

One-Way Travel-Time Inverted Ultra-Short Baseline Localization for Low-Cost Autonomous Underwater Vehicles

Nicholas R. Rypkema¹, Erin M. Fischell² and Henrik Schmidt²

Abstract—This paper presents an acoustic localization system for small and low-cost autonomous underwater vehicles (AUVs). Accurate and robust localization for low-cost AUVs would lower the barrier toward multi-AUV research in river and ocean environments. However, these AUVs introduce size, power, and cost constraints that prevent the use of conventional AUV sensors and acoustic positioning systems, adding great difficulty to the problem of underwater localization. Our system uses a single acoustic transmitter placed at a reference point and is acoustically passive on the AUV, reducing cost and power use, and enabling multi-AUV localization. The AUV has an ultra-short baseline (USBL) receiver array that uses one-way travel-time (OWTT) and phased-array beamforming to calculate range, azimuth, and inclination to the transmitter, providing an instantaneous estimate of the vehicle location. This estimate is fed to a particle filter and graph-based smoothing algorithm to generate a consistent AUV trajectory. We describe the complete processing pipeline of our system, and present results based on experiments using a low-cost AUV. To the authors' knowledge, this work constitutes the first practical demonstration of the feasibility of OWTT inverted USBL navigation for AUVs.

I. INTRODUCTION

Accurate localization for a robotic vehicle is often essential for the purposes of path planning or geo-referencing of scientific measurements. For autonomous underwater vehicles (AUVs), accurate and reliable navigation in unstructured underwater environments is a major challenge: Global Positioning System (GPS) and radio frequency signals do not work underwater, and communication is unreliable and limited in bandwidth. Most commercial AUVs rely on a Doppler velocity log (DVL)-aided inertial navigation system [7] to navigate between periodic GPS surface fixes to limit unbounded error growth. In this work, we are interested in localization for small, low-cost AUVs. Size, power, and cost constraints limit the use of typical sensors and techniques; in particular, dead reckoning can only be performed with a low-cost MEMS inertial measurement unit (IMU) and magnetic compass, and without using an expensive and bulky DVL.

Besides inertial navigation, other approaches generally fall into two categories: acoustic or geophysical positioning [1][2]. Geophysical techniques include terrain relative [8], visually-augmented [9], and sonar-aided navigation [10].

Thanks to Craig Lawrence of DARPA for donation of the SandShark AUV, and Bluefin Robotics. Additional thanks to MIT LAMSS and Sailing Pavilion Personnel for experiment support. This work was funded by ONR under award number N00014-16-1-2081, Lincoln Laboratory under award number PO 7000339130, and Battelle under award number PO 472523.

¹N. R. Rypkema is with the Electrical Engineering and Computer Science Department, MIT, Cambridge, MA 02139, USA. rypkema@mit.edu

²E. M. Fischell and H. Schmidt are with the Mechanical Engineering Department, MIT, Cambridge, MA 02139, USA. {emf43, henrik}@mit.edu

Although terrain relative and visually-augmented navigation can be performed with inexpensive, low-power sensors (altimeter and camera respectively), both have drawbacks; the former requires prior maps of the seafloor, which often lacks features and is vulnerable to environmental change, and the latter can only be performed in clear water, or using high-power lighting. Sonar-aided navigation requires costly sidescan [12] or imaging [11] sonars.

On the other hand, environmental conditions have comparatively less effect on acoustic positioning, and localization is possible with the use of cheap hydrophones. Existing systems typically use two-way travel-time (TWTT) between multiple beacons and an AUV to triangulate an estimate of vehicle position that has an accuracy that does not degrade with time. One limitation of these so-called long or short baseline (LBL/SBL) systems is the time and expense associated with setting up the network of beacons. Ultra short baseline (USBL) systems also exist, which can provide an estimate using a single beacon containing an acoustic transceiver array. The relative range and bearing from the beacon to the vehicle is determined using TWTT and phase differencing.

The problem with using the two-way travel-time is that it requires the AUV to have an active acoustic system. Such a system is undesirable because it uses significant power, decreasing runtime; it adds cost; and it is impractical for multi-AUV systems because each vehicle must transmit signal in a different time slot or frequency band.

The advent of small, low-cost, low-power chip scale atomic clocks has enabled the use of one-way travel-time (OWTT) ranging through clock synchronization of the beacon and AUV. Previous work by Eustice et al. [4] demonstrated range-only OWTT navigation with a single ship-mounted beacon in which OWTT-derived range is fused with dead reckoning via maximum likelihood estimation. Webster et al. [5] presented similar experimental results with range-only OWTT measurements using an extended Kalman filter (EKF). Although large advances have been made in this area, and such systems have been demonstrated in the field to have adequate performance under certain conditions, the main issue with this approach is that a single range-only measurement cannot provide an unambiguous estimate of vehicle position - multiple measurements must be made to reduce positional uncertainty to desired levels.

Our work overcomes this limitation with a technique for real-time on-board navigation correction for low-power, low-cost AUV systems that uses one-way travel-time and phased-array beamforming to calculate vehicle location. This one-way travel-time inverted ultra-short baseline (OWTT-iUSBL)

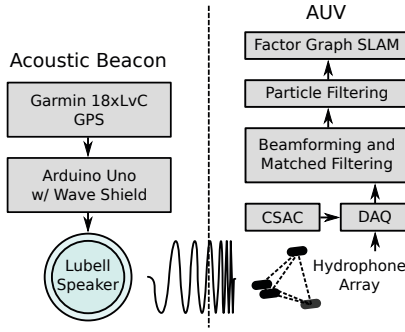


Fig. 1: OWTT-iUSBL system diagram. An acoustic source beacon transmits a signal with a known waveform every second triggered by GPS PPS; a CSAC on the AUV triggers synchronous recording of the signal using a hydrophone array and DAQ; processing of the recorded signal provides instantaneous range and angle-of-arrival measurements, which are fused with a motion model and further processed to estimate vehicle location.

system uses an acoustic beacon that is time-synchronized to a clock on the AUV. This is referred to as inverted USBL because the acoustic array resides on the AUV instead of the beacon. Acoustic data is collected using the array, and processed to find range, azimuth, and inclination to the acoustic source from the array. If the source is at a known location, instantaneous absolute vehicle position can be calculated using this information and AUV compass data.

This OWTT-iUSBL technique is acoustically passive on the AUV - the AUV outputs no acoustic signal. This decreases power draw and cost significantly. It also means that a single acoustic source can be used for localization of multiple AUVs, with no time or frequency sharing required. Jakuba et al. [6] recently analyzed the feasibility of OWTT-iUSBL using a simulated deep-profiling underwater glider, providing an excellent analysis of the advantages of this approach.

To our knowledge, this work presents the first practical demonstration of the feasibility of OWTT-iUSBL navigation for AUVs. This paper first describes the one-way travel-time inverted USBL system, including the required hardware and processing. Results based on deployment of the system on a Bluefin SandShark AUV are presented, along with conclusions and suggestions for future work.

II. ONE-WAY TRAVEL-TIME iUSBL

This section describes the OWTT-iUSBL system, diagrammatically illustrated in Fig. 1. An acoustic beacon emitting a periodic signal with a known waveform is placed at a known location. The AUV is time-synchronized with the beacon and captures the signal using a tetrahedral hydrophone array. OWTT is found using matched filtering and scaled by speed of sound in water to obtain range, and phased-array beamforming processes the captured signal to obtain azimuth and inclination to the beacon. A particle filter fuses these measurements with a motion model of the vehicle. Finally, a graph-based smoothing algorithm uses the particle filter output to generate a more consistent trajectory estimate.

A. System Hardware

The OWTT-iUSBL system hardware consists of time synchronized acoustic source and receiver systems, both built

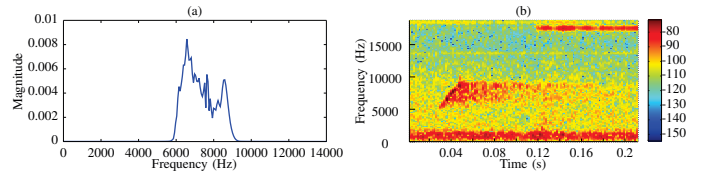


Fig. 2: (a) Magnitude frequency response of the up-chirp signal $h[n]$. (b) In-water spectrogram of up-chirp received and recorded by the AUV.

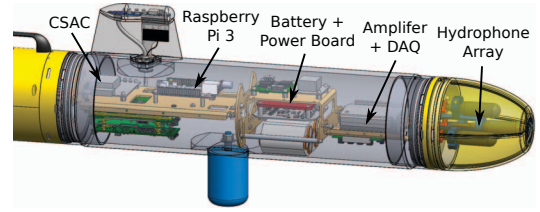


Fig. 3: CAD model of the Bluefin SandShark AUV payload, labeling the main components of the acoustic receiver system. (c/o Bluefin Robotics)

with commercial off-the-shelf components.

1) *Acoustic Beacon*: The acoustic source system includes an Arduino Uno microcontroller with a Wave Shield for audio. The rising edge of the pulse-per-second (PPS) signal from a Garmin 18xLvC GPS unit is used to trigger playback of a pre-recorded 20 ms, 7–9 kHz linear up-chirp signal (Fig. 2) output by the Wave Shield. Oscilloscope calibration and a hard-coded delay cycle ensured that the delay between the PPS rising edge and the onset of the audio signal was consistent with < 2 ms of jitter. A Lubell UW30 underwater speaker is used to play the audio signal into the water. The resulting beacon periodically outputs the up-chirp signal at the start of every GPS second.

2) *Autonomous Underwater Vehicle*: Our acoustic receiver system is implemented as a payload on a Bluefin SandShark AUV. The SandShark platform is equipped with a MEMS IMU with magnetometer, depth pressure sensor, GPS unit, propeller, and control fins. Navigation data, including vehicle attitude and speed, is pre-filtered by the vehicle from IMU and prop RPM information. The payload uses this filtered data to command vehicle depth, heading, and speed.

The payload (Fig. 3) consists of a nose-mounted tetrahedral hydrophone array with 4.5 cm spacing, a data acquisition module (DAQ), a Raspberry Pi 3 computer, and a SA.45 chip scale atomic clock (CSAC). The CSAC is synchronized to the GPS PPS signal before deployment, and maintains time-synchronization while the vehicle is submerged. The CSAC signal rising edge is used to trigger DAQ recordings of the hydrophone array. 8000 samples are collected each second starting at the clock's rising edge at a rate of 37.5 kS/s, effectively recording in sync with the firing of the beacon.

Systemic delay in the source/receiver system was characterized by observing the delay in reception of the transmitted up-chirp with the hydrophone array placed next to the acoustic source. This delay is subsequently accounted for during the calculation of estimated range.

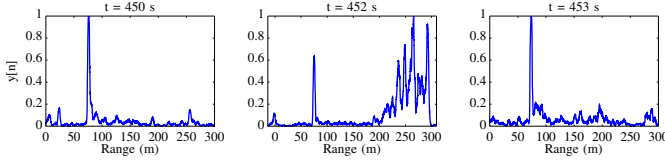


Fig. 4: Three sequential matched filtering outputs. In each case the true range is approximately 75 m, but at $t = 452$ the maximum response is false and at 265 m. Particle filtering is able to track the true range.

B. Matched Filtering

Matched filtering is performed with measurements from each of the four hydrophones to get a range estimate signal. This is in essence a convolution between hydrophone i 's received signal, $x_i[n]$, and a replica of the up-chirp $h[n]$:

$$y_i[n] = \sum_{k=0}^{N-1} h[n-k]x_i[k] \quad (1)$$

The output $y_i[n]$ reaches a maximum at the sample number at which $x_i[n]$ most closely resembles the replica $h[n]$. The standard deviation of this sample number across the four elements can be used as a measure of the validity of the array measurement:

$$\sigma = \sqrt{\frac{1}{3} \sum_{i=1}^4 (\arg\max_n(y_i[n]) - \frac{1}{4} \sum_{i=1}^4 \arg\max_n(y_i[n]))^2} \quad (2)$$

When $\sigma < 5$ we deem the measurement valid. Invalid measurements occur when the beacon signal is not received by all elements, which occurs primarily due to self-occlusion - the body of the AUV obstructs the transmitted up-chirp. To get the range estimate signal we combine the matched filter output from all elements and smooth using moving average:

$$y'[n] = \sum_{i,j=1}^4 y_i[n]y_j[n] \quad i \neq j \quad (3)$$

$$y[n] = \frac{1}{N+1} \sum_{i=-N/2}^{N/2} y'[n+i] \quad N \text{ even} \quad (4)$$

Finally, sample numbers (n) are converted to ranges by subtracting the characterized systemic delay and scaling by the ratio of approximate sound velocity in freshwater over sampling rate ($r = \frac{c}{F_s} n = \frac{1480}{37500} n$). This range estimate signal $y[n]$ is normalized and passed on to the particle filter. Fig. 4 shows typical outputs of our matched filtering process.

C. Phased-Array Beamforming

Phased-array beamforming [3] is also performed using the raw measurements, giving an azimuth-inclination heatmap estimating the angle-of-arrival. Assuming a planar incident sound wave and isotropic hydrophone response, beamforming is done by iterating through various azimuth-inclination combinations (look-angles), using the array geometry to apply time delays (phase shifts) to the received signals, and summing these time-delayed signals. The look-angle with

the maximum response is the likeliest angle-of-arrival of the incident wave, which occurs when the hydrophone signals are in phase and add constructively.

The time delay τ_i of a plane wave arriving from a specific look-angle at each element can be calculated as:

$$\tau_i = -\frac{\vec{u}^T \vec{p}_i}{c} \quad \vec{u} = \begin{bmatrix} \sin(\theta) \cos(\phi) \\ \sin(\theta) \sin(\phi) \\ \cos(\theta) \end{bmatrix} \quad (5)$$

where \vec{p}_i is the position of hydrophone i , ϕ and θ are the look-angle azimuth and inclination, and c is the speed of sound in water. This time delay is constant for a given look-angle, but corresponds to phase shifts in the frequency domain that increase with frequency:

$$f_i[n - \tau_i] \xrightarrow{DFT/IDFT} F_i[\omega] \cdot e^{-j\omega\tau_i} \quad (6)$$

where f_i is the plane wave signal incident on hydrophone i , and $\vec{\omega}$ contains the range of frequencies output by the n-point DFT. Therefore, to reconstruct the plane wave signal and nullify the geometrically-induced time delays, we simply apply opposite phase shifts to the received signals and average over all elements. In addition, before averaging, our beamformer applies a matched filter with the up-chirp replica $h[n]$ to enhance signal detection. The resulting output is:

$$Y_i[\omega] = DFT\left(\sum_{k=0}^{N-1} h[n-k]x_i[k]\right) \quad (7)$$

$$g[n] = \frac{1}{4} \sum_{i=1}^4 IDFT(Y_i[\omega] \cdot e^{j\omega\tau_i}) \quad (8)$$

where $x_i[n]$ is the signal received by hydrophone i and $g[n]$ is the beamformer output for the specific look-angle. Equation 7 performs matched filtering and equation 8 performs phase shifting and averaging. These equations will produce a maximal response when the look-angle is equal to the true angle-of-arrival of the incident wave. We search for this angle-of-arrival by iterating through a number of look-angles, producing an azimuth-inclination heatmap, $g[\phi, \theta]$:

$$g[\phi, \theta] = \max_n(g[n; \tau_i(\phi, \theta)]) \quad (9)$$

The most likely angle-of-arrival is then:

$$(\phi^*, \theta^*) = \arg\max_{\phi, \theta} (g[\phi, \theta]) \quad (10)$$

This heatmap is passed on to the particle filter when $265^\circ \leq \phi^* \leq 95^\circ$ and $45^\circ \leq \theta^* \leq 135^\circ$ to prevent error caused by self-occlusion. Fig. 5 illustrates a typical heatmap.

Real-time on-board beamforming is achieved despite its computational complexity by using two techniques: firstly, the phase shifts associated with each look-angle are precomputed and stored; secondly, instead of the full DFT we use the Chirp Z-transform [13] to limit frequency domain operations to a range of interest (6–10 kHz) - this reduces the number of frequency domain points without loss of resolution.

D. Particle Filter Localization

Matched filtering and beamforming only provide instantaneous and noisy estimates of range and angle-of-arrival respectively. In addition, underwater acoustic propagation frequently exhibits undesirable properties such as multipath and reflections, resulting in outliers and measurement distributions that are non-Gaussian. Consider the three valid matched filtering outputs in Fig. 4 - in this sequence the true range between the beacon and AUV is approximately 75 m; however, the middle measurement has a false maximum at 265 m. This suggests that simply using *argmax* on range and angle-of-arrival measurements could result in significant error in localization. The non-Gaussian nature of these measurements, and the desire to fuse measurements with AUV attitude information and motion model, motivate the use of a particle filter for localization rather than a regular EKF.

Our particle filter makes use of three main coordinate frames: the Forward-Port-Above body-fixed frame (*bff*), in which acoustic angle-of-arrival measurements are made (Eqs. 5–9); the vehicle-carried East-North-Up frame (*vcf*), whose origin is fixed to the center of gravity of the AUV; and the local-level East-North-Up frame (*llf*), whose origin we define to be the beacon position and within which the vehicle navigates. At time instant t , our filter models the azimuth-inclination to the beacon using a set of N particles and associated weights w_i^x which reside on the unit ball in the vehicle-carried frame. A second set of N particles and weights w_i^r residing on a 0–300 m range-line (*rl*) are maintained to estimate range to the beacon using matched filtering measurements (Eqs. 1, 3, 4). Two particle sets are used due to the independent nature of range and azimuth-inclination measurements. The state vectors of the particles in each set are given by:

$$\vec{S}_i^x(t) = [x_i^{vcf}(t)y_i^{vcf}(t)z_i^{vcf}(t)]^T \quad \vec{S}_i^r(t) = [r_i^{rl}(t)] \quad (11)$$

Typically, the particle filter is initialized using GPS measurements when the AUV is on the surface awaiting deployment. AUV GPS position is transformed into the local-level frame by subtracting beacon location, and particles are initialized in this frame centered around GPS position. These particles are then transformed to the vehicle-carried frame and range-line for state vector storage:

$$n_i \sim \mathcal{N}(0, \sigma_{\text{GPS}}^2) \quad w_i^x(0) = w_i^r(0) = 1/N \quad (12)$$

$$r_i^{rl}(0) = \sqrt{(x_{\text{GPS}}(0) + n_i)^2 + (y_{\text{GPS}}(0) + n_i)^2 + (n_i)^2} \quad (13)$$

$$\begin{bmatrix} x_i^{vcf}(0) \\ y_i^{vcf}(0) \\ z_i^{vcf}(0) \end{bmatrix} = \begin{bmatrix} (-x_{\text{GPS}}(0) + n_i)/r_i^{rl}(0) \\ (-y_{\text{GPS}}(0) + n_i)/r_i^{rl}(0) \\ n_i/r_i^{rl}(0) \end{bmatrix} \quad (14)$$

where $(x_{\text{GPS}}, y_{\text{GPS}})$ is the local-level frame GPS position and σ_{GPS} is the standard deviation of GPS measurement noise. The transform from the local-level to the vehicle-carried frame and range-line is denoted by R_{llf}^{vcf} . This transform is performed by negating and normalizing the local-level

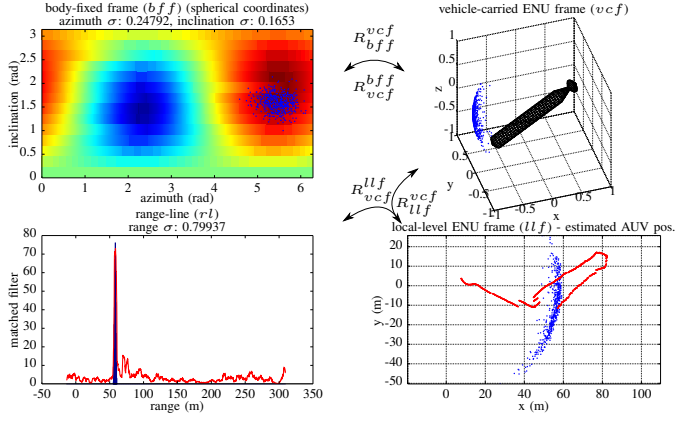


Fig. 5: *Top Left*: typical heatmap resulting from phased-array beamforming, as well as azimuth-inclination particles in the body-fixed frame with spherical coordinates (blue). *Bottom Left*: typical range estimate signal (red) resulting from matched filtering, as well as histogram of range particles on the range-line (blue). *Top Right*: azimuth-inclination particles in the vehicle-carried frame (blue), and current vehicle attitude. *Bottom Right*: combined azimuth-inclination and range particles in the local-level frame (blue), as well as particle filter estimated AUV trajectory (red).

particles to get vehicle-carried particles (Eq. 14), and calculating particle magnitudes to get range-line particles (Eq. 13). Particles are re-initialized whenever a GPS fix is received.

1) *Prediction*: In the prediction step the two particle sets are sorted in ascending order according to their weights. The particles are then transformed to the local-level frame by combining both sets (essentially by element-wise multiplication) - a transform denoted by R_{vcf}^{llf} (Eq. 16). Vehicle pitch (θ^{vcf}) and yaw (ϕ^{vcf}) as well as speed estimated from prop RPM (v) are then used to propagate the combined particles using our motion model:

$$\text{let } \vec{T}_i^x(t) = [x_i^{llf}(t) \ y_i^{llf}(t) \ z_i^{llf}(t)]^T \quad (15)$$

$$\vec{T}_i^x(t - \Delta t) = -\vec{S}_i^x(t - \Delta t) \cdot \vec{S}_i^r(t - \Delta t) \quad (16)$$

$$\vec{T}_i^x(t) = \vec{T}_i^x(t - \Delta t) + \begin{bmatrix} \Delta t v \sin(\theta^{vcf}) \cos(\phi^{vcf}) \\ \Delta t v \sin(\theta^{vcf}) \sin(\phi^{vcf}) \\ \Delta t v \cos(\theta^{vcf}) \end{bmatrix} \quad (17)$$

Finally, the particles are transformed and separated back into the vehicle-carried frame and range-line, and Gaussian noise added to each particle:

$$(\vec{S}_i^x(t), \vec{S}_i^r(t)) = R_{llf}^{vcf}(\vec{T}_i^x(t)) + \vec{n} \sim \mathcal{N}(0, \sigma_{r,x}^2) \quad (18)$$

Storing the particles in the vehicle-carried rather than the body-fixed frame allows us to exclude attitude propagation in the update step. This reduces computation, since attitude updates occur much faster than acoustic measurements.

2) *Update*: Whenever a valid acoustic azimuth-inclination heatmap or range estimate signal is received, weights are updated and the particles resampled. For the range particle set, the update step is simple: particle weights are multiplied by the value of the range estimate signal corresponding to their associated ranges, and resampling and weight normalization is done using systematic resampling. For the azimuth-inclination particle set we first transform the particles into the

body-fixed frame using vehicle pitch (θ^{vcf}), roll (ψ^{vcf}) and yaw (ϕ^{vcf}) - a transformation denoted as R_{vcf}^{bff} :

$$\text{let } \vec{U}_i^\Phi(t) = [\phi_i^{bff}(t) \ \theta_i^{bff}(t)]^T \quad (19)$$

$$\vec{U}_i^\Phi(t) = R_\Phi((R_z(\phi^{vcf})R_y(\theta^{vcf})R_x(\psi^{vcf}))^T \vec{S}_i^x(t)) \quad (20)$$

where R_Φ is the standard Cartesian to spherical transform, and R_z , R_y , and R_x are the standard rotation matrices. In the body-fixed frame the azimuth-inclination particles are represented using spherical coordinates; their weights are multiplied with the corresponding azimuth-inclination heatmap values, and resampling and weight normalization is performed using systematic resampling. Finally the particles are transformed back into the vehicle-carried frame using the inverse rotation matrices and the standard spherical to Cartesian transform - we denote this process as R_{bff}^{vcf} .

3) *Estimation*: Estimation is performed by calculating the weighted means of both the range and azimuth-elevation particle sets in the body-fixed frame. Transformation of the range and azimuth-elevation means into the local-level frame provides an estimate of the AUV position in the local-level frame. In addition, the means transformed into vehicle-carried frame are used during factor graph smoothing.

The output of the particle filter with 500 particles, along with visualizations of the particles in each coordinate frame can be seen in Fig. 5. This figure also illustrates typical outputs of the matched filtering and beamforming processes.

E. Factor Graph Smoothing

Although the particle filter provides an estimate of the vehicle's location, it does so by recursively marginalizing out all previous measurements, resulting in a trajectory that often contains discontinuities. A factor graph smoothing algorithm can improve this by utilizing all particle filter measurements to optimize over the full AUV trajectory. This approach results in a smoother and more consistent trajectory, while still retaining the robustness against acoustic outliers provided by the particle filter.

In our approach we estimate the vehicle pose, $\vec{x}_i = [x_i, y_i, \phi_i]^T$, in the local-level frame using a factor graph smoothing framework to represent the collection of poses over the entire trajectory. Each node \vec{x}_i in the graph corresponds to the pose estimate at time i , and is linked to preceding and subsequent pose nodes by odometry constraints calculated using our motion model, as depicted in Fig. 6. When a valid acoustic measurement appears, the pose node is linked to a beacon node $\vec{b} = [x_b, y_b]^T$ by either an azimuth-range or range-only constraint outputted by our particle filter. In addition, the initial pose is constrained by a prior, which represents surface GPS measurements in the local-level frame. The beacon node is also constrained by a prior, $[0, 0]^T$, as it represents the origin of the local-level frame. We use the GTSAM library [14], and specifically the iSAM2 algorithm [15] to incrementally perform maximum a-posteriori inference over the factor graph as it is constructed. Motion and measurement noise are independent and assumed

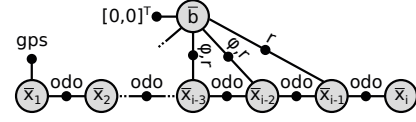


Fig. 6: Illustration of the factor graph - \vec{x}_i are vehicle poses connected by motion model odometry (*odo*), and \vec{b} is the beacon pose connected to vehicle poses by either azimuth-range (ϕ, r) or range-only (r) measurements; the initial vehicle pose has a prior factor from GPS measurements (*gps*), and the beacon has a prior factor of $[0, 0]^T$ placing it at the origin.

to be Gaussian, with the standard deviation of measurement noise estimated directly using the particles from our filter.

III. EXPERIMENTS AND RESULTS

Experiments were performed using our SandShark AUV with acoustic payload (section II-A.2) on a portion of the Charles River by the MIT Sailing Pavilion. Our acoustic beacon (section II-A.1) was submerged to about 0.5 m depth and fastened to the pavilion dock, at a known GPS position. The AUV was pre-programmed with a mission where it was instructed to travel back-and-forth along the dock for 170 m at 2 m depth and a speed of 1.4 m/s. The mission duration was set to 1200 s, and the AUV was instructed to surface for a 120 s GPS fix whenever it was at the end of a 170 m run and the time since the last fix was greater than 150 s.

Both matched filtering and phased-array beamforming were performed on the Raspberry Pi 3 in real-time at approximately 1.25 Hz with 4050 look-angles (15 inclination and 270 azimuth equally-spaced angles). This data was recorded by the payload along with pre-filtered navigation data from the vehicle, which was received by the payload at a rate of about 10 Hz. The payload and AUV system clocks were synchronized using an NTP server running on the payload.

Particle filtering and factor graph smoothing were performed offline. 500 particles were used for both the azimuth-inclination and range set. iSAM2 was used to build and solve the factor graph with vehicle poses added at a rate of 5 Hz and using ranges and azimuths output by the particle filter. A new graph was initialized each time the AUV received a GPS fix, allowing us to monitor the difference in estimated and true position during the underwater to surface transition.

The AUV was deployed for two runs, with the vehicle surfacing for three GPS fixes during the first and four GPS fixes during the second. For the two runs we perform a qualitative comparison between the trajectories resulting from vehicle dead reckoning, particle filtering, and factor graph smoothing. We also use a simple metric to assess the inter-GPS-fix navigation performance of the three methods over both runs: during the underwater to surface transition a discontinuity in position occurs when the AUV gets a GPS fix, which is caused by localization error during underwater navigation; smaller jumps would indicate better performance. Unfortunately, this metric is subject to GPS positional error.

The top row of fig. 7 shows plots of azimuth and range from the AUV to the beacon, as estimated from dead reckoning (blue) and our particle filter (red), along with *argmax* measurements from matched filtering and beamforming in

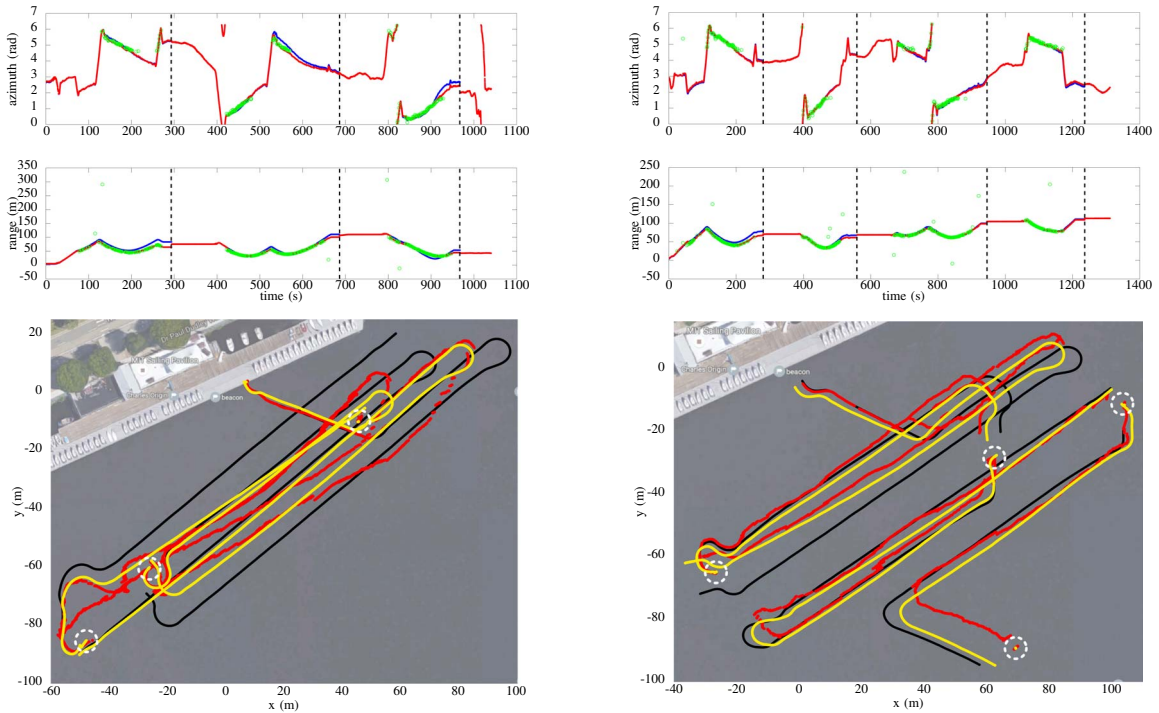


Fig. 7: Run 1 (left) and Run 2 (right) results. *Top*: azimuth and range estimate to beacon in body-fixed frame - dead reckoned (blue), particle filter output (red), and raw acoustic measurement maxima (green), with vertical lines indicating GPS fix times. *Bottom*: AUV trajectory estimates in local-level frame - dead reckoned (black), particle filter output (red), and iSAM2 factor graph smoothing (yellow), with dashed white circles indicating GPS fix locations.

	Run 1 GPS Jumps (m)			Run 2 GPS Jumps (m)				Mean (m)	Std. Dev. (m)
Dead Reckoning	8.6	4.8	33.0	8.3	8.0	6.4	12.7	11.7	9.7
Particle Filter	2.4	11.8	9.4	6.0	8.9	6.1	4.3	7.0	3.2
Factor Graph	2.9	5.4	5.3	10.4	5.8	6.4	8.6	6.4	2.4

TABLE I: Sizes of discontinuity in position for the three methods when AUV receives a GPS fix during the underwater to surface transition, along with mean and standard deviation - smaller jumps indicate better underwater navigation performance.

green. We see that the particle filter successfully fuses the observed acoustic measurements with the dead reckoned motion model, pulling the estimate towards observations. Our validity checks are apparent when looking at the measurements - no azimuth measurements exist between 95° (1.66 rad) and 265° (4.62 rad), which are invalid due to self-occlusion. Even though outliers exist in the range measurements, these are filtered out by the particle filter.

The lower plots of Fig. 7 display the resulting trajectories for the three methods for both runs, with dead reckoning in black, particle filtering in red, and iSAM2 factor graph smoothing in yellow. The three GPS fixes for the first run occur at $(-26, -61)$ (294 s), $(-50, -88)$ (686 s), and $(45, -10)$ (967 s). For the second run, the four fixes occur at $(-27, -65)$ (280 s), $(62, -28)$ (559 s), $(104, -11)$ (945 s), and $(63, -88)$ (1236 s). The positional jumps that occur during these fixes are listed in table I, along with the average jump distance and standard deviation for the three methods. Qualitative examination of the trajectories indicate that the dead reckoned estimates are the least self-consistent, with large discontinuities when the vehicle surfaces for a fix. The particle filter trajectories are better in this respect, but they suffer from non-continuity caused by incorporation of latest observations in the filter's recursive estimate. The trajectories

resulting from iSAM2 on the other hand are both the most self-consistent, and maintain a smooth, continuous trajectory between GPS fixes; this is a result of optimizing over the entire vehicle history, incorporating all acoustic measurements. These observations are supported by the jump distances in table I - the iSAM2 approach has both the smallest average discontinuity, and the lowest standard deviation.

Besides the inherent positional uncertainty associated with GPS measurements, possible sources for the observed differences between GPS and our localization approach include motion due to river currents, which cannot be accounted for; inaccurate characterization of the systemic delay in the source/receiver system; inaccurate measurement of the acoustic array element positions; as well as jitter in the onset of the acoustic beacon signal - a maximal 2 ms delay in the onset of the beacon transmission at a sound speed of 1480 m/s corresponds to a 2.96 m error, which is on the same order of magnitude as the differences observed.

IV. CONCLUSION AND FUTURE WORK

This work has presented a system to localize a small, low-cost AUV using a single acoustic source. It uses OWTT of a known signal emitted by the source to estimate range, and an AUV mounted array to estimate angle to the source using

matched filtering and beamforming. These measurements are fused with an AUV motion model using a particle filter, then smoothed with a factor graph-based algorithm to provide a good-performance AUV localization estimate, without the use of conventional sensors such as a DVL or high-grade INS. It is acoustically passive on the AUV, reducing power use and cost, and enabling multiple AUVs to localize using a single beacon. Future work includes installing a second commercial LBL acoustic localization system for ground-truth verification; simultaneously deploying two source beacons with different chirp signals to remove vehicle dependence on magnetometer for yaw, which limits vehicle deployment to areas devoid of large magnetic anomalies; and implementing online versions of particle filtering and factor graph smoothing to perform closed loop navigation with our factor graph estimate.

REFERENCES

- [1] L. Paull, et al., *AUV Navigation and Localization - A Review*. IEEE Journal of Oceanic Engineering, vol. 39, no. 1, pp. 131-149, 2014.
- [2] J.C. Kinsey, R.M. Eustice, L.L. Whitcomb, *A Survey of Underwater Vehicle Navigation: Recent Advances and New Challenges*. IFAC Conference of Manoeuvring and Control of Marine Craft, vol. 88, pp. 1-12, 2006.
- [3] H.L. Van Trees, *Optimum Array Processing*. John Wiley & Sons, 2002.
- [4] R.M. Eustice, et al., *Experimental Results in Synchronous-Clock One-Way-Travel-Time Acoustic Navigation for Autonomous Underwater Vehicles*. IEEE International Conference on Robotics and Automation, pp. 4257-4264, 2007.
- [5] S.E. Webster, et al., *Advances in Single-Beacon One-Way-Travel-Time Acoustic Navigation for Underwater Vehicles*. The International Journal of Robotics Research, vol. 31, pp. 935-950, 2012.
- [6] M.V. Jakuba, et al., *Feasibility of Low-Power One-Way Travel-Time Inverted Ultra-Short Baseline Navigation*. IEEE OCEANS 2015, pp. 1-10, 2015.
- [7] B. Jalving, et al., *A Toolbox of Aiding Techniques for the HUGIN AUV Integrated Inertial Navigation System*. IEEE OCEANS 2003, vol. 2, pp. 1146-1153, 2003.
- [8] S. Carreno, et al., *A Survey on Terrain Based Navigation for AUVs*. IEEE OCEANS 2010, pp. 1-7, 2010.
- [9] R.M. Eustice, O. Pizarro, H. Singh, *Visually Augmented Navigation for Autonomous Underwater Vehicles*. IEEE Journal of Oceanic Engineering, vol. 33, no. 2, pp. 103-122, 2008.
- [10] I.T. Ruiz, S. de Raucourt, Y. Petillot, *Concurrent Mapping and Localization Using Sidescan Sonar*. IEEE Journal of Oceanic Engineering, vol. 29, no. 2, pp. 442-456, 2004.
- [11] D. Ribas, et al., *SLAM using an Imaging Sonar for Partially Structured Underwater Environments*. IEEE International Conference on Intelligent Robots and Systems, pp. 5040-5045, 2006.
- [12] M. Fallon, et al., *Efficient AUV Navigation Fusing Acoustic Ranging and Side-Scan Sonar*. IEEE International Conference on Robotics and Automation, pp. 2398-2405, 2011.
- [13] L.R. Rabiner, B. Gold, *Theory and Application of Digital Signal Processing*. Prentice-Hall, pp. 393-399, 1975.
- [14] F. Dellaert, *Factor Graphs and GTSAM: a Hands-On Introduction*. Technical Report GT-RIM-CP&R-2012-002, 2012.
- [15] M. Kaess, et al., *iSAM2: Incremental Smoothing and Mapping Using the Bayes Tree*. International Journal of Robotics Research, vol. 31, no. 2, pp. 216-235, 2012.