time is used for the remote nodes.	32
Table 3.	A conversion factor between degrees and distance, specific to the operating area, is found by choosing two points separated by 0.00001 degrees in either direction.	45
Table 4.	The speed of sound in the lake is found using ranges between anchored modems. This method yields a large range of solutions due to errors in the measurements.	46
Table 5.	GPS coordinates for the anchored modems when deployed and recovered.	46
Table 6.	Ranges between anchored modems are calculated using their GPS coordinates.	47
Table 7.	Average localization error for the FP and DLC methods for experimental data.	54
Table 8.	Average localization error for experimental data using the FP method, the DLC method using one condition number, and the DLC method that uses the minimum of all three possible condition values of the inverted A matrix.	57
Table 9.	The number of waypoints calculated for each track depends on the length of the track and number of range data used to find each waypoint.	60
Table 10.	The DL method uses combinations of three range data, whereas the FP method uses combinations of two range data to find a solution. The difference has a large effect on algorithm computation time.	60
Table 11.	Comparison of computation times for the FP and DLC methods using experimental data, depending on the number of ranges used to calculate each waypoint.	61

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

GPS	Global Positioning System
DL	Difference Linearization
FP	Filtered Pairwise
UUV	Unmanned Undersea Vehicle
ISI	Intersymbol Interference
HFM	Hyperbolic Frequency Modulated
RTS	Request to Send
CTS	Clear to Send
SRQ	Selective Automatic Repeat Request
RSSI	Received Signal Strength Indicator
TDOA	Time Difference of Arrival
TOA	Time of Arrival
LS	Least Squares
GDOP	Geometric Dilution of Precision
DLC	Difference Linearization with Condition check
MSFK	Modified Souci/Fachmann/Kraut
LBL	Long Baseline
CTD	Conductivity-Temperature-Depth
UTC	Universal Time Coordinated
EKF	Extended Kalman Filter

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Underwater acoustic modems arranged in a fixed network can be used as a means of navigation for an underwater mobile node operating in the network domain. Ranges between stationary nodes and the mobile node can be obtained as a by-product of through-water acoustic communications. Various algorithms have been developed to calculate the position of an underwater vehicle using node-to-node ranges. Testing these algorithms with a surface vehicle allows direct comparison of the calculated tracks to that provided by the Global Positioning System (GPS).

Seaweb is a Navy-developed acoustic network technology that uses underwater acoustic modems to communicate information to an onshore command and control center via a gateway node. The gateway node can be a moored buoy or a surface vehicle. Range data measured incident to regular transmissions between nodes afford the Seaweb network an additional capability as a navigational aide or tracking system.

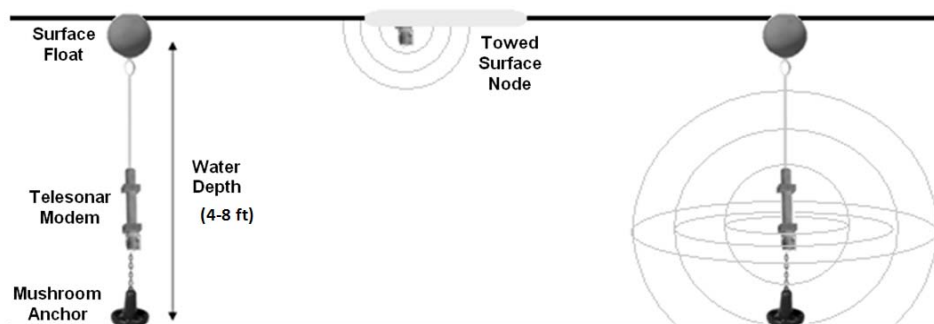


Figure 1. Schematic of deployed underwater nodes and towed mobile surface node in Lake Del Monte.

Ranges are calculated with the round-trip propagation time of transmitted signals and the sound speed in the environment. At regular time intervals during the vehicle's run, the mobile node transmits a broadcast ping. Upon receipt at each fixed node, that node returns an echo including the node's specific address. The mobile node records the ping transit time and node address for each return echo.

The electro-acoustic transducers are omnidirectional. Assuming a two-dimensional system, for each range measurement, there is a range circle about the fixed node that describes the vehicle's possible position. The exact position can be found at the intersections of overlapping range circles. Many sources of systematic error exist including those due to the two-dimensionality assumption, signal processing delays, variations in sound speed, and uncertainty in the fixed node locations. Overlapping range circles will not intersect at one point and will yield ambiguous solutions, as seen in Figure 2. The localization algorithms developed in previous theses use different methods to choose the solution that is most correct.

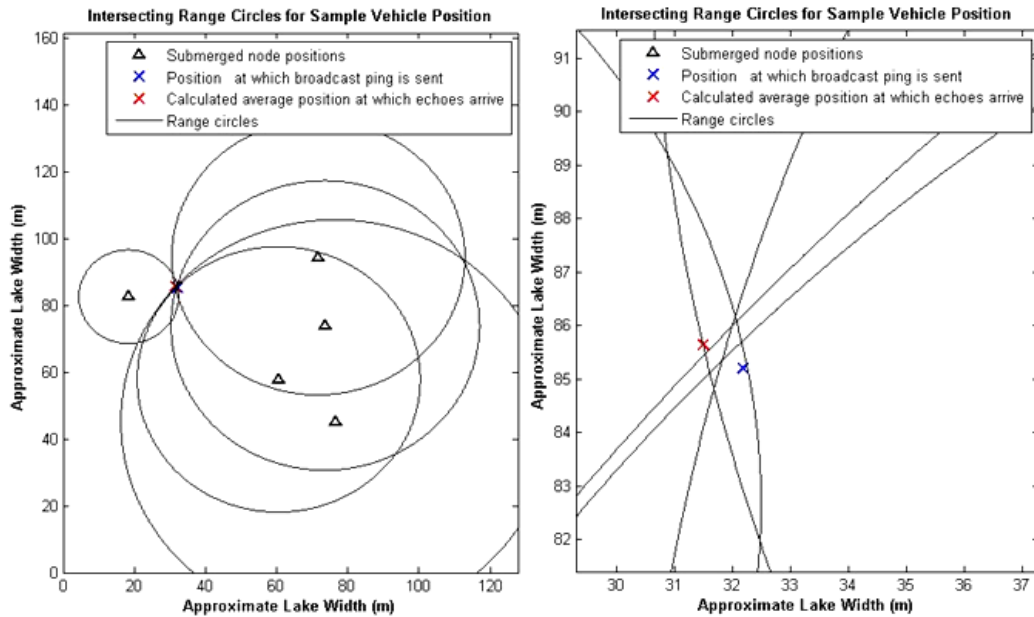


Figure 2. Overlapping range circles from five submerged nodes, and a close-up view, for a given position of the surface node. The average position is calculated with a difference linearization algorithm for synthetic data.

The localization algorithm developed by LT Michael Reed, USN uses a difference linearization method, inspired by the GPS implementation. This method linearizes a set of three range circle equations to find the two unknown x and y coordinates. Solutions are found for all combinations of three range circles and averaged to obtain a final solution.

Pairwise algorithms use sets of two range circles to calculate solutions. The correct solutions must be determined from each pair of ambiguous solutions. Solutions

are found for all combinations of two range circles and averaged to obtain a final solution. The benefits of each method are discussed.

Synthetic data are created to test the accuracy and robustness of the localization algorithms. Assumptions are made for the mathematical models, including straight-line propagation of sound and a two-dimensional representation, which closely approximate the shallow lake. Random error is introduced to each range measurement to test the error tolerance of each algorithm. Synthetic data are generated for random network configurations and random vehicle paths within the bounds of the lake dimensions. Synthetic results are also analyzed to find preferred network geometries.

The difference linearization (DL) algorithm performs better than the pairwise algorithms studied in previous theses, but encounters problems when little range data are available per fix due to fewer available combinations of range equations. The method uses matrix inversion and also yields errors for certain network geometries.

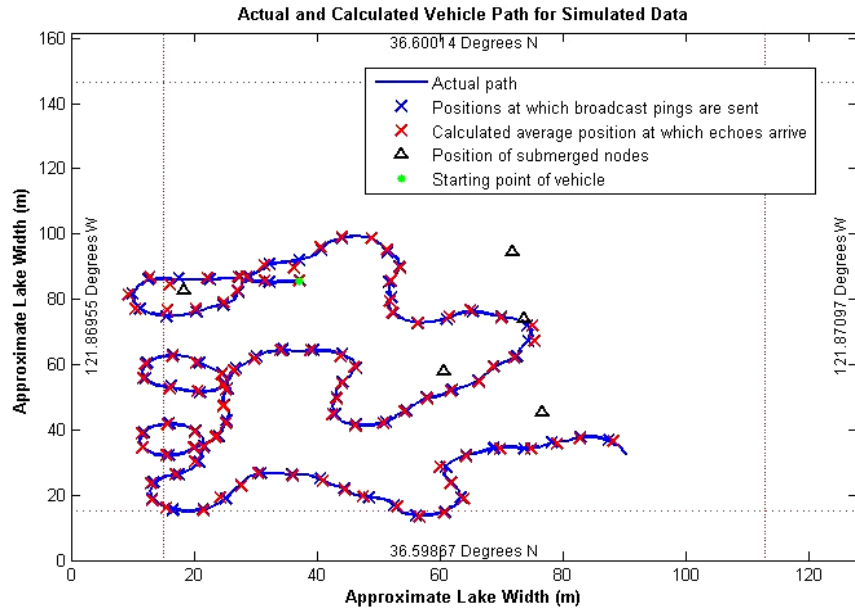


Figure 3. Graphical representation of the performance of the filtered pairwise localization algorithm for a sample synthetic track.

Because of the shortcomings of the existing algorithms, a new “filtered pairwise” localization algorithm is proposed. The filtered pairwise (FP) algorithm uses all available combinations of two range circles to find a position solution. The method of calculating

the solution for two range circles depends on the particular geometry. In cases where the two range circles do not overlap, the point between the two range circles is chosen. In cases where two solutions exist, all solutions are compared to one another. The error between solutions is minimized to eliminate one of the two ambiguous solutions. A simple average of the remaining solutions is done to find the final estimated position.

The algorithm is not mathematically complicated, but uses multiple logic structures for solution comparisons. The computation time is less than that for the difference linearization method. For simulations with ample range data per fix, the average error for the filtered pairwise method is larger. For situations with limited range data per fix, error due to matrix inversion in the difference linearization method makes the filtered pairwise method a good alternative.

A Seaweb acoustic network was established in the Del Monte Lake on the Naval Postgraduate School campus in April 2012. The Seaweb network enabled testing with experimental controls. A series of experiments was performed with a towed surface node. The quiet boat towed a sled fitted with the mobile node and GPS loggers to simultaneously record GPS fixes and range data from the submerged Seaweb nodes. Localization performance with experimental range data can be quantified by comparison to the GPS track.

Previous studies have shown Seaweb positioning to outperform inertial navigation. Submarines and unmanned underwater vehicles must surface periodically to reestablish their position via GPS due to errors inherent in inertial navigation. The goal of this research is increased accuracy of Seaweb range data and tracking algorithms to improve navigation of submerged vehicles. This capability can enable future deployable underwater ranges.

ACKNOWLEDGMENTS

I would like to thank my thesis advisor, Professor Joe Rice, for his help and hard work throughout the year. Secondly, I would like to acknowledge Professor Grace Clark for her advice and constructive criticism. My mentor from SPAWAR Systems Center in San Diego, Chris Fletcher, was incredibly helpful in teaching me how to use the Seaweb equipment and in executing the lake experiments on campus. I would like to thank the physics department machinist, Steven Jacobs, who enthusiastically gave his time to help modify the experimental equipment for use in the lake. Vicky Taber's help was instrumental in coordinating the use of Del Monte Lake and the row boat with the NPS environmental office. Finally, I would like to thank my fiancé, Ensign Michael Moberg, for his help in the lake experiments and support throughout the year.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OVERVIEW

Networks of underwater acoustic sensors are being advanced for a variety of applications including passive environmental data collection and surveillance. Networked underwater sensors can further be used as navigational aids for surface and subsurface vehicles.

U.S. Navy Seaweb networks use fixed and mobile distributed nodes to transport communications to and from onshore command and control centers. Seaweb modem firmware is used with commercial off-the-shelf acoustic modems that are adaptable for shallow or deep ocean operations, with use on surface vehicles, submarines and unmanned underwater vehicles (UUVs). The standard commercial modem firmware supports simple node-to-node communications. The Seaweb firmware supports network architectures and allows sensors to act as repeater nodes with link-layer and network layer protocols, described in Chapter II [1]. A description of Seaweb and references to past Seaweb operations are included in [2].

Seaweb modems can be anchored and suspended above the ocean floor, towed by a surface vehicle, fixed to a buoy, or integrated onto a mobile underwater vehicle. Each modem has a specific address that is included in all data packet transmissions. Once deployed, a network discovery is initialized that autonomously establishes an ad hoc network topology for communications. Anchored modems, referred to as repeater nodes, repeat information through the network to a gateway node on a moored buoy or surface vehicle. The gateway node sends or receives information from a local or onshore operator. All network nodes can be accessed via the gateway node.

In this thesis research, anchored modems communicate acoustically with a modem on a towed surface vehicle that is connected via serial port to a computer. Anchored modems are also referred to as remote nodes. The towed modem is referred to as the local node.

B. OBJECTIVE

Seaweb communications provide node-to-node range information. A mobile vehicle equipped with a network node can utilize the range information from fixed nodes to track its position, using a range-based localization algorithm. Previous studies by [3]–[5] have shown this method to outperform the inertial navigation of an unmanned underwater vehicle and to avoid the need of periodically surfacing to reestablish its position via GPS, due to fix expansion inherent in inertial navigation. Increased accuracy of Seaweb range data and tracking algorithms will aid navigation of manned and unmanned submerged vehicles and can be the basis for future deployable underwater ranges.

C. APPROACH

This thesis reviews existing range-based localization algorithms. Previous thesis work by [3]–[5] included development and testing of localization algorithms with simulated and experimental node-to-node range data. The best-performing algorithm developed in [5] is tested with synthetic data and a new algorithm is proposed.

A Seaweb network was established in the Del Monte Lake on the Naval Postgraduate School campus in April 2012. The Seaweb network enabled testing with experimental controls. A series of experiments were performed with a mobile surface node. A quiet boat towed an instrumented sled, fitted with the mobile node and GPS loggers. Ranges between underwater modems and the towed surface node were recorded while simultaneously recording GPS fixes. The experimental data are used to test the existing and new algorithms. The GPS track is a check on the performance of the localization algorithms.

II. ACOUSTIC RANGING

Underwater ranging with acoustic signals is a challenge due to dynamic environmental conditions. Spatial and temporal variation of the medium influences sound propagation characteristics as described in [6]. Further, anchored references are not stationary due to the dynamic environment. These effects and basic Seaweb acoustic communications are described.

A. UNDERWATER SOUND PROPAGATION

1. Sound Speed

The sound speed in water is five orders of magnitude less than the speed of electromagnetic communications in air. This limits the bit rate of communications. Long round-trip propagation delays also introduce error for mobile sensors, whose movement during the course of ranging may be significant. Sound speed in water is highly variable and depends on salinity, temperature and depth. The surface layer is more dynamic than deeper ocean layers and will have greater temporal and spatial variations in sound speed.

2. Geometric Spreading

Sound energy spreads as it travels out from a source. This effect only depends on range from the source and is independent of frequency. When an acoustic wavefront spreads spherically from a source, the acoustic intensity is proportional to $1/r^2$, where r is the range from the source. Cylindrical spreading occurs at ranges where the sound is trapped in a channel by surface and bottom boundaries. The intensity amplitude of the wavefront is proportional to $1/r$. Geometric spreading limits the range of acoustic communications for a given source strength and receiver sensitivity.

3. Attenuation

In addition to the geometric drop in signal strength, signals are attenuated over a distance due to effects such as scattering and absorption. Attenuation depends on a number of variables including range, frequency, temperature, salinity, pH, viscosity, depth, and bottom and surface roughness.

4. Noise

Man-made and ambient noise can limit the range of acoustic communications. Noise sources include machinery, cavitation, shipping, flow noise, electronic noise, noise due to the sea state, and biologics. Noise levels vary for different frequency bands.

5. Refraction

Waves that are obliquely incident on a boundary between two media will bend according to Snell's law

$$\frac{c_1}{\cos \theta_1} = \frac{c_2}{\cos \theta_2} = \text{constant} \quad (1)$$

where c is the sound speed and θ is the grazing angle of the ray measured from horizontal. For a constant sound speed gradient, waves will follow circular paths with a radius of

$$R = \frac{c(z)}{|g| \cos \theta(z)} \quad (2)$$

where z is the depth, $c(z)$ is the depth-dependent sound speed, g is the sound speed gradient defined as $g = \Delta c / \Delta z$ and θ is the angle from horizontal at the depth z . Typical radii of curvature for sound rays are on the order of kilometers. Thus, for short ranges, straight line propagation of acoustic signals is a good estimate.

6. Multipath

Multipath signals occur when the transmitted signal follows multiple paths to the receiver, each with a different propagation distance and travel time. When using propagation time to determine range, multipath can lead to over- or underestimations. Multipath effects are described in [6]–[8]. The phenomenon occurs due to the omnidirectionality of the transducer and boundary reflections. If the source or receiver is near the bottom or surface boundary, reflections arrive near in time to the direct path signal and can cause destructive interference. Impulse responses show that the amplitude of reflected signals can sometimes be stronger than the direct path signal. Multipath signals

can also arrive earlier than direct path signals. If obstacles obstruct the line of sight from source to receiver, multipath signals can be mistaken for the direct signal.

Increasing the distance of modems from the bottom and surface boundaries usually will decrease multipath interference. Intersymbol interference (ISI) can be minimized by lowering the data rate and increasing symbol duration. Guard times between symbols allow multipath signals to die out.

7. Doppler Spreading

Doppler spreading occurs when the source and receiver are in motion relative to one another. A given frequency is shifted from the source frequency, f_o , according to

$$f = f_o \left(\frac{1 + \frac{v_r}{c}}{1 - \frac{v_s}{c}} \right) \quad (3)$$

where v_r is the receiver velocity, v_s is the source velocity, and c is the sound speed in the medium. Multipath signals that arrive at the receiver at different angles will experience different Doppler shifts due to the projection of the relative motion. Different frequencies also experience a different Doppler shift.

B. FURTHER ENVIRONMENTAL CONSIDERATIONS

1. Location of Deployed Nodes

Difficulties in node deployment cause uncertainty about the node's position on the sea floor. [9] discusses the trajectory of a deployed node as it travels to the sea floor when ocean currents are present.

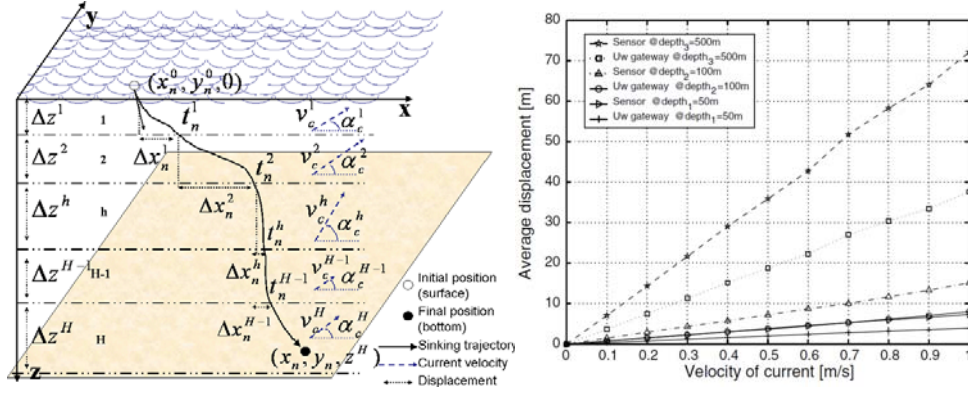


Figure 1. Pictured is a model trajectory of a deployed sensor. The graph shows the average displacement of sensors from computer simulations as the velocity of the current is increased. The trajectories of two types of sensors, labeled “Sensor” and “Uw gateway,” are modeled at three bottom depths. From [9].

The trajectory equations assume complete knowledge of the ocean currents as they vary with depth. This knowledge of deployed nodes is impractical, but simulations give a good estimation of the horizontal displacement to be expected. Shallow sensor deployments are much less impacted by currents. The difference in “Sensor” and “Uw gateway” trajectories, shown in Figure 1, vary based on the shape and weight of each particular object.

2. Node Mobility

When anchored, nodes suspended above the ocean floor will drift due to currents. Buoys can also be affected by wind forces. Tides and currents can affect the location and orientation of modems and surface floats as shown in Figure 2. The area in which the modem might be located is bounded by a watch circle. The radius of the circle is determined by length of the tether and the water depth. The anchor position may also migrate due to a changing sea floor. A modem on a surface buoy can use GPS to update its position, but the exact position of submerged nodes is difficult to determine.

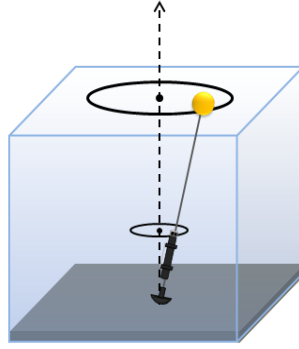


Figure 2. Tides and currents will affect the location and orientation of an anchored modem. The modem can lie anywhere within a watch circle, whose radius depends on the length of tether and the water depth.

C. SEAWEB RANGE DATA

1. Format

The local node can send a number of 9-byte specialized data packets, termed utility packets. In this thesis, the basic ping command is used, which first sends a ranging signal from the local node to a remote node, followed by a PING utility packet. A matched filter at the remote node detects the ranging signal, which is a Hyperbolic Frequency Modulated (HFM) chirp. The time of arrival at the remote node is identified at the peak amplitude of the incoming chirp. In most cases, the multipath signal amplitudes will be lower than the direct path signal and will not lead to false ranges.

The remote node replies with a ranging signal and ECHO utility packet. The local node uses the time difference between the outgoing signal and the incoming signal to calculate the round-trip propagation time, termed the “delay” time, and range. The delay time already accounts for the processing time at the remote node. Procomm, by Symantec, is a terminal emulation tool that can be used for data communications, as described in [10]. Procomm is used to send commands to the local modem and display the received information. A sample ping command and response displayed in the Procomm terminal are shown in Figure 3.

```

atep20
Send Ping-time out: 009.9 secs user:11>
XMT:[20]<[26] PING Time: 19:57:45.6796
RCV:[26]<[20] ECHO Time: 19:57:48.2218
LOCAL CMD RESPONSE(63):
Range Data: Node 26 to Node 20; Delay 151.0 ms; Range 113.2 m

```

Figure 3. Procomm terminal display of information from Seaweb utility packet. The *atep* command sends a ping to Node 20 from Node 26. The “Delay” is the round-trip propagation time for the transmission.

The range value is calculated with the delay time and a sound speed of 1500 m/s. This is intended for use in an ocean environment, but provides an upper estimate of the range when operating in freshwater. The delay time value is given to a resolution of 0.1 milliseconds. Immediate subsequent range measurements, taken between two submerged modems that are approximately stationary, show variation of at most 1 millisecond in the delay times. For a sound speed of 1481 m/s, 1 millisecond error corresponds to a range of 1.481 meters.

2. Time Synchronization

For certain localization schemes, time synchronization among the network nodes is critical. The accuracy of clocks will drift over time. The clock times can vary by an offset or can be skewed by a scalar factor.

When the round-trip delay time is calculated by using both the arrival time at the remote node and then the arrival time at the local node, error from clock differences can be introduced. However, if the round-trip delay time is calculated at the local node using transmission and arrival times, as is done in all methods considered here, no clock synchronization is needed.

3. Basic Range Calculation

The range from the remote node to the local node can be estimated as

$$r_{ij} = c(\Delta t_{ab}/2) \quad (4)$$

where r_{ij} is the range between the local node i and the remote node j , c is the sound speed for the environment, and Δt_{ab} is the round-trip delay time calculated at the local node.

The Broadcast Ping Seaweb command transmits a signal from the local node to neighboring remote nodes in the network. The local node then waits for responses for a set period of time. The remote nodes each have an assigned random dwell time. After receiving a broadcast ping, the remote node waits for its specific dwell time before replying to the local node. The dwell time is used to ensure return signals from the neighboring nodes do not interfere with one another. The dwell time is subtracted from the elapsed time at the local node, which then computes the range.

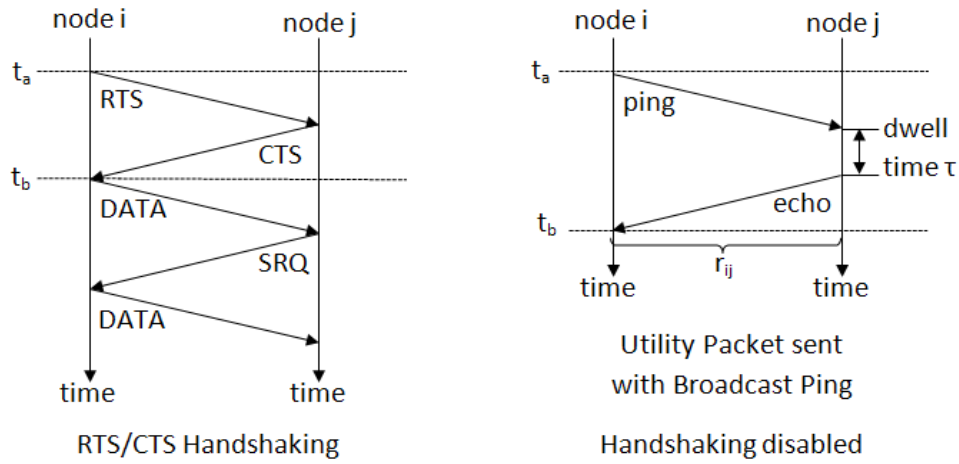


Figure 4. Seaweb handshake operation and broadcast ping with dwell time. After [3] and [5].

Using Equation (4) to calculate ranges requires a number of assumptions. The equation is true if the speed of sound is spatially and temporally constant, sound propagates in a straight line, the remote nodes and the local node are stationary, and there is no error in the calculation of the elapsed time.

The Seaweb firmware uses handshaking when transmitting data. The local node will send a request-to-send (RTS) message to a remote node that takes the remote node out of its idle, low-power state. The remote node replies with a clear-to-send (CTS) message to indicate it is ready to process incoming data. The local node then sends the data packet. If errors exist in the data packet, the remote node sends a selective automatic repeat request (SRQ) for the portion of data with errors. The local node replies with the truncated data. This SRQ/DATA exchange repeats for a set number of times.

When handshaking is used in operational situations, range data are collected for all RTS/CTS transmissions. For the experiment discussed in this thesis, all range data were collected using the PING/ECHO dialogs rather than from RTS/CTS handshaking.

III. LOCALIZATION TECHNIQUES

A. RANGE-BASED APPROACHES

A variety of localization schemes exist for through-air communications, which do not directly translate to the underwater environment. Sensor networks and localization schemes must be designed to work in a spatially and temporally changing medium. Most underwater acoustic localization techniques use range measurements with static references. A 2010 survey of localization schemes by [8] and [11] classify range-based approaches into three categories.

1. Received Signal Strength Indicator (RSSI)

The RSSI approach to localization uses the strength of incoming signals to determine the distance to a source. The source strength must be known and a reliable model for the attenuation over range is needed. The RSSI method for range-determination is not ideal in an underwater environment due to the many variable contributions to transmission loss and the existence of multipath

2. Time Difference of Arrival (TDOA)

The TDOA approach is commonly used in localization problems. This approach requires two separate time-delay measurements between two objects to calculate a range. For example, a node might send an radio transmission and an acoustic signal simultaneously. The range to a receiver is calculated based on the known propagation speeds of the two signals and the difference in arrival times. A related technique is that described in [12]. Distances to an object are found with the time delay between incoming multipath signals from the same transmission. This method requires a good model of the multipath propagation and the geometry of the nodes. In general, TDOA approaches are not well suited for a variable underwater environment.

3. Time of Arrival (TOA)

The commonly used scheme for underwater localization is the TOA approach. This uses transmission times between nodes and the speed of propagation to calculate a distance. If the ranges are calculated using one-way transmissions, clock synchronization is required. The accuracy of this approach requires accurate knowledge of the sound speed and the length of sound propagation paths. The data needed for the TOA approach is already collected with Seaweb sensor network communications, thus the TOA approach is used for thesis.

B. TOA APPROACHES TO LOCALIZATION

1. Intersecting Circles and Spheres

When using omnidirectional transducers, the direction of the remote node with respect to the local node is unknown. The range calculated with Equation (4) is the radius of a sphere centered at the remote node, on which the local node is located. In two dimensions, the range is the radius of a circle, herein called a range circle, centered at the remote node. For localization with Seaweb range data, the positions of the remote nodes are assumed to be known. Overlapping range circles will pin down the location of the local node. In general, three overlapping circles are needed to uniquely find the local node. For three dimensions, four overlapping range spheres are needed. The solutions for overlapping range spheres are worked out in [13]. Solutions for overlapping range circles are found in Chapter IV.

2. Intersecting Hyperbolas and Hyperboloids

An alternative method to using overlapping circles is described in [11]. If the range measured by a vehicle to the node N_1 is R_1 and the range measured to node N_2 is R_2 , the vehicle will be located somewhere on a hyperbola for which the value $(R_1 - R_2)$ is constant. [11] proposes using intersecting hyperbolas, or hyperboloids in three dimensions, to find the vehicle's position. Two intersecting hyperbolas will always yield one solution, versus the intersecting circles approach that gives two ambiguous solutions.

3. Unambiguous Solutions

Three methods for finding an unambiguous solution for overlapping range circles are discussed in Chapter V. A fourth, commonly used approach is the Least Squares (LS) algorithm, which gives best position estimate by finding the solution that minimizes the squared range error. The LS approach is used to solve non-linear equations and to circumvent the problem of ambiguous or no solutions when using overlapping range circles. An LS algorithm is also used for minimizing error with intersecting hyperbolas. The range error, ε_i , for node i can be defined as

$$\varepsilon_i = r_i - \sqrt{(x - x_i)^2 + (y - y_i)^2} \quad (5)$$

where (x, y) is the solution, (x_i, y_i) is the remote node location, and r_i is the measured range. [14] describes various methods of solving for

$$\min \sum_{i=1}^N \varepsilon_i^2. \quad (6)$$

Typically, the range equations are approximately linearized and solved iteratively or directly with matrix calculations. As described in [15], the LS algorithms are good for cases when range error values cannot be assigned a confidence value or weight based on a predictable error model. [15] investigates the sensitivity of the intersecting hyperboloids and intersecting spheres localization approaches with LS algorithms. [11] proposes an alternative to the LS method for intersecting hyperbolas, however [15] and [13] find the intersecting spheres method is much more tolerant to errors in the range data in general.

For this thesis, the localization problem is two dimensional. Intersecting circles are used. An LS algorithm is not used. The handling of ambiguous solutions and non-intersecting circles for sets of two range circles is described in Chapter IV and Chapter VI. The difference linearization method is also investigated and described in Chapter V.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. VEHICLE TRILATERATION WITH RANGE DATA

Ideal range data will yield range circles that overlap at a point. In practice, errors in range data will yield circles that overlap at two positions or range circles that do not overlap. The solutions for each case are calculated and related errors are discussed.

A. GEOMETRY

The intersection points of two overlapping circles with known center positions and known radii can be calculated as shown.

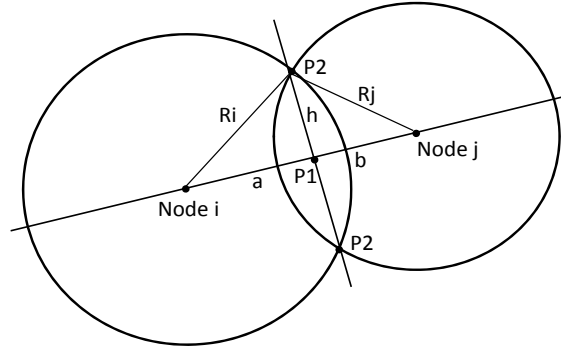


Figure 5. Overlapping range circles can yield two ambiguous solutions, labeled here as $P2$. The positions of Node i and Node j are known as well as the radii. After [16].

Point $P1$ is first defined with coordinates (x_{P1}, y_{P1}) between the two nodes as shown in Figure 5. The distance from $P1$ to Node i , with coordinates (x_i, y_i) , is defined as a . Likewise the distance from $P1$ to Node j , with coordinates (x_j, y_j) , is defined as b . The intersection points are a distance h from point $P1$ as shown. This leads to the relationships

$$\begin{aligned} R_i^2 &= a^2 + h^2 \\ R_j^2 &= b^2 + h^2 \end{aligned}$$

and

$$d = a + b = \sqrt{(\Delta x_{ij})^2 + (\Delta y_{ij})^2} \quad (7)$$

where Δx_{ij} and Δy_{ij} are defined by

$$\Delta x_{ij} = x_i - x_j$$

$$\Delta y_{ij} = y_i - y_j .$$

First solve for the unknown variables a and h .

$$\begin{aligned} (d-a)^2 &= b^2 \\ d^2 - 2ad + a^2 &= R_j^2 - h^2 \\ d^2 - 2ad + a^2 &= R_j^2 - R_i^2 + a^2 \\ a &= \frac{(R_i^2 - R_j^2 + d^2)}{2d} \\ h &= \sqrt{R_i^2 - a^2} \end{aligned} \tag{8}$$

The center point can now be found using the relationship

$$\frac{(P_1 - Node_i)}{(Node_j - Node_i)} = \frac{a}{d} ,$$

from which it can be shown

$$\begin{aligned} x_{P1} &= x_i + \frac{a}{d}(x_j - x_i) \\ y_{P1} &= y_i + \frac{a}{d}(y_j - y_i) . \end{aligned}$$

A relationship for the perpendicular lines in terms of $P2$ and known variables is written

$$\frac{(x_{P2} - x_{P1})}{(y_j - y_i)} = \frac{-(y_{P2} - y_{P1})}{(x_j - x_i)} = \frac{\pm h}{d}$$

to find

$$x_{P2} = x_{P1} \pm \frac{h}{d}(y_j - y_i) \tag{9}$$

$$y_{P2} = y_{P1} \mp \frac{h}{d}(x_j - x_i) \tag{10}$$

This formulation from [16] is general for any choice of Node i and Node j .

If the circles do not intersect, the value of h , calculated with Equation (8) will be imaginary. For such cases, a variable c is defined to be the distance between the two circles. Using the distance d calculated in Equation (7), c is

$$c = d - R_i - R_j .$$

If c is greater or equal to zero, the circles are oriented as shown in Figure 6.

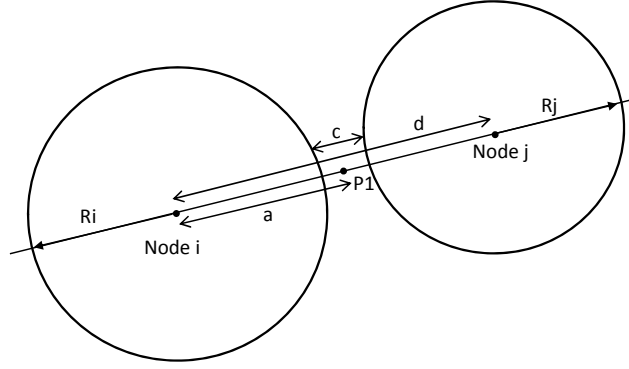


Figure 6. Range circles may not intersect. In the case shown, the point between the two range circles is chosen as the solution.

In this case, the solution is the point that is located halfway between the circles. The distance a is now defined as

$$a = R_i + (c/2).$$

The relationship

$$\frac{a}{d} = \frac{(x_{P1} - x_i)}{(x_j - x_i)} = \frac{(y_{P1} - y_i)}{(y_j - y_i)} \quad (11)$$

is used to find

$$x_{P1} = x_i + \frac{a}{d}(x_j - x_i) \quad (12)$$

$$y_{P1} = y_i + \frac{a}{d}(y_j - y_i) \quad (13)$$

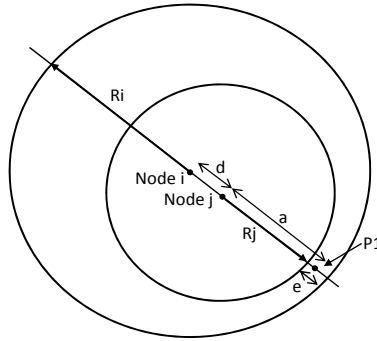


Figure 7. Range circles can lie within one another as shown. The point at which the circles are closest is chosen as the solution.

Cases where h is imaginary and c is less than zero correspond to two non-intersecting circles that are inside one another. The solution is located at the point halfway between the circles where the circles are closest. Assuming R_i is greater than R_j , the distance e is defined as

$$e = R_i - R_j - d$$

from which it is found

$$a = R_j + (e/2)$$

$$x_{p1} = x_j + \frac{a}{d}(x_j - x_i) \quad (14)$$

$$y_{p1} = y_j + \frac{a}{d}(y_j - y_i) . \quad (15)$$

B. INHERENT AND SYSTEMATIC ERRORS

1. Geometric Dilution of Precision

Small errors in just one range can quickly degrade the accuracy of these calculations as shown in Figure 8. Geometric dilution of precision (GDOP) refers to the geometry-sensitive change in error. The position solutions from certain sensor network geometries are less sensitive to error than others. Thus, it is difficult to calculate a confidence value for each solution.

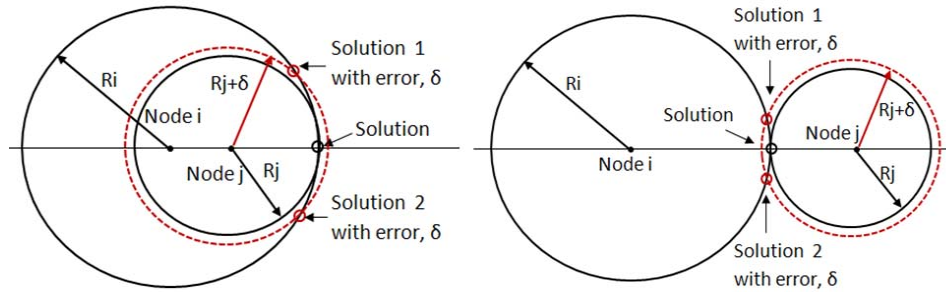


Figure 8. An example of GDOP. A small error in one range measurement will lead to a large error in the calculated solutions for certain geometries.

2. Two-dimensional Model

All algorithms used in this thesis assume a two-dimensional system. This approximation is valid in the shallow lake. Fathometer measurements showed the lake to

be between 4 and 8 feet deep. The mobile local node was suspended 13 inches beneath the lake surface. The deepest node was suspended 48 inches below the surface. Thus, if the boat is maneuvered over the deepest node, the range measurements have an error of 35 inches. All other errors due to the two-dimensional estimation are less than this.

Because straight-line propagation of sound is assumed, this error is corrected by projecting all ranges onto a two-dimensional plane. The converted ranges are used as inputs to the localization algorithms.

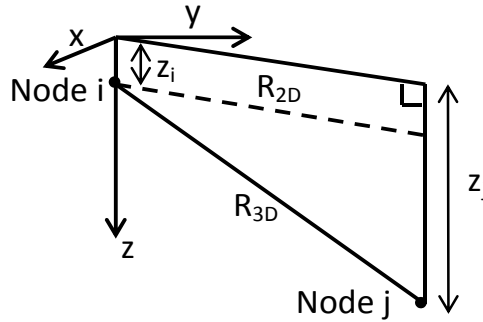


Figure 9. Projection of three-dimensional range onto two-dimensional coordinate system.

A range between nodes i and j , defined as r_{ij3D} , has magnitude

$$r_{ij3D} = \sqrt{\Delta x_{ij}^2 + \Delta y_{ij}^2 + \Delta z_{ij}^2} \quad (16)$$

The projection of the range onto the x-y plane is defined as r_{ij2D} and is shown in Figure 9.

We can rearrange Equation (16) to solve for the projected range

$$r_{ij2D}^2 = x_{ij}^2 + y_{ij}^2 = r_{ij3D}^2 - z_{ij}^2 \quad (17)$$

$$r_{ij2D} = \sqrt{r_{ij3D}^2 - z_{ij}^2} \quad (18)$$

This correction is used, but makes a minimal difference in calculating the path of the vehicle for such a shallow geometry.

3. Vehicle Motion

If the broadcast ping is used to communicate with N remote nodes, the remote nodes have no built-in dwell times, and the vehicle is stationary, the echoes will return to the local node at different times due to the differing ranges. If the vehicle is moving, the

vehicle will receive the N echoes at N different positions along its track. The return propagation times will change, and will be reflected in the round-trip propagation time. The time delay data received at the N different positions are used to calculate one waypoint that corresponds to the time the broadcast ping was sent. Thus, the vehicle's motion introduces two separate errors to the calculation of position: the ranges are inexact, and they do not correspond to the same position and time.

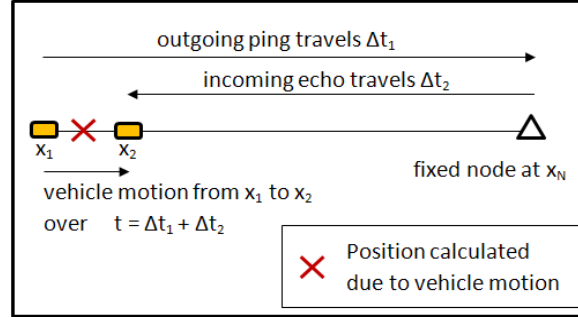


Figure 10. Schematic of the effect of vehicle motion on ranging.

The most simple and dramatic example of the first error occurs when the vehicle is moving directly toward or away from a fixed node. As pictured in Figure 10, the range calculated corresponds to neither the position when the ping was sent, nor the position when the echo was received. If the echo is sent the moment the ping arrives at the remote node, these errors are usually negligible because the propagation speed of the signal is much greater than the vehicle speed. Note that Figure 10 is not to scale.

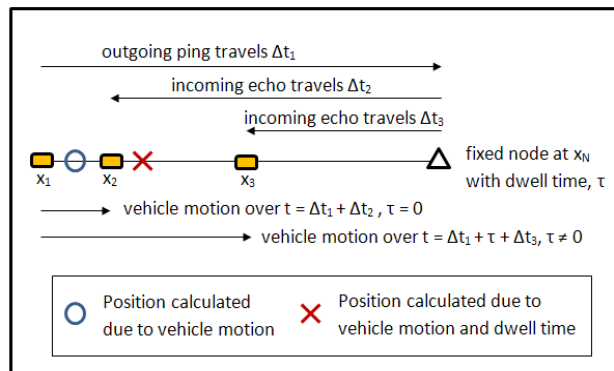


Figure 11. Schematic showing the effect of built-in dwell times for the remote nodes. The addition of a dwell time yields a range error equivalent to that found when the vehicle speed is increased.

When a Broadcast Ping command is used, the remote nodes send a reply after a built-in dwell time, τ . This dwell time increases the ranging error as shown in Figure 11. Processing time at the local and remote nodes increases the ranging error in the same way, as does increasing the vehicle speed. All of these variables should be minimized for experimentation.

THIS PAGE INTENTIONALLY LEFT BLANK

V. PREVIOUS WORK ON VEHICLE LOCALIZATION ALGORITHMS

A. WEIGHTING METHOD

The algorithm developed by [3] uses a weighting method to estimate the position of the local node. Each possible solution from two overlapping range circles is assigned a weight based on the proximity of all other solutions. Outlier solutions are given smaller weights. Clustered solutions are assigned higher weights. The solutions are multiplied by the weights, summed, and divided by the sum of the weights to find the estimated position.

$$\vec{R} = \begin{pmatrix} x \\ y \end{pmatrix} = \left[\sum_{j=1}^N W_j \begin{pmatrix} x_j \\ y_j \end{pmatrix} \right] / N \quad (19)$$

In Equation (19), (x, y) is the estimated position, the (x_j, y_j) 's represent all the raw solutions, and the W_j 's are the calculated weights.

B. CENTER OF MASS METHOD

The center-of-mass method developed by [4] calculates the vehicle position by averaging all the solution values to find the center of mass for the system. Solutions distant from the center of mass are discarded and the center of mass is recalculated.

C. DIFFERENCE LINEARIZATION METHOD

The difference linearization (DL) method, adapted from a GPS localization method, is investigated in [5] and found to perform better than the weighting or center of mass algorithms. The algorithm uses combinations of three range circles to find a solution. Three range equations can be written as

$$\begin{pmatrix} r_1^2 \\ r_2^2 \\ r_3^2 \end{pmatrix} = \begin{pmatrix} (x-x_1)^2 + (y-y_1)^2 \\ (x-x_2)^2 + (y-y_2)^2 \\ (x-x_3)^2 + (y-y_3)^2 \end{pmatrix} = \begin{pmatrix} x^2 - 2xx_1 + x_1^2 + y^2 - 2yy_1 + y_1^2 \\ x^2 - 2xx_2 + x_2^2 + y^2 - 2yy_2 + y_2^2 \\ x^2 - 2xx_3 + x_3^2 + y^2 - 2yy_3 + y_3^2 \end{pmatrix} \quad (20)$$

Subtracting the second range equation from the first gives

$$r_1^2 - r_2^2 = -2xx_1 + x_1^2 - 2yy_1 + y_1^2 + 2xx_2 - x_2^2 + 2yy_2 - y_2^2,$$

which is linear in x and y . A pair of simultaneous linear equations gives the matrix equation

$$\begin{pmatrix} r_1^2 - r_2^2 \\ r_2^2 - r_3^2 \end{pmatrix} = 2 \begin{pmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} x_1^2 - x_2^2 + y_1^2 - y_2^2 \\ x_2^2 - x_3^2 + y_2^2 - y_3^2 \end{pmatrix} \quad (21)$$

that has an exact solution of

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{2} \begin{pmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix}^{-1} \begin{pmatrix} r_1^2 - r_2^2 + x_2^2 - x_1^2 + y_2^2 - y_1^2 \\ r_2^2 - r_3^2 + x_3^2 - x_2^2 + y_3^2 - y_2^2 \end{pmatrix}. \quad (22)$$

This can be written as

$$\vec{R} = \frac{1}{2} \mathbf{A}^{-1} \vec{C} \quad (23)$$

where $\vec{R} = \begin{pmatrix} x \\ y \end{pmatrix}$, $\mathbf{A} = \begin{pmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix}$, and $\vec{C} = \begin{pmatrix} r_1^2 - r_2^2 + x_2^2 - x_1^2 + y_2^2 - y_1^2 \\ r_2^2 - r_3^2 + x_3^2 - x_2^2 + y_3^2 - y_2^2 \end{pmatrix}$.

This method also works for overdetermined cases where there are more than three ranges available. If there are N fixed nodes, the matrix $\mathbf{A}$ will have size $(N-1 \times 2)$ and the vector C will have $(N-1)$ elements. It was found in [5] that calculating one solution with this method for more than three nodes was less accurate than averaging the solutions from all combinations of three nodes. Combinations of three nodes leave the matrix $\mathbf{A}$ a square matrix that can be inverted for an exact solution.

1. Eigenvalues

The DL algorithm runs into errors given certain configurations of the anchored modems. As seen in Equation (23), to calculate an x and y position, the matrix

$\mathbf{A} = \begin{pmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix}$ must be inverted, and depends solely on the positions of three

remote nodes in relation to one another. If the nodes are co-linear, the matrix will be singular with a zero eigenvalue, and cannot be inverted. If the nodes are arranged in approximately a straight line, the matrix will have a small eigenvalue. When solving linear equations with matrix coefficients, for data with non-zero error, a small eigenvalue will inflate that error. [17] shows that the inverse of a matrix can be represented by

$$\mathbf{M}^{-1} = \sum_{i=1}^n \frac{1}{\lambda_i} \cdot e_i e_i^T \quad (24)$$

where $\mathbf{M}$ is an $n \times n$ matrix and e_i is the eigenvector of the matrix corresponding to the eigenvalue λ_i , and T indicates the transpose of the vector. If the vector C from Equation (23) has error, this error is magnified with a small eigenvalue. In overdetermined situations, solutions can be filtered out that correspond to eigenvalues much less than one can be discarded before averaging.

2. Condition Number of Matrix

Errors also occur when the magnitude of the eigenvalues are well behaved. These errors can be avoided by checking the condition of matrix $\mathbf{A}$. The condition number of a matrix is a measure of the accuracy of linear calculations using that matrix. The condition is a characteristic of the matrix itself, not of the calculation. Consider the calculation of solution vector x with the square coefficient matrix $\mathbf{A}$ and the constant vector b .

$$\mathbf{A} \vec{x} = \vec{b} \quad (25)$$

As described in [18], if there is a small error in the values of the coefficient matrix or in the constant vector, a well-conditioned matrix will have small errors in the solution vector. An ill-conditioned matrix yields large errors in x for small perturbations of the values in $\mathbf{A}$ or b . The condition number of a matrix is the same for the matrix inverse. The condition of an $n \times n$ matrix is defined as

$$\text{cond}(\mathbf{A}) = \|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\| \geq 1 \quad (26)$$

where $\|\mathbf{A}\|$ is the 2-norm, or Euclidean vector norm, of the matrix. As with vector norms, the norm of a matrix is a measure of the size of the matrix. The 2-norm is defined by [19] as

$$\|\mathbf{A}\|_2 = \max_{\vec{x} \neq 0} \frac{\|\mathbf{A}\vec{x}\|_2}{\|\vec{x}\|_2} \quad (27)$$

or more simply

$$\|\mathbf{A}\|_2 = \sqrt{\sigma_1^2} \quad (28)$$

where σ_1^2 is the maximum eigenvalue of $\mathbf{A}^\dagger \mathbf{A}$, $\mathbf{A}^\dagger$ being the conjugate transpose of $\mathbf{A}$.

Reference [18] proves via the definition and properties of norms that the error in vector x can be related to a perturbation in matrix $\mathbf{A}$ by

$$\frac{\|\Delta \vec{x}\|}{\|\vec{x} + \Delta \vec{x}\|} \leq (\|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|) \frac{\|\Delta \mathbf{A}\|}{\|\mathbf{A}\|} \quad (29)$$

and similarly for a perturbation of vector b

$$\frac{\|\Delta \vec{x}\|}{\|\vec{x} + \Delta \vec{x}\|} \leq (\|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|) \frac{\|\Delta \vec{b}\|}{\|\vec{b}\|} \quad (30)$$

This gives a mathematical representation of the condition number $(\|\mathbf{A}\| \cdot \|\mathbf{A}^{-1}\|)$, which is the upper limit of how error in one measurement scales error in x .

[20] describes a rule of thumb to estimate error in a measurement. If m is defined as

$$m = \log_{10}[\text{cond}(\mathbf{A})] \quad (31)$$

m represents the number of significant figures lost in the solution vector x due to a perturbation. Condition numbers close to unity are best. For overdetermined situations, the DL algorithm is edited to keep only those matrices with condition numbers less than 10, corresponding to a one digit loss in accuracy. Matrices with low eigenvalues will also be ill-conditioned. Thus, the condition of a matrix can be used as a metric to identify good node geometries. The DL algorithm edited to check the $\mathbf{A}$ matrix condition number is referred to as the difference linearization with condition check (DLC) method.

3. Number of Available Ranges

A further limitation of the DL method is the need for three range circles to calculate one position. If the $\mathbf{A}$ matrix is ill-conditioned, it will be bad for all points on the track. Whereas in an over-determined situation, solutions from ill-conditioned matrices can be ignored, if only three remote nodes are available, all solutions will have inflated error.

VI. THE FILTERED PAIRWISE ALGORITHM

A. ALGORITHM DESCRIPTION

As an alternative to the difference linearization (DL) method and the previously developed pairwise algorithms, a filtered pairwise (FP) algorithm was developed. With only three nodes available for ranging, pairwise algorithms generate three solutions from the three combinations of two range circles. Error is reduced in the pairwise algorithms by averaging the three solutions. The number of combinations available can be described with

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}, \quad (32)$$

read as ‘ n choose k .’ With five nodes, each method has an equal number of combinations of nodes. With six nodes, the difference linearization method has more combinations for averaging. In most underwater acoustic networks, six nodes will not be available for ranging.

The pairwise methods do not encounter the complications of matrix inversion. The FP method uses simple geometry to calculate all solutions and uses logic loops to discard the erroneous solutions.

B. ALGORITHM STRUCTURE

For each waypoint, an estimated position is calculated from all combinations of range circles as described.

1. Computing All Solutions

Solutions for range circles that do not intersect are calculated first, as shown in Chapter IV. These combinations have only one solution that, even with GDOP error, is a good initial guess at the vehicle position. The solutions are saved. Next, for all cases with intersecting range circles, the two ambiguous solutions are calculated and saved.

2. Comparing All Solutions

It is assumed that for every pair of ambiguous solutions, one solution is ‘correct’ and the other solution is incorrect, and the correct solutions are near one another. For the first pair of ambiguous solutions, S_{1a} and S_{1b} , the distance between each solution and every other solution, correct or incorrect, is calculated. For example, given two ambiguous solutions for a different combination, S_{ia} and S_{ib} , the distances

$$A_i = (S_{1a} - S_{ia}) \quad B_i = (S_{1a} - S_{ib}) \quad C_i = (S_{1b} - S_{ia}) \quad D_i = (S_{1b} - S_{ib})$$

are calculated. One of the four distances represents the distance between two correct solutions. The smallest distance of A_i and B_i is saved for point S_{1a} , and the smallest distance of C_i and D_i is saved for S_{1b} . This is repeated for all other pairs of ambiguous solutions and combinations of pairs. Finally, the saved distance values for S_{1a} are summed and compared to the sum of the distances for S_{1b} . The smallest sum corresponds to the correct solution. This one correct solution is then used to choose the correct solutions from all other pairs of ambiguous solutions, using the distances already calculated.

3. Averaging All Filtered Solutions

The solutions from non-intersecting range circles and the chosen correct solutions are averaged to find a single position for the waypoint. No weights are applied, and at this point, further outliers are not discarded.

VII. ALGORITHM IMPLEMENTATION FOR SYNTHETIC DATA

A. SETUP AND ASSUMPTIONS

To test the algorithms, random tracks are created that are continuous and stay within given bounds. The dimensions of the simulation area are the approximate dimensions of the anticipated experimental area in a shallow lake. A set number of nodes are placed in the domain and are given random fixed positions. A mobile node issues broadcast pings at a given period along the track. The nodes reply with an echo after a set dwell time. The echo arrival times are used to calculate ranges. Random error or bias can be added to the range data. The range data are saved to be used as input to the algorithms. All variables can be edited as needed in the Matlab script.

The simulations assume a completely two-dimensional scenario. There is some inherent error due to interpolation and limited resolution in Matlab when making the tracks and simulating the traveling pings and echoes. The errors are small and have a minimal effect on the performance of the algorithms.

B. CASE STUDIES

Case studies comparing the filtered pairwise (FP) method, difference linearization (DL) method, and the difference linearization with condition check (DLC) method are run using five remote nodes. Again, with five nodes, there is an equal number of combinations of nodes, and thus, an equal number of possible solutions for each type of algorithm. A script is created that generates a random path with random node positions. Range data are generated and the average localization error for the algorithms is found by averaging the difference between the actual vehicle positions and calculated positions for the entire track. 100 iterations are performed for the script.

1. Data Sets with Zero Range Error

Simulations are first run with zero added range error and no dwell times for the nodes. Each algorithm is run with the same data. There is still error in the data due to the

motion of the vehicle. This error is minimal however, as the echoes are allowed to arrive at the same time and the pings take only milliseconds to and from the remote nodes.

Localization Error for 100 Iterations of Error-Free Synthetic Data			
	FP Method	DL Method	DLC Method
Minimum Error (m)	0.1305	0.1001	0.0926
Maximum Error (m)	0.7141	4.8981	0.3004
Average Error (m)	0.1978	0.4743	0.1458

Table 1. Comparison of localization error for the three algorithms using error-free synthetic data.

For error-free synthetic data, the FP method performs better than the DL method. The DLC algorithm has the lowest average localization error. The difference between the DL and DLC methods shows the effectiveness of checking the condition number of the inverted matrices.

2. Data Sets with Random Range Error

Error can scale with range or be an additive factor, depending on the source of the error. Ranging error can be due to indirect sound propagation paths, refracted propagation, motion of the remote nodes, motion of the vehicle, processing time at the remote or local node, and a number of other factors. Errors also occur due to interpolation and limited resolution in Matlab when creating synthetic data. Because the error is not predictable, random error is simply added to the calculated ranges. The *rand* command in Matlab is used to generate a random number between 0 and 1. The value is shifted and scaled to give a random number between -1 and 1. This number is then multiplied by a variable scalar factor. The scalar factor represents the maximum range error in meters. The equation used in Matlab and the equivalent expression are

$$R_E = R_o + 2E(\text{rand} - 0.5)$$

$$|R_E - R_o| \leq E \quad (33)$$

where R_E is the range with error, R_o is the original range, and E is the maximum added range error. Simulations are run for scalar factors of 0.5, 1, 1.5 and 2 meters.

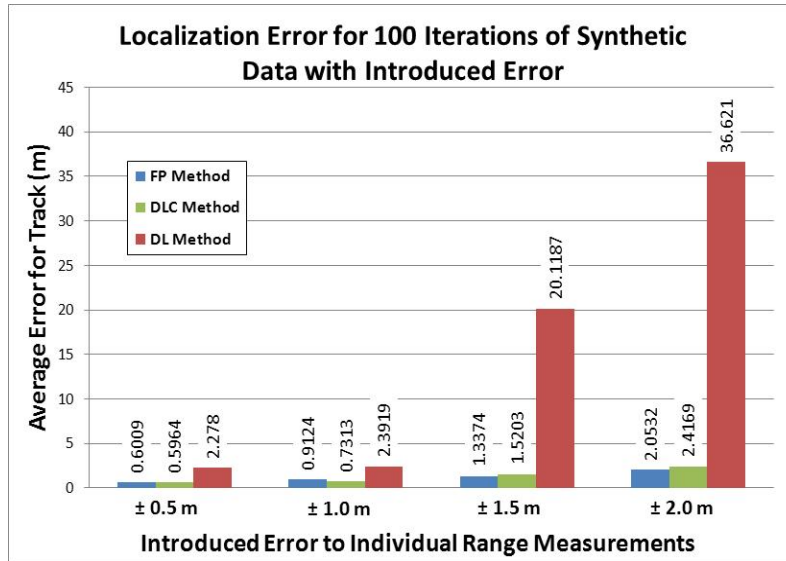


Figure 12. Comparison of localization error for the three algorithms using synthetic data with introduced error.

When error is added to the synthetic range data, the FP and DLC methods have similar average localization errors that scale almost linearly with the maximum error added to individual range values. The improvement of the DLC over the DL method is evident. The FP appears to perform better with greater error. To investigate this, 10 iterations of the algorithm are run over a range of introduced error values.

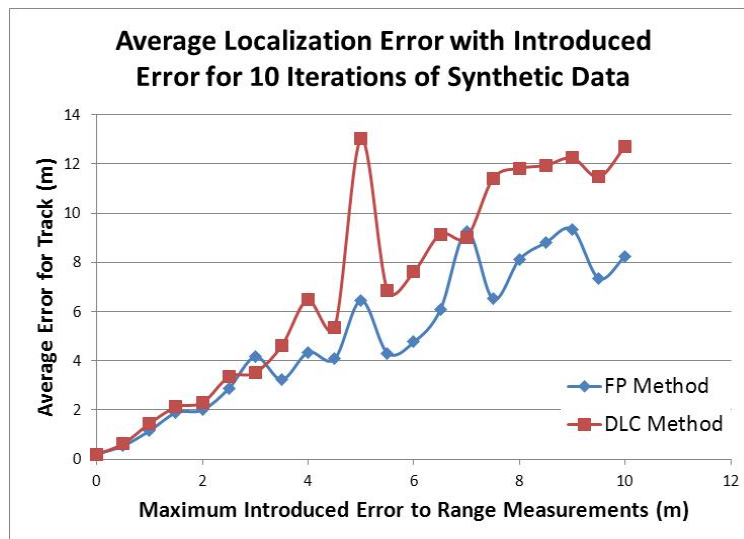


Figure 13. A measure of the robustness of each algorithm is found by plotting the average localization error as it increases with the amount of error introduced to each individual range measurement.

The graph in Figure 13 gives a rough measure of the robustness of each algorithm. The average localization error for the FP method grows approximately linearly with the amount of error introduced to individual measurements. The growth of average error for the FP method is less than that for the DLC method, indicating that it may be the better choice for practical implementation.

3. Data Sets with Dwell Time

Data are created for which the remote nodes are assigned a random dwell time. The maximum dwell time for a node is set to be one half the time between pings, or 2.5 seconds for these simulations. This is an arbitrary choice for the synthetic data.

Localization Error for 100 Iterations of Synthetic Data with Dwell Times			
	FP Method	DL Method	DLC Method
Minimum Error (m)	0.2596	0.1956	0.1744
Maximum Error (m)	1.0612	55.6238	0.8228
Average Error (m)	0.4802	1.8873	0.4542

Table 2. Average localization error for the three methods when a dwell time is used for the remote nodes.

The data have no artificially introduced error. The DLC method again performs best in this case. The effect of adding dwell times can be seen by comparing the error values with those from error-free range data in Table 1.

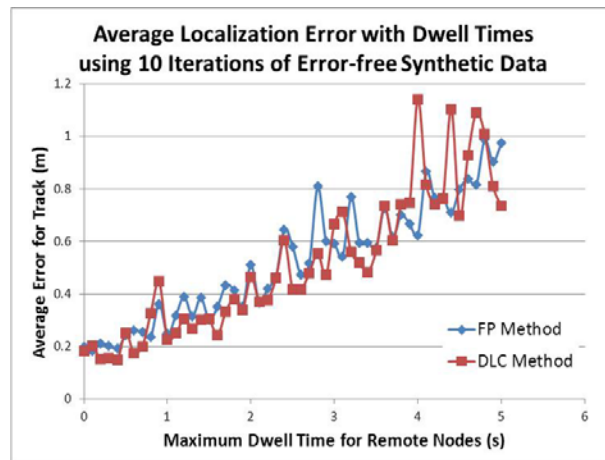


Figure 14. The performance of each algorithm with dwell times at the remote nodes is plotted. The synthetic data are created for a vehicle moving 0.5 m/s.

Ten iterations of the algorithms are performed with a range of allowed dwell times to roughly illustrate the effect on localization error. The error is due to motion of the local node over the duration of the transmission, as described in Chapter IV. The trend in Figure 14 is similar to that expected when the vehicle speed is increased or processing time at the modems is increased. From ten iterations, it appears that neither the FP nor the DLC method is affected more than the other by this type of error.

4. Data Sets with Nodes in a Straight Line

As described in Chapter V, if there are only three nodes, the difference linearization algorithm must use all available data, even in cases with a bad sensor network geometry. If the nodes are in a straight line, the DLC method gives no solutions. The FP algorithm yields two ambiguous solutions for each waypoint. If neither solution has less error than the other, the algorithm arbitrarily chooses a solution. Half of the solutions will be correct and the other half will be mirror images of the correct solution, reflected about the axis on which the nodes are collinear, as shown in Figure 15. In this particular case, further filtering and comparison to previous positions can yield the correct position. For this thesis, the algorithms are limited to using only the current range information from the nodes. The previous vehicle position, speed and orientation are not used. Of note is the fact that the pairwise method yields solutions where the difference linearization method yields none. Figures showing the DL and FP method outputs for the same track, but with reference nodes in an approximately straight line, are included in Appendix F.

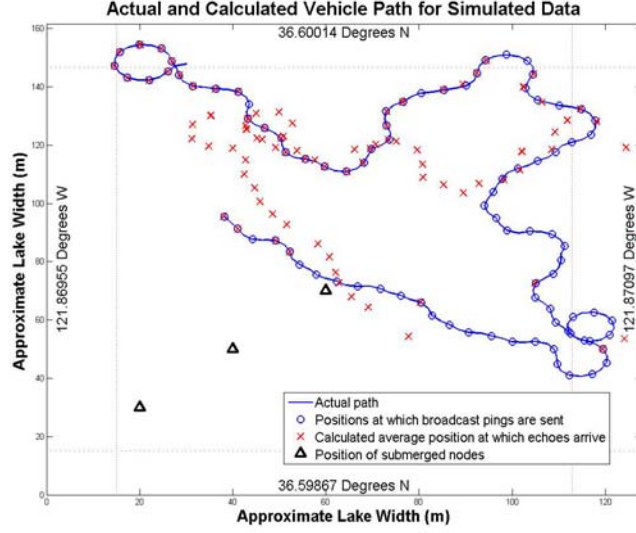


Figure 15. Nodes in a straight line will yield ambiguous solutions using the FP method. Incorporating information about the previous vehicle position, speed and orientation can resolve the ambiguity.

Most schemes for node deployments, including that in [9], have reference nodes arranged in a straight line. In practice, this is not usually achievable. Nevertheless, deploying nodes in a straight line should be actively avoided if they are to be used for localization.

5. Data Sets Varying the Number of Nodes

Synthetic network geometries are generated with three to six remote nodes and the average localization error is evaluated for the three algorithms. The average localization error decreases as more remote nodes are made available for ranging, as shown in Figure 16.

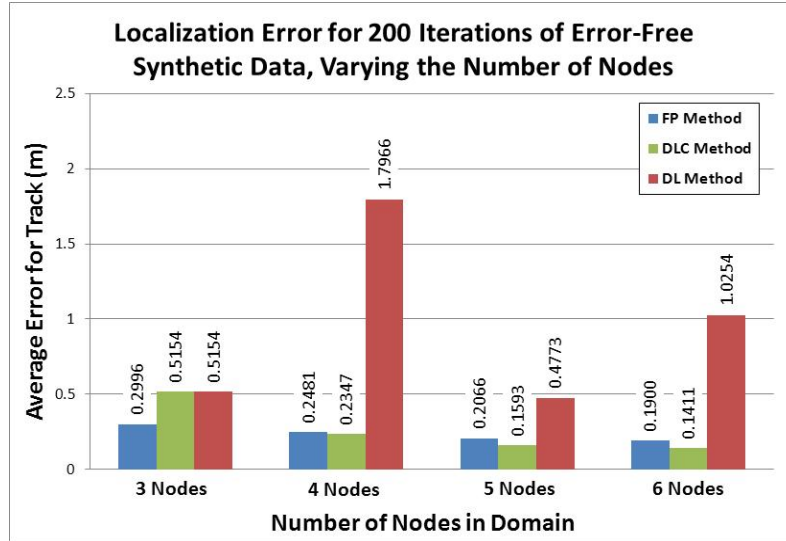


Figure 16. Average localization error for the three methods, varying the number of remote reference nodes.

In overdetermined cases, the DLC method performs better than the FP method. If simulations are run for 3 remote nodes in the domain, the DL method and DLC method have the same output, as no data can be discarded in the case of a bad geometry, and the FP method performs best. When data from only good geometries were analyzed for 100 iterations, the DLC method has an average localization error of 0.2473 meters, whereas the FP method has an average error of 0.3007 meters. For the randomly generated node positions, 24% of the geometries had bad condition numbers and were ignored in this calculation. The DLC method therefore performs better than the FP method for error-free data with any number of nodes, given a good sensor network geometry. With a bad geometry and only three remote reference nodes, the FP method performs better than the DLC method.

THIS PAGE INTENTIONALLY LEFT BLANK

VIII. DESCRIPTION OF EXPERIMENT

A. EXPERIMENTAL PLAN OVERVIEW

An experiment was conducted in Del Monte Lake on the Naval Postgraduate School campus on 3–6 April and 16 April 2012. An aluminum boat is driven on the lake using a battery-powered trolling motor. The boat tows a sled behind it, fitted with a Seaweb modem and two GPS loggers. The towed modem, or local node, is controlled via a serial connection to a laptop computer on the boat. Commands to the local node are sent via a Procomm terminal on the laptop. Six Seaweb modems are deployed as remote nodes in the lake at the positions shown in Appendix A. As the boat transits the lake, the GPS loggers record fixes and the local node is commanded to issue pings to the remote nodes. The data packets returned in the echoes are recorded on a capture file for later analysis. A description of the equipment used and the experimental plan are found in Appendix G.

B. SHALLOW LAKE ENVIRONMENT

Del Monte Lake is a shallow fresh-water lake, approximately 150 meters by 150 meters. During the experiment conducted in April 2012, fathometer measurements showed the depth to vary between 4 and 8 feet. The lake is fed by surface runoff water and has a silt bottom. A fountain in the center of the lake can be turned off to reduce ambient noise. A drain pump runs constantly to circulate water through the lake.

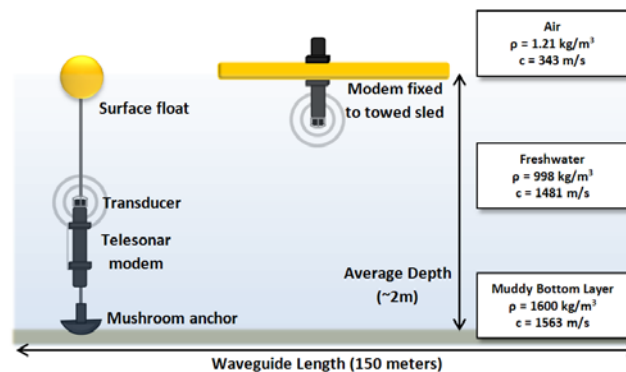


Figure 17. Diagram of fixed node and mobile node towed by a surface vehicle. Sound speeds, densities and the lake dimensions are estimated for modeling sound propagation in Del Monte Lake.

The lake's small volume, shallow depth and the drain pump make the lake susceptible to sound speed variability. The temperature of the water is affected by the amount of sunlight. The second week of tests occurred after a week of rain, which changed the lake's salinity. The influx of sediment and drain water also influences the salinity. The constant motion of water and sediment into and out of the lake affects the bottom composition and bathymetry.

For this analysis, the temperature and salinity are assumed to be spatially constant in the lake, and thus the sound speed is constant with range and depth. The zero-valued gradient leads to an infinite radius of curvature, or straight-line unrefracted propagation.

For the short ranges in the lake, the transmission loss due to geometric spreading and attenuation is neglected. Noise can be a limiting factor. Noise sources in the lake experiments included the motor noise, flow noise across the transducer, flow noise across the body of the boat and towed sled, noise from the pump and fountain, and noise due to wind. The wind increased the lapping noise of water against the edge of the aluminum boat and sled. This was significant due to the shallow depth and proximity of the local transducer. When active, the fountain created broadband noise that interfered with the measurements. When collecting data, the fountain was turned off and the boat was operated at low speeds to minimize motor and flow noise. Data were taken on days when the local wind speeds were low and the surface was calm.

Multipath is a significant problem in the shallow lake where there are many reflections from the bottom and surface. The MSFK modulation of the acoustic modems is designed to tolerate the impact of multipath. The shallow lake constrains the distance from the bottom and surface that the transducers can be suspended.

For initial estimates of sound propagation, the lake is modeled as a waveguide with a depth of 2 meters and a length of 150 meters. The upper boundary is represented as a pressure release boundary. Density and sound speed values of 1.21 kg/m^3 and 343 m/s are used for air. The lower bound is also a pressure release boundary. Typical density and sound speed values for a muddy bottom are 1600 kg/m^3 and 1563 m/s . For freshwater a density of 998 kg/m^3 and a constant sound speed of 1481 m/s are assumed. The remote

source depth is set to 1 meter and is the approximate average position of transducers on the anchored modems. The receiver depth of 0.5 meters represents the local transducer mounted under the towed sled.

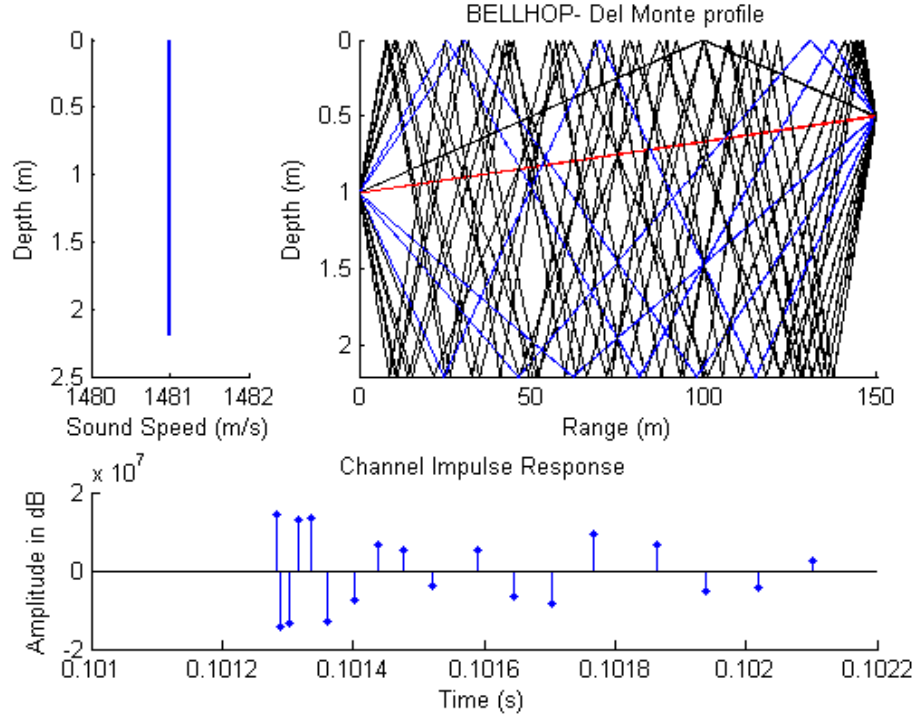


Figure 18. Bellhop sound propagation modeling software is used to produce eigenray and impulse response plots given a sound speed profile, medium characteristics, source and receiver locations, and dimensions of the operating space. From [21].

The eigenrays for sound propagation in the lake are modeled using the Bellhop acoustic propagation model. Bellhop is a numerical ray-tracing program run via Matlab that can output ray amplitudes, paths, travel times, and transmission loss, given an estimated sound speed profile for the lake environment. A program description and user manual is provided in [22]. Figure 18, created by Pascal Gagnon with Bellhop, shows a graph of the sound speed profile, the eigenrays for a finite number of launch angles, and the impulse response at the receiver for those eigenrays. The ray plot and impulse response show the amount of multipath in the lake. Of significance is the similarity in arrival time and amplitude of the direct path and the path with one surface reflection. The

impulse response shows that the amount of time it takes before the multipath signals die out in the channel is about 1 millisecond.

C. IN-AIR TESTING OF EQUIPMENT

Before the modems were deployed in the lake, the acoustic communications were tested in air. The Teledyne Benthos modems have a number of settings that can be adjusted for a specific mode of operation. Settings are easily changed in air at short distances, when the acoustic transmissions are fully reliable. For this experiment, settings were chosen to minimize the time between transmissions. The in-air testing was useful for familiarization with the equipment, the Procomm coding software, and the Seaweb commands and data packets.

IX. PRACTICAL RANGING APPROACHES

A. BROADCAST PING

During in-air testing, it was determined that the broadcast ping cannot be used for ranging in the lake experiments. The broadcast ping command is designed for ocean applications in which the nodes are separated by long distances and propagation times are longer. The default settings for a broadcast ping cause the local node to wait 1 minute for up to 16 responses. This time can be shortened to approximately 30 seconds if waiting for only six nodes.

This is a problem for the planned tracking experiment. The 30-second wait time limits the update frequency at which waypoints can be calculated. In the ocean, where the distance between nodes can be 1–2 kilometers, a waypoint every 30 seconds is ideal. In the lake, the largest distance between the fixed and mobile nodes is approximately 200 meters. The boat travels approximately 2 miles per hour and can therefore cover about 30 meters in 30 seconds. 30 meters is a significant portion of the network domain. Further, all ranges calculated during the 30-second wait time are used to plot one waypoint. This and non-stationary node geometry lead to significant localization errors. Finally, though the wait time for the local node is reduced to 30 seconds, the dwell times for the remote nodes cannot be changed by the user. Not all remote nodes return an echo due to dwell times greater than 30 seconds.

The lowest possible dwell and wait times are desired to calculate ranges that approximately represent the same position, and to allow for a high update frequency of waypoint calculations.

B. SEQUENTIAL PINGING

An alternative to the broadcast ping command is to use the addressed ping command between two nodes. This method was used for the lake experiments. This function does not allow the local node to communicate with all remote nodes simultaneously. Each ping is addressed to a specific remote node. With six remote nodes,

six separate pings need to be sent out, versus one with a broadcast ping. The echoes return immediately, however, which ultimately reduces processing time.

There are several constraints on the dwell and wait times, including environmental limits, signal processing limits, and operating limits of the equipment and coding software. The round-trip propagation time limits the wait time for the local node. In the lake, the round-trip times are on the order of milliseconds and are a negligible limitation. The processing time at local and remote nodes can be reduced by adjusting the modem settings. Communications can be impaired if the modems issue signals before the incoming multipath signals die out. A delay is automatically built into the processing time of the modems to avoid multipath interference. The lowest total processing time achieved was 4.2 seconds. For six nodes, echoes can be collected in 25.2 seconds, which is an improvement from the 30 seconds required for the broadcast ping. If an echo is not received, the local node waits for a set amount of time before continuing with the next ping. The lowest wait time achieved by adjusting modem settings was 10 seconds. This causes large gaps in the data due to one bad transmission. The 4.2 seconds between outgoing pings yields an error equivalent to the vehicle motion error described in Chapter IV. This method of data collection is not ideal, but it is more reliable than the broadcast ping for the lake experiments.

C. LONG BASELINE MODE

A Long Baseline (LBL) mode of operation for acoustic modems will resolve issues with the broadcast and sequential pinging modes. Short and Ultra Short Baseline systems also exist. The LBL systems, described in [8], are designed for ranges of 50 meters to 2 kilometers. They are used commercially in the same manner as the Seaweb sensor networks. The systems use signal propagation times to calculate ranges and locate an object. The difference between the two lies in the form of communications. The baseline systems use transponders that receive a signal in one band of frequencies and reply in a different band. The proposed adaptation of the LBL systems for Seaweb networks would have each remote node respond to a broadcast ping using a unique set of frequencies. This would allow the echoes to return to the local node simultaneously and

would minimize errors due to vehicle motion over the duration of the transmissions. It would also permit a higher update frequency for ranging calculations.

THIS PAGE INTENTIONALLY LEFT BLANK

X. FURTHER EXPERIMENTAL CONSIDERATIONS

A. SPEED OF SOUND CALCULATION

While conducting the experiment on Del Monte Lake, ranging commands were sent between the stationary, remote nodes to estimate the speed of sound in the lake. GPS positions were recorded when the anchored modems were deployed and recovered. Assuming the GPS coordinates are the exact locations of the modems, we can calculate the range between remote nodes and then back-calculate the sound speed using measured delay times. Formulae for calculating the distance between GPS points can be found online. The Haversine Formula used by [23] calculates the great-circle distance between two points, assuming a spherical earth with a radius of 6,371 kilometers.

Coordinate-to-distance conversions depend on the particular latitude and longitude of the points. Choosing the coordinate position for Node #21 in the middle of the lake, located at 36°35.953'N and 121°52.203'W, one can find a conversion factor by calculating the distance between this point, and a point that varies by a given number of degrees north and west. The conversion factors were found as shown in Table 3, using the formula from [23].

	Deg/Min N	Deg/Min W	Degrees N	Degrees W	Range (m)
Node 21	36°35.953'	121°52.203'	36.59922	121.87005	
Point 2			36.59923	121.87005	
Δ°			0.00001	0.0	1.112

	Deg/Min N	Deg/Min W	Degrees N	Degrees W	Range (m)
Node 21	36°35.953'	121°52.203'	36.59922	121.87005	
Point 2			36.59922	121.87006	
Δ°			0.0	0.00001	0.8927

Table 3. A conversion factor between degrees and distance, specific to the operating area, is found by choosing two points separated by 0.00001 degrees in either direction.

The conversion factors are used to calculate the ranges between each combination of nodes given their GPS coordinates. The ranges shown are also corrected for the depth of the nodes. Reversing the calculations in Chapter IV, the node-to-node ranges are converted from two dimensions to three.

The transmission delay times were collected between nodes on 16 April 2012. The sound speed is found by dividing the GPS-derived range by half the round-trip delay time. Table 4 shows the ranges, delay times and sound speeds calculated for nine node combinations.

Node # - Node #	Ranges for 16 APR (m)	Round-trip Delay for 16 APR (s)	Sound Speed (m/s)
20-21	140.21	0.1988	1410.56
20-23	127.56	0.1835	1390.31
20-25	164.07	0.2281	1438.62
21-22	123.45	0.1725	1431.32
21-23	120.13	0.1645	1460.60
22-24	103.08	0.1438	1433.62
22-25	123.24	0.1691	1457.65
23-24	134.44	0.185	1453.39
24-25	114.80	0.1552	1479.42
Average			1439.50

Table 4. The speed of sound in the lake is found using ranges between anchored modems. This method yields a large range of solutions due to errors in the measurements.

The average calculated sound speed is 1439.5 m/s with the individual sound speeds varying over a range of 89.11 m/s. The average speed is far from the 1481 m/s sound speed estimate for freshwater, but within the range of typical sound speed values for freshwater.

B. GPS ERROR

Table 5 contains the GPS coordinates recorded on 3–4 April 2012, when the nodes were deployed, and those from 16 April 2012, recorded during recovery.

Date		Node 20	Node 21	Node 22	Node 23	Node 24	Node 25
3-4 APR	N	36°36.009'	36°35.953'	36°36.018'	36°36.015'	36°35.962'	36°35.966'
	W	121°52.270'	121°52.203'	121°52.228'	121°52.182'	121°52.246'	121°52.171'
16-Apr	N	36°36.005'	36°35.951'	36°36.015'	36°36.014'	36°35.962'	36°35.966'
	W	121°52.268'	121°52.202'	121°52.225'	121°52.183'	121°52.246'	121°52.169'

Table 5. GPS coordinates for the anchored modems when deployed and recovered.

Table 6 compares the ranges between the nine sets of nodes for the deployment and recovery dates. A comparison of the initial and final GPS locations shows that the nodes remained approximately stationary throughout the testing, however differences in the GPS measurements correspond to range differences of up to 5.13 meters between the same two nodes. From visual inspection, it was clear that the anchored modems were not displaced by this amount. The range difference can be attributed to error in the GPS measurements.

Node # - Node #	Ranges for 3-4 APR (m)	Ranges for 16 APR (m)	ΔR (m)
20-21	143.91	140.21	3.70
20-23	131.40	127.56	3.84
20-25	167.47	164.07	3.40
21-22	126.08	123.45	2.63
21-23	119.08	120.13	1.06
22-24	107.19	103.08	4.11
22-25	128.37	123.24	5.13
23-24	136.80	134.44	2.37
24-25	111.83	114.80	2.97

Table 6. Ranges between anchored modems are calculated using their GPS coordinates.

Using the calculated average sound speed reduces the localization error for both algorithms compared to the error found with the typical freshwater sound speed of 1481 m/s. The error in the GPS readings significantly affects the calculated sound speed. For this reason, a secondary method for calculating sound speed is desired. A common practice is to use a conductivity-temperature-depth (CTD) probe to directly measure the water properties that determine sound speed.

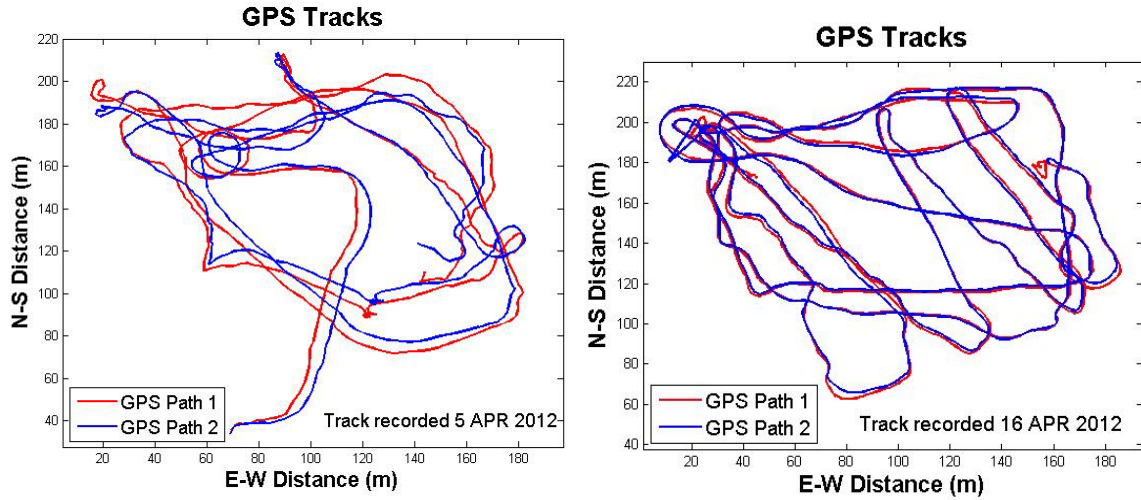


Figure 19. Plots of the two simultaneously logged GPS tracks shows that significant error can exist in the reference track. Tracks with good agreement were chosen for data analysis.

The modem positions were recorded with a Garmin GPSmap76CS logger. The two G-Log 760 GPS loggers used to record the vehicle track also had errors, as seen by plotting their output. The GPS loggers were placed approximately half a meter apart, but recorded positions up to 20 meters apart. For this reason, the tracks used to analyze the error in the localization algorithms were limited to those with good agreement between the two GPS loggers.

XI. ALGORITHM IMPLEMENTATION FOR EXPERIMENTAL DATA

A. PROGRAM DESCRIPTION

1. Initial Data Processing

GPS fixes are retrieved from the loggers using the accompanying software and saved as Microsoft Excel files. The displayed data from node-to-node communications, shown in Figure 3, are saved by Procomm in a text-formatted capture file. A Matlab script reads in the GPS data. For consistency, the GPS coordinates of the nodes recorded on 16 April 2012 are used when analyzing all tracks. For ease of plotting and computation, the latitude and longitude fixes of the GPS-logged tracks, and the locations of the remote nodes are converted to x and y coordinates with units of meters and with the origin defined to be a set position southwest of the operating area in Del Monte Lake. The x axis runs west to east and the y axis runs south to north. Again, the conversion factors between GPS coordinates and meters were found as shown in Table 3, using the formula from [23]. The text files with range data are sorted through with a second Matlab script to extract the required information.

After GPS data and data from Procomm capture files are converted to Matlab data files, they are read into a Matlab script specific to each algorithm, where the roundtrip propagation times are converted to ranges. The depths of the transducers are then used to project the ranges onto a two-dimensional plane.

The track is manually truncated to only include lengths of track for which there is a sufficient amount of recorded node-to-node ranges. For the experiment, the individual remote nodes were pinged using a programmable hotkey via Procomm. An infinite loop script was less efficient due to irregular transmission times and errors. Each modem can only store a given number of bits in a temporary cache. If the local node is commanded via a hotkey to send a ping before the incoming ping is processed, the cache will fill and

the modem either pauses or automatically reboots. The sections of sparse data shown in Figure 20 occurred when transiting around the fountain or when sequential pings were sent too quickly.

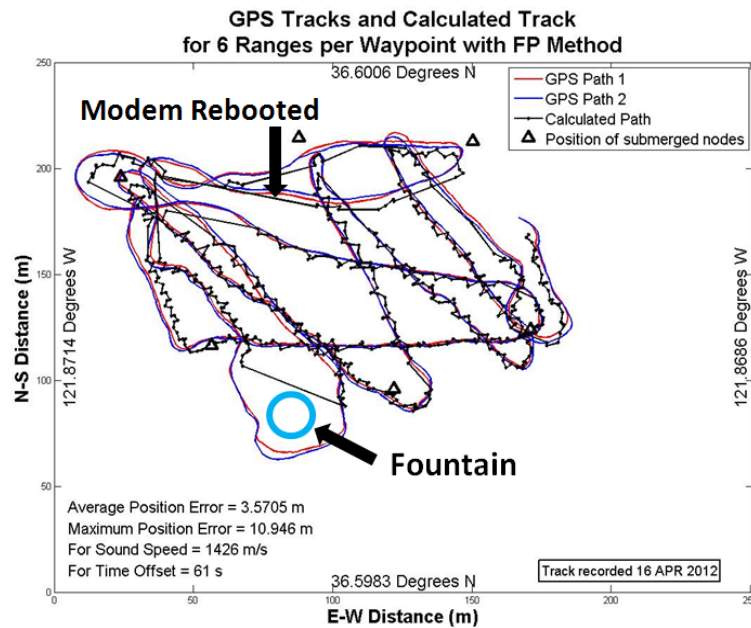


Figure 20. The FP algorithm output is shown for data taken on 16 April 2012. Sections of sparse data occurred when transiting around the fountain or when sequential pings were sent too quickly.

Both the GPS data and Seaweb data are recorded with Universal Time Coordinated (UTC) time. It is assumed that the times from both GPS loggers time and the local node are correct. This cannot be exactly verified using the equipment. Data from the local node are passed to the Procomm terminal that has its own clock, based on the computer clock. The time it takes to pass this information will skew any calibration. The GPS loggers used do not allow data to be viewed as they are recorded. Imprecise methods of checking the time agreement show that the times do not differ more than a few minutes, thus the assumption is maintained for initial algorithm testing. The GPS track is truncated to correspond to the Seaweb data times and the times are shifted to begin at zero.

2. Sorting Data

The necessary use of sequential pinging as the vehicle moves introduces significant error when using trilateration. An example of this error is shown in Figure 21.

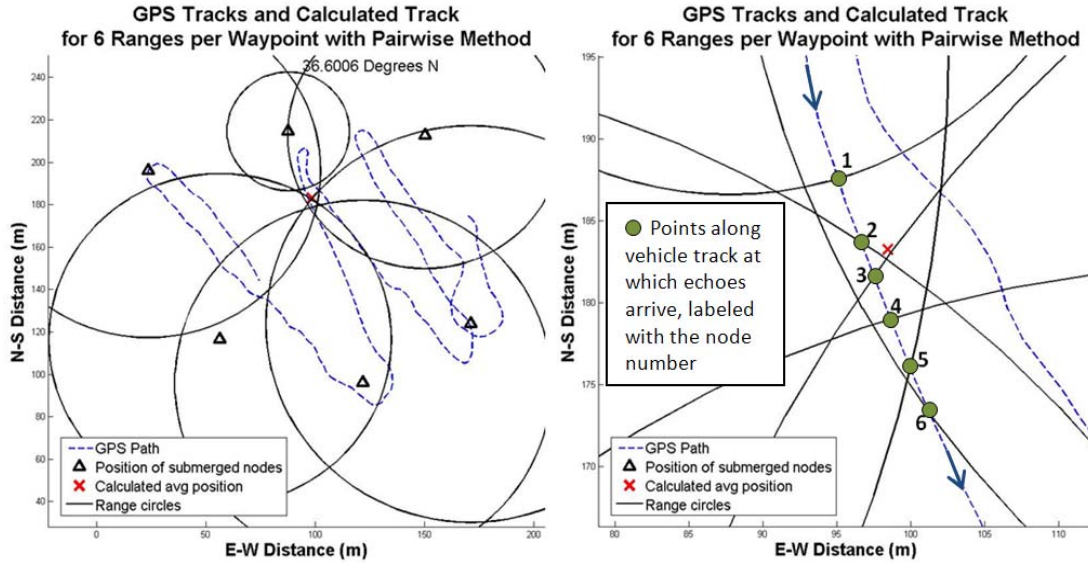


Figure 21. The first image shows range circles from six sequential pings, the vehicle track, and the average position calculated with the filtered pairwise algorithm with the six returns. The second image shows the same at a higher resolution.

The figure shows that the returns are spaced almost evenly along the track. They are each separated by approximately 4.2 seconds. The even spacing indicates that little error exists in the individual node-to-node range data. The average position is calculated using the range circle crossings, most of which are shown. One can see that the range circles from Node 1 and Node 4 do not intersect, as well as for Node 1 and Node 3. For the first set, the approximate solution described in Chapter IV is a good approximation, whereas it is worse for the second node set.

To minimize error due to vehicle motion, and maximize the amount of waypoints calculated, the range data are broken up in overlapping sections of lengths 3 through 6, as shown in Figure 22.

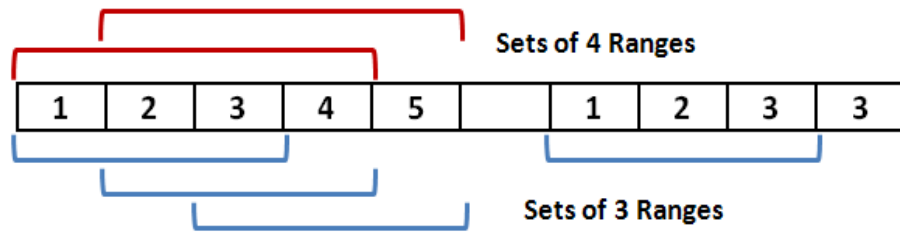


Figure 22. The schematic shows successful echo returns from nodes 1–5. The return from node 6 was unsuccessful. Two echoes from node 3 were recorded in a row. These errors limit the division of data for analysis.

Data sections with three ranges are closer in time. Data sections with six ranges will be more spaced out, but will allow for more averages. If an echo is not received from a node, the returns before and after the data gap will not be used to calculate a waypoint. This is due to the fact that the local node waits for a return for approximately ten seconds before continuing with the next ping, yielding a total of about 15 seconds between ranges. On a few occasions, pings to the same node were sent in sequence. The two ranges are unusable, as they will yield range circles that are an equal distance apart in all directions. These errors limit the sections of six ranges more so than the sections of three ranges. The effect is evident in the plotted results included in Appendix B.

3. Determining Average Localization Error

After each Matlab script sections the data, the filtered pairwise (FP) and the difference linearization with a condition check (DLC) localization algorithms are implemented to calculate the average position for each waypoint. The localization error is found by computing the distance in meters from each calculated waypoint to the corresponding waypoint of the GPS track. All distances are summed and then divided by the number of waypoints used to find the average localization error for the track. For consistency, when calculating the localization error for each algorithm, the same number of waypoints is used for the same length of track.

Difficulties arise when trying to find the true position and the corresponding calculated waypoint due to GPS logger error, the inability to synchronize GPS logger

time with Seaweb node clocks and Procomm time, the resolution of GPS tracks, and interpolation in Matlab, and the fact that the return echoes used to calculate one waypoint occur at different times.

The time chosen to represent each waypoint is the median time that the echoes return. The Matlab *timeseries* function is used to interpolate GPS track positions at these median times. The positions from the two GPS tracks are averaged at each time for comparison with the algorithm output. The GPS loggers are set to take one fix every second. The interpolation error in Matlab is minimal.

B. EMPIRICAL CALIBRATIONS

Once the algorithms output an average localization error for the track, a time offset is added to the Seaweb echo times to align them with the GPS UTC time, and the average localization error for the track is minimized.

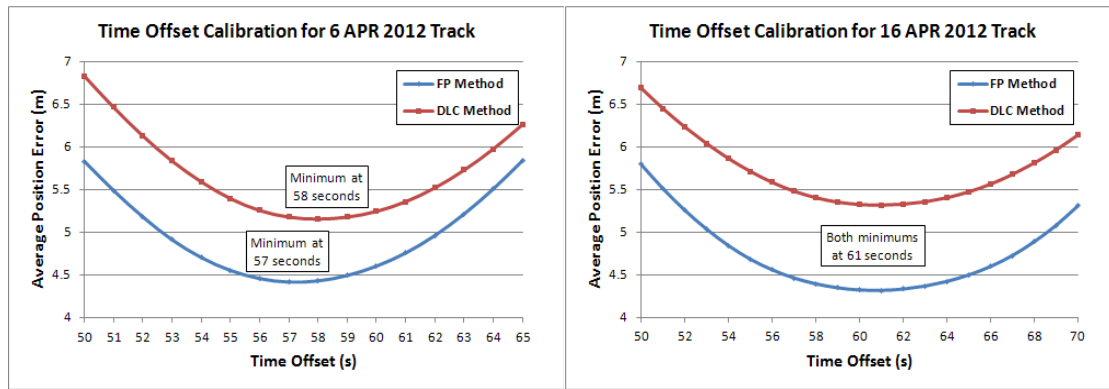


Figure 23. Calibration for the time offset between Seaweb clocks and GPS logger clocks.

Testing the algorithm over a range of time offsets, it is found for the April 6 data that the Seaweb clocks are approximately 57.5 seconds ahead of the GPS times, and 61 seconds ahead for the April 16 data, as shown in Figure 23. For the calibration shown, the algorithms are run with sets of three ranges per waypoint. Further examination shows that the error is minimized for any sized set of ranges for these time offsets.

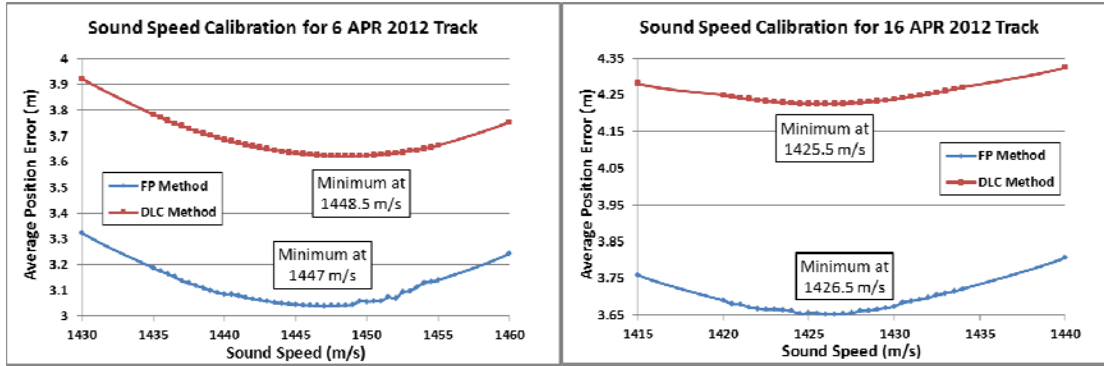


Figure 24. Determination of the sound speed that minimizes localization error.

After the time offset is applied to the data, a similar calibration is done to find the sound speed that yields the least average error. For the calibration shown as shown in Figure 24, the algorithms are run with sets of six ranges per waypoint, as this setting yields the minimal localization error at any sound speed. The sound speeds are expected to differ for the two days. For the April 6 data, a sound speed of approximately 1447.75 m/s minimizes the localization error. For the April 16 data a sound speed of approximately 1426 m/s minimizes the error.

C. RESULTS DISCUSSION

The plotted results from the algorithms are included in Appendix B and summarized in Table 7. Both algorithms performed best when using combinations of six ranges to find each waypoint. Though the vehicle positions are calculated using echo returns over a range of approximately 25 seconds, versus about 13 seconds for sets of three ranges, the amount of averaging allowed with more range information has a greater effect on algorithm performance.

Average Localization Error in Meters for Each Method				
	Data from 6 APR 2012		Data from 16 APR 2012	
Ranges Used	DLC Method	FP Method	DLC Method	FP Method
3	4.99	4.12	5.30	4.34
4	4.64	3.63	4.93	3.95
5	4.14	3.32	4.52	3.75
6	3.62	3.04	4.20	3.65

Table 7. Average localization error for the FP and DLC methods for experimental data.

The plotted tracks show erratic discontinuities that appear to be semi-periodic in localized regions. These discontinuities are not due to errors in the data, but are results of sequential pinging. The semi-periodicity is a result of using overlapping sections of data for nearby waypoints. The synthetic simulations were run assuming an LBL-type mode of operation for which a single ping communicates with all remote nodes. The remote nodes reply with no delay and can arrive at the local node simultaneously. If the synthetic data are made to mimic the sequential pinging mode of communications, plots similar to that in Figure 25 are produced.

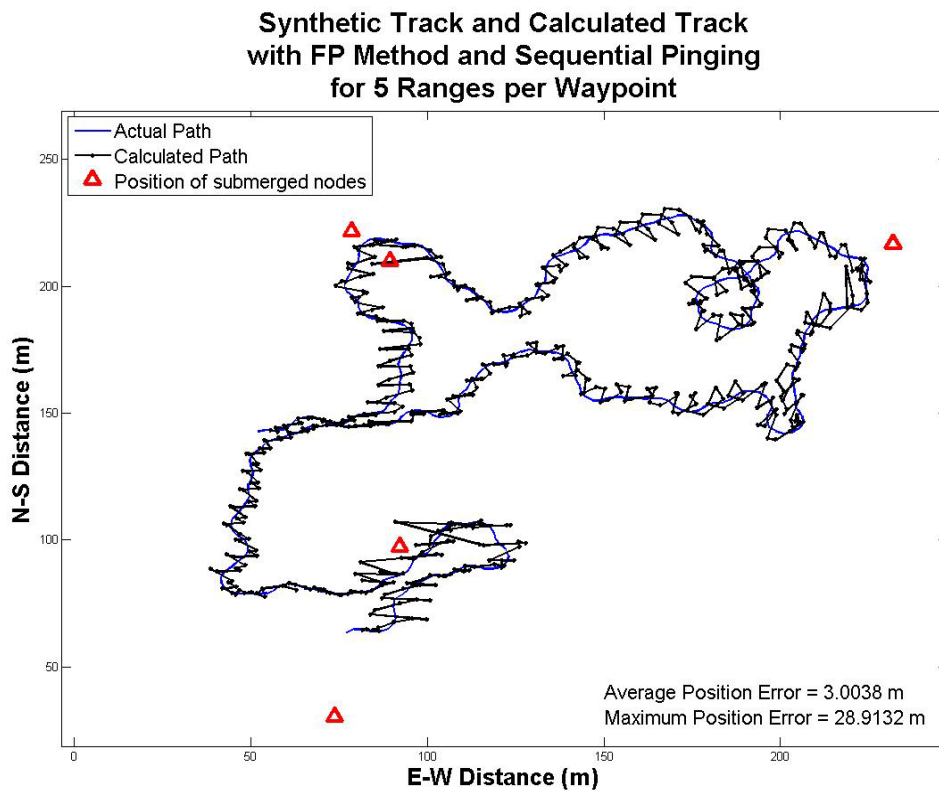


Figure 25. Example of discontinuities over a synthetic track due to sequential pinging.

The figure shows the same discontinuities over the randomly generated synthetic track as seen in the experimental data. The simulation was run for error-free synthetic data. There were no delays added at the remote nodes. Pings were sent once every 3 seconds. Even with the high ping frequency and no error, the pairwise method yields an

average error of about 3 meters. The average localization error of the FP and DLC methods with experimental data cannot be expected to be better than 3 meters.

The FP algorithm outperformed the DLC algorithm regardless of the number of reference nodes. This can be explained by the FP method's robustness with regards to range error. In most simulations, the DLC algorithm has a lower average localization error, though by a small margin. Figure 13 shows that when error in the individual range measurements is present, the FP method performs better than the DLC on average. When pinging remote nodes sequentially, the displacement of the vehicle over the period of time can translate to individual range measurement errors.

D. CONDITION NUMBERS REVISITED

In looking at the eigenvalues and condition numbers of the matrices for the deployed nodes, it was discovered that each algorithm had different condition numbers using the same node locations. This led to the realization that for every three nodes, there are in fact three condition numbers possible.

$$\begin{aligned}\mathbf{P} &= \begin{pmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix} \\ \mathbf{Q} &= \begin{pmatrix} (x_3 - x_1) & (y_3 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix} \\ \mathbf{R} &= \begin{pmatrix} (x_3 - x_1) & (y_3 - y_1) \\ (x_1 - x_2) & (y_1 - y_2) \end{pmatrix}\end{aligned}$$

The $\mathbf{A}$ matrix from Equation (23) can be expressed as one of the three matrices shown, each of which has a different condition number. Of the modems deployed in the lake, Nodes 20, 22 and 23, pictured in Appendix A, are approximately in a straight line, and their $\mathbf{A}$ matrices yield the largest condition numbers. The three condition values are 16.16, 15.74 and 6.38. The condition numbers for all $\mathbf{A}$ matrices for all node combinations are included in Appendix C.

The DLC algorithm can be edited to compute the three condition numbers and find the minimum of the three. The range equations will then be summed in the way that corresponds to the lowest number. Editing the DLC code in this way approximately

doubles the computation time. It also does not guarantee that the error will be reduced. The calculated condition of a matrix is an upper limit, and as seen in Equation (30), the fractional error in a solution is less than or equal to the condition number multiplied by the fractional error in the measurement. In Table 8, the average localization errors are compared for the FP method, the DLC method using the **P** matrix, and the DLC method using the minimum condition value.

Average Localization Error in Meters for Each Method						
	Data from 6 APR 2012			Data from 16 APR 2012		
Ranges	DLC Method with <i>P</i> Matrix	DLC Method Min Condition #	FP	DLC Method with <i>P</i> Matrix	DLC Method Min Condition #	FP
3	4.99	4.99	4.12	5.30	5.30	4.34
4	4.64	4.16	3.63	4.93	4.81	3.95
5	4.14	4.37	3.32	4.52	4.78	3.75
6	3.62	3.63	3.04	4.20	4.41	3.65

Table 8. Average localization error for experimental data using the FP method, the DLC method using one condition number, and the DLC method that uses the minimum of all three possible condition values of the inverted **A** matrix.

The table shows that using the matrices with the lowest condition numbers does not necessarily reduce the overall localization error for the experimental data. When using three ranges, the average error is the same to 14 decimal places, for example. The Node 20, 22, and 23 combination conveniently never occurs when using three ranges due to sequential pinging. The error was only reduced when using combinations of four ranges. The plotted results are shown in Appendix D. The FP method still has the lowest error in all cases.

Finally, the only area in which the FP method performed better than the DLC method in simulations was for synthetic data with error introduced to each individual range measurement. To verify that the FP algorithm performs better than the DLC method using the smallest condition numbers, 50 iterations are repeated with the corrected DLC method.

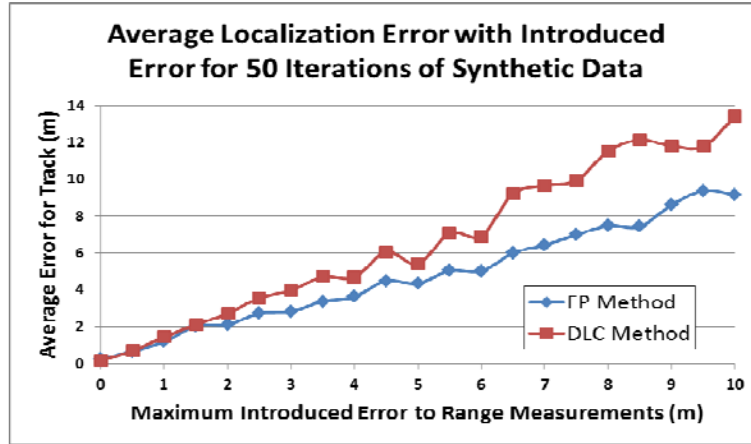


Figure 26. Average localization error as it increases with the amount of error introduced to each individual range measurement. The FP method is compared to the DLC method that uses the minimum matrix condition number.

In Figure 26 it is seen that the localization error using the FP method still grows at a slower rate than for the DLC method. Thus, making this final correction to the DLC method does not guarantee that the localization error will be reduced.

XII. PRACTICAL IMPLEMENTATION OF LOCALIZATION ALGORITHMS

Though the simulations are run for ideal data and the experimental data are taken in a controlled environment, the development of the filtered pairwise (FP) and the difference linearization with a condition check (DLC) localization algorithms is done with practical implementation in mind. In evaluating localization algorithms, important performance metrics are the computation time, the robustness to error in the ranges, and the operational limits and cost of the existing equipment.

A. DEPLOYMENT CONSIDERATIONS

Modems deployed in the ocean will typically be as sparsely distributed as possible, so as to cover the greatest area with minimal cost. Trilateration requires the communication area of three neighboring nodes to overlap. Only three nodes are available for ranging in most cases, versus the six nodes used in this thesis, thus it is critical that those three nodes are not deployed in a geometry that results in bad matrix condition numbers.

B. REAL-TIME IMPLEMENTATION CAPABILITY

The computation time of each algorithm is an important consideration for real-time implementation. The computation times for the experimental data are summarized in Table 11. The times shown are for the computation of the entire track. Depending on the length of the track, the algorithm used, and the number of ranges per waypoint, each algorithm performs a different number of calculations. Table 9 shows the number of waypoints calculated for each track, varying with the number of range data used to find each waypoint. The data from the April 6th track has a greater decrease in calculated waypoints as the ranges per waypoint increase, than the data from the April 16th track. This can be attributed to the sparseness of data.

Number of Waypoints Calculated for Both FP and DL Methods		
Ranges per Waypoint	Data from 6 APR 2012	Data from 16 APR 2012
3	239	364
4	214	352
5	189	341
6	162	330

Table 9. The number of waypoints calculated for each track depends on the length of the track and number of range data used to find each waypoint.

Table 10 shows the total number of combinations used to calculate the waypoints for each track and for each localization algorithm, as the number of ranges per waypoint increase. The numbers of waypoints from Table 9 are multiplied by the number of combinations of range circles used to find each solution. The multiplicative factors are calculated with the ' n choose k ' formula in Equation (32). The DL and DLC methods use combinations of three range data, whereas the FP method uses combinations of two range data to find a solution. This difference, combined with the number of ranges used to calculate each waypoint, is the greatest contribution to algorithm computation time.

Number of Range Data Combinations Used for Calculation of Waypoints over Entire Track				
	Data from 6 APR 2012		Data from 16 APR 2012	
Ranges per wp	DL Method	FP Method	DL Method	FP Method
3	$239 \times 1 = 239$	$239 \times 3 = 717$	$364 \times 1 = 364$	$364 \times 3 = 1092$
4	$214 \times 4 = 856$	$214 \times 6 = 1284$	$352 \times 4 = 1408$	$352 \times 6 = 2112$
5	$189 \times 10 = 1890$	$189 \times 10 = 1890$	$341 \times 10 = 3410$	$341 \times 10 = 3410$
6	$162 \times 20 = 3240$	$162 \times 15 = 2430$	$330 \times 20 = 6600$	$330 \times 15 = 4950$

Table 10. The DL method uses combinations of three range data, whereas the FP method uses combinations of two range data to find a solution. The difference has a large effect on algorithm computation time.

If five range data are used per waypoint calculation, both algorithms have the same amount of combinations to compute. Thus, the computation times compared for this case reflect the complexity of each algorithm's calculations. Table 11 shows that the FP method has the lowest computation time per combination. The multiplicative factors in

Table 10 are also the number of averages that each method can use to calculate a waypoint, assuming no data is discarded from bad geometries.

Computation Time in Seconds for Each Method						
	Data from 6 APR 2012			Data from 16 APR 2012		
Ranges per wp	DLC Method with <i>P</i> Matrix	DLC Method Min Condition #	FP	DLC Method with <i>P</i> Matrix	DLC Method Min Condition #	FP
3	0.033	0.053	0.066	0.046	0.079	0.085
4	0.077	0.139	0.083	0.121	0.223	0.113
5	0.150	0.268	0.086	0.260	0.475	0.138
6	0.244	0.448	0.103	0.487	0.889	0.186

Table 11. Comparison of computation times for the FP and DLC methods using experimental data, depending on the number of ranges used to calculate each waypoint.

Table 11 also shows that the DLC method that minimizes the matrix condition numbers has a computation time almost twice that of the DLC method using only the **P** matrix. The computation times in the table do not include the time needed to preprocess or post-process the data. Again, in practical situations, only three nodes are typically available for ranging. For real-time implementation of the localization algorithms, the trade-off between computation time and the amount of averaging enabled by each algorithm should be considered.

C. CRITICAL ASSUMPTIONS

Assumptions about the controlled lake environment will not hold in an ocean environment. In particular, the constant sound speed and two-dimensionality approximations are critical assumptions that become less accurate as the range and depth of the operational domain increase. Additional complexity of the environmental model is required, but is hampered by computation time considerations and spatially and temporally changing parameters. For this reason, a robust localization algorithm is required, versus one that requires near-ideal data to converge on a solution.

D. AVAILABILITY OF ADDITIONAL INFORMATION

Finally, when operating a sensor node on a moving vehicle, it is likely that the vehicle's heading and speed will be known. One might also know the depth of the vehicle and the local sound speed. This additional information can aid in localization by reducing or constraining the degrees of freedom.

1. Vehicle Speed and Heading

A vehicle's speed and heading are, of course, helpful in determining its future location. The plot in Figure 27 was created by iteratively solving a system of nonlinear range circle and distance interval equations, assuming a constant heading and a constant time interval between pings.

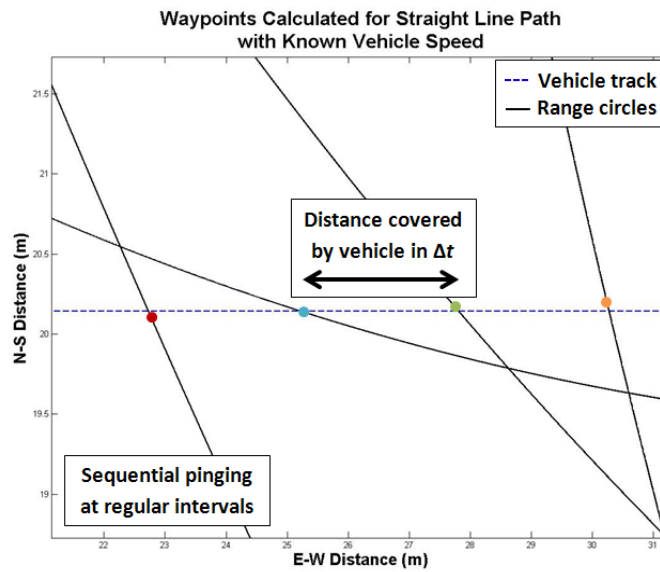


Figure 27. A vehicle's speed can be used to solve for its future positions.

Initial testing of this method, inspired by Ensign Michael Moberg, showed that for ideal synthetic data, the iterative nature of the function solver yielded inaccurate solutions in about 50% of the simulations. When the function solver converged on a correct solution, the results were highly accurate. This iterative approach was computationally inefficient and was not further investigated, but it serves as a simple example of incorporating additional information into the localization algorithm.

2. Previous Positions

The previous position of a vehicle is valuable information, and can be used to filter out ambiguous solutions. In many cases, the incorrect ambiguous solution from intersecting range circles is more distant from the previous vehicle position than is physically possible, given the vehicle's speed. Further, if the pings are sent at a sufficient frequency or the track of the vehicle is unchanging, the previous two positions of the vehicle can be used to estimate the vehicle heading.

THIS PAGE INTENTIONALLY LEFT BLANK

XIII. CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

A. THESIS CONCLUSIONS

Simulations of a mobile sensor node moving in the domain of a fixed sensor network were created. Synthetic node-to-node range data were generated to test existing range-based localization algorithms.

Simulations showed that difficulties arise when using the previously developed difference linearization (DL) algorithm due to matrix inversion. When solving linear matrix equations, the condition number of a matrix characterizes the solution's sensitivity to error in the equation inputs. The inverted matrix in the DL algorithm includes only the fixed positions of remote nodes. Thus, the condition number can be used to evaluate good network geometries. For bad network geometries, the localization error is large for the DL method. A correction to the difference linearization method is proposed that checks the condition of each inverted matrix and discards solutions calculated with ill-conditioned matrices. Difference linearization with a condition check (DLC) greatly improves the performance of the DL method in over-determined situations.

The filtered pairwise (FP) algorithm developed in this thesis is a good alternative to the DL and DLC methods previously developed. The FP method does not require matrix inversion. Comparisons of the two localization algorithms in simulation showed that for good network geometries, the DLC method yielded lower average localization error with error-free data, when varying the number of remote nodes, and when dwell times are added at the remote nodes. The FP method is shown to be more robust when error is introduced to the individual range measurements.

In practice, only three nodes will typically be available for ranging. If the three nodes are arranged in a geometry that yields ill-conditioned matrices, the FP method performs better than the DL and DLC methods.

The sensor network deployed in Del Monte Lake on the NPS campus in April 2012 was used to collect node-to-node ranges between the submerged, anchored remote

modems and a local modem towed behind a boat operating on the surface. Experimental data were collected and analyzed using the FP and the DLC algorithms. GPS fixes were recorded simultaneously to provide a reference track for analysis.

During in-air testing of equipment, it was determined that the broadcast ping command was not suited for the lake experiments. The experimental data were taken by sequentially pinging individual remote nodes with the mobile local node. A simulation of sequential pinging with error-free synthetic data shows that the average localization error with the FP method has a lower limit of approximately 3 meters.

In experimentation, the FP method calculated the track of the mobile node with more accuracy than the DLC method and had a lower computation time.

The superior performance of the FP method over the DLC method with only three reference nodes available and the algorithm's robustness make the simple FP method an attractive option for localization. The FP algorithm can be further improved by incorporating more information about the mobile vehicle, such as its speed, heading and depth.

B. FUTURE WORK

1. Kalman Filters

Kalman filters are commonly used for tracking and navigation of autonomous vehicles with uncertain or noisy measurements. A Kalman filter algorithm is used to iteratively estimate the state of a linear system that is subject to Gaussian noise. The filter estimates a parameter vector based on previous measurements of the system, and minimizes the mean squared error of the estimate. New information is fed into the algorithm as it is received. A forgetting factor is used to truncate the data from previous measurements that are fed into the Kalman filter. The basic Kalman filter, forgetting factor and sample coded applications are described in [24].

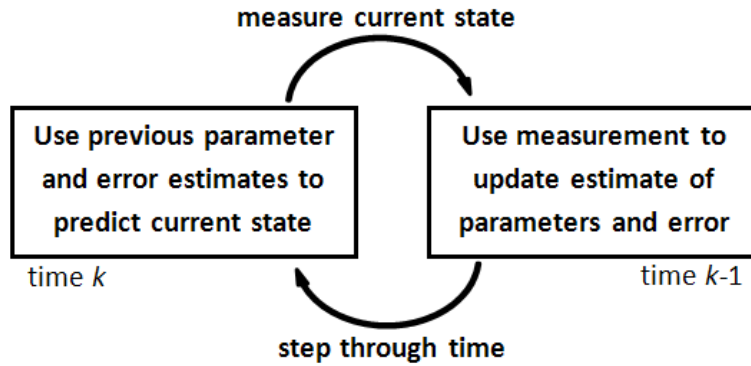


Figure 28. Schematic of a Kalman filter procedure. After [25].

An Extended Kalman Filter (EKF) is used for non-linear systems. The application of an EKF to localization is described in [25]. For a two-dimensional scenario, the system, or behavior of a vehicle, can be described with a linear matrix equation derived from basic kinematic equations. Shareef and Zhu [25] step through the procedure of an EKF algorithm for a trilateration problem with the range equations expressed in Equation (20). An EKF is used to make a position estimate, a position-velocity estimate, and a position-velocity-acceleration estimate depending on the known information. The benefit of using additional information depends on the particular scenario. The formulation in [25] can be extended to create a position-velocity-bearing model of an EKF. Ideally, the depth of a vehicle can be used for a three-dimensional model of an EKF. An Extended Kalman Filter can be tested using the experimental data already collected and no additional information. Further simulations and experimentation can investigate the benefit of adding speed and acceleration information, as discussed in [25], as well as other parameters.

2. Experimental Controls

When conducting future experiments in Del Monte Lake, further experimental controls are possible. The uncertainty in the sound speed can be resolved by taking conductivity-temperature-depth (CTD) probe measurements to determine the sound-speed-dependent medium characteristics. More accurate GPS loggers can be used, or their agreement can be checked prior to conducting lake tests. The GPS logger agreement

depends in part on the satellites that the loggers choose to use as references. The satellites can be explicitly selected with the accompanying software.

A thorough survey of the lake, specifically its bathymetry and the bottom type, would aid in modeling the sound propagation. The remote nodes could also be more accurately located. The weights should be anchored and monitored to ensure their locations do not drift over time. The tethers can be better rigged to limit the watch circles of the underwater nodes. Recordings of the acoustic signals received at the local node can be studied to determine the effect of multipath on the accuracy of the range data. Finally, a statistical analysis of the accuracy of range data can be conducted by ranging between stationary anchored nodes over a period of time.

The environmental characterization and added experimental controls will enable better quantization of the localization error based on range measurements.

C. APPLICATIONS

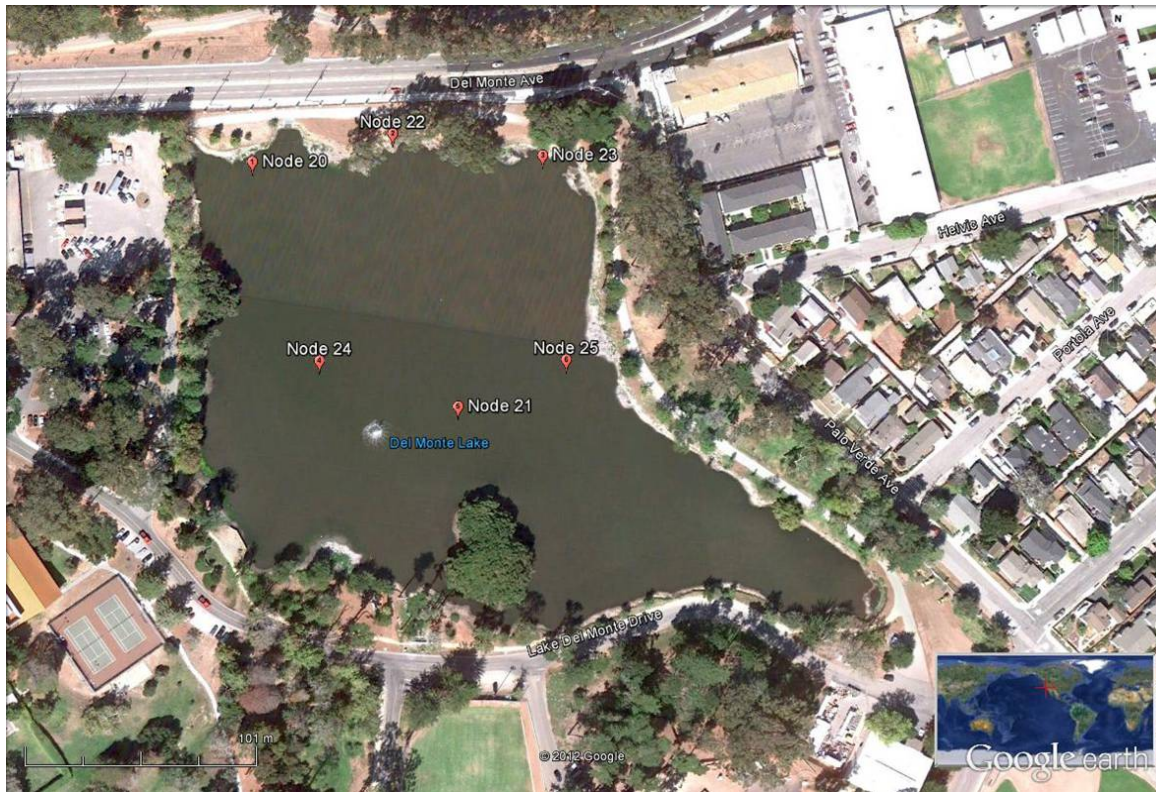
The node-to-node range data for Seaweb acoustic sensor networks are collected as a by-product of normal acoustic communications. Accurate localization and tracking of mobile sensor nodes can be the basis for deployable underwater ranges in the future.

When submerged, UUV's and submarines must use inertial navigation to estimate their position. Fix expansion occurs over time, requiring the vehicles to surface and obtain a GPS fix. A deployable range of nodes can serve as an aide to submerged navigation. Submerged or surface vehicles fixed with a local node can use node-to-node range data to establish a position estimate in a GPS-denied environment.

Localization algorithms developed for mobile nodes can also be implemented in a network to localize the stationary nodes. In deep water it can be difficult to accurately measure the location of nodes. Self-localization in a network is desired to initialize and update the positions of remote network nodes.

The development of further practical applications for Seaweb sensor networks will benefit from accurate, robust, and real-time range-based localization algorithms.

APPENDIX A: DEL MONTE LAKE AND MODEM LOCATIONS

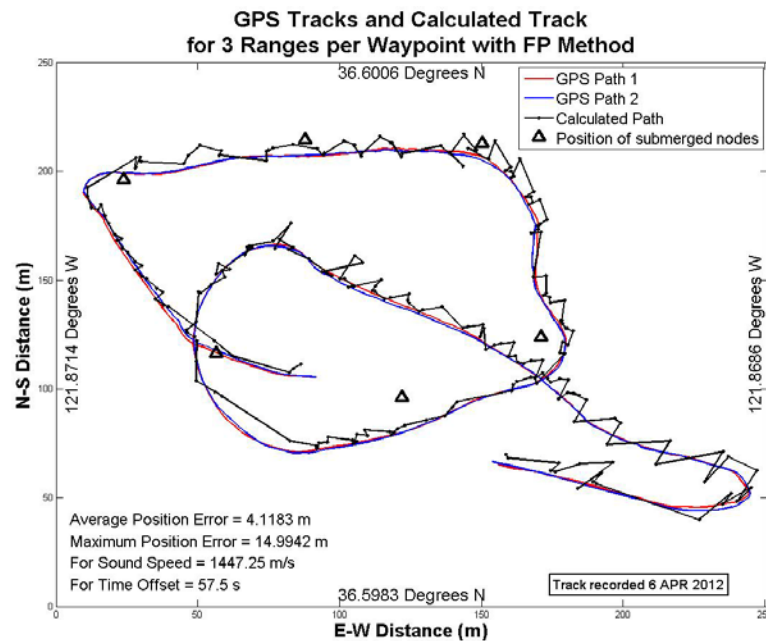
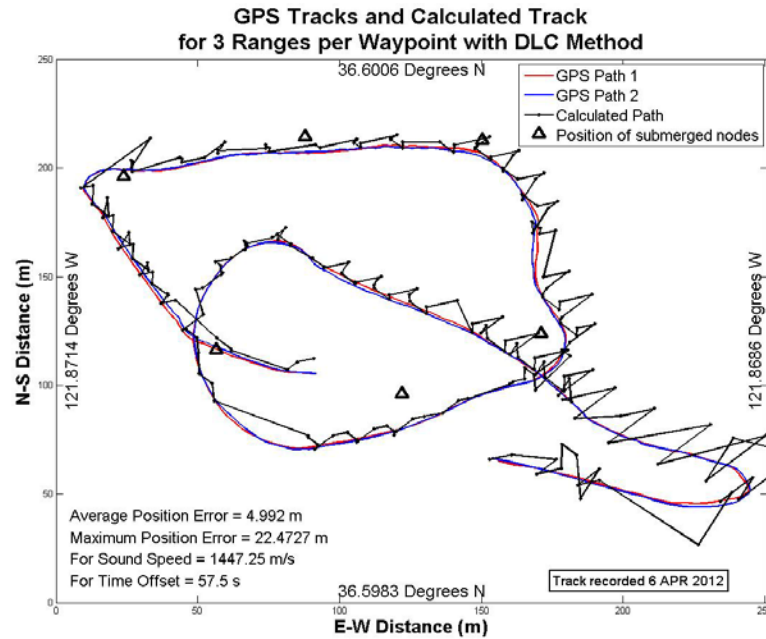


The Google Earth image shows the approximate locations of the deployed modems in Del Monte Lake on the NPS campus. The GPS coordinates from the Garmin GPSmap76CS logger were entered into Google Earth. The pictured location of node 22 indicates an error either in the logged GPS locations or in plotting the points via Google Earth. The node numbers refer to the node ID numbers used for node-to-node communications. Nodes 20–25 are referred to throughout this thesis as nodes 1–6. The ID number for local node, fixed to the towed sled, is 26.

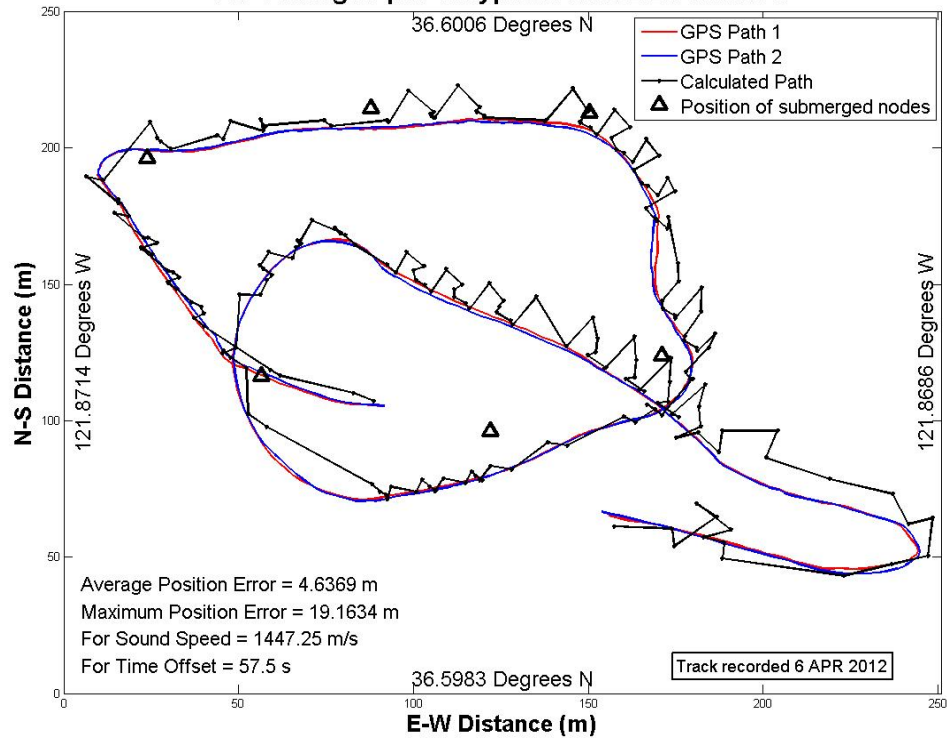
THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B: PLOTTED EXPERIMENTAL RESULTS

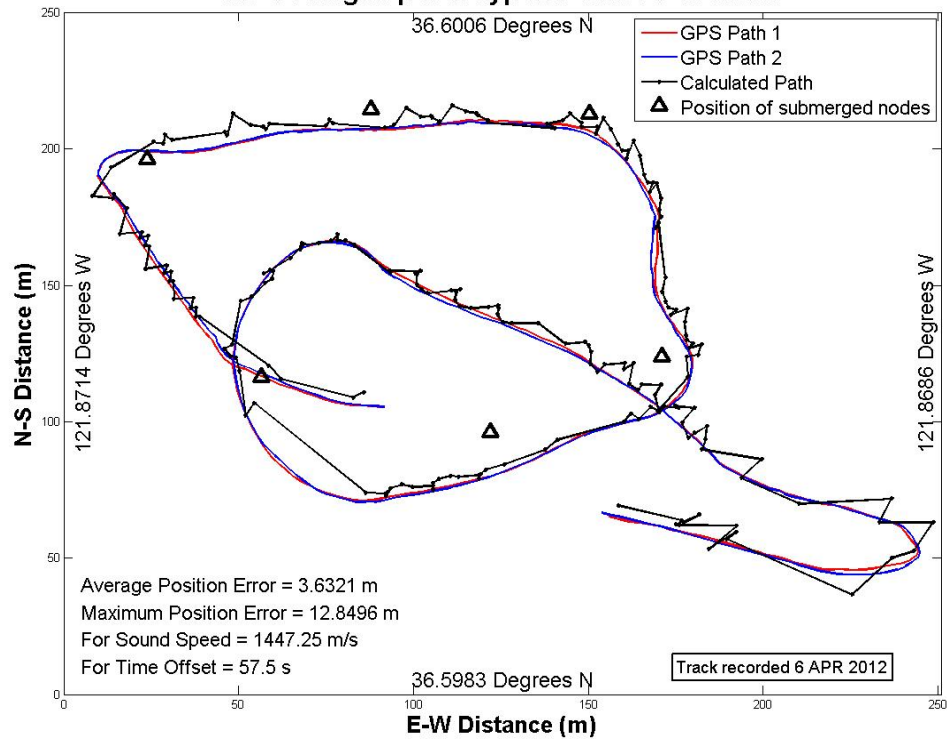
Contact professor J.A. Rice at jarice@nps.edu for a copy of the experimental data and the Matlab codes used to implement the localization algorithms and create these plots.



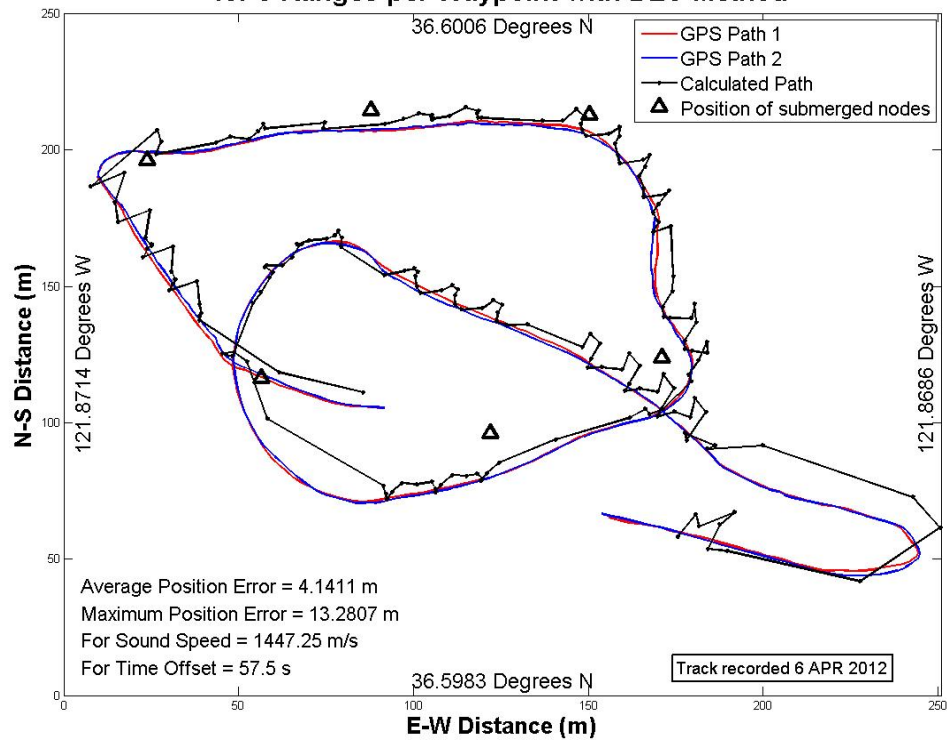
GPS Tracks and Calculated Track for 4 Ranges per Waypoint with DLC Method



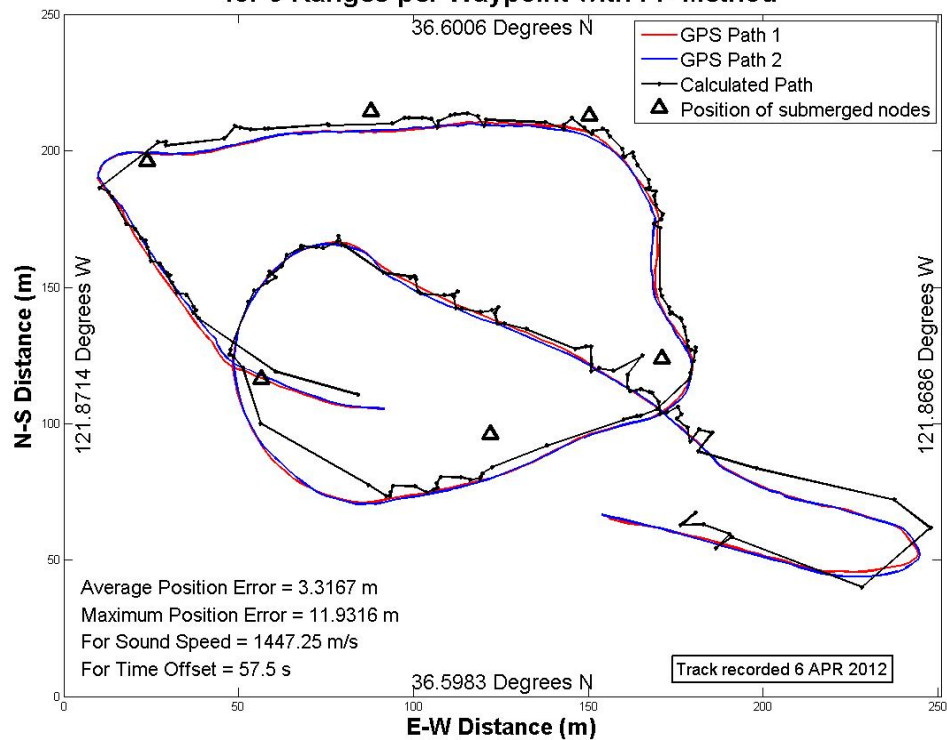
GPS Tracks and Calculated Track for 4 Ranges per Waypoint with FP Method



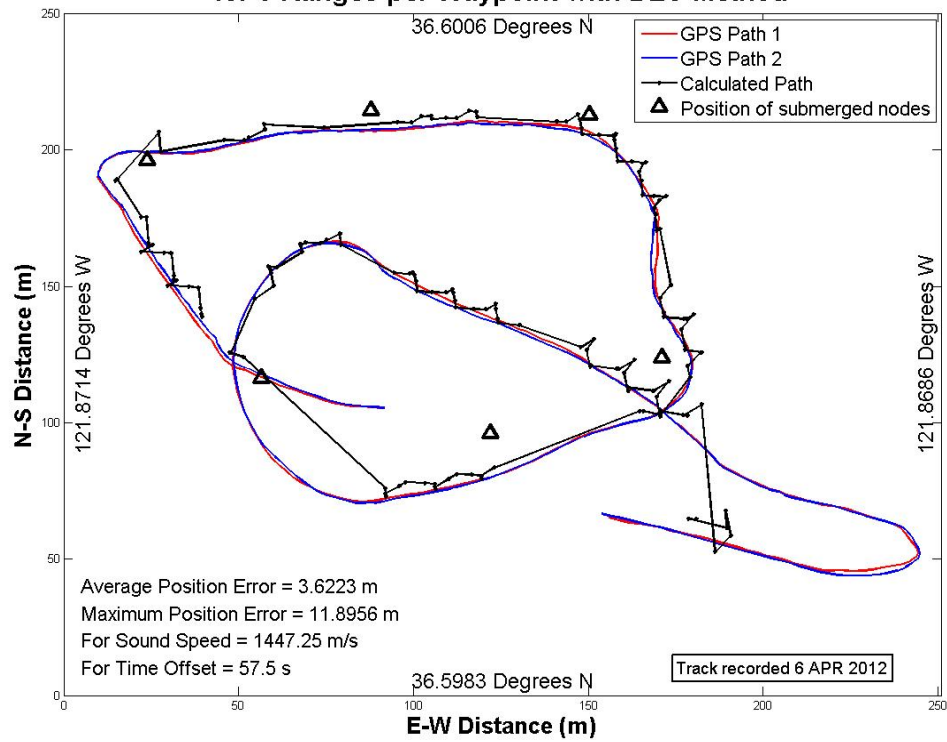
GPS Tracks and Calculated Track for 5 Ranges per Waypoint with DLC Method



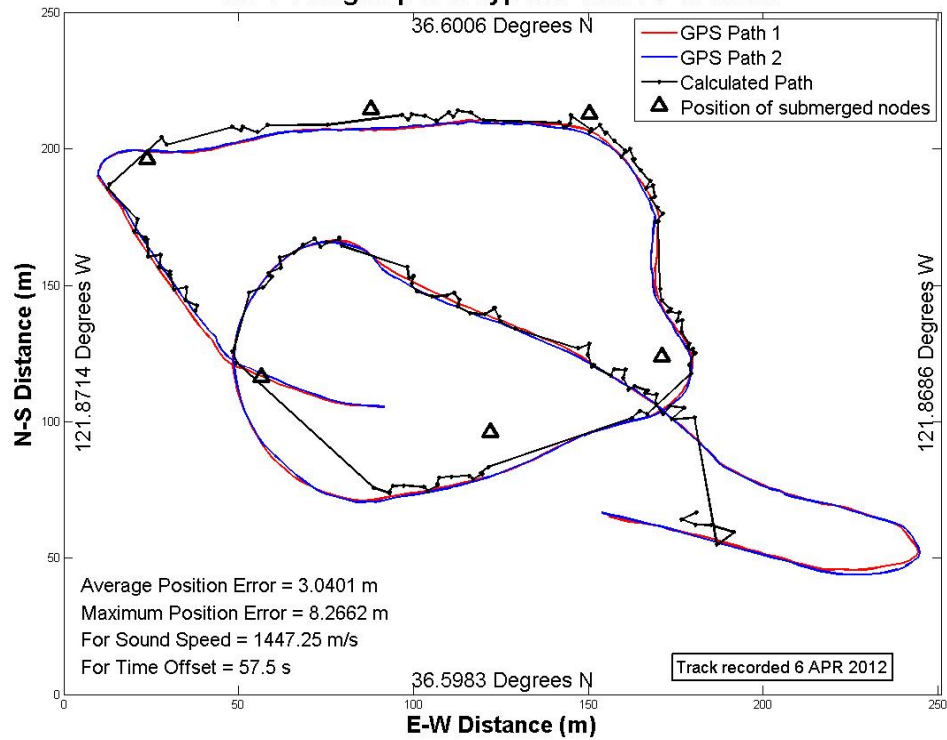
GPS Tracks and Calculated Track for 5 Ranges per Waypoint with FP Method



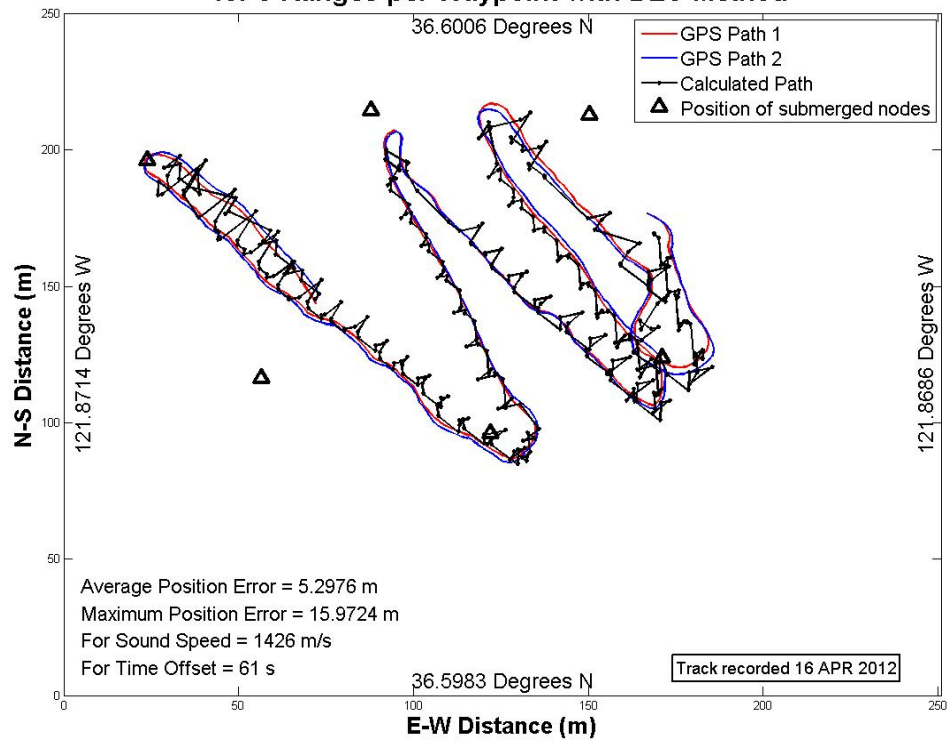
GPS Tracks and Calculated Track for 6 Ranges per Waypoint with DLC Method



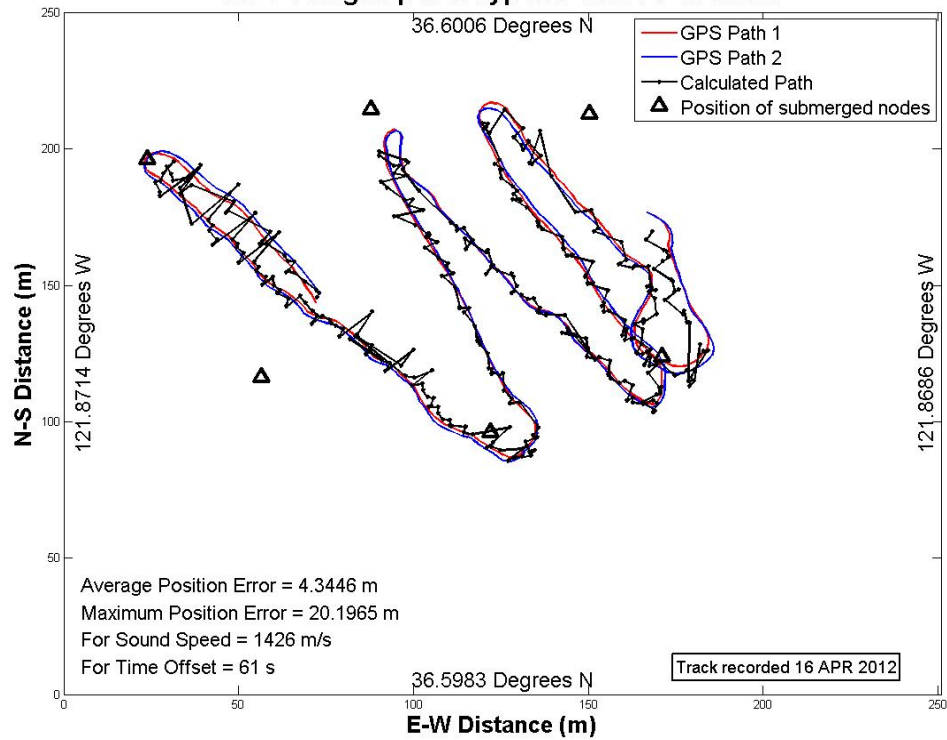
GPS Tracks and Calculated Track for 6 Ranges per Waypoint with FP Method



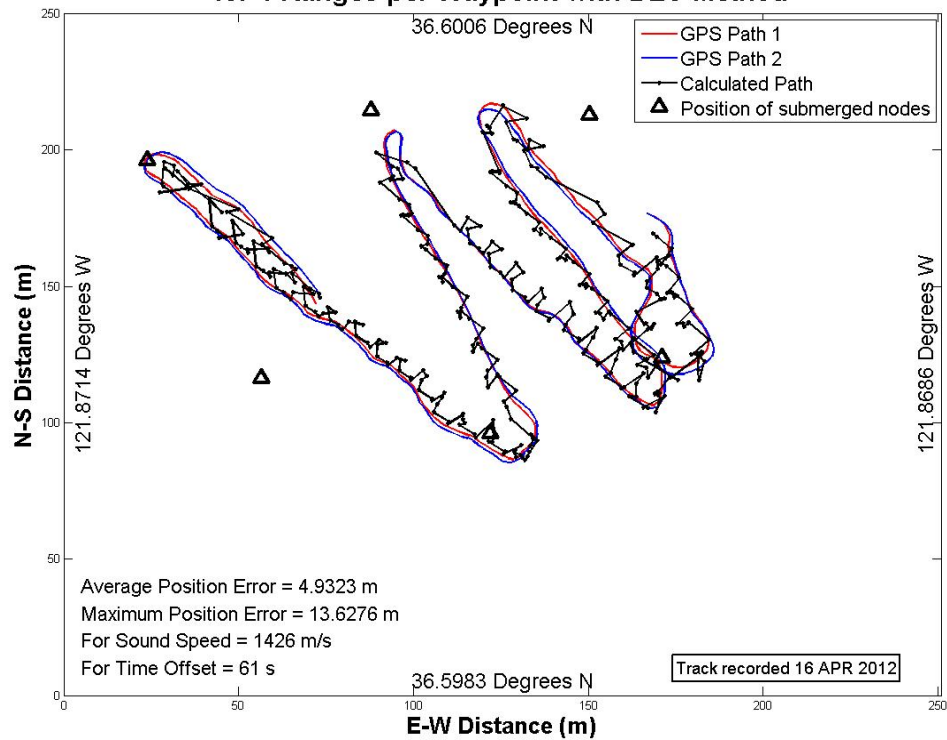
GPS Tracks and Calculated Track for 3 Ranges per Waypoint with DLC Method



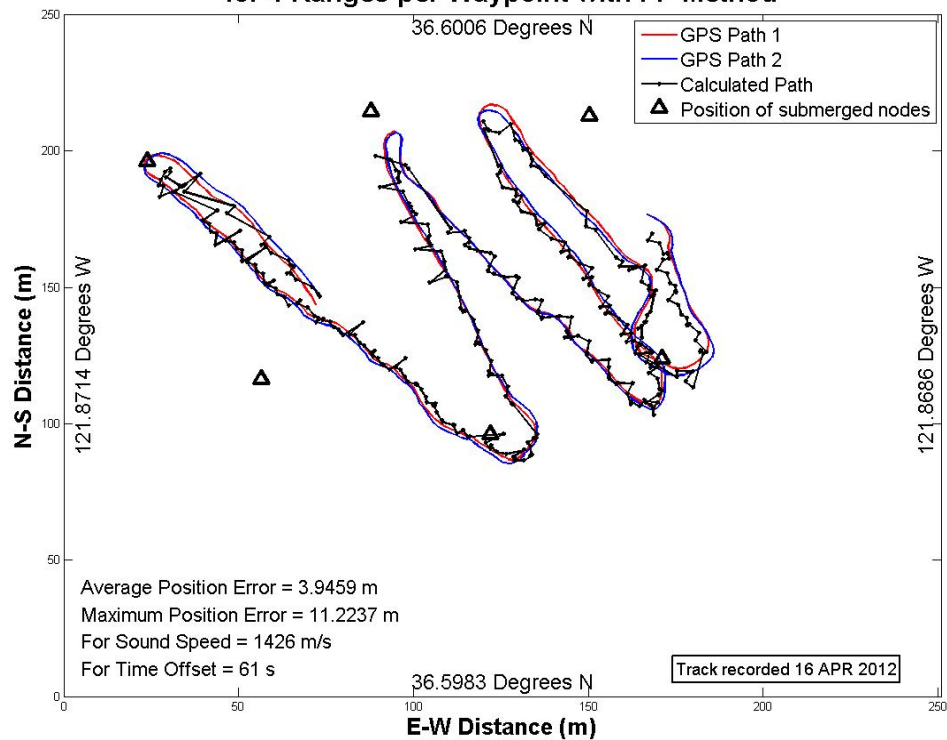
GPS Tracks and Calculated Track for 3 Ranges per Waypoint with FP Method



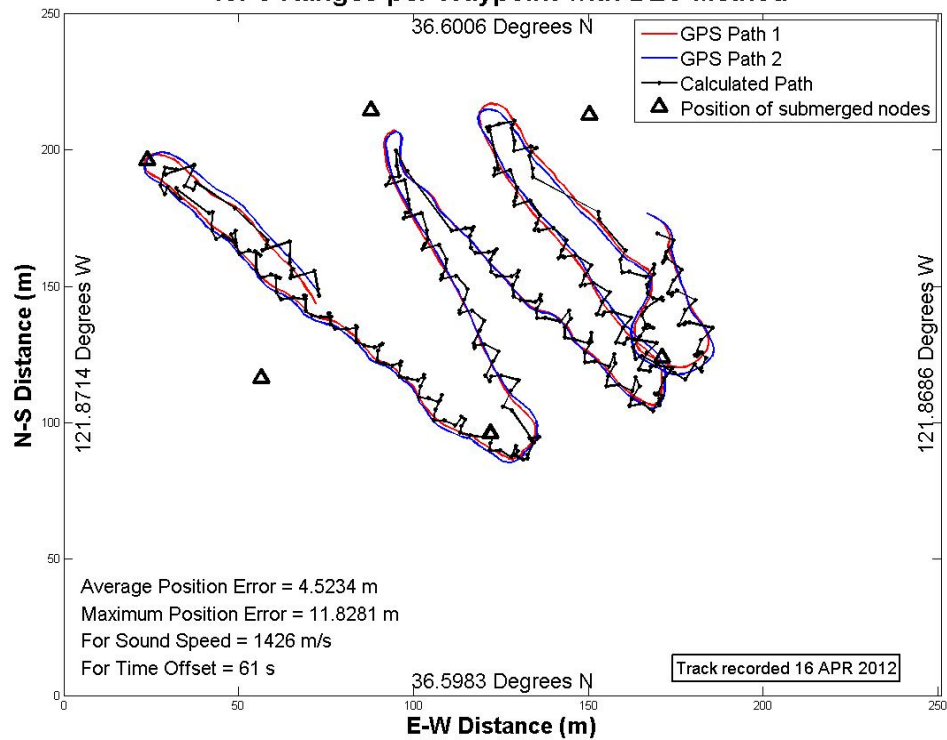
GPS Tracks and Calculated Track for 4 Ranges per Waypoint with DLC Method



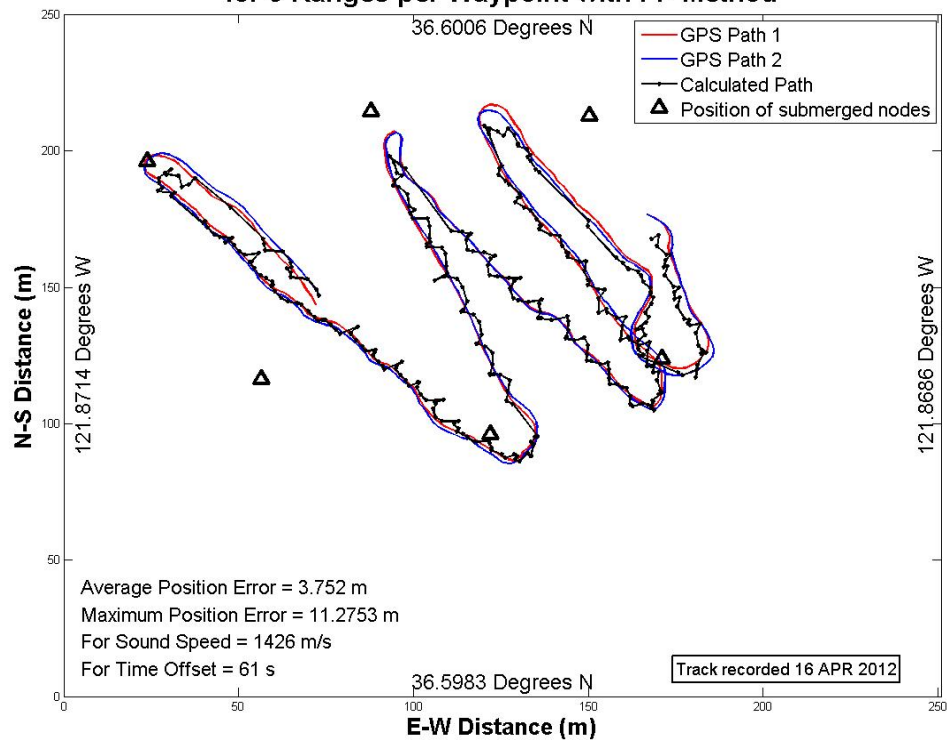
GPS Tracks and Calculated Track for 4 Ranges per Waypoint with FP Method



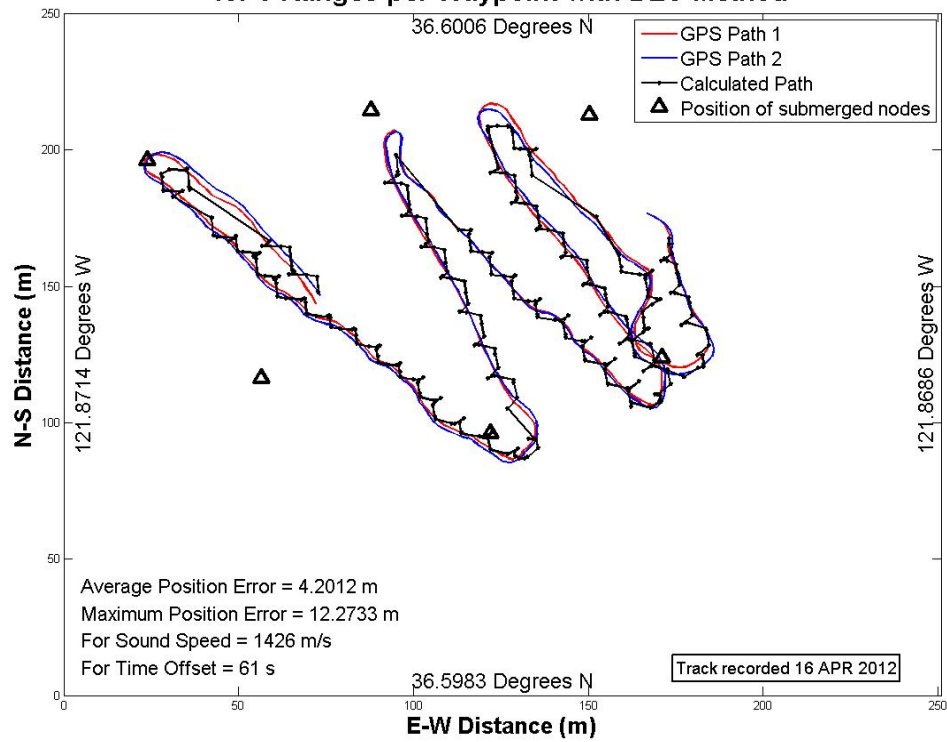
GPS Tracks and Calculated Track for 5 Ranges per Waypoint with DLC Method



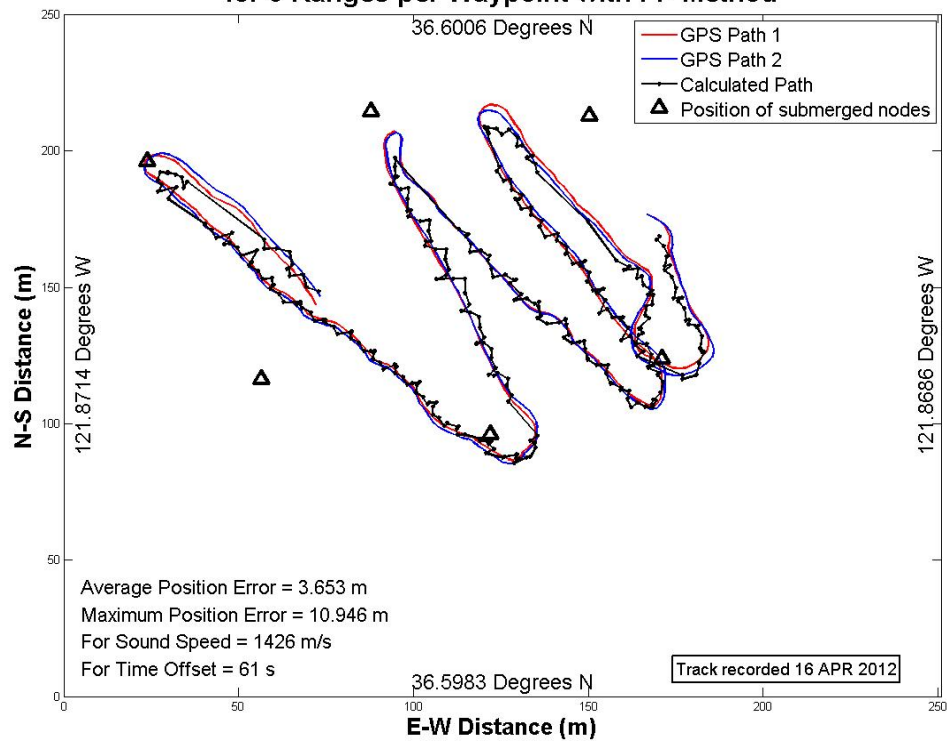
GPS Tracks and Calculated Track for 5 Ranges per Waypoint with FP Method



GPS Tracks and Calculated Track for 6 Ranges per Waypoint with DLC Method



GPS Tracks and Calculated Track for 6 Ranges per Waypoint with FP Method



APPENDIX C: CONDITION NUMBERS OF MATRICES

Condition numbers	1	2	3	4	5	6	7	8	9	10
Node Combos	20	20	20	20	20	20	20	20	20	20
	21	21	21	21	22	22	22	23	23	24
	22	23	24	25	23	24	25	24	25	25
for P matrix	3.99	1.84	5.16	2.60	6.38	2.18	2.22	2.89	1.42	1.56
for Q matrix	2.54	2.02	5.78	5.93	16.16	1.32	4.01	1.61	3.43	3.37
for R matrix	1.85	1.47	2.21	3.67	15.74	2.81	5.55	1.86	2.67	4.03

Condition numbers	11	12	13	14	15	16	17	18	19	20
Node Combos	21	21	21	21	21	21	22	22	22	23
	22	22	22	23	23	24	23	23	24	24
	23	24	25	24	25	25	24	25	25	25
for P matrix	2.14	3.36	4.26	3.68	4.38	6.18	3.24	1.59	1.49	2.64
for Q matrix	3.77	2.41	2.28	1.76	3.25	2.37	1.79	3.14	1.78	2.07
for R matrix	1.99	1.50	2.27	2.34	1.76	5.62	4.40	4.01	2.10	1.33

The **P**, **Q** and **R** matrices are defined as:

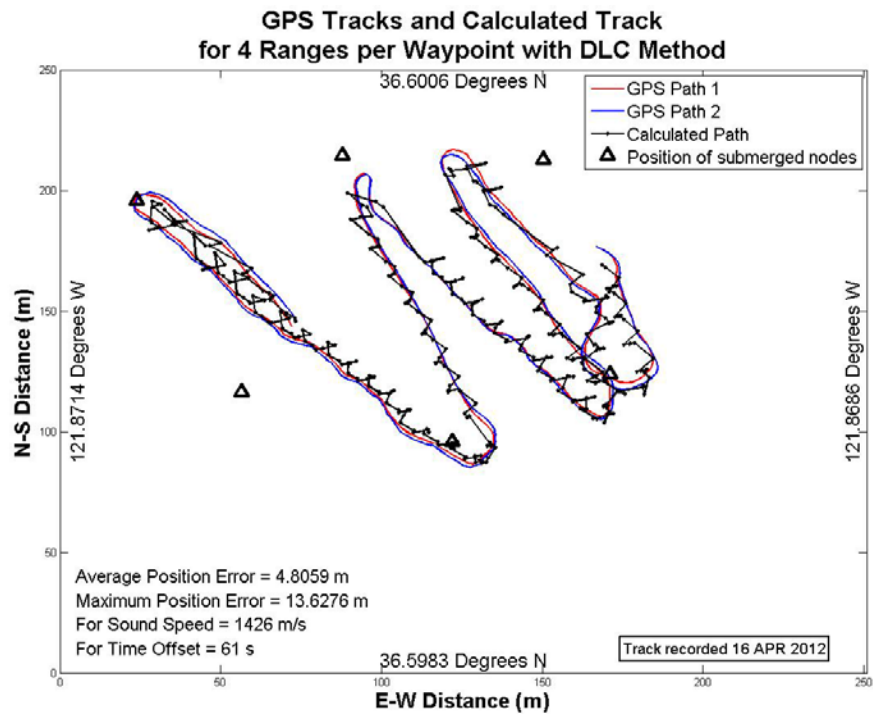
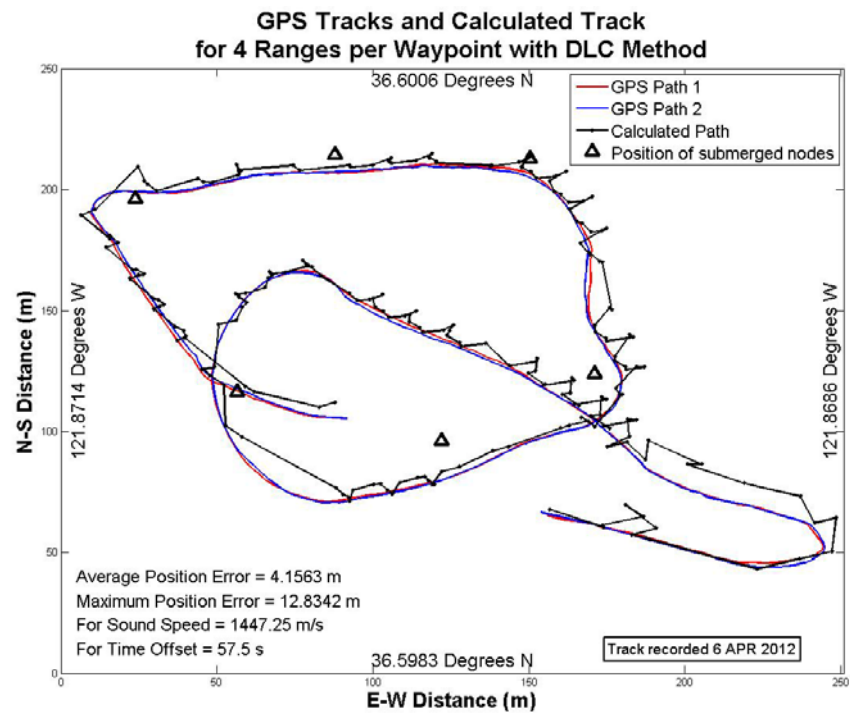
$$\mathbf{P} = \begin{pmatrix} (x_2 - x_1) & (y_2 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix}$$

$$\mathbf{Q} = \begin{pmatrix} (x_3 - x_1) & (y_3 - y_1) \\ (x_3 - x_2) & (y_3 - y_2) \end{pmatrix}$$

$$\mathbf{R} = \begin{pmatrix} (x_3 - x_1) & (y_3 - y_1) \\ (x_1 - x_2) & (y_1 - y_2) \end{pmatrix}$$

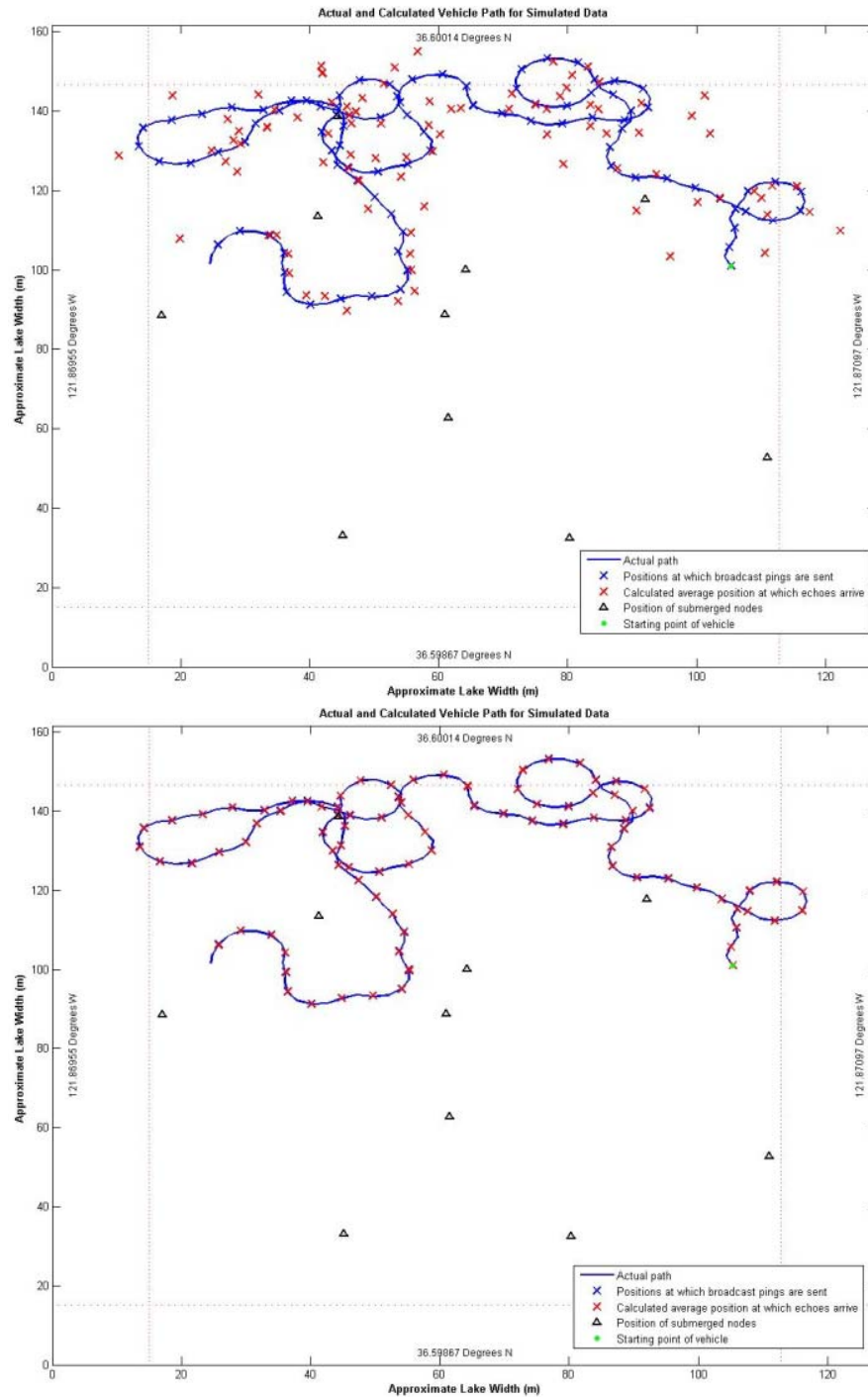
THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D: RESULTS FOR CONDITION NUMBER MINIMUMS



THIS PAGE INTENTIONALLY LEFT BLANK

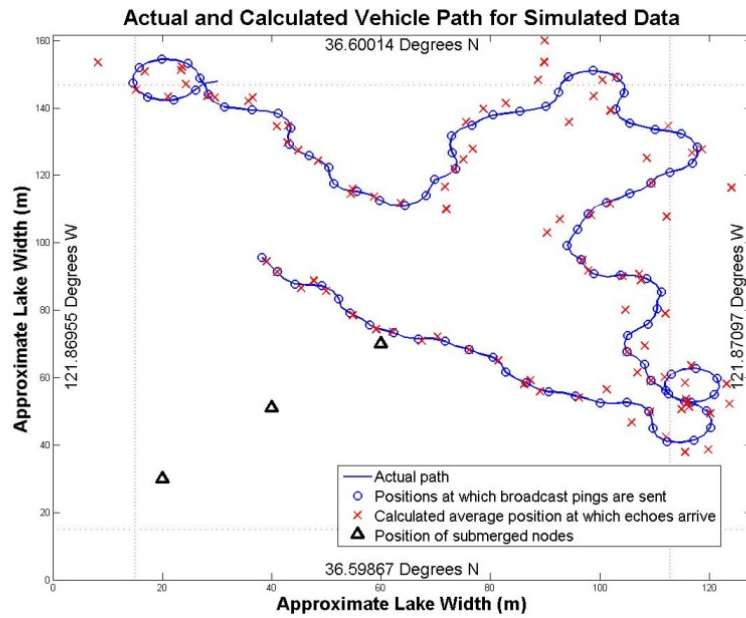
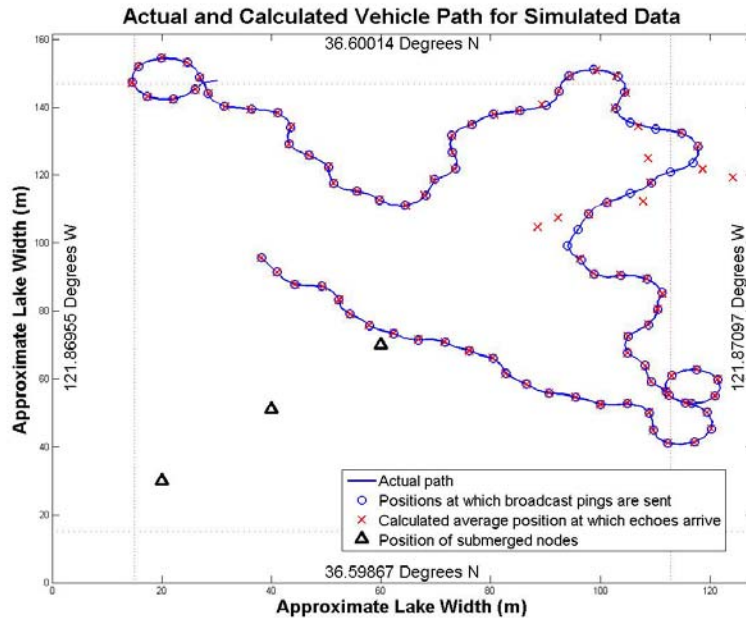
APPENDIX E: CORRECTING EIGENVALUE ERROR



Example of DLC algorithm output after data from matrices with eigenvalues less than 1 are filtered out. The plots were created using synthetic data with 10 reference nodes.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX F: STRAIGHT LINE NODE DEPLOYMENT



When three reference nodes are deployed in approximately a straight line, the FP method performs well, whereas the DL method has a much higher average positioning error due to inversion of an ill-conditioned matrix.

THIS PAGE INTENTIONALLY LEFT BLANK

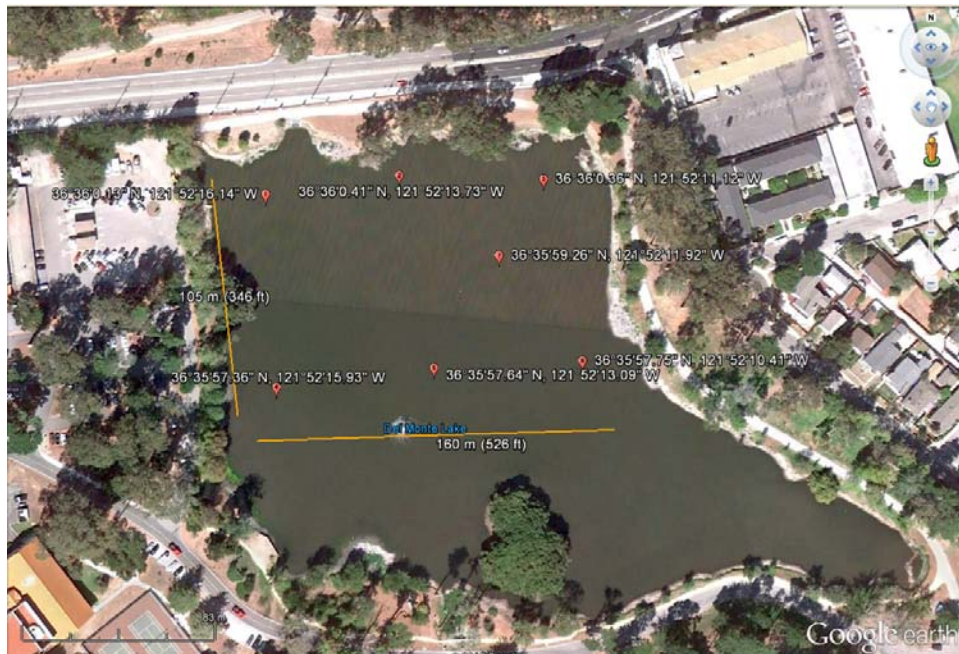
APPENDIX G: PRELIMINARY EXPERIMENTAL PROPOSAL

Del Monte Lake Initial Experiments
ENS R. E. King, USN

APR 2012

ESTABLISHING A SEAWEB NETWORK IN DEL MONTE LAKE & COLLECTING RANGE DATA AND GPS FIXES FOR A SURFACE VEHICLE

OBJECTIVE: To deploy Seaweb nodes in lake and establish a network. To collect range data between submerged Seaweb nodes and mobile node, towed by surface vehicle, and record GPS fixes along vehicle track.



[Fig. 1] Sample Arrangement of Submerged Nodes in Del Monte Lake (Image from Google Earth)

TEST SITE, ENVIRONMENTAL IMPACT & SAFETY: The Del Monte Lake on the NPS campus, shown in Appendix (B), is a shallow fresh-water lake, approximately 500 ft by 500 ft. The depth is estimated to be between 4 and 7 ft. The lake is fed by drain water and has a silt bottom. Historically, this body of water was part of an estuary system until it was isolated from the sea by the construction of a railway. The lake was dredged in 2009. The lake features an island and a fountain that will introduce ambient noise to the experiment. The lake is advantageous for Seaweb experiments due to its ready access by students, its security as part of the NPS campus, and its lack of marine mammals and other species that might otherwise be impacted by acoustic transmissions. Acoustic

transmissions are within human hearing range and will not negatively impact the local wildlife. The equipment used in this lab is made of non-corrosive materials. Lake tests for this experiment are estimated to last one week after which all equipment will be removed from the lake. All equipment, including the boat motor, uses clean battery power. All provided instructions for motor and battery operation will be followed. Two people will be present at all times in the boat. Life vests will be worn at all times.

READING REFERENCES:

- (a) Endura Pro C2 Trolling Motor Owner's Manual
- (b) DieHard Automatic Battery Charger Owner's Manual
- (c) LifeSled Specifications Sheet, www.lifesled.com
- (d) G-Log 760 GPS Recorder User's Manual, www.transystem.com.tw
- (e) Acoustic Telemetry Modems User's Manual, P/N 003452, Rev. E
- (f) UDB-9000 Universal Deck Box User's Manual, P/N M-270-10, Rev. B

EQUIPMENT: 13 foot Aluminum Boat; Minn Kota ENDURA Pro C2 55 lb Transom-Mounted Trolling Freshwater Motor (Model #11910023); DieHard Marine Deep Cycle Battery, Group Size 27M (Model #27524); DieHard Automatic Battery Charger (Model #71222); LS1 towed sled; G-Log 760 GPS Recorders (S/N 8410000916 and 8410000917); Garmin GPSmap 76CS logger (S/N 10R-022508); Teledyne Benthos Transducer (Part #C-270-167-1, Model #AT-440-LF-0, S/N 45444); Teledyne Benthos Deck Box (Part #013189 Model # (XDCR) UDB-9000M, S/N 45450); Seaweb repeater nodes (Part #013308, Model #ATM-885); 15 lb Greenfield Products Mushroom Weights (Part #515-E-UPC); 24 lb Surface Floats.

Also needed: Oars; Life Preservers; Rope; Battery Cables; Transducer Cables; Assorted Tools



[Fig. 3] The boat used in this experiment is similar to the Sears Gamefisher aluminum boat shown. (Image from GovernmentAuctions.org)

The aluminum boat used for this experiment is borrowed from the Public Works Environmental group at NPS. It is approximately 13 ft long and 4 ft wide, intended to carry up to 300 lb.



[Fig. 4] Minn Kota ENDURA Pro C2 Transom Mounted Trolling Freshwater Motor. (Image from www.minnkotamotors.com)

The trolling motor is a transom-mount design. The depth of the propeller is adjustable. The shaft is 30 inches. The shaft is sturdy, but designed to flex in case of impact on rocks or other objects in the lake. The propellers are Weedless Wedge 2 type, designed to avoid tangling in weeds on the bottom of the lake. The motor is battery-powered and has 55 lb of thrust. “Pounds thrust” is the common rating for trolling motors. For a 13 foot boat, 30 pounds of thrust are required, assuming the boat is carrying two people and little to no additional weight. The motor will be used for towing the LifeSled, and the boat will be loaded with the weight of the batteries and the modem’s deck box. Operating with additional weight, high winds, currents, weedy bottoms, or other adverse conditions requires additional thrust. Adverse conditions are not expected on the lake. The 55 lb thrust motor should be more than capable of moving the boat and towed equipment. All motors will achieve speeds of only 2-4 miles per hour. Trolling motors are designed to be quiet so as not to scare fish. The quiet operation will be beneficial for our experiment. The motor operates on 12 Volts. It requires one 12-volt deep cycle marine battery.



Battery Type:	
Group Size:	27M
Temperate Zone:	All climates
Transportation Type:	Marine and recreational vehicle
Power Configuration:	
Voltage:	12.0
Power Ratings:	
Amp Hours at 20 Hour Rate:	105
Cold Cranking Amps (CCA at 0 deg.F):	575
Reserve Capacity (RC):	180 min.

[Fig. 5] DieHard Marine Deep Cycle Battery, Group Size 27M. (Image and table from www.sears.com)

The batteries used are 12-volt and intended for use with marine vehicles. They each weight 57 lb The weight must be taken into consideration for the total weight of the boat. Instructions for battery maintenance, operation and safety are included in the DieHard Automatic Battery Charger Owner’s Manual (b).

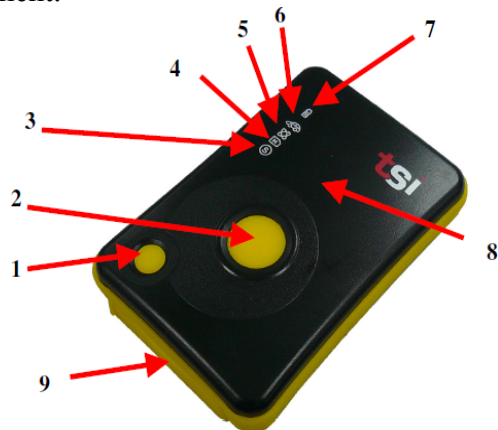


Amps:	10/2/50
Auto Rev. Polarity Protection:	Yes
Automatic:	Yes
Engine Start Amps:	50
Fast Charge Amps:	10
Item Type:	Tools
Manual Operation:	Yes
Overall Color:	Black
Primary Type:	Switch
Spark Proof Auto Mode:	Yes
Trickle Charge Amps:	2

[Fig. 6] DieHard Fully Automatic Battery Charger and Engine Starter, for 12-volt Batteries. (Image and table from www.sears.com)

The batteries will be charge on shore over a period of 8-12 hours. The battery charger plugs into a standard wall outlet. All instructions and safety measures should be observed as outlined in the Owner’s Manual.

The LifeSled LS1 is 62 in x 36 in x 4.5 in and weighs 42 lb. It has over 450 lb of deadweight buoyancy. The sled has a 3-point hook up system for towing. It has 10 internally anchored handles for securing equipment.



1. Power bottom
2. Push to log (POI) bottom
3. Smart mode Led (Blue)
4. Memory capacity LED (Red)
5. GPS status LED (Orange)
6. Car mode LED (Blue)
7. Battery status LED (Red/Green)
8. Built-in patch Antenna
9. USB connector & Reset button (with cover)

[Fig. 8] LifeSleds

[Fig. 9] (Image from User’s Manual(c))

The G-Log 760 GPS recorder is a waterproof, all-enclosed device that is 52 x 82 x 19 mm, and therefore easily mountable to the towed sled. It can log up to 50,000 waypoints with a 1 Hz refresh rate and can operate for 25 hours. It uses a rechargeable lithium-ion battery and charges via USB connection to a computer. The operating parameters for the lake tests, including maximum speed, acceleration, temperature, altitude and humidity, are well within the limitations of the logger and the lithium-ion battery. The GPS logger records date, time, longitude, latitude, altitude, speed and heading for every fix. It is accurate to within 3 meters.

The G-Log 760 has three modes: normal, smart and car mode. Normal mode records a fix at the set interval and remains on until manually powered off. Smart mode adjusts the fix frequency based on the current speed and turns off the logger if the logger is stationary for 3 minutes to conserve power. Car mode is intended for higher speeds. Data is taken every 5 seconds, and only while logger receives external power, usually from the car cigarette lighter. Loggers will be used in normal mode to avoid accidentally being powered down if stationary during a run, and to ensure a constant fix frequency.

If logger is in car or smart mode, pushing the reset button (9) will return the logger to normal mode. **NOTE:** Pushing and holding the reset button for 5 seconds will return the GPS logger to factory default settings and *will erase any logged data*. Be sure NOT to hold reset button if returning the logger to normal mode.



[Fig. 12] Mushroom Weight
(Image from www.westmarine.com)



[Fig. 13] Surface Float

The mushroom weights are 15 lb cast iron weights with no-mar plastisol protectant coating to avoid corrosion. They are intended for use with sand, clay and firm soil bottom types. The lake is known to have a primarily silt bottom. The 15 lb should adequately weight down the transducer so that there is no slack in the rope.

The surface float provides 24 lb buoyancy. The surface floats will serve to keep the rope taught and mark the location of the nodes for use when transiting the lake. The length of rope used will vary depending on the depth at the site.



[Fig. 14] Repeater Nodes

We are using 7 Teledyne Benthos ATM-885 Acoustic Telemetry Modems (labeled #20-26) that each include a built-in omnidirectional transducer. The modem battery pack and electronics are contained in a hard coat anodized aluminum housing. The modem has a built-in piezoceramic transducer encased in urethane also with a protective anodized aluminum housing. The modems can operate at depths up to 2000 m. The transducers operate in the low frequency band of 9-14 kHz with a toroidal beam pattern that is near-omnidirectional. The attainable range depends on the source level, signal frequency, beam pattern, depth of modem, transducer orientation relative to other transducers, and environmental characteristics. Typical horizontal ranges are 2-3 km for omnidirectional transducers. The modems can operate independently on battery power or can be connected to a host processor and dc power source via a serial connection. The modems will be fixed to the anchor weights and surface floats so that they are suspended approximately above the lake bottom.

PROCEDURE:

Preparing the GPS Recorder for Use – Detailed instructions for G-Log 760 operation, software installation and data viewing can be found in the User's Manual (d). Settings and instructions required to complete this lab are detailed below. A summary of LED display meanings is included at the end of the experimental procedure document. Refer to [Fig. 3] for button and LED locations.

1. Before leaving the lab, install driver and software for GPS device. With the recorder plugged in and powered on, open the TSI LogView program. Click on "Configure GPS Module." Set the fix frequency to 1 fix per second. Do this for each GPS device.
2. Open the GpsView Program. Baud rate is already set. To find the COM number (for Windows), right click My Computer -> Properties -> Hardware -> Device Manager -> Ports (COM&LPT). GPS device should be shown with its COM number.
3. In GpsView Program Status tab, click On. Switch to Setup tab. Fix Update Rate should be 1. In AGPS section, click Update. Once complete, you can close the program. Do this for each GPS device.

NOTE: Be sure to turn off the GPS recorder before unplugging the USB cable.

NOTE: GPS recorders are charged via USB. Ensure both recorders are fully charged before leaving the lab.

4. Power on the GPS logger by pressing and holding the power button (1) for 3 seconds. Press the power to check the battery status when the power is on before each run. Orange GPS status LED (5) will stay on while recorded finds satellite signal. This may take up to 1.5 minutes. Be sure to power on the device in an open space away from buildings with a clear view of the sky. Remain stationary during satellite signal acquisition. GPS status LED will blink when signal is found and recorder will begin logging.
5. Push POI button (2) to record Point Of Interest. Memory status LED (4) will blink 3 times to confirm POI was recorded. If POI is recorded at a known location on the lake perimeter, GPS data can be compared to the physical location on a Google Earth display. Record a POI where the boat is launched and retrieved for every run. Make note of physical location for later comparison.
6. Fix GPS logger to sled.
NOTE: A third, more accurate GPS unit will be used in addition to the two G-Log 760 recorders.

Attaching the LifeSled and Mobile Node – No tools are required for attaching the node to the LifeSled other than a crescent wrench to tighten the lock nuts. This can be done in the lab, before heading out to the lake.

7. The transducer is fixed to the LifeSled as depicted:



[Fig. 15] Towed sled fixed with acoustic modem

8. Attach the 25 m cable to the deck box after the sled is sitting in the water.

Fixing the Motor to the Boat and Attaching Battery Supply –

9. Fix the motor to the stern of the boat using the simple clamp mechanism included.
10. When in deep enough water, adjust the motor height to ensure the propeller is under the bottom of the boat.

NOTE: Be sure to pull the motor up when in shallow water to avoid damage to the propeller.

11. Connect the cable leads from the motor to the exposed connectors on the battery box.
12. Use the battery box cables to hook up the marine battery inside. The LED indicators should light up indicating that the battery is charged.

Deploying the Seaweb Nodes – Surface vehicle will be operated by two people. Life vests should be worn at all times.

13. Secure one mushroom weight to the end of a rope at least 10 feet long. Mark the rope every 6 inches from the bottom of the mushroom weight for at least 10 feet. Use this rope for depth measurements.
14. Load the equipment into the boat, keeping note of the total weight. The boat should not carry much over 300 lb total, including the motor, battery, and passengers.
15. Once the boat is in the desired node location, lower the measurement rope and weight into the water until the weight is resting on the bottom of the lake. Record the water depth and retrieve the weight. **The water depth can also be recorded using a handheld fathometer.**
16. Adjust the rope length on the first node based on the local depth. Lower the node into the water. **Make note of the node number written on the modem.** Once the weight is settled on the lake bottom, record the node position with all GPS loggers used in the experiment. Make visual note of the location of the node for comparison.
17. Repeat for all seven nodes.

Operating the Surface Vehicle for Data Collection – Experiment plans will change based on the data received from initial tests. The following tests are planned:

Run #	Run Description	Purpose
Deploy Nodes	Record GPS log data while deploying nodes.	Ensure GPS recorders are logging data as expected.
1	Vehicle stationary in center of lake for extended period of time. Send 4-5 pings and observe return data. Repeat for multiple positions about lake.	Initial test to ensure equipment is working. Repeated ping and echo measurements from same location can be compared to check that measurements are reliable and to get an idea of expected error in measurements.
2	Navigate vehicle around perimeter of lake, navigating close to, but not over, nodes.	Observe what echoes are available and reliable at bounds of experimental area. Test algorithm performance when close to nodes.
3	60 minute random path with 12 ping/min	Collect random path GPS and range data for use with Matlab algorithms. Collect data that spans the entirety of the operational area.

NOTE: Be sure to regularly check battery level for motor, deck box, and GPS units.

Symbol	Function / Color	Status	Description
	Battery Status / Red	Blinking	Low battery level
		On	During power charging via USB port
	Battery Status / Green	On / Off	Power gauge when both POI and Power buttons are pressed at the same time
	Battery Status / Green & Red	All LED On	Fully charged
	GPS Status / Orange	On	Searching for satellite
		Blinking	GPS fix & start log
	Car mode status / Blue	On	In car mode, device will go into sleep mode when no power supply from USB and device will wake up and start logging when there is power supply from USB.
	Memory capacity / Red	Blinking	The device is logging position
		Blink 3 times	POI button is pressed
		On	The memory is >80% full
		Keep flashing	Erasing data
	Smart Log mode / Blue	On	Smart log mode on/ vibration sensor on
		Off	Normal mode on/ vibration sensor off
		2 sec flashing	Sleep mode
		1/3 sec flashing	Reset button pressed to rest the device

[Fig. 16] LED Display Summary from G-Log 760 GPS Recorder User's Manual (d)

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

- [1] *Seaweb Modem User's Guide*, Version 20.5.6, Benthos, Inc., North Falmouth, MA, 2010.
- [2] J.A. Rice, "Seaweb acoustic communication and navigation networks," in *Underwater Acoustic Measurement: Technologies & Results*, Heraklion, Crete, Greece, 2005.
- [3] M. J. Hahn, "Undersea navigation via a distributed acoustic communications network," M.S. thesis, Dept. of Engineering Acoustics, Naval Postgraduate School, Monterey, CA, 2005.
- [4] S. P. Ouimet, "Undersea navigation of a glider UUV using an acoustic communications network," M.S. thesis, Dept. of Engineering Acoustics, Naval Postgraduate School, Monterey, CA, 2005.
- [5] M. S. Reed, "Use of an acoustic network as an underwater positioning system," M.S. thesis, Dept. of Engineering Acoustics, Naval Postgraduate School, Monterey, CA, 2006.
- [6] D. Kapolka, *Underwater Acoustics for Naval Applications*. Monterey: Naval Postgraduate School, 2012.
- [7] I. F. Akyildiz, D. Pompili and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad Hoc Networks*, vol. 3, pp. 257–279, 2005.
- [8] H.P. Tan et al., "A survey of techniques and challenges in underwater localization," *Ocean Engineering*, vol. 38, no. 14-15, pp.1663–1676, 2010.
- [9] D. Pompili et al., "Three-dimensional and two-dimensional deployment analysis...", *Ad Hoc Netw.*, p. doi:10.1016/j.adhoc.2008.07.010, 2008.
- [10] *Symantec Procomm Plus User's Guide*, Version 4.8, Symantec Corp., Cupertino, CA, 1999.
- [11] T. Bian et al., "Design and evaluation of a new localization scheme for underwater acoustic sensor networks," in *IEEE Global Communications Conf. Proc.*, Honolulu, HI, 2009, pp. 1–5.
- [12] B. Friedlander, "Accuracy of source localization using multipath delays," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 4, pp. 346–359, 1988.
- [13] H. C. Schau and A. Z. Robinson, "Passive source localization employing intersecting spherical surfaces from time-of-arrival differences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-35, no. 8, pp. 1223–1225, 1987.
- [14] F. Reichenbach et al., "A distributed linear least squares method for precise localization with low complexity in wireless sensor networks," in *IEEE Int. Conf. Distributed Computing in Sensor Systems*, San Francisco, CA, 2006, pp. 514–528.

- [15] Y. Huang et al., "Real-time passive source localization: A practical linear-correction least-squares approach," *IEEE Transactions on Speech and Audio Processing*, vol. 9, no. 8, pp. 943–956, 2001.
- [16] P. Bourke, "Intersection of two circles," April 1997. [Online]. Available: <http://paulbourke.net/geometry/2circle/>. [Feb. 7, 2012].
- [17] B. K. Horn, "Quick review of Eigenvalues and Eigenvectors," 30 December 2004. [Online]. Available: <http://people.csail.mit.edu/bkph/articles/Eigenvectors.pdf>. [May 15, 2012].
- [18] A. K. Kaw, "Adequacy of solutions," in *Introduction to Matrix Algebra*. Tampa: University of South Florida, 2012, ch. 9, pp. 148–164.
- [19] J. Stensby, "Analytical and computational methods: Matrix norms," 1998. [Online]. Available: <http://www.ece.uah.edu/courses/ee448/chapter4.pdf>. [May 16, 2012].
- [20] J. Keesling, "The condition number for a matrix," [Online]. Available: <http://www.math.ufl.edu/~kees/ConditionNumber.pdf>. [May 15, 2012].
- [21] P. Gagnon, private communication, May 2012.
- [22] M. B. Porter, *The BELLHOP Manual and User's Guide*, Preliminary Draft, Heat, Light, and Sound Research, Inc., La Jolla, CA, 2011.
- [23] C. Veness, "Movable type scripts: Calculate distance, bearing and more between latitude/longitude points," 2012. [Online]. Available: www.movable-type.co.uk/scripts/latlong.html. [April 20, 2012].
- [24] L. Ljung, *System Identification Toolbox User's Guide: Algorithms for Recursive Estimation*, R2012a ed., The MathWorks, Inc., Natick, MA, 2011, pp. 6-13.
- [25] A. Shareef and Y. Zhu, "Localization using extended Kalman filters in wireless sensor networks," in *Kalman Filter Recent Advances and Applications*, InTech, 2009, pp. 297–320.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. RDML Jerry Ellis
Naval Postgraduate School
Monterey, California
4. RDML Rick Williams
Naval Postgraduate School
Monterey, California
5. Prof. Joseph Rice
Naval Postgraduate School
Monterey, California
6. Prof. Grace Clark
Naval Postgraduate School
Monterey, California
7. Prof. Clyde Scandrett
Naval Postgraduate School
Monterey, California
8. CAPT Jeff Kline
Naval Postgraduate School
Monterey, California
9. Prof. Tim Chung
Naval Postgraduate School
Monterey, California
10. CAPT Carol O'Neal
Naval Postgraduate School
Monterey, California
11. Ms. Lisa Trawick
Naval Postgraduate School
Monterey, California

12. Dr. Bob Headrick
Office of Naval Research
Arlington, Virginia
13. Dr. Dave Johnson
Office of Naval Research
Arlington, Virginia
14. Mr. Dave Everhart
Naval Surface Warfare Center
Panama City, Florida
15. Ms. Rita Painter
SPAWAR Systems Center Pacific
San Diego, California
16. Mr. Chris Fletcher
SPAWAR Systems Center Pacific
San Diego, California
17. Mr. Bob Creber
SPAWAR Systems Center Pacific
San Diego, California
18. Mr. Jerry Dejaco
SPAWAR Systems Center Pacific
San Diego, California
19. Dr. Paul Gendron
SPAWAR Systems Center Pacific
San Diego, California
20. Mr. Dale Green
Teledyne Benthos
North Falmouth, Massachusetts
21. Mr. Ken Scussel
Teledyne Benthos
North Falmouth, Massachusetts
22. Mr. Mike Coryer
Teledyne Benthos
North Falmouth, Massachusetts

23. Mr. Rob Pinelli
Teledyne Benthos
North Falmouth, Massachusetts
24. LT Ryan Hilger
Naval Postgraduate School
Monterey, California
25. Mr. Steven Jacobs
Naval Postgraduate School
Monterey, California