Bathymetric Factor Graph SLAM with Sparse Point Cloud Alignment

Vittorio Bichucher, Jeffrey M. Walls, Paul Ozog, Katherine A. Skinner, and Ryan M. Eustice

***Abstract*— This paper reports on a factor graph simultaneous localization and mapping framework for autonomous underwater vehicle localization based on terrain-aided navigation. The method requires no prior bathymetric map and only assumes that the autonomous underwater vehicle has the ability to sparsely sense the local water column depth, such as with a bottom-looking Doppler velocity log. Since dead-reckoned navigation is accurate in short time windows, the vehicle accumulates several water column depth point clouds—or submaps—during the course of its survey. We propose an *xy*-alignment procedure between these submaps in order to enforce consistent bathymetric structure over time, and therefore attempt to bound long-term navigation drift. We evaluate the submap alignment method in simulation and present performance results from multiple autonomous underwater vehicle field trials.**

I. Introduction

Precision autonomous underwater vehicle (AUV) navigation often relies on integrating noisy Doppler velocity log (DVL) body-frame velocity and attitude along with pressure-depth to compute a dead-reckoned (DR) navigation solution [1]. However, integrating noise in velocity and attitude observations leads to unbounded error growth in xy-position estimates. A global positioning system (GPS) provides accurate geo-referenced coordinates to vehicles while at the surface but is unavailable subsea due to the attenuation of electromagnetic signals. Acoustic positioning systems such as ultra-short-baseline (USBL) and long-baseline (LBL) provide relative measurements from the AUV to a support vessel or static beacon [2]. However, acoustic systems limit the range of operations and require additional infrastructure.

Terrain-aided navigation (TAN) consists of using a prior water column height map, or bathymetric map, to perform AUV localization while surveying the underwater environment. Terrain-aided simultaneous localization and mapping (SLAM) consists of concurrently using the sensed bathymetry as information for AUV localization and mapping.

In this work, we propose a method to reduce long-term navigation drift using only onboard vehicle sensors to exploit bathymetry information. Moreover, our method does not rely on a previous bathymetric map. If the environment exhibits topographic relief and the survey trajectory is designed wisely, our method can achieve bounded-error navigation for AUVs.

*This work was supported in part by the Office of Naval Research under awards N00014-12-1-0092 and N00014-11-D-0370.

V. Bichucher, J. Walls, P. Ozog, K. Skinner, and R. Eustice are with the University of Michigan, Ann Arbor, Michigan 48109, USA {vittbich, jmwalls, paulozog, kskin, eustice}@umich.edu.

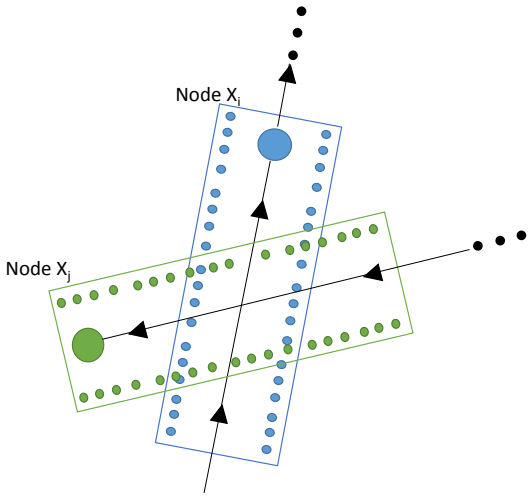


Fig. 1: Example of submap alignment. Two submap nodes, $\mathbf{x}_i$ and $\mathbf{x}_j$ have intersecting submaps, representing a collection of water column observations. The alignment procedure computes an xy-translation between $\mathbf{x}_i$ and $\mathbf{x}_j$ such that the bathymetric structure within the overlapping region is consistent. The black triangles show direction of travel.

We use factor graph SLAM to estimate the AUV trajectory—a factor graph discretizes the AUV trajectory into nodes (AUV poses at some instance in time) and factors (measurements that constrain nodes). We introduce a new submap alignment factor type: rigid-body transformations between nodes calculated by intersecting submaps (Fig. 1) generated by accumulation of sparse point-cloud DVL altitude measurements.

To compute the rigid-body transformation, scan matching techniques are used. We modified the Generalized Iterative Closest Point (GICP) [3] algorithm to only optimize over the xy-position, since all other degrees of freedom have bounded error and our submap is sparse. This alignment procedure can be posed as a quadratic program and can, therefore, be efficiently solved.

Specifically, the contributions of this paper are:

- A factor-graph-based bathymetric SLAM framework that exploits a sparse point cloud representation of the underlying bathymetry measured by a DVL;
- A modified GICP algorithm to provide a translation constraint between overlapping submaps.

We evaluate our algorithm in simulation and show that our alignment solution provides an accurate xy translation observation between submaps. We then provide performance results using data collected from several AUV field trials.

II. Previous Work

Navigation systems for AUVs leverage multiple sensor modalities; recent surveys include the work by Paull et al. [4] and Kinsey et al. [5]. Some algorithms involve the use of acoustic beacons to provide range-only observations to the AUV [2, 6]. Others use camera-based methods to derive relative-pose measurements [7, 8].

TAN represents a popular method for AUV localization, and it has been approached through many avenues. TAN for AUVs can be traced back to Di Massa and Stewart [9] and Kulander [10] who applied derivatives of terrain contour matching (TERCOM)—an application of TAN toward missile guidance—to AUV navigation. These approaches used a correlative search between bathymetry measurements and a prior bathymetric map. More recently, Nygren and Jansson [11] developed a similar framework but for scenarios with weak priors in AUV pose estimation.

Particle filters are now a standard approach to TAN because they can express multi-modal distributions and naturally represent multiple data association hypotheses. Karlsson et al. [12] and Bergman et al. [13] proposed a navigation method for AUVs by comparing the sensed bathymetry to a prior bathymetric map using a particle filter. Karlsson et al. [12] also investigated the use of the Cramer-Rao lower bound to estimate sensor noise and necessary bathymetry excitement for particle filter convergence. Williams and Mahon [14] similarly applied a particle filter to an existing bathymetric map to estimate position and velocities in unstructured natural terrain. Claus and Bachmayer [15] used a jittered bootstrap particle filter to avoid particle degeneracy while performing TAN with a prior bathymetric map. Finally, Kimball and Rock used both particle filters [16] and nonlinear least squares [17], along with an *a priori* iceberg sonar map, to compute the relative position of an AUV with respect to the floating mass, while Eustice et al. [18] used an *a priori* ship-borne bathymetric map to post-process DR DVL navigation error for an AUV.

State-of-the-art SLAM frameworks are smoothing approaches wherein the entire vehicle trajectory is estimated. The method was first presented for AUVs by Eustice et al. [19], who applied a smoothing, or delayed-state, approach to visually augmented underwater navigation. Later, Kim and Eustice [20] extended these capabilities for improved front-end and back-end performance by considering the visual saliency of each underwater image. Fallon et al. [6] also demonstrated the benefits of full trajectory smoothing over standard filter approaches for AUV navigation.

Barkby et al. [21] and Roman and Singh [22] performed bathymetry-based SLAM for AUVs. Barkby et al. [21] introduced, using a particle filter, a novel map representation using Gaussian processes (GP) to reduce computational costs and to perform map alignments when no, or little, overlap is present. Roman and Singh [22] used dense submaps, collected using multibeam sonars, and scan matching algorithms to create rigid-body transformations between those submaps. Our work is an extension of [22] within a factor graph estimation framework, which includes full trajectory smoothing, and a new submap alignment technique.

Scan matching techniques consist of computing the rigid-body transformation between two vehicle poses in the form of point clouds. The GICP algorithm developed by Segal et al. [3] has been a common approach in terrestrial robotics. However, its use typically relies on laser range sensors to produce a dense point cloud representation of the environment. We have modified this algorithm to optimize only over xy-translation, instead of computing a rigid-body 6 degree of freedom (DOF) transformation since other DOFs are well instrumented. For scan matching in the AUV community, Ozog et al. [23] aligned accumulated DVL range observations of a ship hull to a prior CAD model and used GICP for metric comparison. Roman and Singh [22] used a point-to-plane scan matching algorithm to register the dense submap alignments. Finally, VanMiddlesworth et al. [24] mapped the underwater portion of ship hulls using a profiling sonar. Though they also used a factor graph and aligned submaps using iterative closest point (ICP), our approach uses point clouds that contain far fewer points, and a different cost function that only optimizes over the xy-direction.

III. Methodology

Our approach augments typical DR AUV navigation with terrain-based measurements from sensed depth and altitude to yield an observation of total water column depth. We rely on a factor graph estimation framework [25] to compute the most likely vehicle trajectory given all observations. The factor graph is a smoothing framework that estimates the set of poses along a vehicle trajectory. We introduce a factor that enforces consistency in observed water column depth (i.e., AUV poses at different times, but overlapping spatial area, should register similar water column depth).

Within our framework, AUVs have no prior bathymetric map. We assume that DR navigation drift is negligible over small time scales such that we can create 'submaps'—or sparse point clouds composed of water column depth observations. An appropriate choice for the time scale of a submap may vary according to the performance of DR. We found that a time scale of $10\,\mathrm{s}$ works well in our application. For a vehicle pose at the ith time index, $\mathbf{x}_i$, we accumulate a point cloud, $\mathbf{P}^i = \{\mathbf{p}_n^i\}_{n=1}^m$, where each water column depth point, $\mathbf{p}_n^i$, is expressed relative to $\mathbf{x}_i$. We then use a modified GICP algorithm to align overlapping submaps. The alignment step between two submaps, $\mathbf{P}^i$ and $\mathbf{P}^j$, provides a relative measurement between two poses, $\mathbf{x}_i$ and $\mathbf{x}_j$. The alignment is only computed over xy-translation because attitude and depth are directly instrumented with bounded error.

In the following, we first review factor graph estimation. We then derive an xy-alignment procedure as a quadratic program from GICP alignment. Finally, we demonstrate the implementation of the alignment factor within an AUV factor graph.

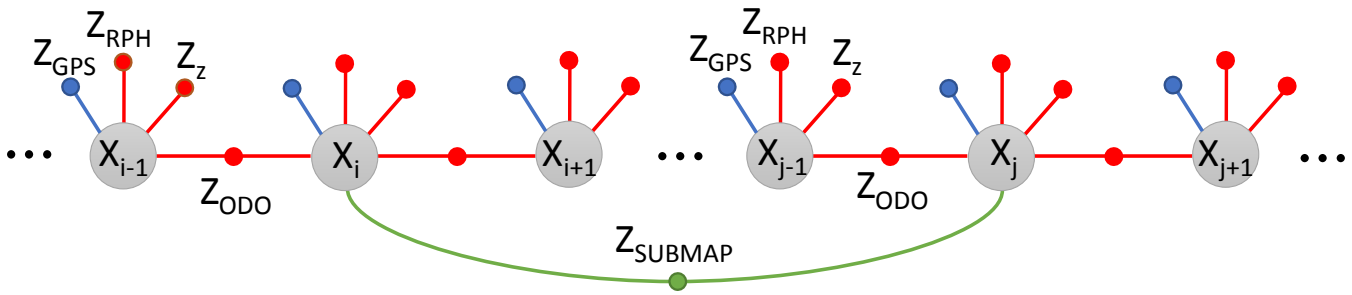


Fig. 2: Factor graph representation of our SLAM problem. Red factors are DR navigation, blue factors are GPS constraints (only available at the surface), and green factors illustrate submap alignment. The color scheme represents the three graphs we will be comparing throughout the paper: red factors consist of the DR solution, red and blue factors create a proxy ground truth graph, and red and green factors form the bathymetric SLAM graph we propose.

A. Factor Graph Estimation

Factor graph estimation, popular within the SLAM community, is a general smoothing framework. Here, we only consider nodes that consist of 6-DOF poses, often referred to as pose graphs.

Factor graph estimation optimizes the set of all vehicle poses along a trajectory, $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, using least squares optimization. Approaching SLAM as a least squares problem leverages the inherent sparsity of the SLAM problem, which enables efficient inference. Further, implementations of factor graph SLAM such as incremental smoothing and mapping (iSAM) [25] facilitates integration of other factor types developed in other literature.

The factor graph represents the joint distribution over poses given all measurements, $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_n\}$,

$$p(\mathbf{X}|\mathbf{Z}) \propto p(\mathbf{x}_1) \prod_i p(\mathbf{z}_i|\mathbf{X}_{j\in\mathcal{C}_i}), \tag{1}$$

where $\mathcal{C}_i$ represents the set of pose indices constrained by factor $\mathbf{z}_i$. We discuss the factor types (each $\mathbf{z}_i$) considered within this paper below and depicted in Fig. 2. The maximum *a posteriori* (MAP) estimate over the vehicle trajectory can be computed from the joint distribution (1) as

$$\begin{aligned} \mathbf{X}^* &= \underset{\mathbf{X}}{\operatorname{argmax}}\, p(\mathbf{X}|\mathbf{Z}) \\ &= \underset{\mathbf{X}}{\operatorname{argmin}} -\log p(\mathbf{X}|\mathbf{Z}). \end{aligned} \tag{2}$$

For Gaussian noise models, the above optimization is reduced to a least squares problem [26]. Moreover, the solution can be computed efficiently due to the sparse nature of factors (measurements typically involve very few poses in the graph).

Each factor represents a measurement over a set of poses. Within this work, we consider three factor types: odometry factors, unary pose factors, and submap alignment factors. We detail the first two factors below and submap alignment in Section III-B.

1) Odometry factors: Odometry factors model the transformation between sequential poses over time. We compute odometry factors by integrating DVL body-frame linear velocities [1] which results in translational odometry measurements with respect to the world coordinate frame,

$$\mathbf{z}_{\text{odo}} = \mathbf{t}_i - \mathbf{t}_{i-1} + \mathbf{w}_{\text{odo}}, \tag{3}$$

where $\mathbf{w}_{\text{odo}} \sim \mathcal{N}\big(\mathbf{0}, \mathrm{R}_{\text{odo}}(\mathrm{d})\big)$ is an additive zero-mean Gaussian noise perturbation that is a function of distance traveled (d), and $\mathbf{t}_k$ is the xyz-location of node $\mathbf{x}_k$.

2) Unary pose factors: Attitude, depth, and GPS (while at the surface) are modeled by unary pose factors. Each unary pose factor involves only a single pose node,

$$\mathbf{z}_{\text{rph}} = \text{rph}(\mathbf{x}_i) + \mathbf{w}_{\text{rph}}, \tag{4}$$

$$\mathbf{z}_{\text{depth}} = \text{depth}(\mathbf{x}_i) + \mathbf{w}_{\text{depth}}, \tag{5}$$

$$\mathbf{z}_{\text{gps}} = \mathbf{x}_i + \mathbf{w}_{\text{gps}}, \tag{6}$$

where each measurement model includes an additive zero-mean Gaussian noise term, $\mathbf{w}_{\text{rph}}$, $\mathbf{w}_{\text{depth}}$, and $\mathbf{w}_{\text{gps}}$, respectively. Attitude is directly observed by an inertial measurement unit (IMU) and magnetic compass, depth is measured by a pressure sensor, and GPS measures xy vehicle position.

Attitude, odometry, and depth factors provide the necessary constraints for DR navigation. Moreover, the addition of GPS factors to our DR graph generates our ground truth solution. We will compare our results of DR navigation combined with the submap alignments to these two factor graphs. These graphs are shown in Fig. 2, where the color of each factor represents to which graph the factor belongs.

Since our observation models are linear functions of the state, this implies that (2) can be written as a *linear* least squares problem, such that the MAP estimate can be computed in exactly one Gauss-Newton iteration.

B. Map Alignment Factors

Submaps are generated by accumulating DVL altitude measurements between nodes. When submaps of different nodes spatially overlap, they should register similar water column depths. We exploit this shared information to compute a 2D transformation between submaps by enforcing water column depth consistency. We add the rigid-body transformation to our factor graph as a submap factor,

$$\mathbf{z}_{ij} = \mathbf{x}_j - \mathbf{x}_i + \mathbf{w}_{ij}, \tag{7}$$

where $\mathbf{z}_{ij}$ is taken from a modified GICP algorithm and $\mathbf{w}_{ij} \sim \mathcal{N}\big(\mathbf{0}, \mathrm{R}_{ij}\big)$ is zero-mean Gaussian noise.

Iterative closest point (ICP) is a simple and efficient method to compute the rigid-body transformation that best aligns two point clouds. Early methods of ICP [27] addressed point-to-point registration, while Chen and Medioni [28] proposed a point-to-plane variant. These ICP methods alternate

between two steps: computing point correspondence between the two point clouds, and computing the transformation that minimizes the distance between the corresponding points. The drawbacks of such methods are the implicit assumptions of full overlap between geometric surfaces being matched or that points are taken from a measured, not sensed, geometric surface.

Motivated by a maximum likelihood estimate (MLE) approach, GICP modified the cost function when computing the best transformation given the point correspondence between the two point clouds. With such an approach, GICP has shown to be robust to poor point correspondences and to improve results over ICP algorithms while maintaining the simplicity and efficiency of ICP [3].

In detail, GICP assumes the measured points are drawn from independent Gaussians centered at the point's location. Therefore, given two points clouds, $\mathbf{P}^i = \{\mathbf{p}_n^i\}_{n=1}^m$ and $\mathbf{P}^j = \{\mathbf{p}_n^j\}_{i=1}^m$, associated to nodes $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively, GICP computes the rigid-body transformation ($\mathbf{x}_{ij}$) such that $\mathbf{x}_j = \mathbf{x}_i \oplus \mathbf{x}_{ij}$. Where the $\oplus$ operator corresponds to the head-to-tail operation as described by [29] It does so by first assuming the existence of two point clouds $\hat{\mathrm{P}}^i = \{\hat{\mathbf{p}}^i\}$ and $\hat{\mathrm{P}}^j = \{\hat{\mathbf{p}}^j\}$ such that

$$\mathbf{p}_n^i \sim \mathcal{N}\big(\hat{\mathbf{p}}_n^i, \mathrm{C}_n^{P^i}\big)$$
$$\mathbf{p}_n^j \sim \mathcal{N}\big(\hat{\mathbf{p}}_n^j, \mathrm{C}_n^{P^j}\big),$$

where $\mathrm{C}_n^{P^i}$ and $\mathrm{C}_n^{P^j}$ are the sample covariance matrices associated with point clouds $\mathbf{P}^i$ and $\mathbf{P}^j$, respectively. GICP defines the correct rigid-body transformation matrix between the two submaps as T^*—making $\mathbf{x}_{ij}$ the parametrized version of the matrix T^*,

$$\mathrm{T}^* = \begin{bmatrix} \mathrm{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix}, \tag{8}$$

where R is an orthonormal rotation matrix and $\mathbf{t}$ is a vector describing translation. It then defines, $\hat{\mathbf{p}}_n^j = \mathrm{T}^* \hat{\mathbf{p}}_n^i$. Therefore, given an arbitrary transformation T, the error between aligned point clouds is

$$\mathbf{d}_n^{(\mathrm{T})} = \mathbf{p}_n^j - \mathrm{T}\mathbf{p}_n^i$$
$$\mathbf{d}_n^{(\mathrm{T})} \sim \mathcal{N}\big(\hat{\mathbf{p}}_n^j - \mathrm{T}\hat{\mathbf{p}}_n^i, \mathrm{C}_n^{P^j} + \mathrm{T}\mathrm{C}_n^{P^i}\mathrm{T}^\top\big). \tag{9}$$

The MLE alignment is then computed as

$$\begin{aligned} \mathrm{T}^* &= \operatorname*{argmax}_{\mathrm{T}} \prod_n p(\mathbf{d}_n^{(\mathrm{T})}) \\ &= \operatorname*{argmax}_{\mathrm{T}} \sum_n \log p(\mathbf{d}_n^{(\mathrm{T})}) \\ &= \operatorname*{argmax}_{\mathrm{T}} \sum_n \mathbf{d}_n^{(\mathrm{T})^\top} (\mathrm{C}_n^{P^j} + \mathrm{T}\mathrm{C}_n^{P^i}\mathrm{T}^\top)^{-1} \mathbf{d}_n^{\mathrm{T}}. \end{aligned} \tag{10}$$

Attempting to optimize a 6-DOF rigid-body transformation between two point clouds using sparse DVL data could result in erroneous results due to a lack of constraints in the cost function, since our submap intersection is incredibly sparse ($\sim$25 points). Morever, since navigation drift is in the xy-plane, we modified the GICP algorithm to optimize only over xy-translation (t_x, t_y) by assuming the rotation and z-translation of T are known. Expanding (10) with the assumptions mentioned above we obtain

$$\mathbf{t}^* = \operatorname*{argmin}_{\mathbf{t}=[t_x,t_y,t_z]^\top} \sum_n \mathbf{t}^\top \Lambda_n^{-1} \mathbf{t} + 2(\mathrm{R}\mathbf{p}_n^i - \mathbf{p}_n^j)^\top \Lambda_n^{-1} \mathbf{t}, \tag{11}$$

subject to t_z being equal to the measured depth difference between the two nodes, and R being the rotation matrix taken from the attitude sensor. Moreover, $\Lambda_n = \mathrm{C}_n^{P^j} + \mathrm{R}\mathrm{C}_n^{P^i}\mathrm{R}^\top$. Equation (11) is a quadratic program and can be computed efficiently. The alignment constraint is computed from the optimized translation vector, $\mathbf{Z}_{\text{align}} = [t_x^*, t_y^*]^\top$.

C. Data Association

The submap alignment procedure produces a relative factor between two poses. Data association refers to the step for suggesting submaps that *should* align. We propose a set of simple heuristics in order to select submaps to align:

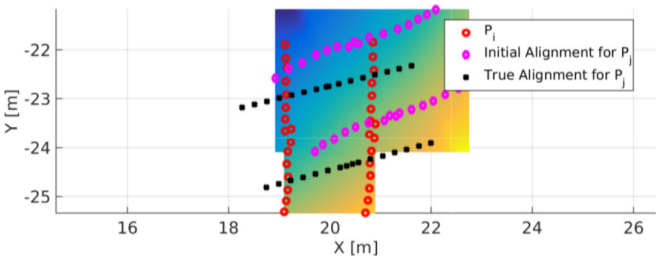
- The distance between submap centroids must be within a mahalanobis distance threshold.
- The intersecting regions of submaps must have water column depths within a threshold range of each other. Since water column depth is measured with bounded error, overlapping submaps must contain similar water column depths.
- Adjacent nodes do not produce alignments, since submaps are sparse and have small overlapping areas.

For each new node added to the graph, we select potential alignments based on the above criteria. If more than one submap passes all the criteria above, we choose the submap with the closest centroid. We then perform modified GICP as outlined in Section III-B.

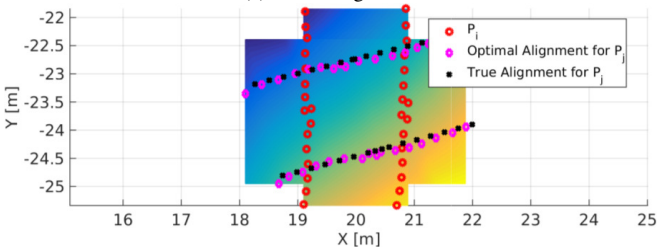
With the above criteria, incorrect alignment factors will be added to the graph. To handle erroneous data association, we use dynamic covariance scaling (DCS) [30]. DCS weakens the influence of factors that are not consistent with other measurements by dynamically increasing the factor noise covariance. Field trial results (Section IV) demonstrate that DCS is able to correctly scale problematic false submap matches.

Using the heuristics mentioned above, data association is sensitive to the quality of our pose estimation: until our first alignment, our estimated trajectory will exactly match the DR solution. If DR estimation error is large, our data association scheme will only propose erroneous alignment factors since the distance to the correct alignment will be larger than other submaps. Enforcing good alignments can be achieved by designing trajectories that contain spatial submap intersection before DR navigation error grows too large.

In the following section, we present results for field experiments with known data association and with the data association scheme mentioned above. We also show the effect of large DR navigation error on our data association heuristics.



(a) Initial alignment



(b) Optimal alignment

Fig. 3: Top view of submap alignment results in simulation. The top figure shows the initial condition alignment (pink) versus the correct alignment (black). The bottom figure shows the optimized submap alignment (pink) matching the correct alignment (black). Red is the submap of the first node.

IV. Results

A. Numerical Simulation

We first evaluate the alignment factor produced by (11) in simulation. We generated a bowl-shaped environment with water column depth varying $2\,\mathrm{m}$ over a $100\,\mathrm{m}$ area. Within the area, we randomly selected $100\,000$ overlapping submap regions and generated submap point clouds. Finally, we aligned each submap pair and compared the alignment measurement with the relative ground truth positions of the submaps.

Fig. 3 illustrates one alignment between a simulated pair of submaps. The red point cloud ($\mathbf{P}_i$) is the submap from a node $\mathbf{x}_i$, the pink point cloud ($\mathbf{P}_j$) is the estimated submap from a node $\mathbf{x}_j$, and the black point cloud is the ground truth submap of $\mathbf{x}_j$. The surfaces below the point clouds $\mathbf{P}_i$ and $\mathbf{P}_j$ represent a section near the nodes' locations in the simulated map. These sections are color-coded by depth. In Fig. 3a, it is noticeable that the surfaces do not have a smooth color transition. However, the two submaps pass our data association heuristics and so we propose an alignment between them. The results are shown in Fig. 3b. The surfaces show a smooth transition, and the pink submap lies nearly on top of the black submap.

After $100\,000$ runs, the average alignment error is an approximate Gaussian zero-mean distribution with standard deviation of $1.5\,\mathrm{m}$. Therefore, simulation results for our measurement model validate our assumption of zero-mean Gaussian noise in (7). Due to non-modeled error sources, such as a non-smooth bathymetry and heading biases, and to avoid overconfidence, we increase the standard deviation from $1.5\,\mathrm{m}$ to $4\,\mathrm{m}$ when performing submap alignment in our field trials.

Fig. 4: Ocean Server, Inc., Iver2 AUV used during field trials at the University of Michigan Biological Station. Sensors include a $600\,\mathrm{kHz}$ Teledyne RDI Explorer DVL, a Microstrain 3DM-GX25 AHRS, and a Desert Star SSP-1 pressure-depth sensor

B. AUV Field Trials

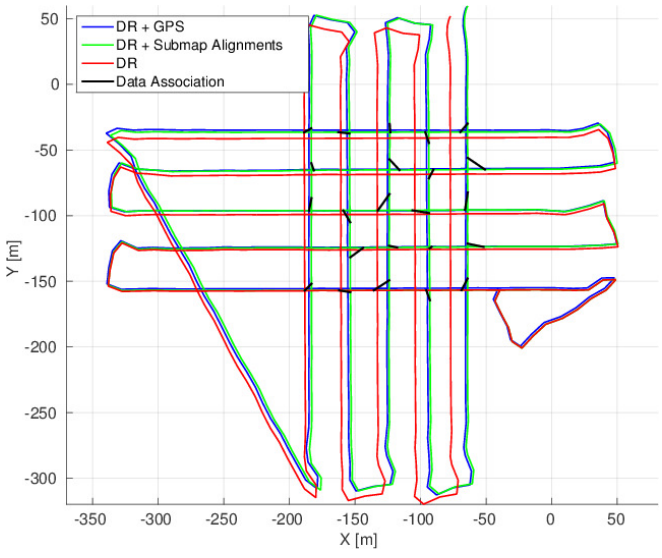
We fielded a modifed Ocean Server, Inc., Iver2 AUV (Fig. 4). The AUV is equipped with a $600\,\mathrm{kHz}$ Teledyne RDI Explorer DVL, a Microstrain 3DM-GX25 AHRS, and a Desert Star SSP-1 pressure-depth sensor. More details regarding the vehicle configuration can be found in [31].

We collected data from two field trials. For both, the AUV survey mission consisted of performing an East-West lawn mower pattern (x-direction) followed by a North-South pattern (y-direction). The difference between the trials was in their total length. One run consisted of 5 legs in each direction, while the other consisted of over 10 legs for each lawn mower. During each trial, the vehicle moved approximately $1\,\mathrm{m/s}$ at a fixed depth of $5\,\mathrm{m}$. The total water column depth over the surveyed area varied between $7\,\mathrm{m}$ and $21\,\mathrm{m}$. Each submap consisted of $10\,\mathrm{s}$ of accumulated DVL altitude observations and contained at least $1\,\mathrm{m}$ difference in water column depth.

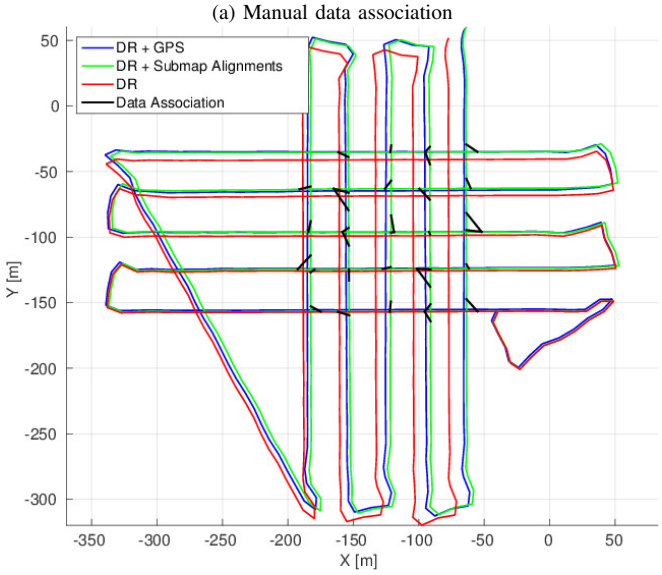
The AUV followed two overlapping lawn mower survey patterns for each trial consisting of $300\,\mathrm{m}$ tracklines spaced $25\,\mathrm{m}$ apart. The AUV surfaced at the end of each lawn mower trackline in order to observe GPS (for ground-truth evaluation). We post-processed the vehicle navigation using standard DR, and DR with our submap alignment factors. Additionally, we estimated the vehicle trajectory using GPS as a proxy for ground-truth.

1) Trial 1: Fig. 5 shows the estimated AUV trajectory for the shorter trial. We first hand-labeled aligning submaps (Fig. 5a) and then compared the solution to automatic data association (Fig. 5b). Many more links were proposed with automatic data association. As shown in Fig. 6, DCS helped to reject incorrectly proposed submap alignments by dynamically scaling the covariance of the submap factor based on the factor's influence to the chi-squared error of each submap alignment factor. The DCS scale factor varies between 1 (standard observation) and 0 (does not influence the optimized trajectory).

Fig. 7 was plotted under automatic data association, and it demonstrates the benefit of the submap aligment-based approach compared to standard DR navigation. Note that the error and covariance plots are the same for each estimator during the first half of the trial. Submap alignments are only



(a) Manual data association



(b) Automatic data association

Fig. 5: Results for Trial 1 field experiments. Red consists of the DR navigation solution, the blue trajectory is the optimized graph that consists of DR navigation with GPS factors, and the green trajectory is the optimized graph that contains DR navigation and submap alignments factors. The black lines connect nodes that share the same submap alignment.

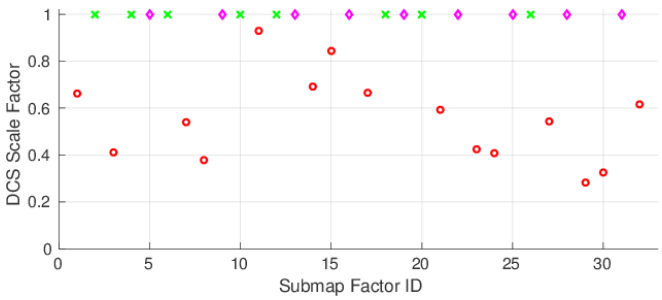
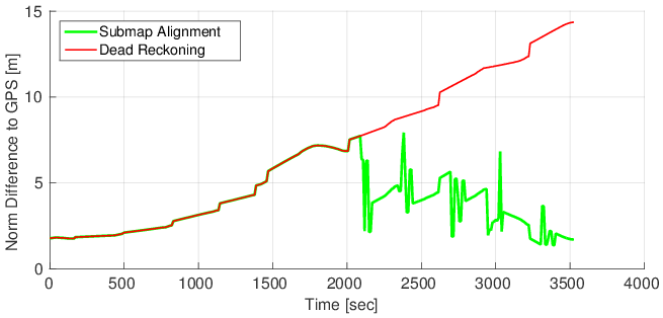
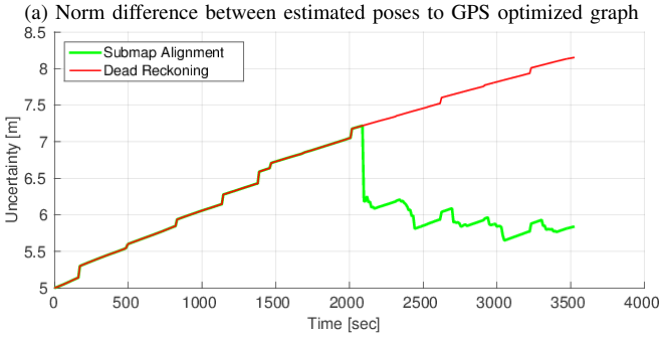


Fig. 6: Scaling factor computed by DCS for Trial 1 experiments. Green crosses are automatic alignments that are correct, magenta diamonds are proposed alignments that the spatial submap overlap is partial, and red circles are false positive submap alignments.

proposed after the AUV has executed its first lawn mower and transitions to the second lawn mower during which it



(a) Norm difference between estimated poses to GPS optimized graph



(b) Estimator uncertainty (4th root of determinant of xy covariance)

Fig. 7: Estimator comparison to GPS optimized graph for Trial 1.

overlaps the first.

2) Trial 2: The second trial demonstrates our proposed algorithm on a longer mission. We first present results with hand-labeled data association (Fig. 8a) followed by automatic data association (Fig. 8b). Fig. 8b demonstrates the failure mode for our data association heuristics. The submap technique does not propose many correct links, since DR navigational error is large when the first alignment is proposed, and our method will lead to an estimated trajectory that is overconfident and erroneous.

V. Conclusion

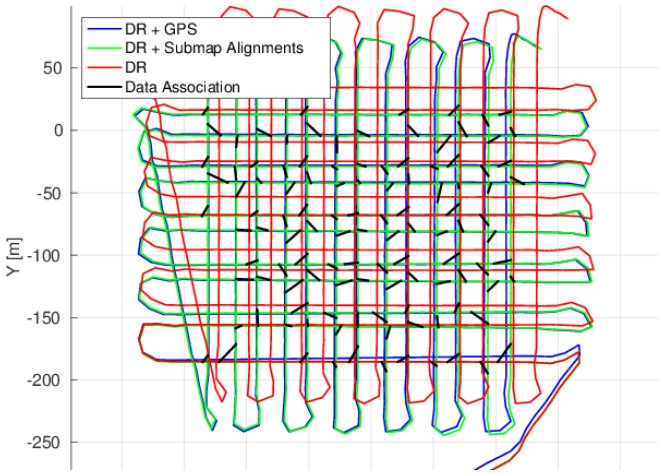
We have presented an algorithm for terrain-aided navigation without the use of a prior map. We proposed an xy submap alignment procedure derived from GICP. The submap alignment contraints are used within a factor graph framework to enforce trajectory estimates with consistent bathymetric structure. Our field results demonstrate the ability of our algorithm to produce accurate AUV position estimates using onboard vehicle sensors.

For the data association, our heuristic-based approach performed well when the first submap alignment occurs prior to a large error in DR navigation. Therefore, good survey trajectory design is required. Further, DCS is included in submap alignment factors to enforce the removal of potentially wrong data associations.

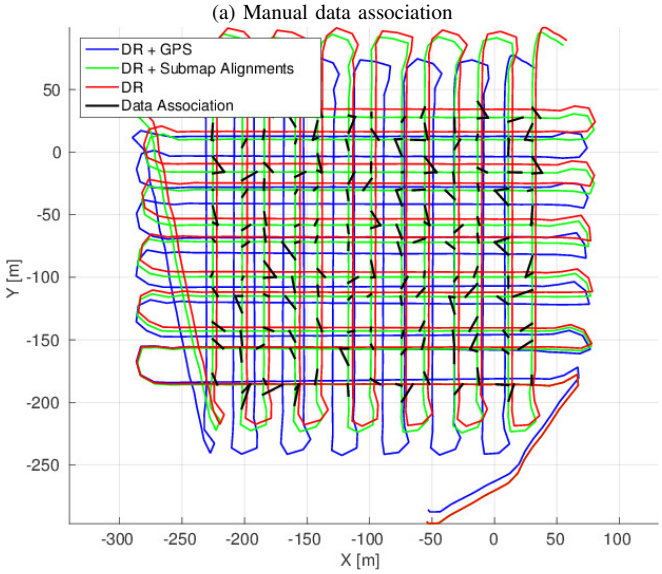
References

[1] R. M. Eustice, H. Singh, and L. L. Whitcomb, "Synchronous-clock one-way-travel-time acoustic navigation for underwater vehicles," *J. Field Robot.*, vol. 28, no. 1, pp. 121–136, 2011.

[2] L. L. Whitcomb, D. R. Yoerger, and H. Singh, "Combined doppler/LBL based navigation of underwater vehicles," in *Proc. Int. Symp. Unmanned Untethered Subm. Tech.*, Durham, NH, 1999, pp. 1–7.



(a) Manual data association



(b) Automatic data association

Fig. 8: Results for Trial 2 field experiments. Red consists of the DR navigation solution, the blue trajectory is the optimized graph that consists of DR navigation with GPS factors, and the green trajectory is the optimized graph that contains DR navigation and submap alignments factors. The black lines connect nodes that share the same submap alignment.

[3] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," in *Proc. Robot.: Sci. & Syst. Conf.*, Jun. 2009.

[4] L. Paull, S. Saeedi, M. Seto, and H. Li, "AUV navigation and localization: A review," *IEEE J. Ocean. Eng.*, vol. 39, no. 1, pp. 131–149, 2014.

[5] J. C. Kinsey, R. M. Eustice, and L. L. Whitcomb, "Underwater vehicle navigation: Recent advances and new challenges," in *IFAC Conf. Manoeuvring and Cont. Marine Craft*, Lisbon, Portugal, Sep. 2006.

[6] M. F. Fallon, M. Kaess, H. Johannsson, and J. J. Leonard, "Efficient AUV navigation fusing acoustic ranging and side-scan sonar," in *Proc. IEEE Int. Conf. Robot. and Automation*, Shanghai, China, May 2011, pp. 2398–2405.

[7] R. M. Eustice and H. Singh, "Visually mapping the RMS Titanic: Conservative covariance estimates for SLAM information filters," *Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1223–1242, 2006.

[8] A. Kim and R. M. Eustice, "Active visual SLAM for robotic area coverage: Theory and experiment," *Int. J. Robot. Res.*, vol. 34, no. 4-5, pp. 457–475, 2015.

[9] D. E. Di Massa and W. Stewart, "Terrain-relative navigation for autonomous underwater vehicles," in *Proc. IEEE/MTS OCEANS Conf. Exhib.*, vol. 1, Halifax, Canada, Oct. 1997, pp. 541–546.

[10] L. Kulander, "Development of a terrain navigation system for AUVs," in *Proc. Unmanned Untethered Submersible Technology*, Durham, NH, USA, Jun. 1989, pp. 494–501.

[11] I. Nygren and M. Jansson, "Terrain navigation for underwater vehicles using the correlator method," *IEEE J. Ocean. Eng.*, vol. 29, no. 3, pp. 906–915, 2004.

[12] R. Karlsson, F. Gusfafsson, and T. Karlsson, "Particle filtering and cramer-rao lower bound for underwater navigation," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 6, Apr. 2003, pp. VI–65.

[13] N. Bergman, L. Ljung, and F. Gustafsson, "Terrain navigation using bayesian statistics," *Control Systems, IEEE*, vol. 19, no. 3, pp. 33–40, 1999.

[14] S. B. Williams and I. Mahon, "A terrain-aided tracking algorithm for marine systems," in *Proc. Int. Conf. Field Service Robot.*, ser. Springer Tracts in Advanced Robotics, 2006, pp. 93–102.

[15] B. Claus and R. Bachmayer, "Terrain-aided navigation for an underwater glider," *J. Field Robot.*, 2015.

[16] P. Kimball and S. Rock, "Sonar-based iceberg-relative AUV localization," in *Proc. IEEE/OES Autonomous Underwater Vehicles Conf.*, 2008, pp. 1–6.

[17] ——, "Mapping of translating, rotating icebergs with an AUV," *IEEE J. Ocean. Eng.*, vol. 40, no. 1, pp. 196–208, 2015.

[18] R. Eustice, R. Camilli, and H. Singh, "Towards bathymetry-optimized doppler re-navigation for AUVs," in *Proc. IEEE/MTS OCEANS Conf. Exhib.*, Washington, D.C., May 2005.

[19] 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.

[20] A. Kim and R. M. Eustice, "Real-time visual SLAM for autonomous underwater hull inspection using visual saliency," *IEEE Trans. Robot.*, vol. 29, no. 3, pp. 719–733, 2013.

[21] S. Barkby, S. Williams, O. Pizarro, and M. Jakuba, "Bathymetric SLAM with no map overlap using gaussian processes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, 2011, pp. 1242–1248.

[22] C. Roman and H. Singh, "A self-consistent bathymetric mapping algorithm," *J. Field Robot.*, vol. 24, no. 1-2, pp. 23–50, 2007.

[23] P. Ozog, N. Carlevaris-Bianco, A. Kim, and R. M. Eustice, "Long-term mapping techniques for ship hull inspection and surveillance using an autonomous underwater vehicle," *J. Field Robot.*, 2015, in Press.

[24] M. VanMiddlesworth, M. Kaess, F. Hover, and J. J. Leonard, "Mapping 3d underwater environments with smoothed submaps," in *Field and Service Robotics*. Springer, 2015, pp. 17–30.

[25] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robot.*, vol. 24, no. 6, pp. 1365–1378, 2008.

[26] F. Dellaert and M. Kaess, "Square root SAM," *Int. J. Robot. Res.*, vol. 25, no. 12, pp. 1181–1204, 2006.

[27] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Robotics-DL tentative*. International Society for Optics and Photonics, 1992, pp. 586–606.

[28] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," in *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*. IEEE, 1991, pp. 2724–2729.

[29] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous robot vehicles*, I. J. Cox and G. T. Wilfong, Eds. New York, NY, USA: Springer-Verlag New York, Inc., 1990, pp. 167–193.

[30] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *Proc. IEEE Int. Conf. Robot. and Automation*, May 2013.

[31] H. C. Brown, A. Kim, and R. M. Eustice, "An overview of autonomous underwater vehicle research and testbed at PeRL," *Marine Tech. Soc. J.*, vol. 43, no. 2, pp. 33–47, 2009.