Degeneracy-Aware Factors with Applications to Underwater SLAM

Akshay Hinduja[1], Bing-Jui Ho[3], and Michael Kaess[2]

***Abstract*— Simultaneous Localization and Mapping (SLAM) is commonly formulated as an optimization over a graph. A popular approach is the pose graph, which seeks to solve for robots poses that are constrained by pose-to-pose measurements, such as odometry measurements or loop closures. For range sensors, these pose-to-pose constraints can be achieved by performing scan matching techniques, such as Iterative Closest Point (ICP). However, in environments with insufficient or degenerate geometric features, the ICP solution can be unreliable and lead to significant drift in the trajectory of the graph optimization solution.**

In this paper, we propose a degeneracy-aware approach which has two stages: (1) a degeneracy-aware ICP algorithm and (2) a partially constrained loop closure factor to incorporate the results from (1) into the SLAM pose graph optimization. Our approach performs updates and optimizes both ICP and the pose graph in only the well constrained directions of the state space. These directions are selected on the basis of a dynamic threshold, which updates at each iteration. We apply the proposed algorithm to autonomous underwater mapping with sonar. To evaluate the performance of this algorithm, we conduct experiments in both simulation and real world scenarios, and show the method's robustness to navigational drift and ability to reject poor loop closures in degenerate environments, which would otherwise degrade the accuracy of the trajectory and the quality of the resulting map.

I. Introduction and Related Work

SLAM systems utilizing laser range finders or RGB-D sensors have become commonplace over the past decade. A multitude of algorithms have been developed using these sensors to provide accurate state estimates and consistent maps in real-time [20, 21, 23], without the need for absolute measurements from sensors such as a GPS. However, most of these methods are prone to failure in certain scenarios. One particular failure mode is when the scene in consideration has degenerate geometry or insufficient features for the method to get correspondences. Cadena et al. [1] note that the absence of observable features, even temporarily, can hinder the SLAM solution and necessitates the need for techniques to deal with the lack of observability.

A variety of methods have been proposed to combat degeneracy in SLAM. We can broadly categorize them as either (1) limiting or (2) actively handling degenerate situations. Limiting the occurrence of degenerate situations can be achieved by effectively supplementing the system with more information to minimize failure. Hsiao et al. [7] use an IMU in their planar SLAM framework to maintain the sensor pose for short durations when faced with texture-less scenes. In their point-plane SLAM system, Taguchi et al. [16] analyze the possible correspondences found to satisfy certain properties, and reject the pairs likely to be degenerate based on those properties. Tribou et al. [18] describes the use of multiple cameras with different fields of view such that at a given time at least one camera's observation can be used for localizing the system effectively, even if in a visually poor environment. Other ways to limit degeneracy could be by switching to different features or methods. For the mapping of texture-less pipe systems, Ma et al. [11] substitutes visual features with voids detected via ultrasonic scanners as landmarks, effectively avoiding degeneracies which would occur through visual measurements. Hu et al. [8] switch between feature matching methods for RGB-D and RGB to deal with different environments during mapping.

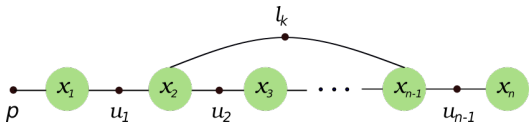


Fig. 1: Pose graph representation of a SLAM problem: Green circles represent the pose nodes x_k to be estimated. The small black dots represent measurement factors (odometry u_i or loop closures l_k) that are connected through edges to one or more pose nodes. The prior p constrains the first pose to remove a gauge freedom.

Actively handling degeneracy in a SLAM system requires analyzing the constraints imposed by measurements, identifying the degeneracies, and appropriately compensating for them. Pathak et al. [12] describe a novel method of scan matching using planes, where they determine and minimize the geometric uncertainty of the configuration space. Cho et al. [3] build upon Pathak et al.'s method and propose the detection of degeneracy between two sets of plane features by comparing the ratios of eigenvalues in the second moment matrix. They then project the value of the IMU recorded orientations in the directions estimated to be degenerate, which is then used to optimize their state estimate. Rong et al. [13] use the eigenvalues of the Empirical Observability Gramian to represent the observability of the system parameters. They then use the local condition number to determine the degeneracy online, and detect future motions which may result in degenerate cases. As Zhang et al. [22] define, an optimization problem in SLAM is degenerate if there exist one or more directions in the state space of the optimization which are not well constrained. However, even if some directions in the optimization are degenerate, the well constrained directions can still be used to update in a subspace of the original optimization. They separate the degenerate and non-degenerate directions and introduce a technique called solution remapping to perform incremental

The authors are with the Department of Mechanical Engineering[1], the Robotics Institute[2], Carnegie Mellon University, Pittsburgh, PA 15213, USA and Aptiv AMoD[3], Pittsburgh, PA 15238. {akshayh, kaess}@cmu.edu, bing-jui.ho@aptiv.com

This work was partially supported by the Office of Naval Research under awards N00014-16-1-2103 and N00014-18-1-2843.

updates only in the non-degenerate directions. To solve for the occurrence of degeneracy, our contributions presented in this paper are as follows:

1) A degeneracy-aware point-to-plane ICP algorithm, based on the solution remapping technique [22].
2) A new degeneracy-aware partial factor, which allows for the optimization of the pose graph only in the well constrained directions, by using the results from the degeneracy-aware ICP.

II. Problem Statement

We will now introduce the non-linear least squares optimization that is used to solve the maximum a posteriori (MAP) formulation of the SLAM problem. MAP estimation tries to achieve the most likely state $\mathbf{x}$ of a system given a set of measurements $\mathbf{z}$:

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmax}}\ p(\mathbf{x}|\mathbf{z}) \tag{1}$$

$$= \underset{\mathbf{x}}{\operatorname{argmax}}\ p(\mathbf{x})\, p(\mathbf{z}|\mathbf{x}) \tag{2}$$

$$= \underset{\mathbf{x}}{\operatorname{argmax}}\ p(\mathbf{x}) \prod_{i=1}^{N} p(\mathbf{z}_i|\mathbf{x}) \tag{3}$$

The likelihood term $p(\mathbf{z}_i|\mathbf{x})$ arises from a generative model for the measurement $\mathbf{z}_i$ with assumed additive Gaussian noise so that $p(\mathbf{z}_i|\mathbf{x}) = \mathcal{N}(\mathbf{h}_i(\mathbf{x}), \mathbf{\Sigma}_i)$. Here $\mathbf{h}_i(\mathbf{x})$ predicts the measurement $\mathbf{z}_i$ based on a state estimate $\mathbf{x}$. MAP inference under Gaussian noise is equivalent to solving a non-linear least-squares optimization [4]

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} -\log \prod_{i=1}^{N} p(\mathbf{z}_i|\mathbf{x}) \tag{4}$$

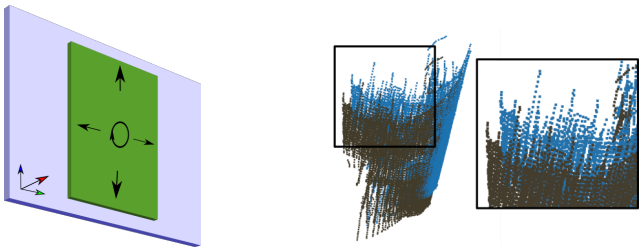
$$= \underset{\mathbf{x}}{\operatorname{argmin}} \sum_{i=1}^{N} \|\mathbf{h}_i(\mathbf{x}) - \mathbf{z}_i\|^2_{\mathbf{\Sigma}_i} \tag{5}$$

The prior $p(\mathbf{x})$ can be dropped on the assumption that we have no prior information about the state.

This optimization can be represented in graphical form using a pose graph or factor graph [4]. Pose graphs are a special form of factor graphs where all variable nodes are robot poses, and the factors represent pose-to-pose constraints. A typical pose graph is shown in Fig. 1 with robot poses $x_1, ..., x_n$. Between adjacent poses we have pose-to-pose odometry constraints, $u_1, ..., u_{n-1}$. Loop closure constraints l_k connect two arbitrary poses. The variables nodes in the graph are estimated using nonlinear least-squares using Gauss-Newton or a similar algorithm.

Equation 5 gives us the general nonlinear least squares form of the pose graph optimization. We now represent an expanded version of the nonlinear least squares formulation for the SLAM system represented in Fig. 1.

We assume a motion model $\mathbf{f}(\mathbf{x}, \mathbf{u})$, where $\mathbf{u}_i = u_1, u_2, ..., u_n$ are the added odometry factors between poses x_{i+1} and x_i, along with a measurement model for loop closures $\mathbf{g}(x_i, x_j)$. Then, for a sequence of poses $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$, the MAP estimate can be written as:



(a) (b)

Fig. 2: Examples of degeneracy:- (a) Planar registration: The smaller plane while constrained in X (into the plane), pitch and yaw, is unconstrained in the Y, Z and roll directions. (b) Degeneracy in mapping simulated pilings: The optimization is unconstrained in the vertical direction as well as along the curved surface, resulting in an incorrect registration when using point-to-plane ICP.

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} \Big(\sum_{i=1}^{N} \|\mathbf{f}_{i,i+1}(\mathbf{x}_i, \mathbf{x}_{i+1}) - \mathbf{u}_i\|^2_{\mathbf{\Sigma}_i} + \sum_{i=1}^{N} \|\mathbf{g}_i(x_i, x_j) - \mathbf{l}_{ij}\|^2_{\mathbf{\Sigma}_{ij}}\Big) \tag{6}$$

To acquire a loop closure constraint $\mathbf{l}_{ij}$ between poses x_i and x_j, a popular method for measurements from range scanners and RGB-D sensors is the point-to-plane ICP algorithm [2]. Here we introduce the linear least-squares optimization for point-to-plane ICP as described by Low [10].

Let $\mathbf{s}_i = (s_{ix}, s_{iy}, s_{iz}, 1)^\top$ and $\mathbf{d}_i = (d_{ix}, d_{iy}, d_{iz}, 1)^\top$ represent the source and their corresponding destination points. $\mathbf{n}_i = (n_{ix}, n_{iy}, n_{iz}, 0)^\top$ is the unit normal vector at $\mathbf{d}_i$. The resulting error metric can be represented as follows

$$T_{opt} = \underset{T}{\operatorname{argmin}} \sum ((T \cdot \mathbf{s}_i - \mathbf{d}_i) \cdot \mathbf{n}_i)^2 \tag{7}$$

where T is the 3D rigid body transformation for transforming the source points such that the total error is minimized. As derived in [10], on further simplification our original metric can be re-arranged into a matrix expression as

$$\mathbf{x}_{opt} = \underset{\mathbf{x}}{\operatorname{argmin}} |A\mathbf{x} - \mathbf{b}|^2 \tag{8}$$

Here, A is our covariance matrix obtained from our source points and the corresponding normals, and $\mathbf{x}$ is the 6×1 vector of the unknown transformation.

In controlled environments, coupled with only small increments in orientation, point-to-plane ICP converges well when a unique global minimum exists and thus provides an optimal solution. However, complications arise when the optimization might get trapped in a local minimum. This is common in cases of degeneracy due to planar and textureless environments, in which all 6 degrees of freedom (DoF) are not constrained. In these situations the registration either fails completely and aborts, or even worse, gives an incorrect solution in some directions. Visual examples of such degeneracies are shown in Fig. 2.

Using the background provided by Equations 8 and 5, Section III describes the formulation of the degeneracy-awareICP and partial loop closure factor, which help in avoiding the situations described above.

Algorithm 1 Degeneracy-aware ICP

Input: Initial relative transformation from odometry T_{odom}
Output: Relative ICP transformation T_{icp}

$T_{icp} \leftarrow T_{odom}$
while nonlinear operations **do**
  Linearize the optimization problem at T_{icp} to get $A^\top A$
  Compute eigenvalue λ_i and eigenvector v_i of $A^\top A$ for $i = 1...6$
  Determine an eigenvalue threshold λ_t
  Construct matrix V containing all the eigenvectors
  Construct matrix V_f containing only well conditioned directions based on λ_t
  Compute update $\Delta x_f \leftarrow (A^\top A)^{-1} A^\top b$
  $T_{icp} \leftarrow T_{icp} + V^{-1} V_f \Delta x_f$
end while
return T_{icp}

III. Approach

A. Degeneracy-Aware ICP and Solution Remapping

As mentioned earlier in Section II, the point-to-plane ICP can provide an incorrect registration due to degeneracy, which can result in a very poor trajectory estimate when the loop closure is added to the pose graph. To combat these errors, we need to detect the degenerate, or poorly constrained, directions of the state space, and make use of only the well-constrained directions in the optimization-step for ICP. Zhang et al. [22] show how to iteratively solve a nonlinear system by performing updates only in the well constrained directions. After determining the values of eigenvalues λ_i and eigenvectors v_i from the A matrix of a linearized system, the method of Zhang et al. constructs three matrices V_d, V_f and V. The three matrices are constructed such that V_d contains the degenerate direction's eigenvectors, V_f the fully constrained direction's eigenvectors and V the complete set of eigenvectors. This division is based on the comparison of the individual eigenvalues λ_i against a common threshold λ_t. The linearized system as usual is solved as $\mathbf{x}_f = (A^\top A)^{-1} A^\top b$. Then, this standard solution is projected onto the space of well-constrained directions. This incremental update is added to the state estimate, and the procedure repeats at the new linearization point.

In our degeneracy-aware ICP system, we only use the well conditioned directions to update the optimization solution, and ignore the directions determined as degenerate. The degeneracy-aware ICP algorithm is described in pseudo-code in Algorithm 1.

The degeneracy-aware ICP was implemented by extending the Point Cloud Library's [14] point-to-plane ICP method. The linearization of the point-to-plane error metric to obtain $A^\top A$ and $A^\top b$ was achieved by the method described in [10]. The fundamental difference between the solution remapping technique presented by [22] and our proposed degeneracy-aware ICP is that at each iteration we re-compute the well conditioned directions on the basis of a different covariance matrix ($A^\top A$). By doing so we acknowledge the fact that at each iteration where we linearize at an updated T_{icp}, the degenerate directions might differ.

B. Thresholding

To determine which directions are degenerate, we need to set a threshold λ_t to be compared to the eigenvalues we receive from the degeneracy-aware ICP. Zhang et al. [22] use a sample dataset with predetermined degenerate and non degenerate sections, and determine their threshold based on the mid point of the minimum eigenvalue distribution they get. However this method is not suitable for real-time evaluation. Cho et al. [3] compare the ratios of eigenvalues amongst the different directions and set up a ranked list, using their sensor properties to determine their fixed threshold. Similar to [13] we use the condition number, i.e. the ratio between the maximum and minimum eigenvalues, λ_{max} and λ_{min}, for each iteration as our eigenvalue threshold for the loop closure candidates. The benefit of this threshold is largely seen in highly degenerate cases, where λ_{min} might be an extremely small value, along with other directions, thereby making λ_t much larger. This automatically rejects that constraint on account of it being unreliable.

C. Partial Loop Closure Factor

After obtaining the matrix of fully constrained eigenvectors V_f through the degeneracy-aware ICP, we incorporate this result into the pose graph formulation with our proposed partial loop closure factor. Given a measurement model for loop closures $\mathbf{g}(x_i, x_k)$, and referring to our least squares approach to SLAM optimization in Section II, we can use the matrix V_f as follows:

$$\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} \sum_{(i,k)\epsilon L} \left\| V_f \left(\mathbf{g}(x_i, x_k) - \mathbf{l}_{ik} \right) \right\|^2_{V_f \Lambda_{ik} V_f^\top} \quad (9)$$

where L is the set of all tuples (i, k) for which we consider loop closure constraints, and Λ_{ik} is the covariance for the registration between x_i and x_k. This formulation of the measurement model enables the generation of a partial factor for the well constrained directions of the tuple (i, k). The covariance for loop closure constraints, Λ_{ik}, is updated at each iteration as $V_f \Lambda_{ik} V_f^\top$ to reflect the selected well constrained directions. Upon incorporating our partial factor, we can perform MAP inference on the resulting pose graph as to determine the value of unknown poses x_i that maximally adhere to the information given the uncertain measurements.

IV. Application and Evaluation

We apply our degeneracy-aware framework to underwater mapping using multi-beam sonar. To evaluate the performance of the degeneracy-aware loop closures, we use a volumetric submap-based pose graph approach as described in [17], where we implement the degeneracy-aware ICP formulation through solution remapping, and incorporate the partial factor for loop closures. This system is typically used for the inspection of ship hulls and structural pilings at harbors. A description of the system is presented in the following subsections.

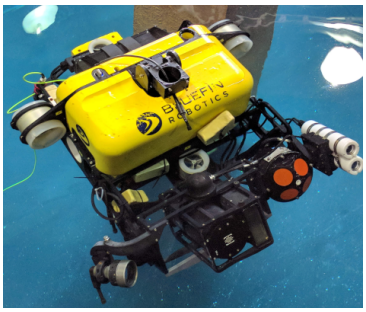


Fig. 3: Bluefin Hovering Autonomous Underwater Vehicle (HAUV)

A. Platform

Our experiments were performed using the Hovering Autonomous Underwater Vehicle (HAUV) [5]. The HAUV is equipped with five thrusters enabling control of all degrees of freedom barring roll and pitch. For on-board navigation, the HAUV houses a 1.2MHz Teledyne/RDI Workhorse Navigator Doppler velocity log (DVL), a Paroscientific Digiquartz depth sensor, and an attitude and heading reference system (AHRS) fitted with a Honeywell HG1700 IMU. The onboard computer of the HAUV is capable of providing odometry estimates by fusing readings from these sensors. The depth sensor, and AHRS are capable of providing highly accurate and drift free measurements of Z, pitch, and roll. However, in comparison the measurements for X, Y and yaw rotation are subject to drift as they are estimated through dead reckoning from the DVL and AHRS readings.

For the purpose of mapping, we utilize a dual-frequency identification sonar (DIDSON) [15]. This sonar has 96 beams, using a 2D transducer array. We modify it for use in "profiling" mode by fitting a concentrator lens which reduces its vertical field of view to 1°. Similar to [6, 17], we assimilate a fixed number of these sonar profile scans to form a single submap, which is used as the basis of our mapping system.

B. Pose Graph Formulation

For each time instance t_i, we have a new pose estimate x_i and sonar scan S_i available. The odometry between two subsequent scans is given by the difference between their poses

$$u_{i,i+1} = x_{i+1} \ominus x_i \tag{10}$$

where $\ominus$ expresses the first pose in the local coordinate frame of the second pose.

As shown in Fig. 4, our factor graph is formulated such that each subsequent pose x_i, $x_{i+1}, \ldots$ is represented as a node in the graph, and the odometry constraint between consecutive poses, $u_{i,i+1}$, is a factor that is connected to these pose nodes. The second row depicts the submap formulation where we combine a fixed number of scans together to form a submap. Each individual scan is registered to the coordinate frame for the first scan and is referred to as the base pose of submap m_i. The base poses for consecutive submaps m_i and m_k are connected via a factor representing the accumulated odometry constraints for submap m_i, $u_{i,k}$. The third row shows our degeneracy-aware loop closure factor $l_{i,j}$ that can

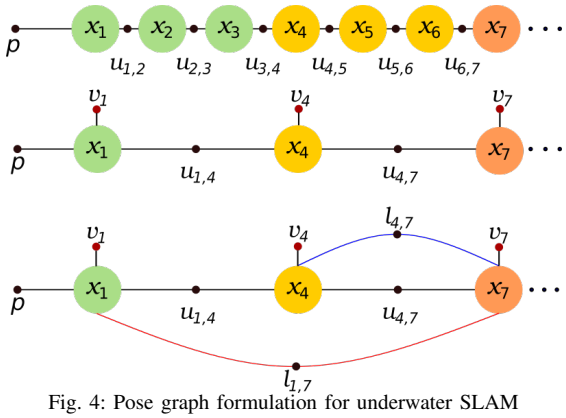


Fig. 4: Pose graph formulation for underwater SLAM

be added between any two submaps satisfying the necessary conditions [17].

We use an Euler-angle representation for a pose x_i as a vector $\left[\phi_{x_i}, \theta_{x_i}, \psi_{x_i}, t^x_{x_i}, t^y_{x_i}, t^z_{x_i}\right]^\top$, where $t^x_{x_i}, t^y_{x_i}$ and $t^z_{x_i}$ represent the translation in $X, Y$and Z, and ϕ_{x_i}, θ_{x_i}, and ψ_{x_i} are the heading, pitch, and roll angles respectively. Our odometry factors are split into $u_{i,i+1}$ and v_i. $u_{i,i+1}$ is a 3 degree of freedom (DoF) XYH factor, represented as $u_{i,i+1} = [t^x_{u_{i,i+1}}, t^y_{u_{i,i+1}}, \phi_{u_{i,i+1}}]$. v_i is a unary ZPR constraint linked to the base pose of the submap m_i, represented as $v_i = [t^z_{v_i}, \theta_{v_i}, \psi_{v_i}]$. A more detailed representation of the XYH and ZPR factors can be found in [19]. We use the iSAM library [9] for our factor graph framework optimization.

1) Odometry Prior: As previously mentioned in Section IV-A, the HAUV's AHRS and depth sensor give highly accurate and drift free measurements. To take further advantage of this information, we take the readings for depth $t^z_{x_i}$, pitch θ, and roll ϕ and improve the performance of the 6 DoF ICP algorithm by biasing the transform with a priori odometry information.

Section II introduced the linearization of the ICP transformation T_{icp} between two poses x_i and x_j, which can be represented as an over determined system $A\mathbf{x} = \mathbf{b}$.

We use $\mathbf{\Delta}$ to represent the vector difference between the depth z, pitch θ, and roll ϕ of the two poses. Thus $\mathbf{\Delta} = (t^z_{x_j} - t^z_{x_i}, \theta_{x_j} - \theta_{x_i}, \phi_{x_j} - \phi_{x_i})$. Let $\boldsymbol{X}_{pred}$ denote predicted measurements for the same $\mathbf{X}_{pred} = (t^z_{pred}, \theta_{pred}, \phi_{pred})$.

To add the prior odometry information in this case is equivalent to adding more constraints to the original matrix A and vector $\mathbf{b}$ from our linearized system. Let the matrix P and vector $\mathbf{d}$ be used to encode the prior constraints. Through whitening we simplify to get [4]:

$$P = \Sigma^{-1/2}\mathbf{X}^\top_{pred} \tag{11}$$

$$\mathbf{d} = \Sigma^{-1/2}\mathbf{\Delta}^\top \tag{12}$$

Thus, our system can now be represented as:

$$\begin{pmatrix} A \\ P \end{pmatrix} \mathbf{x} = \begin{pmatrix} \mathbf{b} \\ \mathbf{d} \end{pmatrix} \tag{13}$$

Upon solving this system, we get a 6 DoF ICP transformation estimation biased towards our odometry prior.

2) Partial Factor for Loop Closure: Consider two poses x_i and x_j with overlapping submaps which satisfy the conditions for a loop closure to be triggered. The relative transformation between them through odometry measurements can be represented as $l_{i,j}^{odom} = x_i \ominus x_j$. Let the resulting measurement from point-to-plane ICP (with solution remapping) be denoted as $l_{i,j}^{icp}$. As explained in Section III-A, we can find the eigenvalues and eigenvectors of the resulting of $A^\top A$ matrix. Let V be the resulting matrix of eigenvectors for all 6 DoF, and V_f the well constrained matrix. To prevent double-counting due to the added odometry prior in the ZPR directions, we introduce a constant matrix C, such that the diagonal for X, Y and yaw are 1, and the rest are 0. Referring to Section III-C, our final loop closure measurement model can be represented as:

$$\mathbf{g}(x_i, x_j) = \left\| V_f(C[(x_i \ominus x_j) \ominus l_{i,j}^{icp}]) \right\|^2_{V_f \Lambda_{ij} V_f^\top} \tag{14}$$

Our factor is constrained to be a maximum of 3 DoF for the X, Y and yaw directions on account of the matrix C.

C. Simulated Datasets

To evaluate the performance of the degeneracy-aware ICP and loop closure factor, we compare the maps generated by: (1) odometry only, (2) point-to-plane ICP with odometry prior (PTP-OP) and full loop closure factor, and (3) our proposed degeneracy-aware ICP with the partial loop closure factor, supplemented with the odometry prior.

The two datasets we used are the simulated pilings and propeller. The pilings dataset in particular is an extreme example of degeneracy given the nature of the loops made around the pilings and submaps formed by the system. We model the motion of our simulated robot to mimic the real HAUV. We also add additional Gaussian noise to the odometry to compare how well the accumulated navigational drift can be corrected. Fig. 5 shows the top down view of the simulated pilings reconstruction. Here, the reconstruction in black corresponds to the ground truth, and the SLAM solution is shown in a progressive color scheme from blue to yellow, signifying increasing error.

Fig. 6 show the maps generated for the propeller dataset. The Root Mean Square Error (RMSE) comparisons for the two datasets is shown in Table I. The results for our proposed algorithm show a marked improvement over the standard point-to-plane ICP formulation, which tends to substantially drift from the ground truth. For comparison, the PTP-OP results had 34 loop closures, whereas our proposed algorithm only accepted 9 for the simulated pilings dataset. Similarly, for the simulated propeller we see 45 loop closures for PTP-OP and 17 for the proposed algorithm. This is expected on account of the threshold we set, which rejects loop closures with very small λ_{min} values. Both algorithms were provided with identical parameters.

D. Real World Datasets

Our algorithm was also tested for real world datasets induced with additional Gaussian noise. We compare our results with an odometry only solution as well as results from Teixeira's [17] original submap SLAM formulation.

TABLE I: RMSE values for simulated datasets. The proposed algorithm gives better results over the standard point-to-plane ICP formulation, as well as considerable improvements over the odometry solution.

Dataset	Odometry	PTP-OP	Degeneracy-aware
Pilings	0.183	0.618	**0.173**
Propeller	0.346	0.429	**0.168**

We first test against a similar dataset used by Teixeira which is the running gear (rudder + propeller) of the *SS Curtis*. We use the odometry only solution, without added noise, as the reference solution for each of the three SLAM solutions which are under additional noise. Fig. 7 shows the three results. The reference point cloud is in gray and the 3 solutions are represented by a progressive color scheme ranging from blue to yellow, representing increasing error from the reference, similar to the simulated datsets.

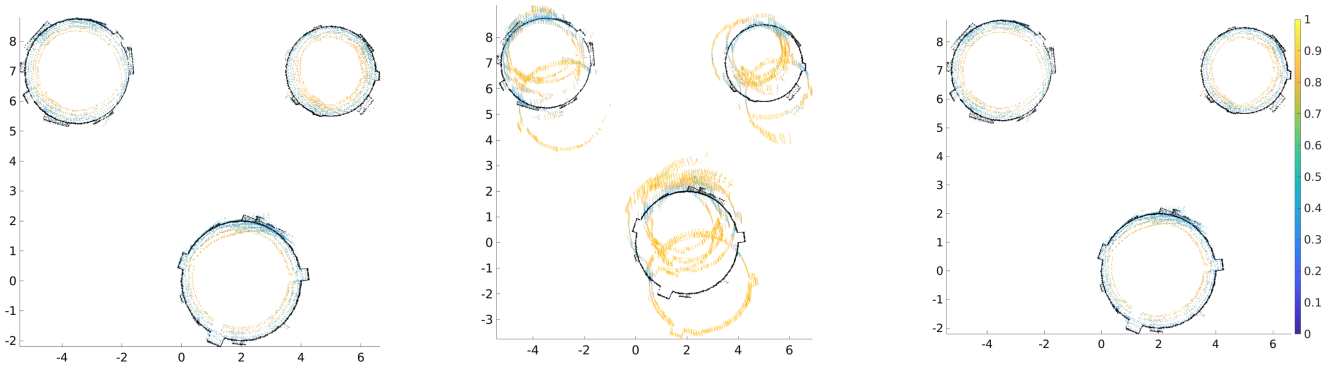
The second dataset is of pilings from the pier. Fig. 8 depicts the environment of the dataset. Each piling is identical and rotational motion of the HAUV to navigate around them increases the chances of degenerate observations. The visual results are shown in Fig. 9. The same color scheme as before applies here as well. The original algorithm by Teixeira et al. was specifically formulated for the mapping of ship hulls which are continuous surfaces that are locally mostly planar, explaining the poor result for the pilings. Our method shows its versatility for the mapping of different types of structures, giving consistently better results.

Fig. 8: Environment in which the pilings dataset was created

V. Conclusion

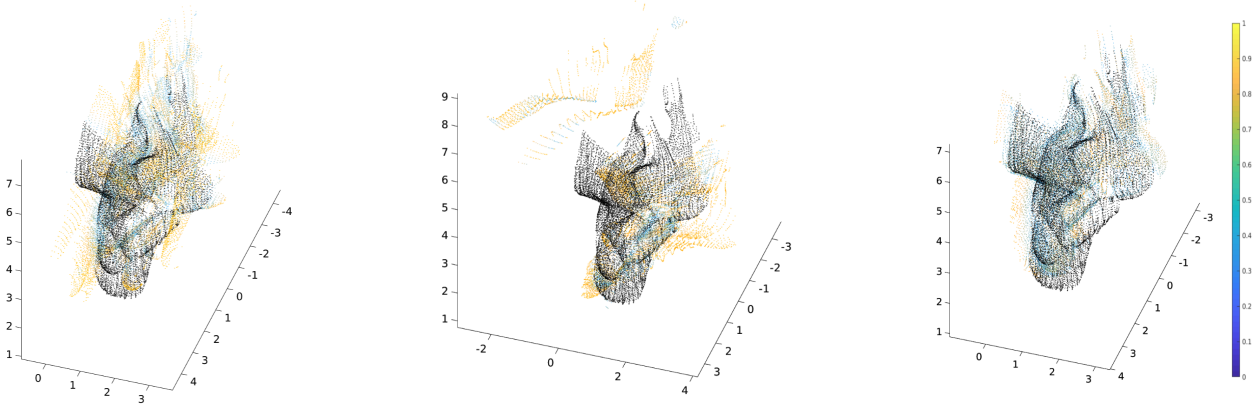
To create a pose graph optimization robust to degeneracies in the sensor data, we present a degeneracy-aware ICP algorithm for loop closure registration and show how to create partial factors that only incorporate well-constrained dimensions. For evaluation we use an underwater volumetric submap SLAM framework by integrating the degeneracy-aware ICP and loop closure formulation. We demonstrate the ability and robustness of our approach to correct for navigational drift by showing improvements over standard point-to-plane ICP for loop closure registration.

In future work, automatic selection of the threshold λ_t should be investigated. Our experiments have shown that different types of environments benefit from different threshold values, indicating the need to calculate the threshold from properties of the sensor data.



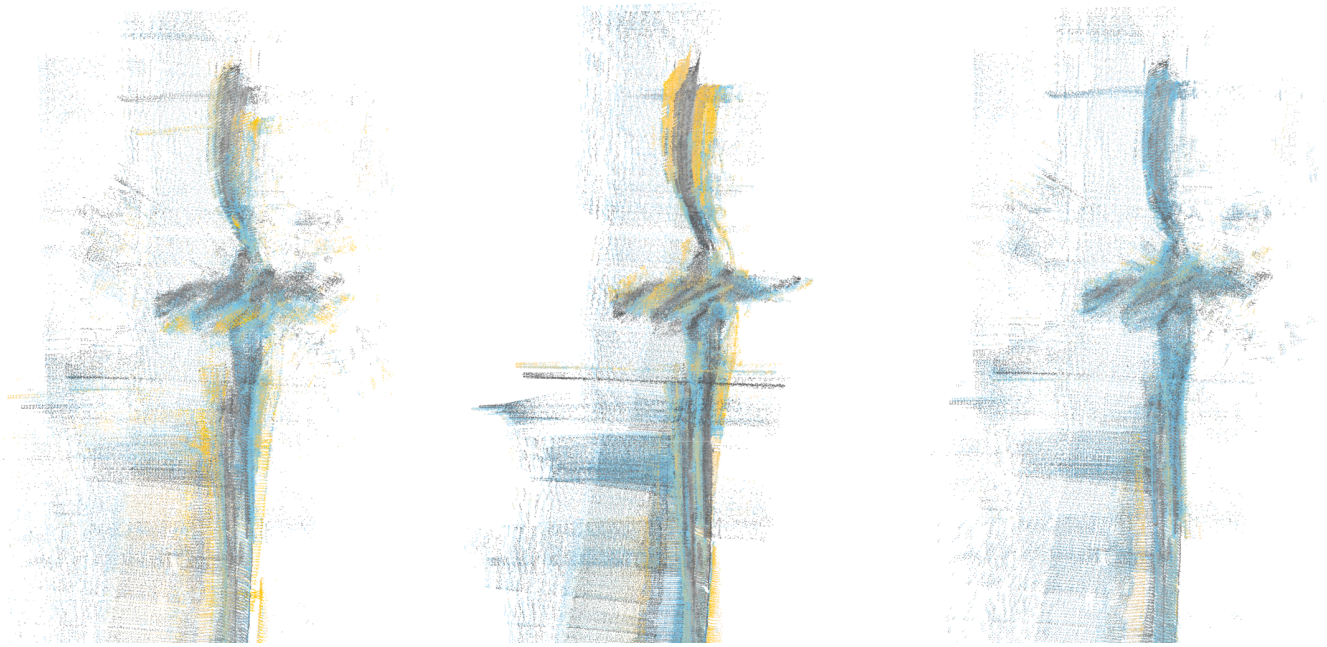
(a) Odometry solution (b) PTP-OP solution (c) Proposed solution

Fig. 5: Results for simulated pilings, top down view (RMSE)



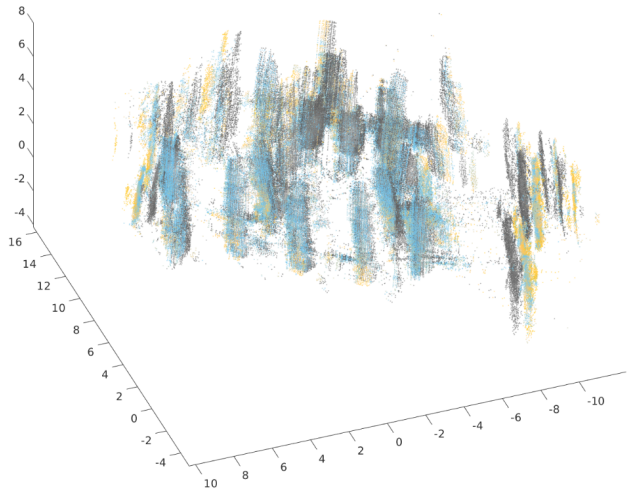
(a) Odometry solution (b) PTP-OP solution (c) Proposed solution

Fig. 6: Results for simulated propeller (RMSE)

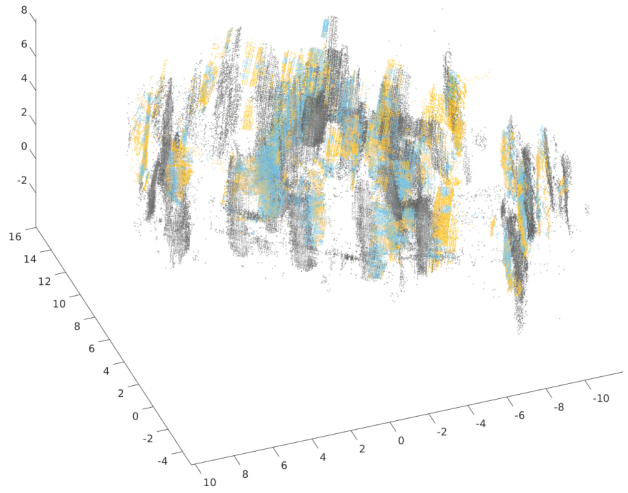


(a) Odometry only solution (b) Texeira et al. solution (c) Proposed solution

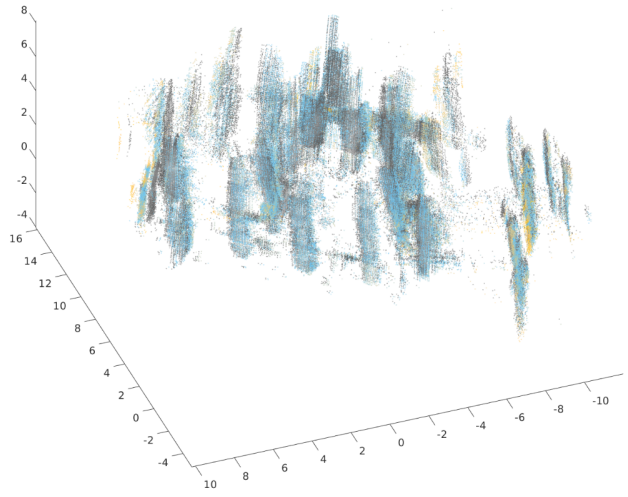
Fig. 7: Results for running gear dataset, bottom up view



(a) Odometry solution



(b) Teixeira et al. solution



(c) Proposed solution

Fig. 9: Results for pilings dataset

REFERENCES

[1] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," in *IEEE Trans. Robotics*, Dec. 2016, pp. 1309–1332.

[2] Y. Chen and G. Medioni, "Object modeling by registration of multiple range images," *Image Vision Comput.*, vol. 10, pp. 145–155, Jan. 1992.

[3] H. Cho, S. Yeon, H. Choi, and N. Doh, "Detection and compensation of degeneracy cases for IMU-Kinect integrated continuous SLAM with plane features," *Sensors*, vol. 18, no. 4, 2018. [Online]. Available: http://www.mdpi.com/1424-8220/18/4/935

[4] F. Dellaert and M. Kaess, "Factor graphs for robot perception," *Foundations and Trends in Robotics*, vol. 6, no. 1-2, pp. 1–139, Aug. 2017, http://dx.doi.org/10.1561/2300000043.

[5] General Dynamics Mission Systems, "Bluefin HAUV," https://gdmissionsystems.com/products/underwater-vehicles/bluefin-hauv.

[6] B.-J. Ho, P. Sodhi, P. V. Teixeira, M. Hsiao, T. Kusnur, and M. Kaess, "Virtual occupancy grid map for submap-based pose graph SLAM and planning in 3D environments," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Oct. 2018.

[7] M. Hsiao, E. Westman, and M. Kaess, "Dense planar-inertial SLAM with structural constraints," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2018.

[8] G. Hu, S. Huang, L. Zhao, A. Alempijevic, and G. Dissanayake, "A robust RGB-D SLAM algorithm," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Oct. 2012, pp. 1714–1719.

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

[10] K. Low, "Linear least-squares optimization for point-toplane ICP surface registration," Tech. Rep., 2004.

[11] K. Ma, J. Zhu, T. J. Dodd, R. Collins, and S. R. Anderson, "Robot mapping and localisation for feature sparse water pipes using voids as landmarks," in *Towards Autonomous Robotic Systems*, C. Dixon and K. Tuyls, Eds., Cham, 2015, pp. 161–166.

[12] K. Pathak, A. Birk, N. Vaskevicius, and J. Poppinga, "Fast registration based on noisy planes with unknown correspondences for 3-D mapping," *IEEE Trans. Robotics*, vol. 26, no. 3, pp. 424–441, 2010.

[13] Z. Rong and N. Michael, "Detection and prediction of near-term state estimation degradation via online nonlinear observability analysis," in *IEEE International Symposium on Safety Security and Rescue Robotics (SSRR)*, Oct. 2016, pp. 28–33.

[14] R. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Shanghai, China, May 2011.

[15] SoundMetrics Corporation, "SoundMetrics DIDSON 300 specifications," http://www.soundmetrics.com/products/DIDSON-Sonars/DIDSON-300-m/.

[16] Y. Taguchi, Y.-D. Jian, S. Ramalingam, and C. Feng, "Point-plane SLAM for hand-held 3D sensors," May 2013, pp. 5182–5189.

[17] P. Teixeira, M. Kaess, F. Hover, and J. Leonard, "Underwater inspection using sonar-based volumetric submaps," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Daejeon, Korea, Oct. 2016, pp. 4288–4295.

[18] M. J. Tribou, D. W. L. Wang, and S. L. Waslander, "Degenerate motions in multicamera cluster SLAM with non-overlapping fields of view," *CoRR*, vol. abs/1506.07597, 2015. [Online]. Available: http://arxiv.org/abs/1506.07597

[19] E. Westman and M. Kaess, "Underwater AprilTag SLAM and calibration for high precision robot localization," Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-18-43, Oct. 2018.

[20] T. Whelan, J. McDonald, M. Kaess, M. Fallon, H. Johannsson, and J. J. Leonard, "Kintinuous: Spatially extended KinectFusion," in *RSS Workshop on RGB-D: Advanced Reasoning with Depth Cameras*, July 2012.

[21] T. Whelan, S. Leutenegger, R. F. Salas-Moreno, B. Glocker, and A. J. Davison, "ElasticFusion: Dense SLAM without a pose graph," in *Robotics: Science and Systems (RSS)*, Rome, Italy, Jul. 2015.

[22] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Stockholm, Sweden, May 2016.

[23] J. Zhang and S. Singh, "Visual-lidar odometry and mapping: Low-drift, robust, and fast," *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2015.