

# Toward Informative Planning for Cooperative Underwater Localization

Jeffrey M. Walls and Ryan M. Eustice

**Abstract**—This paper reports on an algorithm for planning a practical trajectory for a surface vehicle that provides range measurements to an autonomous underwater vehicle (AUV). We consider server-client cooperative localization in which a server vehicle provides relative range constraints to minimize the uncertainty of a client vehicle. Our approach assumes the nominal client mission plan is available and draws potential server trajectories from a set of parameterized trajectory classes. We provide a comparative evaluation over several simulations, for both a single client and multiple clients, demonstrating that our algorithm computes operationally practical server paths and performs well relative to existing planning frameworks.

## I. INTRODUCTION

Ocean science has greatly benefited from the quantity and breadth of data collected by autonomous underwater vehicles (AUVs) [1]. Bounded-error navigation methods are required to safely operate the vehicles and georeference this data. Range-only underwater cooperative localization promises a flexible and infrastructure-free alternative to static beacon networks (e.g., long-baseline (LBL)) for bounded-error precision navigation at area scales currently not achievable.

In a synchronous-clock network, one-way-travel-time (OWTT) observations measure the range between a server time-of-launch (TOL) pose and a client time-of-arrival (TOA) pose. Previous work in underwater cooperative localization has mainly focused on distributed position estimation across a bandwidth limited communication channel [2], [3], [4], [5], [6]. The quality of the navigation solution strongly depends upon the relative vehicle trajectories. Fig. 1 illustrates the dependence of client localization quality for two simple server trajectories. Accordingly, we are interested in planning server trajectories that provide informative range measurements with which a set of client vehicles can localize.

In this paper we explore the cooperative localization planning problem through the lens of informative path planning. In general, the optimal planning problem can be expressed

$$\mathcal{P}^* = \arg \max_{\mathcal{P} \in \Psi} I(\mathcal{P}), \quad (1)$$

where  $\mathcal{P}$  is a trajectory in the set of candidate trajectories,  $\Psi$ , and  $I(\cdot)$  represents the utility or reward of a path. We will compute practical server trajectories that minimize the client’s position uncertainty given the client’s mission plan.

Planning consists of two coupled tasks: (i) generating the set of candidate trajectories  $\Psi$ , and (ii) finding the best path in  $\Psi$ . Various planning algorithms tackle these two

\*This work was supported in part by a grant from the National Science Foundation under award IIS-0746455, and in part by the Office of Naval Research under award N00014-12-1-0092.

J. Walls and R. Eustice are with the University of Michigan, Ann Arbor, Michigan 48109, USA {jmwalls, eustice}@umich.edu.

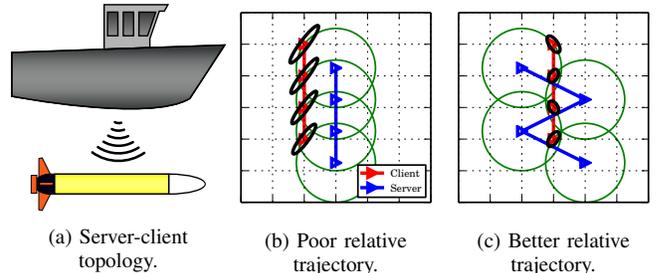


Fig. 1: Illustration of cooperative localization with different server trajectories. Range circle measurements are shown in green, client in red, and server in blue. Clearly (c) reduces the client navigation uncertainty better than (b), as evidenced by the size of the posterior covariance ellipses (shown in black).

tasks in a variety of ways including dynamic programming [7] and sampling-based paradigms [8]. The quality of the optimal path largely depends upon the candidate set  $\Psi$ . In this work, the candidate set is provided by classes of parameterized trajectories. While these paths do not provide concrete optimality guarantees, we show that a search over practical trajectories greatly reduces the search complexity and performs well relative to an upper bound on the objective function.

Specifically, the contributions of this work include:

- A mutual information objective function for a single server localizing multiple clients.
- A search procedure for planning useful server trajectories to localize a set of client AUVs.

We validate our algorithm in simulation over several trials. We also provide a comparison with previously reported planning algorithms for underwater cooperative localization.

## II. PRIOR WORK

### A. Informative path planning

Informative path planning represents a growing literature concerned with optimally planning paths in order to gather evidence about some uncertain variable. Prior work has generally employed sampling-based planners at their core.

Sampling-based motion planning algorithms such as the probabilistic road map (PRM) [9] and the rapidly-exploring random tree (RRT) [10] can quickly explore high-dimensional configuration spaces and offer probabilistic guarantees on completeness given start and goal states. The rapidly-exploring random graph (RRG) algorithm and its derivative RRT\* [11] provide both probabilistic completeness and asymptotic optimality. Essentially, their proof relies on showing that in the limit the RRG contains *all* possible paths through the environment.

Optimal sampling-based algorithms inevitably result in a combinatorial optimization problem—find the path through the graph that satisfies the constraints and maximizes (minimizes) an objective. Searching the set of possible paths through a graph is, in general, an NP-hard problem.

Robot exploration strategies represent a similar problem to our cooperative localization problem. In this scenario, robots seek paths that allow them to observe landmarks used for localization. Sim and Roy [12] presented an A-optimal exploration strategy relying on a breadth-first search over a grid. The belief road map (BRM) algorithm [13] similarly searched over a PRM for finding minimum uncertainty trajectories through a known environment.

Binney and Sukhatme [14] presented a branch and bound procedure for informative path planning. The rapidly-exploring random belief tree (RRBT) algorithm [15] leveraged the probabilistic guarantees of the RRG to plan minimum uncertainty paths to a goal that converge to the global optimal over the configuration space. At each iteration, the RRBT updates a tree over the RRG that contains all potentially optimal paths. The search space is pruned by specifying a partial ordering of nodes in the tree. Following each iteration, the tree contains the optimal path through the set of RRG vertices. The rapidly-exploring information gathering (RIG) algorithm [8] extended the RRBT to an information gathering objective.

### B. Planning for range-only localization

Control theoretic observability analysis has been applied to the range-only estimation problem [16], [17], [18], [4]. Observability, however, can only point to a few trivial server-client relative trajectories for which the client position cannot be estimated. The planning framework developed herein establishes trajectories that are observable, but also minimizes the uncertainty of the estimated client position.

Martinez and Bullo [19] explored two problems: (1) optimal range sensor placement for localizing a static target and (2) motion coordination for localizing a moving target. They define their objective as the determinant of the Fischer information matrix (minimizing the trace of the Cramer-Rao lower bound). The authors discovered that positioning the sensors around the target at equal angular intervals provided the optimal solution. For the dynamic target, the authors take a suboptimal (but arguably good) approach: they control mobile sensors to arrange themselves around the target as in the static case.

Surveying LBL networks involves determining unknown beacon locations given ranges to a topside ship with known position. Hunt et al. [20] evaluated the resulting beacon uncertainty for various ship survey configurations. In a sense, this evaluation constitutes optimal planning for static beacon localization. Jakuba et al. [21] recommend that the topside ship circle the beacons at a fixed radius (up to water depth). Olson et al. [22] briefly described a greedy exploration strategy to disambiguate LBL beacon position hypotheses when observed by a survey AUV.

For client AUV localization, Fallon et al. [4] suggested the server follow zig-zag or encirclement patterns around the client. Tan et al. [7] proposed a finite-horizon exhaustive search algorithm for planning server trajectories with a preplanned AUV client trajectory. They employ a dynamic programming framework over a discrete action set to minimize the sum of the trace of client covariance over client poses. Bahr et al. [23] similarly compute the single next best action (greedy approach) to localize a group of AUV clients optimizing the same trace-based objective function. The authors assume that the client trajectories are unknown and estimate their positions from reported acoustic broadcasts. German et al. [24] suggested a heuristic algorithm to localize a subsea client. Their algorithm is simple and adapts to the real-time location of the client. While their method requires knowing the client’s mission plan, they do not explicitly optimize an information objective or quantify the quality of the achievable solution.

Charrow et al. [25], [26] consider localizing an unknown mobile target using radio-based range-only observations by maximizing the mutual information between the target prior belief and expected range measurements. Most notably, the authors use a Monte Carlo based approximation to the mutual information instead of assuming a linear Gaussian noise model for range observations as is typical.

## III. RANGE-ONLY COOPERATIVE PLANNING

Client AUVs travel through the environment collecting myriad scientific and navigation observations. The client vehicles we consider here have a predetermined mission plan or survey. Our goal is to accurately localize the client vehicle along its nominal trajectory exploiting range-only observations to a server (Fig. 1).

We represent a client trajectory by a set of expected TOA poses, for example, a set of  $N$  client poses is denoted  $\mathbf{X}_c = [\mathbf{x}_{c_0}, \dots, \mathbf{x}_{c_{N-1}}]$  where each pose is the horizontal-plane vehicle position  $\mathbf{x}_{c_i} = [x_{c_i}, y_{c_i}]^\top$ . We assume that depth is well instrumented and range observations can be projected into the horizontal plane (Appendix I). We assume that the expected prior belief over this set of poses is available and can be represented by a Gaussian distribution

$$\begin{aligned} \mathbf{X}_c &\sim \mathcal{N}(\boldsymbol{\mu}_c, \Sigma_c) = \mathcal{N}^{-1}(\boldsymbol{\eta}_c, \Lambda_c) \\ \Lambda_c &= \Sigma_c^{-1} \\ \boldsymbol{\eta}_c &= \Lambda_c \boldsymbol{\mu}_c, \end{aligned}$$

where  $\boldsymbol{\mu}_c$  and  $\Sigma_c$  are the mean and covariance, and  $\boldsymbol{\eta}_c$  and  $\Lambda_c$  are the corresponding information vector and matrix, respectively. The prior belief can be constructed from expected odometry and GPS measurements, for example, given the preplanned client mission. Eustice et al. [27] discuss constructing information within a delayed-state information filter. Generally,  $\Lambda_c$  will be sparse, but we place no restrictions on the information matrix other than it be nonsingular (i.e., well-constrained).

### A. Single Client Range Measurement Update

We first demonstrate a single range update event between an uncertain server and client vehicle to derive a geometry independent measurement update rule for the client. Consider the server TOL pose,  $\mathbf{x}_{s\text{TOL}}$ , and client TOA pose,  $\mathbf{x}_{c\text{TOA}}$ . A range observation is modeled as the Euclidean distance perturbed by zero-mean Gaussian noise

$$\begin{aligned} z_r &= h_r(\mathbf{x}_{c\text{TOA}}, \mathbf{x}_{s\text{TOL}}) + w_r \\ &= \|\mathbf{x}_{c\text{TOA}} - \mathbf{x}_{s\text{TOL}}\|_2 + w_r, \end{aligned}$$

where  $w_r \sim \mathcal{N}(0, \sigma_r^2)$ . The observation model,  $h_r(\cdot)$ , is a nonlinear function over the client and server states. We assume that the observation model behaves linearly in a neighborhood around the current state estimate. The range-observation Jacobian for a 2D state  $\mathbf{x} = [x, y]^\top$  is given

$$\begin{aligned} \mathbf{J}_c &= \left. \frac{\partial h_r}{\partial \mathbf{x}_c} \right|_{\hat{\mathbf{x}}_c} = [\cos \theta \quad \sin \theta] \\ \mathbf{J}_s &= \left. \frac{\partial h_r}{\partial \mathbf{x}_s} \right|_{\hat{\mathbf{x}}_s} = [-\cos \theta \quad -\sin \theta], \end{aligned} \quad (2)$$

where  $\theta = \tan^{-1} \frac{y_c - y_s}{x_c - x_s}$  is the relative angle between the client and server.

Since the client and server distributions are initially independent, the prior joint distribution is  $p(\mathbf{x}_c, \mathbf{x}_s) = \mathcal{N}^{-1}(\boldsymbol{\eta}, \Lambda)$  where  $\boldsymbol{\eta} = [\boldsymbol{\eta}_c^\top, \boldsymbol{\eta}_s^\top]^\top$  and  $\Lambda = \text{diag}(\Lambda_c, \Lambda_s)$ . We can compute the information matrix corresponding to the joint posterior distribution,  $p(\mathbf{x}_c, \mathbf{x}_s | z_r) \sim \mathcal{N}^{-1}(\boldsymbol{\eta}', \Lambda')$ , as the additive update

$$\Lambda' = \Lambda + \frac{1}{\sigma_r^2} \mathbf{J}^\top \mathbf{J},$$

where  $\mathbf{J} = [\mathbf{J}_c, \mathbf{J}_s]$ . We obtain the posterior information over the client vehicle alone by marginalizing out the server pose

$$\begin{aligned} \Lambda'_c &= \Lambda_c + \frac{1}{\sigma_{\text{eq}}^2} \mathbf{J}_c^\top \mathbf{J}_c \\ \sigma_{\text{eq}}^2 &= \sigma_r^2 + \mathbf{J}_s \Lambda_s^{-1} \mathbf{J}_s^\top, \end{aligned}$$

where  $\sigma_{\text{eq}}^2$  is the equivalent variance accounting for both the range and prior server uncertainty. The above formulation provides the client update rule for ranges to an independent server pose. Note that when the server's prior covariance is isotropic  $\Sigma_s = \Lambda_s^{-1} = \sigma_s^2 \mathbf{I}$ , then the equivalent variance is independent of the relative vehicle geometry, i.e.

$$\sigma_{\text{eq}}^2 = \sigma_r^2 + \sigma_s^2. \quad (3)$$

Range measurement updates add information along the vector between the client and server vehicle. The eigenvector corresponding to the nonzero eigenvalue of the information update,  $\Lambda_r = \frac{1}{\sigma_{\text{eq}}^2} \mathbf{J}_c^\top \mathbf{J}_c$ , points along  $\theta$ . Intuitively, then, the server should position itself to add information along directions for which the client is most uncertain.

We now consider the full set of client TOA poses,  $\mathbf{X}_c$ . Let  $\mathbf{Z}_r(\mathcal{P}) = \{z_{r_0}, \dots, z_{r_{N-1}}\}$  represent the set of  $N$  range-only observations as a function of the server path  $\mathcal{P}$  containing TOL poses  $\mathbf{X}_s = [\mathbf{x}_{s_0}, \dots, \mathbf{x}_{s_{N-1}}]$ . We assume that the server positions are independent and have identical isotropic prior uncertainty. In general, server poses are *not* independent or

isotropic but can be so approximated with regular access to an absolute position reference, for example a topside support ship with GPS [28]. We can therefore consider each range observation independently with uncertainty independent of the relative vehicle geometry as in (3).

Since each range observation is independent and identically distributed, we can write the total range information as a sum of the information due to each observation,

$$\begin{aligned} \Lambda_r(\mathcal{P}) &= \sum_i \Lambda_{r_i} \\ &= \frac{1}{\sigma_{\text{eq}}^2} \begin{bmatrix} \mathbf{J}_{c_0}^\top \mathbf{J}_{c_0} & & \\ & \ddots & \\ & & \mathbf{J}_{c_{n-1}}^\top \mathbf{J}_{c_{n-1}} \end{bmatrix}, \end{aligned}$$

where  $\mathbf{J}_{c_i}$  is the range measurement Jacobian in (2) for  $\mathbf{x}_{c_i}$  and  $\mathbf{x}_{s_i}$ . The posterior client information,  $p(\mathbf{X}_c | \mathbf{Z}_r(\mathcal{P})) \sim \mathcal{N}^{-1}(\boldsymbol{\eta}'_c, \Lambda'_c)$ , is then simply computed

$$\Lambda'_c = \Lambda_c + \Lambda_r(\mathcal{P}).$$

### B. Information-based Objective

A mutual information objective seeks informative range observations while simultaneously considering the prior belief over the client. For example, mutual information will seek to add observations along directions for which the client is most uncertain. We maximize the mutual information between the client and expected range-only observations,

$$\begin{aligned} \text{MI}[\mathbf{X}_c | \mathbf{Z}_r(\mathcal{P})] &= \text{H}[\mathbf{X}_c] - \text{H}[\mathbf{X}_c | \mathbf{Z}_r(\mathcal{P})] \\ &= -\frac{1}{2} \log |\Lambda_c| + \frac{1}{2} \log |\Lambda_c + \Lambda_r(\mathcal{P})|, \end{aligned}$$

where  $\text{H}[\cdot]$  represents the entropy of the distribution [29]. We can drop constant and multiplicative terms to arrive at our objective function

$$I(\mathcal{P}) = \log |\Lambda_c + \Lambda_r(\mathcal{P})| = \log |\Lambda'_c|. \quad (4)$$

Since log is a monotonically increasing function, maximizing mutual information is equivalent to maximizing (minimizing) the volume of the posterior information (covariance). This objective function is therefore equivalent to evaluating the determinant of the Fischer information matrix considered previously by [19] and [30] for static beacon localization.

We can modify our objective function to support multiple client vehicles. Each client vehicle is only informed about the server vehicle (and not the other clients) during real-time operation, so that the  $j$ th client tracks a distribution  $p(\mathbf{X}_{c_j} | \mathbf{Z}_r(\mathcal{P}))$ . Therefore, each client is considered independently and the objective for  $M$  clients is simply the sum over each individual client

$$I(\mathcal{P}) = \sum_{j=0}^{M-1} \log |\Lambda'_{c_j}|.$$

We would obtain a different objective function if we considered the joint distribution over *all* vehicles,  $p(\mathbf{X}_{c_0}, \dots, \mathbf{X}_{c_{M-1}} | \mathbf{Z}_r(\mathcal{P}))$ , which models correlation that develops between clients as a result of range observations through the server. Due to communication constraints, this centralized estimator is realizable in post-process only.

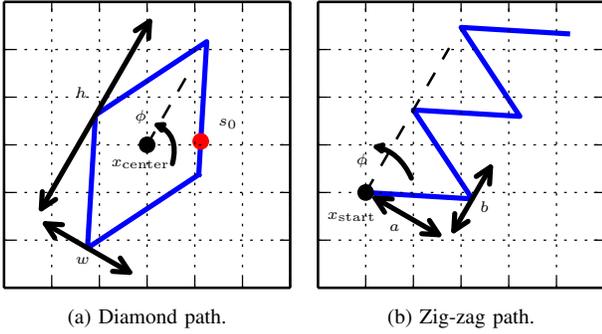


Fig. 2: Illustration of the parameterized trajectories that we consider. Arrows and dots indicate free parameters listed in Table I.

### C. Mutual Information Upper Bound

We define an upper bound on the maximum achievable information objective to serve as a yardstick with which to compare our performance. Computing an exact bound, however, is as hard as the original planning problem (1). To reduce the complexity, we relax the constraint that server poses lie along a feasible trajectory and instead allow the server to ‘teleport’ between TOL positions.

Each observation, in this case, can be represented by the relative angle between server and client alone. The upper bound solves for the set of  $N$  relative server angles  $\Theta = \{\theta_0, \dots, \theta_{N-1}\}$  that maximize the objective,

$$I_{\max} = \max_{\Theta} \log |\Lambda_c + \Lambda_r(\Theta)|. \quad (5)$$

The relaxed optimization problem involves solving a difficult nonlinear program. This bound is *not* necessarily achievable since we have relaxed the constraint that server poses lie along a feasible trajectory. A trajectory that achieves this bound may exist, however, if the server can reach  $\theta_i$  for each TOL.

We can quickly obtain a local optimal using gradient descent. The gradient of the objective function with respect to  $\theta_i$  is

$$\frac{\partial \log |\Lambda'_c|}{\partial \theta_i} = \text{tr} \left( \Lambda_c'^{-1} \frac{\partial \Lambda'_c}{\partial \theta_i} \right).$$

We initialize pairwise sequential  $\theta_i$ 's to be orthogonal. We expect sequential angles to be orthogonal depending on the prior information based on static beacon localization analysis [19]. In practice, we expect that the local optimal should still act as an upper bound on the actual objective function.

Our upper bound is again easily extended to the multiple client case by computing an independent relative angle for each client. We expect the bound to be more conservative in this case since it is unlikely or even impossible for the server to reach each client's  $\theta_i$  at every TOL.

### D. Trajectory Search

Sampling-based planning algorithms are based on graph search with exponential worst-case complexity. For certain objective functions, however, heuristics exist for much more efficient search. Since our cooperative planning problem involves a time-varying information field (since the client

TABLE I: Trajectory parameters.

Diamond parameters:	
Center position	$x_{\text{center}} \in \mathbb{R}^2$
Width and height	$w, h \in \mathbb{R}^+$
Rotation	$\phi \in [0, \pi)$
Starting position along diamond	$s_0 \in [0, 1)$
Zig-zag parameters:	
Start position	$x_{\text{start}} \in \mathbb{R}^2$
Amplitude and wavelength	$a, b \in \mathbb{R}^+$
Rotation	$\phi \in [0, 2\pi)$

is dynamic) and a submodular objective function, these heuristics do not apply.

Prior methods have dealt with exponential complexity by only planning over a finite horizon [7]. However, we would like to leverage the full *a priori* client trajectory when available. Therefore, instead of searching over the full space of potential server trajectories, we search over the parameter space for several classes of trajectories. The smaller parameter space results in a computationally tractable solution—exponential complexity in the number of parameters, which is far less than exponential complexity in the number of server poses. Additionally, the trajectory classes we consider are both simple to plan and execute: diamond patterns, and zig-zags (Fig. 2). We chose these simple, easily parameterized paths for their ability to support a variety of client operations, although any trajectory class that can be easily parameterized could be included within the search.

1) *Search Algorithm:* We first describe our basic algorithm defined in Algorithm 1. We perform a parameter sweep for each trajectory class,  $M_i$ , with the parameter range and resolution defined by the user. Each iteration inside the parameter sweep executes a local search. For each parameter set, we find the local maximum of the mutual information objective and the corresponding optimized parameters. The algorithm returns the trajectory class and parameters that result in the highest objective.

2) *Trajectory Classes:* We rely on two simple trajectory classes in order to find useful server paths: diamond patterns, and zig-zags. We choose the diamond pattern because it displays periodic behavior, consists of straight line tracks, and is described by relatively few parameters. Similarly, zig-

---

#### Algorithm 1 Server Cooperative Path Planning

---

**Require:**  $\Lambda_c, \eta_c$  {Initial client distribution}

**Ensure:**  $M_{\text{best}}, \Theta_{\text{best}}$  {Optimal trajectory class and parameters}

```

1:  $I_{\text{best}} = 0, M_{\text{best}} = \emptyset, \Theta_{\text{best}} = \emptyset$ 
2: for  $M_i \in \{M_0, \dots, M_m\}$  do
3:   for  $\Theta \in \text{dom } M_i$  do
4:      $\Theta^* = \arg \max_{\Theta} I(\mathcal{P}(M_i, \Theta))$ 
5:   if  $I(\mathcal{P}(M_i, \Theta^*)) > I_{\text{best}}$  then
6:      $I_{\text{best}} = I(\mathcal{P}(M_i, \Theta^*))$ 
7:      $M_{\text{best}} = M_i$ 
8:      $\Theta_{\text{best}} = \Theta^*$ 
9:   end if
10: end for
11: end for
12: return  $M_{\text{best}}, \Theta_{\text{best}}$ 

```

---

TABLE II: Percent achieved information gain relative to upper bound— $1 - \Delta I / \Delta I_{\max}$ . Smaller fraction indicates closer to upper bound.

	Single A	Single B	Multiple A	Multiple B
Centroid [24]	24.005%	8.484%	8.690%	9.483%
Finite-horizon [7]	5.851%	8.238%	9.673%	8.097%
Proposed	<b>0.003%</b>	<b>3.956%</b>	<b>5.604%</b>	<b>5.793%</b>

zag patterns consist of straight line tracks and are easily parameterized. Fallon et al. [4] previously suggested zig-zag paths for range-only localization with cooperative navigation aids. We assume that the server travels at a constant prescribed velocity with deterministic control, so that the TOL poses are evenly spaced along its path. The total server path length is determined by the duration of the client survey. The parameters for each trajectory class are illustrated in Fig. 2 and listed in Table I.

#### IV. EXPERIMENTS

We present several examples to demonstrate the ability of our algorithm to provide informative server trajectories to support client vehicles. We also compare our solution to the centroid method proposed by German et al. [24], and a finite-horizon planner using dynamic programming similar to the one reported in [7]. The centroid method is a heuristic that simply positions each server TOL at the centroid of the remaining client survey area. We modified the objective function evaluated in [7] for the finite-horizon method to our objective (4) for fair comparison. We initialized the server position at the origin for the finite-horizon planner, as it does not optimize over the initial position.

We solved the upper bound optimization problem using the BFGS optimization procedure [31]. The parameter optimization was performed using a derivative-free iterative quadratic approximation algorithm [32]. The client’s prior information  $\Lambda_c$  was block tri-diagonal (Markov chain dynamic model) for all experiments, and therefore we were able to employ efficient routines for computing the determinant of the posterior information.

We evaluate the performance of each planned path by comparing the fraction of the achieved information gain to the upper bound objective

$$1 - \frac{\Delta I}{\Delta I_{\max}} = 1 - \frac{I(\mathcal{P}) - I(\emptyset)}{I_{\max} - I(\emptyset)},$$

where  $I(\emptyset) = \log |\Lambda_c|$  represents the prior objective value (i.e., the mutual information objective with no range measurements). This metric indicates how close the planned path came to providing the upper bound information, (5).

##### A. Single Client Simulations

Here we explore two typical AUV client usage scenarios: client travels in a straight line (Single A), and client performs a lawn mower survey (Single B). Expected TOA poses were simulated with a fixed time division multiple access (TDMA) schedule, adding roughly one OWTT constraint per minute. Single client results are summarized in Table II and Fig. 3.

The planned server path for the straight line scenario (Single A) is shown in Fig. 3. The computed zig-zag server

path alternately places server TOL poses directly along the N-S and E-W axes relative to the client TOA pose. This solution essentially meets the upper bound optimal solution by providing sequentially orthogonal relative angles. The centroid method does not perform well in this configuration as expected because information is continually added in the same direction. The finite-horizon method chooses a path similar to the zig-zag but is unable to reach the optimal configuration because the discrete action set does not contain such a path.

The server computes another zig-zag path for the client lawn mower survey (Single B), Fig. 3. The server path is roughly centered within the N-S area covered by the client survey area and continually travels E-W across the lawn mower. The centroid method slowly moves E-W along the N-S center of the server area. Both our algorithm and the centroid produce similar paths, relying on the client to produce N-S relative motion. The finite-horizon algorithm continually encircles the the survey area.

##### B. Multiple Client Simulations

We demonstrate two two-client AUV scenarios: clients perform overlapping and orthogonal lawn mower surveys (Multiple A), and clients perform adjacent lawn mower surveys (Multiple B). Again, we used a fixed TDMA schedule adding one OWTT constraint to each client per minute. Fig. 3 and Table II summarize the planner performance.

For two client vehicles performing overlapping and orthogonal lawn mower surveys (Multiple A), the server finds a diamond trajectory near the centroid of the total survey area that compromises providing variation in relative angle to both clients. The centroid method finds a similar path, and actually outperforms the finite-horizon planner relative to the upper bound information (without performing any optimization). The finite-horizon planner circles the survey area around both clients. The upper bound in this scenario is almost certainly not realizable, since we anticipate our bound performance to be more conservative for multiple vehicles.

For two clients performing adjacent lawn mower surveys (Multiple B), the server chooses a zig-zag pattern maintaining a position in between the two clients along the E-W direction. The centroid method similarly moves E-W. Finally, the finite-horizon planner encircles the survey area. Our planner again computes the best path relative to the upper bound.

#### V. DEEP WATER DISCUSSION

In Section III, we assumed that 3D slant ranges provided by OWTTs could be projected into the local horizontal plane with a fixed measurement covariance. In shallow water environments, the slant range is generally much larger than the relative depth. For deep water AUV surveys with surface OWTT support, the relative depth is a significant fraction of the slant range (Appendix I). Using the correct noise model for pseudo ranges has important repercussions for planning.

As illustrated in Fig. 4, the information added in the horizontal plane by a range measurement decreases as the

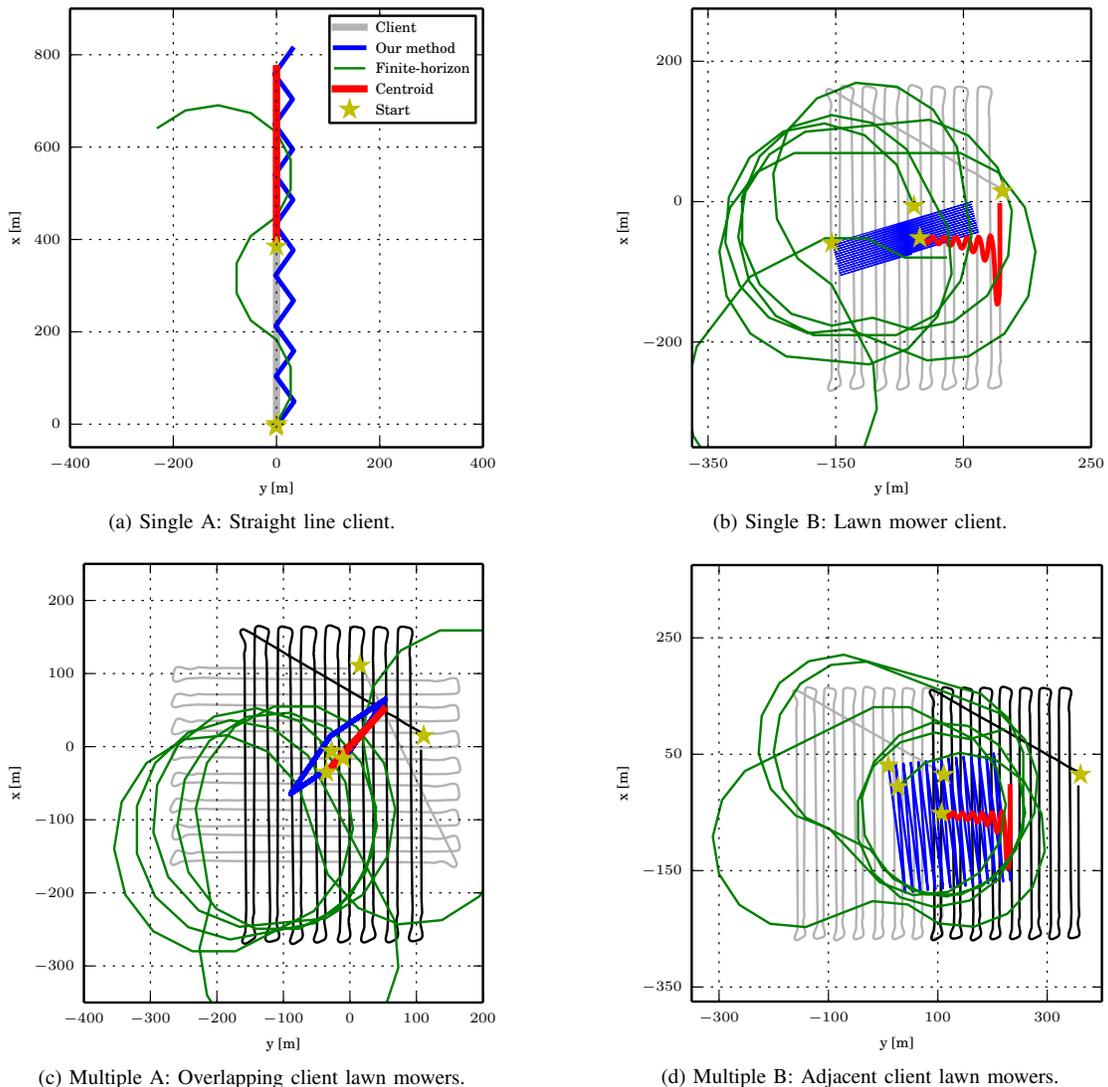


Fig. 3: Planned server trajectories for single and multiple clients. Legend shown in (a).

ratio of depth to slant range increases. When the depth dependent range uncertainty is considered, we expect the server to seek paths with a small depth to range ratio (i.e., a wide baseline away from the client). If a fixed pseudo range uncertainty is naively employed, resulting server paths will not execute wide baseline changes, and the client will not be well informed.

The vertical acoustic channel provides the best reception characteristics. Communication and navigation, therefore, represent conflicting goals [24]. We anticipate that this tradeoff could be managed by modeling the acoustic channel within our objective function (4), and taking the expectation with respect to the relative vehicle paths. This remains the subject of future and ongoing work.

## VI. CONCLUSIONS

We have shown that our planning framework produces operationally practical server trajectories for supporting client AUVs. Moreover, these paths perform well relative to an upper bound on the objective and compared to existing planning

frameworks. Several avenues remain open for future work including modeling the acoustic channel and considering nondeterministic server control.

## APPENDIX I PSEUDO RANGE NOISE VARIANCE

Many prior estimation and planning frameworks for OWTT localization have projected 3D slant ranges into the 2D horizontal plane, obtaining a set of *pseudo* ranges used with a fixed measurement covariance. Here, we give a formulation for the pseudo range as a function of the measured slant range and relative depth between the server and client and provide a first-order covariance approximation.

We model a slant range between 3D client and server states  $\mathbf{x}_c, \mathbf{x}_s \in \mathbb{R}^3$  as

$$z_{\text{slant}} = \|\mathbf{x}_s - \mathbf{x}_c\|_2 + w_{\text{slant}} \quad (6)$$

where  $w_{\text{slant}} \sim \mathcal{N}(0, \sigma_{\text{slant}}^2)$ . We assume that we have access

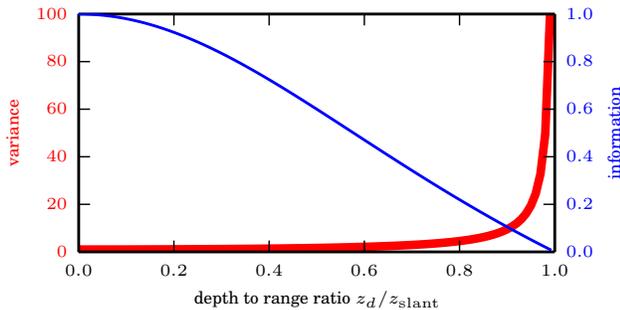


Fig. 4: Pseudo range variance  $\sigma_{\text{pseudo}}^2$  and information  $1/\sigma_{\text{pseudo}}^2$  for increasing depth to range ratio. As the relative depth increases relative to the slant range, the variance increases corresponding to a decrease in information in the horizontal plane.

to a relative depth observation

$$z_d = z_s - z_c + w_d,$$

where  $z_s$  and  $z_c$  are the depth components of  $\mathbf{x}_s$  and  $\mathbf{x}_c$ , respectively, and  $w_d \sim \mathcal{N}(0, \sigma_d^2)$ . Slant ranges are projected into the horizontal plane given the relative depth

$$z_{\text{pseudo}} = \sqrt{z_{\text{slant}}^2 - z_d^2},$$

for  $z_d < z_{\text{slant}}$ . We can model  $z_{\text{pseudo}}$  as a Gaussian random variable with variance approximated through the first-order Taylor expansion, i.e.

$$\sigma_{\text{pseudo}}^2 = \frac{z_{\text{slant}}^2 \sigma_{\text{slant}}^2 + z_d^2 \sigma_d^2}{z_{\text{slant}}^2 - z_d^2}. \quad (7)$$

The variance is *not* a simple weighted combination of the depth and slant range variances, nor is it a fixed value. The variance explodes as  $z_d \rightarrow z_{\text{slant}}$ , regardless of the depth and slant range uncertainties. Linearized range observations add information along the vector between the server and client positions. As  $z_d \rightarrow z_{\text{slant}}$ , almost all information is added along the depth direction, and little information is added in the horizontal plane. This relationship is illustrated in Fig. 4.

## REFERENCES

- [1] J. G. Bellingham and K. Rajan, "Robotics in remote and hostile environments," *Science*, vol. 318, no. 5853, pp. 1098–1102, 2007.
- [2] J. Vaganay, J. J. Leonard, J. A. Curcio, and J. S. Wilcox, "Experimental validation of the moving long base-line navigation concept," in *Proc. IEEE/OES Autonomous Underwater Vehicles Conf.*, Sebasco, ME, June 2004, pp. 59–65.
- [3] A. Bahr, J. J. Leonard, and M. F. Fallon, "Cooperative localization for autonomous underwater vehicles," *Int. J. Robot. Res.*, vol. 28, no. 6, pp. 714–728, 2009.
- [4] M. F. Fallon, G. Papadopoulos, J. J. Leonard, and N. M. Patrikalakis, "Cooperative AUV navigation using a single maneuvering surface craft," *Int. J. Robot. Res.*, vol. 29, no. 12, pp. 1461–1474, 2010.
- [5] S. E. Webster, J. M. Walls, R. M. Eustice, and L. L. Whitcomb, "Decentralized extended information filter for single-beacon cooperative acoustic navigation: Theory and experiments," *IEEE Trans. Robot.*, vol. 29, no. 4, pp. 957–974, 2013.
- [6] J. M. Walls and R. M. Eustice, "An origin state method for communication constrained cooperative localization with robustness to packet loss," *Int. J. Robot. Res.*, 2014, In Print.
- [7] Y. T. Tan, R. Gao, and M. Chitre, "Cooperative path planning for range-only localization using a single moving beacon," *IEEE J. Ocean. Eng.*, vol. 39, no. 2, pp. 371–385, 2014.
- [8] G. Hollinger and G. Sukhatme, "Stochastic motion planning for robotic information gathering," in *Proc. Robot.: Sci. & Syst. Conf.*, Berlin, Germany, June 2013.
- [9] L. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, 1996.
- [10] S. M. LaValle and J. J. Kuffner, "Randomized kinodynamic planning," *Int. J. Robot. Res.*, vol. 20, no. 5, pp. 378–400, May 2001.
- [11] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *Int. J. Robot. Res.*, vol. 30, no. 7, pp. 846–894, 2011.
- [12] R. Sim and N. Roy, "Global A-optimal robot exploration in SLAM," in *Proc. IEEE Int. Conf. Robot. and Automation*, Barcelona, Spain, Apr. 2005, pp. 661–666.
- [13] S. Prentice and N. Roy, "The belief roadmap: Efficient planning in belief space by factoring the covariance," *Int. J. Robot. Res.*, vol. 28, no. 11–12, pp. 1448–1465, 2009.
- [14] J. Binney and G. Sukhatme, "Branch and bound for informative path planning," in *Proc. IEEE Int. Conf. Robot. and Automation*, Saint Paul, MN, USA, May 2012, pp. 2147–2154.
- [15] A. Bry and N. Roy, "Rapidly-exploring random belief trees for motion planning under uncertainty," in *Proc. IEEE Int. Conf. Robot. and Automation*, Shanghai, China, May 2011, pp. 723–730.
- [16] T. L. Song, "Observability of target tracking with bearings-only measurements," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 4, pp. 1468–1472, 1996.
- [17] A. S. Gadre and D. J. Stilwell, "A complete solution to underwater navigation in the presence of unknown currents based on range measurements from a single location," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Edmonton, Canada, 2005, pp. 1420–1425.
- [18] X. S. Zhou and S. I. Roumeliotis, "Robot-to-robot relative pose estimation from range measurements," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1379–1393, 2008.
- [19] S. Martinez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," *Automatica*, vol. 42, no. 4, pp. 661–668, 2006.
- [20] M. Hunt, W. Marquet, D. Moller, K. Peal, W. Smith, and R. Spindel, "An acoustic navigation system," Woods Hole Ocean. Inst., Tech. Rep. WHOI-74-6, Dec. 1974.
- [21] M. V. Jakuba, C. N. Roman, H. Singh, C. Murphy, C. Kunz, C. Willis, T. Sato, and R. A. Sohn, "Field report: Long-baseline acoustic navigation for under-ice AUV operations," *J. Field Robot.*, vol. 25, no. 11–12, pp. 861–879, 2008.
- [22] E. Olson, J. J. Leonard, and S. Teller, "Robust range-only beacon localization," *IEEE J. Ocean. Eng.*, vol. 31, no. 4, pp. 949–958, 2006.
- [23] A. Bahr, J. J. Leonard, and A. Martinoli, "Dynamic positioning of beacon vehicles for cooperative underwater navigation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, Vilamoura, Portugal, Oct. 2012, pp. 3760–3767.
- [24] C. R. German, M. V. Jakuba, J. C. Kinsey, J. Partan, S. Suman, A. Belani, and D. R. Yoerger, "A long term vision for long-range ship-free deep ocean operations: Persistent presence through coordination of autonomous surface vehicles and autonomous underwater vehicles," in *Proc. IEEE/OES Autonomous Underwater Vehicles Conf.*, Southampton, UK, Sept. 2012, pp. 1–7.
- [25] B. Charrow, V. Kumar, and N. Michael, "Approximate representations for multi-robot control policies that maximize mutual information," in *Proc. Robot.: Sci. & Syst. Conf.*, Berlin, Germany, June 2013.
- [26] B. Charrow, N. Michael, and V. Kumar, "Cooperative multi-robot estimation and control for radio source localization," *Int. J. Robot. Res.*, vol. 33, no. 4, pp. 569–580, 2014.
- [27] R. M. Eustice, H. Singh, and J. J. Leonard, "Exactly sparse delayed-state filters for view-based SLAM," *IEEE Trans. Robot.*, vol. 22, no. 6, pp. 1100–1114, 2006.
- [28] J. M. Walls and R. M. Eustice, "Experimental comparison of synchronous-clock cooperative acoustic navigation algorithms," in *Proc. IEEE/MTS OCEANS Conf. Exhib.*, Kona, HI, USA, Sept. 2011, pp. 1–7.
- [29] R. Horn and C. Johnson, *Matrix Analysis*. Cambridge University Press, 1990.
- [30] A. N. Bishop, B. Fidan, B. D. Anderson, K. Dogancay, and P. N. Pathirana, "Optimality analysis of sensor-target localization geometries," *Automatica*, vol. 46, no. 3, pp. 479–492, 2010.
- [31] J. Nocedal and S. Wright, *Numerical Optimization*. Springer, 2006.
- [32] M. J. D. Powell, "The BOBYQA algorithm for bound constrained optimization without derivatives," University of Cambridge, Tech. Rep. NA2009/06, 2009.