

Received 19 August 2024; revised 13 December 2024 and 23 March 2025; accepted 26 April 2025.
Date of publication 3 June 2025; date of current version 17 July 2025.
This article was recommended by Executive Editor Sanjiv Singh.

Digital Object Identifier 10.1109/TFR.2025.3576053

Informed, Constrained, Aligned: A Field Analysis on Degeneracy-Aware Point Cloud Registration in the Wild

TURCAN TUNA¹ (Graduate Student Member, IEEE),
JULIAN NUBERT^{1,2} (Student Member, IEEE),
PATRICK PFREUNDSCHUH³ (Student Member, IEEE), CESAR CADENA¹ (Member, IEEE),
SHEHRYAR KHATTAK⁴ (Member, IEEE), AND MARCO HUTTER¹ (Member, IEEE)

¹Robotic Systems Lab, Swiss Federal Institute of Technology (ETH Zürich), 8092 Zürich, Switzerland

²Max Planck Institute for Intelligent Systems, 70569 Stuttgart, Germany

³Autonomous Systems Lab, Swiss Federal Institute of Technology (ETH Zürich), 8092 Zürich, Switzerland

⁴Jet Propulsion Laboratory (JPL), California Institute of Technology (Caltech), Pasadena, CA 91011 USA

CORRESPONDING AUTHOR: TURCAN TUNA (tutuna@ethz.ch)

This work was supported in part by Sony Research under Grant 2023; in part by the European Union (EU) Horizon 2020 Program under Agreement 852044, Agreement 101016970, Agreement 101070405, and Agreement 101070596; in part by the National Centre of Competence in Research (NCCR) Digital Fabrication; in part by the ETH Zürich Research under Grant 21-1 ETH-27; in part by the Swiss National Science Foundation (SNSF) under Project 227617; and in part by the Max Planck ETH Center for Learning Systems.

Project Page/Supp./Code/Datasets: <https://sites.google.com/leggedrobotics.com/perfectlyconstrained>

(Regular Article)

ABSTRACT The iterative closest point (ICP) registration algorithm has been a preferred method for light detection and ranging (LiDAR)-based robot localization for nearly a decade. However, even in modern simultaneous localization and mapping (SLAM) solutions, ICP can degrade and become unreliable in geometrically ill-conditioned environments. Current solutions primarily focus on utilizing additional sources of information, such as external odometry, to either replace the degenerate directions of the optimization solution or add additional constraints in a sensor-fusion setup afterward. In response, this work investigates and compares new and existing degeneracy mitigation methods for robust LiDAR-based localization and analyzes the efficacy of these approaches in degenerate environments for the first time in the literature at this scale. Specifically, this work investigates i) the effect of using active or passive degeneracy mitigation methods for the problem of ill-conditioned ICP in LiDAR degenerate environments and ii) the evaluation of truncated singular value decomposition (TSVD), inequality constraints (Ineq. Con.), and linear/nonlinear Tikhonov regularization for the application of degenerate point cloud registration for the first time. Furthermore, a sensitivity analysis for the least-squares minimization step of the ICP problem is carried out to better understand how each method affects the optimization and what to expect from each method. The results of the analysis are validated through multiple real-world robotic field and simulated experiments. The analysis demonstrates that active optimization degeneracy mitigation is necessary and advantageous in the absence of reliable external estimate assistance for LiDAR-SLAM, and soft-constrained methods can provide better results in complex ill-conditioned scenarios with heuristic fine-tuned parameters. The code and data used in this work are made publicly available to the community.

INDEX TERMS Field robotics, light detection and ranging (LiDAR) degeneracy, localization, robust iterative closest point (ICP).

I. INTRODUCTION

MODERN sensors such as light detection and ranging (LiDAR) provide accurate distance measurements at

high rates, making them suitable for reliable robot pose estimation and accurate mapping in large-scale environments. Simultaneous localization and mapping (SLAM) is a crucial

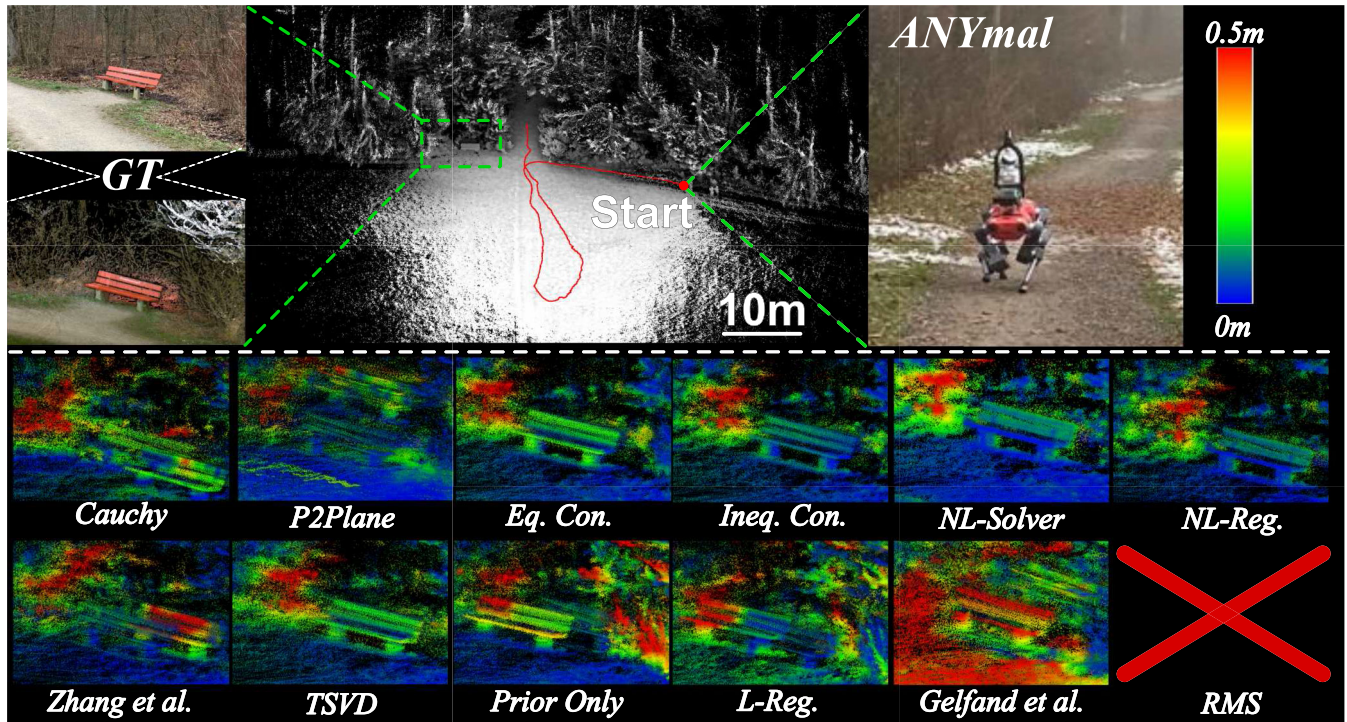


FIGURE 1. Mapping results of the ANYmal forest open-field degeneracy experiment (see Section V-B) are shown. Top: The map and robot trajectory generated by the Eq. Con. method is shown next to the ground truth bench and image of the region of interest. Rest: The point cloud error map for each method is shown. The color coding reflects the point-to-point distances to the ground truth map of the environment. The RMS method diverged due to the irregularities of the natural environment.

tool to operate robots in challenging and degenerate environments [1]. While SLAM and robot localization can also be achieved with other sensing modalities [2], [3], [4], the field robotics community has been focusing on improving LiDAR-based SLAM techniques for better accuracy and robustness in large-scale environments [1], [5], [6], [7].

Most widely accepted LiDAR-SLAM frameworks leverage the efficiency of point cloud registration to provide an accurate environment map. These algorithms align pairs of point clouds with sufficient overlap by finding the 6-degrees-of-freedom (DoF) rigid transformation. A well-established point cloud registration method is the iterative closest point (ICP) algorithm, often utilized in computer vision, 3-D pose estimation, and robotics [8]. In a typical LiDAR-SLAM framework [9], [10], LiDAR scan-to-map registration is performed, which is vital to align the new information to the already generated map of the environment.

A. ALGORITHMIC LIMITATIONS

Despite the success of ICP and the algorithm's broad applicability, there are still some limitations. These limitations include i) the risk of convergence to local minima, ii) the sensitivity to inaccurate initial transformation prior, iii) being susceptible to point-wise feature extraction noise, and iv) robustness against the lack of geometric constraints in the underlying optimization problem [11], [12]. Notably, while the first three sources of error can be directly

addressed by modifying the LiDAR-SLAM framework, the underlying optimization problem's ill-conditioning stems directly from the environment's lack of information along specific directions and cannot be overcome easily.

B. DEFINING LiDAR DEGENERACY

According to the literature [13], [14], [15], and [16], the term LiDAR degeneracy refers to the ill-conditioning of the optimization within the ICP algorithm. If the LiDAR point cloud does not contain sufficient information, e.g., insufficient geometric constraints in a self-similar environment, a linear combination of some of the optimization states might become unobservable. In the presence of such unobservable optimization states, the ICP optimization is said to become *ill-conditioned*, *degenerate*, or *nonlocalizable*.

C. DEGENERATE POINT CLOUD REGISTRATION

While the robotic community has developed LiDAR-based mapping frameworks [10], [17], [18] that function robustly in most environments, point cloud registration and, subsequently, mapping in degenerate environments (without a sufficient amount of distinctive features) still pose a challenge to these frameworks due to a lack of present information in the data. State-of-the-art methods [13], [19], [20], particularly targeted to those scenarios, divide the problem into two steps: 1) analysis and detection of the degenerate directions and 2) mitigation of the adverse effects on the optimization.

In particular, while the first point received considerable attention from the research community [13], [19], [21], [22], [23], [24], [25], [26], [27], the number of works salvaging the information in the best possible way given the detected degenerate directions is far more limited [13], [19], [28]. The latter methods try to mitigate the effect of the degeneracy on the optimization and share the ability to fully benefit from the pose updates, at least in the well-constrained optimization directions, while the former methods entirely skip the optimization and rely on external odometry to propagate the ICP pose estimates. As discussed in [13], skipping the optimization entirely, if ill-conditioned, is a plausible strategy only for short periods, particularly when the registration initial guess is unreliable and the additional fine-registration of the point cloud is beneficial.

D. LiDAR DEGENERACY MITIGATION IN THE FIELD

The robotics community has recently increased their attention to robotic operations in challenging and LiDAR-degenerate environments, as LiDAR-degenerate robotic datasets from field experiments [29], [30], [31], [32], [33] become more readily available to researchers. The robotics community is still actively developing methods [13], [14], [16], [34] to improve the robustness and accuracy of the LiDAR-SLAM and point cloud registration approaches. Despite this, there remains a fundamental research gap on the efficacy of degeneracy mitigation methods *once LiDAR degeneracy is detected*. Many methods have been proposed to mitigate LiDAR degeneracy using external information, such as sensor fusion with factor-graph optimization [35], [36] and filters [37]. However, addressing the optimization step in ICP-based point cloud registration [13] or mitigating LiDAR degeneracy within the ICP method remains an open question. This work focuses on methods to mitigate LiDAR degeneracy within the ICP formulation to narrow the research gap mentioned above. The approaches for degeneracy mitigation in ICP are classified into two categories, namely, *passive* and *active* degeneracy mitigation.

- 1) *Active Degeneracy Mitigation*: A method is classified as an active degeneracy mitigation method if the output of any explicit LiDAR detection from the method (e.g., Section II-B) is used to take action toward mitigating the degeneracy within the ICP cost function optimization.
- 2) *Passive Degeneracy Mitigation*: In contrast to the previous item, a method is called to perform passive degeneracy prevention if the LiDAR degeneracy information is not explicitly used within the ICP formulation to mitigate the degeneracy.

E. CONTRIBUTIONS

In response to the research gap mentioned above, this work studies and compares new and existing degeneracy mitigation methods with various simulated and real-world experiments, such as in Fig. 1, to investigate the efficacy of degeneracy mitigation in point cloud registration in complex degenerate

environments. In addition to the study results, this work introduces three distinct methods to constrain the optimization problem for robust point cloud registration. Finally, the study's results are put into context with theoretical derivations and the limitations of the state-of-the-art methods and validated through multiple robotic field experiments and simulated examples. The contributions of this work are as follows.

- 1) A thorough study on the effectiveness of different constraint types is performed for the degenerate point cloud registration task in the context of robot operation in geometrically ill-conditioned real-world environments such as open natural fields, urban tunnels, and complex construction sites.
- 2) The introduction, investigation, and discussion of subspace Tikhonov regularization, truncated singular value decomposition (TSVD), and inequality-constrained ICP methods for the field of degenerate point cloud registration for the first time in the literature.
- 3) Comparison of the discussed methods in a fair and isolated fashion through simulated examples and multiple detailed robotic field experiments conducted with a legged robot, a walking excavator, and a hand-held setup with different LiDARs.
- 4) The consolidation of open-source methods and all investigated methods are going to be open-sourced¹ in a single framework to foster future research on developing degeneracy-aware LiDAR-SLAM systems.

II. RELATED WORK

A. (ROBUST) POINT CLOUD REGISTRATION

Fast and accurate pose estimation through point cloud registration has been a central research topic in robot perception over multiple decades [38]. Different families of solutions have been explored in this field, such as ICP and its variants [39], [40], [41], [42], feature-based [43], [44], probabilistic [45], [46], [47], learned [48], [49], [50], or robust global registration [51], [52] methods.

Despite the majority of advancements, the simpler ICP variants using point-to-point [53], point-to-plane [54], symmetric point-to-plane [42], and G-ICP [39] are among the most widely used LiDAR-based point cloud registration algorithms due to their simplicity and practicality [8], [55]. All ICP variations have in common to find the refined 6-DoF transformation between two point clouds by repeatedly finding sets of corresponding point pairs, followed by iterative minimization of a predefined alignment metric.

On the other hand, the susceptibility of the ICP method to local minima encouraged researchers to investigate globally optimal and certifiable algorithms for robust point cloud registration [8], [52], [56], [57], [58], [59]. These algorithms ensure that the estimated solution corresponds to the globally optimal solution, even in outlier-dominant cases. Similarly, robust M-estimators have been proposed to deal with heavy noise-dominated point cloud registration problems. These

¹<https://sites.google.com/leggedrobotics.com/perfectlyconstrained>

estimators down-weight the outliers to mitigate their influence on the cost function. The well-known M-estimators such as Welsch, Tukey, Cauchy, Huber, and Geman–McClure have been used by the robotics community [17], [60], [61]. Recently, Bapin et al. [61] investigated the effectiveness of different robust norms on nondegenerate registration problems and concluded that, if tuned correctly, most robust norms perform similarly by increasing the signal-to-noise ratio in the optimization problem. In addition, to further mitigate the need for manual hand-tuning, researchers proposed adaptive M-estimators [62] to be robust against different distributions of inputs and adaptive graduated nonconvexity [52], [56], [59], [63], [64] to improve robustness against large perturbations. Furthermore, driven by the success of the ICP algorithm toward practical applications, researchers have focused on improving its robustness through statistical methods [47], [65]. These improvements include better feature sampling [14], [66], [67], adaptive matching techniques [17], and optimization acceleration methods [60].

However, none of these methods particularly tackle situations where insufficient information is contained in the underlying point cloud due to the symmetric or self-similar nature of the environment, which is often present in practice in environments such as tunnels, open planes, or narrow corridors [13], [25], [68].

B. OPTIMIZATION DEGENERACY DETECTION

1) POINT CLOUD REGISTRATION UNCERTAINTY

Previous methods have been proposed to model the uncertainty of the registration and capture the degeneracy of the optimization problem by estimating the covariance of the pose estimation process [11], [12]. Although this may be a good indicator of ill-conditioning, the obtained measures often provide overly optimistic estimates [69]. Recently, Talbot et al. [70] provided additional insights on accurately modeling the uncertainty of LiDAR point cloud registration information for use in a multimodal sensor fusion framework as a loosely coupled measurement. More specifically, the authors evaluated the ICP uncertainty formulation in [71] in the context of LiDAR-SLAM by simplifying the formulation to omit the cross-covariance terms for compute limitation purposes.

Similarly, in LOG-LIO2 [72], authors provide an incremental methodology for efficient calculation of measurement uncertainty of LiDAR measurements. Their proposed uncertainty model captures the range and bearing uncertainty as well as the uncertainty arising from LiDAR beam incident angle and surface roughness. Later, the authors integrated this uncertainty model in a LiDAR inertial odometry framework to illustrate the benefits during deployment. As one of the most recent works, Hatleskog and Alexis [16] proposed a novel probabilistic formulation to estimate the probability of optimization directions being degenerate by taking the measurement and surface normal estimation noise into

account. As the increasing amount of recent work shows, the estimation of unified metric uncertainty of point cloud registration is a promising direction in quantifying optimization degradation. These methods further allow the fusion of the ill-conditioned poses into sensor-fusion frameworks through the metric uncertainty, as discussed in Section II-C3.

2) LEARNING-BASED APPROACHES

Recent research has investigated data-driven approaches to identify degeneracy in LiDAR information. Gao et al. [73] introduced a deep learning-based entropy metric by fusing covariance estimation with the information level of the environment. To avoid complicated Monte-Carlo sampling during robot operation, Nubert et al. [25] suggested utilizing synthetic data for training while relying solely on the current LiDAR scan to compute a 6-DoF localizability metric. Although data-driven methods have the potential to outperform traditional methods by modeling higher degree dynamics of the problem, the interpretability of these methods still presents a challenge.

3) OPTIMIZATION STABILITY-BASED APPROACHES

Other works [23], [36], [74] also propose to use the optimization condition number as a single combined degeneracy metric for the full 6-DoF pose estimation. Tangentially, CompSLAM [9] employs the D-optimality criterion [75] as a metric to detect optimization degeneracy in a multimodal sensor fusion framework. Following a similar formulation, Hinduja et al. [76] used the optimization's relative condition number as the degeneracy detection threshold in contrast to the minimum eigenvalue. Among pioneering methods for degeneracy detection, Zhang et al. [19] introduced a detection metric called *degeneracy factor*. This metric identifies the degenerate directions of the optimization by analyzing the minimum eigenvalue of the optimization's Hessian matrix. Similarly, DAMS-LIO [77] follows the footsteps of [19] and uses the Hessian of the optimization to acquire the degenerate directions by thresholding the eigenvalues for a filtering-based state estimation method. Furthermore, Bai et al. [78] argued that using only the minimum eigenvalue of the optimization Hessian is impractical due to the required tuning of thresholds. To mitigate this problem, the authors proposed simultaneously utilizing the minimum eigenvalue and the condition number. Recently, Jin and Jiang [35] proposed a degeneracy metric called *degenerate degree* by using the minimum eigenvalue in combination with the partial condition number of the optimization Hessian. Considering the challenges of setting a threshold for minimum eigenvalues, Lee et al. [79] proposed to use the Chi-squared test to calculate a nonheuristic threshold of normalized eigenvalues. Later, the authors use these normalized eigenvalues to identify the degenerate directions for LiDAR as well as vision-based optimization problems. In recent work, Ji et al. [34] proposed a point-to-distribution ICP degeneracy detection method. Leveraging the probability distribution

from their adaptive voxelization scheme, the authors show that their proposed metric is inherently correlated with the eigenvalues of the optimization Hessian.

4) FINE-GRAINED APPROACHES

Similarly, Zhen and Scherer [27] proposed directly estimating an environment's localizability using the Hessian eigenspace by measuring the sensitivity of a point and surface-normal pair constraint with respect to the optimization states. Using the formulation introduced in [27], Zou et al. [80] generated an information matrix from the LiDAR measurements to identify the degenerate directions of the optimization, which is later propagated to a factor-graph for accurate robot pose estimation. Toward more fine-grained degeneracy handling, X-ICP [13] proposed a correspondence-based degeneracy detection method to constrain the optimization problem for point cloud registration. This method independently considered the individual observability contribution of each point-to-plane correspondence. Following the formulation of X-ICP [13], Chen et al. [37] proposed to detect LiDAR degeneracy using correspondence information in the registration step.

C. DEGENERACY MITIGATION IN ROBOTICS

For robot operation in more complex and unstructured environments, numerous approaches have been proposed to mitigate the effect of optimization ill-conditioning for LiDAR-based robot pose estimation methods. These approaches can be categorized as either *passive* or *active*, based on how degeneracy mitigation is achieved. More specifically, the categorization is based on whether an approach changes the underlying optimization cost function or the immediate relative LiDAR pose estimation of the registration, using the detected degeneracy information. Technical details are provided in Section IV.

1) PASSIVE DEGENERACY MITIGATION

Most passive degeneracy mitigation methods use a complementary sensor modality in a sensor fusion framework [31], [78], [81] or rely on an external odometry source [9], [23], [36], [68] to propagate the robot pose estimate in the case of LiDAR degeneracy. Uniquely, Lim et al. [82] employed an adaptive parameter setting strategy to mitigate the effects of LiDAR degeneracy. In similar reasoning, Ferrari et al. [83] focused on improving the correspondence search of the ICP algorithm to improve the quality of the ICP and, tangentially, the robustness to degeneracy. Based on the analysis of the optimization variables, Gelfand et al. [15] investigated the geometric stability of the point-to-plane ICP by proposing a sampling-based method to extract the most informative scan points to improve the conditioning of the optimization process. Petracek et al. [14] proposed to minimize information redundancy in the input point cloud. To achieve this, the authors utilized an elaborate

and degeneracy-aware sampling strategy and alleviated the adverse effects of the LiDAR degeneracy.

2) ACTIVE DEGENERACY MITIGATION

The work in [84] is among the first to perform active degeneracy mitigation by focusing on point cloud registration through additional constraints to reduce incorrect point cloud registration. Similarly, Olsson and Eriksson [85] concentrated on reducing the error originating from the linearization of the rotation variables by introducing non-linear equality constraints (Eq. Con.) in the optimization. Recently, Zhang et al. [19] proposed *solution remapping* to project the relative LiDAR pose updates from the point cloud registration only along the well-constrained directions. In X-ICP [13], the addition of hard constraints along the degenerate directions was proposed to reduce the robot pose estimate error. Nashed et al. [28] proposed using additional regularization terms in a nonlinear least-squares problem to mitigate the effect of optimization degeneracy. Furthermore, to the best of the authors' knowledge, [28] is the only existing work that compares different degeneracy mitigating methods. Utilizing the constrained optimization from X-ICP [13], Chen et al. [37] proposed to use the detected degeneracy information in their error-state iterated extended Kalman filter to prevent constrain degenerate LiDAR measurement updates in degenerate directions. However, the corresponding comparison is limited and does not generalize to the applications of LiDAR-SLAM and point cloud registration. Moreover, the method proposed in [28] does not conduct field experiments and uses an RGBD camera instead of a 3D-LiDAR. As discussed, a considerable body of work focuses on LiDAR degeneracy. Despite this fact, the literature lacks a comprehensive analysis of how to effectively use the LiDAR degeneracy information. This work thoroughly analyzes and compares different methods in simulated and real-world settings to bridge this gap.

3) SENSOR FUSION AND MULTIMODAL DEGENERACY MITIGATION

In addition to LiDAR-centric methods, other approaches have focused on utilizing complementary sensing modalities to alleviate the adverse effects of LiDAR degeneracy. COIN-LIO [68] proposed to complement the uninformative directions in the LiDAR inertial framework with LiDAR intensity image using visual feature detection techniques. Similarly, Jin and Jiang [35] complemented the ICP degenerate directions with constraints from an ultrawideband positioning approach. Furthermore, Nissov et al. [31] used LiDAR and radar modalities in a factor-graph-based windowed smoother with explicit degeneracy information for each modality. Recently, Lee et al. [79] combined LiDAR, inertial, and visual information by actively switching between modalities in a pose graph optimization setting. The authors utilized separate detection of optimization degeneracy for LiDAR and camera modalities compared to more generic covariance estimation methods such as dynamic covariance

scaling [86], fallback graphs [87], or global frame alignment [88]. In recent years, the DARPA Subterranean (SubT) Challenge [1] highlighted the importance of degeneracy-aware LiDAR-SLAM. Among the methods used in the challenge, CompSLAM [9] used the detected degeneracy to switch between sensor modalities. In contrast, DARE-SLAM [36] used degeneracy detection to prevent the addition of LiDAR factors into the pose graph to prevent degradation. Another method, LION [23], utilized a measurement multiplexer to estimate the uncertainty of the LiDAR factors to robustly fuse with the IMU measurements in a sliding-window factor graph. To minimize the dependency on a single sensing modality, Zhao et al. [89] suggested running three parallel pose graphs: one for LiDAR inertial estimation, another for visual-inertial estimation, and a central IMU pose graph dedicated to sensor fusion. This approach, which depends on precise measurement uncertainties, highlights the effectiveness of factor-graph-based methods. Extending the sensor fusion framework [89] to actively utilize degeneracy information, Zhao et al. [90] proposed a framework to predict the observability of a LiDAR scan and incorporate it as a covariance (uncertainty) matrix.

Although these methods improve the robustness of the overall odometry or pose estimation of the system, they leave the original problem of LiDAR degeneracy untouched and rely on external sensor fusion for system-level robustness. This work focuses on investigating methods that improve the robustness and quality of a LiDAR-based pose estimation within the degenerate point cloud registration framework.

III. PROBLEM ANALYSIS AND PRELIMINARIES

This work investigates the advantages and disadvantages of different optimization degeneracy mitigation methods. Specifically, the problem of degenerate point cloud registration is considered under the scope of LiDAR-SLAM in geometrically challenging environments. This section presents the preliminaries on point cloud registration and degeneracy detection.

A. POINT CLOUD REGISTRATION AND ICP

The point cloud registration problem aims to align two point clouds and obtain their relative rigid transformation as shown in Fig. 2. To achieve this, various techniques and metrics are proposed, as discussed in Section II. In this work, the ICP algorithm with the point-to-plane [54] cost function is utilized as it is widely adopted by the robotics community and many point cloud registration frameworks [9], [10], [18], [23], [91], [92].

Specifically, point cloud registration focuses on estimating the homogeneous transformation $T_{ML} \in SE(3)$ between a *source* and a *reference* point cloud of sizes N_p and N_q , respectively, given an initial guess $T_{ML,init}$. The *source* point cloud ${}_{\mathcal{L}}P \in \mathbb{R}^{3 \times N_p}$ is expressed in the LiDAR frame \mathcal{L} , while the *reference* point cloud ${}_{\mathcal{M}}Q \in \mathbb{R}^{3 \times N_q}$ is expressed in the (sub)map frame \mathcal{M} . The homogeneous transformation $T_{ML} = [R_{ML} | t_{ML}]$, consists of a rotation matrix $R \in SO(3)$,

and a translation vector $t \in \mathbb{R}^3$, denoted as $t = [t_x, t_y, t_z]^T$. For the N_p *source* points ${}_{\mathcal{L}}p_i \in \mathbb{R}^3$ in ${}_{\mathcal{L}}P$, the closest reference point ${}_{\mathcal{M}}q_i \in \mathbb{R}^3$ in ${}_{\mathcal{M}}Q$, is found through a K -nearest neighbors correspondence search given initial transformation $T_{ML,init}$ as shown in Fig. 2 matching block. This transformation is often provided as an initial guess and applied to all points in ${}_{\mathcal{L}}P$ to obtain ${}_{\mathcal{M},init}P$ to improve the matching process and the optimization convergence characteristics [12] through the iterative process of ICP. In the remainder of the section, all variables are expressed in the (sub)map frame \mathcal{M} , and for the sake of brevity, the notation is simplified to omit the frame. The result of the correspondence search can be summarized as a one-to-many matching $\{p_i, Q(p_i)\}$ for $i \in \{1, \dots, N\}$, where p_i and $q_i := Q(p_i)$ are the $N \leq N_p$ matched point pairs from *source* and *reference* point clouds, respectively. Moreover, $n_i \in \mathbb{R}^3$, $\|n_i\| = 1$ is the surface normal of reference point q_i , as needed for the point-to-plane [54] ICP cost function. The number of matched points N indicates the size of the optimization problem. The cost function of the point-to-plane ICP minimization problem is defined as

$$\min_{R, t} \sum_{i=1}^N \left\| ((Rp_i + t) - q_i) \cdot n_i \right\|_2. \quad (1)$$

The problem shown in (1) can be reformulated as a quadratic cost minimization problem following the derivation shown in [38] after introducing the scalar triple product and rotation matrix linearization:

$$\begin{aligned} \min_{x \in \mathbb{R}^6} x^T & \underbrace{\left(\sum_{i=1}^N \begin{bmatrix} p_i \times n_i \\ n_i \end{bmatrix} \begin{bmatrix} (p_i \times n_i)^T & n_i^T \end{bmatrix} \right)}_A x \\ & - 2x^T \underbrace{\left(\sum_{i=1}^N \begin{bmatrix} p_i \times n_i \\ n_i \end{bmatrix} n_i^T (q_i - p_i) \right)}_b + \text{Constant}. \quad (2) \end{aligned}$$

Here, $x = [r^T, t^T]^T \in \mathbb{R}^6$ are the optimization variables, where the translational part is $t \in \mathbb{R}^3$. Moreover, the rotation vector $r \in \mathbb{R}^3$ where $R \approx r^\wedge + I$. The \wedge ‘‘hat operator’’ transforms a vector to its skew-symmetric matrix representation, $r^\wedge \in \mathfrak{so}(3)$. Here, $\mathfrak{so}(3)$ is the Lie algebra of $SO(3)$ group, which consists of skew-symmetric matrices of size 3×3 .

Substituting A and b into (2) results in the regular linear least-squares optimization formulation

$$E(x) = \min_{x \in \mathbb{R}^6} \left\| Ax - b \right\|_2 \quad (3)$$

with $E(x)$ indicating the residual error of the optimization and minimization of this cost function results in the Frobenius norm (2-norm) solution. Here, $A \in \mathbb{R}^{6 \times 6}$ denotes the Hessian of the optimization problem, and $b \in \mathbb{R}^6$ contains the constraints imposed by the registration. The optimal translation t and rotation r vectors can be calculated at each linearization point. Due to errors in linearization and correspondence matching, the matching and minimization operations of ICP are repeated iteratively until convergence.

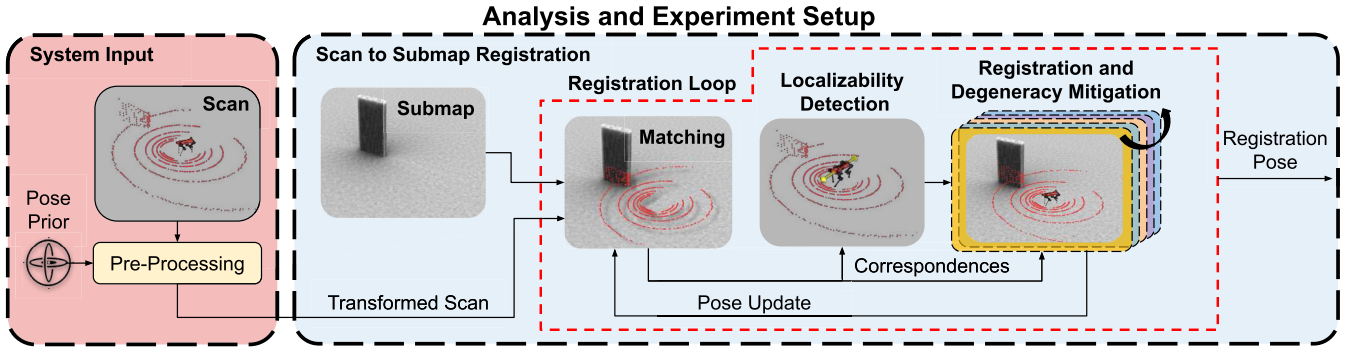


FIGURE 2. Overview of the analysis framework and experiment setup. The pose prior is used to transform and undistort the input point cloud, fed to the iterative registration optimization loop with the existing point cloud map. The localizability of the optimization at the current iteration is detected based on the method proposed in X-ICP [13] and fed to the analyzed degeneracy mitigation methods (explained in Section IV). The final registration pose within the registration loop is calculated iteratively as in ICP fashion.

The minimization problem (1) can be solved with nonlinear solvers, such as Gauss–Newton or Levenberg–Marquardt (LM) optimization [93], or using closed-form solvers, such as singular value decomposition (SVD) [94] or lower–upper (LU)-decomposition to solve (3). This operation corresponds to the degeneracy mitigation block shown in Fig. 2. Except for Sections IV-A6 and IV-B1, this work uses SVD to solve (3) for the reasons introduced in the following.

The minimum of (3) is found when the cost function gradient with respect to \mathbf{x} is zero in all directions. Following the previous definitions

$$\begin{aligned} \frac{\partial E(\mathbf{x})}{\partial \mathbf{x}} &= \frac{\partial}{\partial \mathbf{x}} \left[\mathbf{b}^\top \mathbf{b} - \mathbf{b}^\top \mathbf{A} \mathbf{x} - \mathbf{x}^\top \mathbf{A}^\top \mathbf{b} + \mathbf{x}^\top \mathbf{A}^\top \mathbf{A} \mathbf{x} \right] \\ &= -2\mathbf{A}^\top \mathbf{b} + 2\mathbf{A}^\top \mathbf{A} \mathbf{x} = 0 \end{aligned} \quad (4)$$

resulting in the normal equation after rearranging the terms

$$\left(\mathbf{A}^\top \mathbf{A} \right) \mathbf{x} = \mathbf{A}^\top \mathbf{b}. \quad (5)$$

The solution of the least-squares problem with (5) can be interpreted geometrically as the projection of \mathbf{b} onto the span of matrix \mathbf{A} , as shown in Fig. 3. The residual of the optimization can be seen as the orthogonal distance of \mathbf{b} to the span of \mathbf{A} .

B. SENSITIVITY OF THE LEAST-SQUARES PROBLEM

This section contextualizes the previously introduced methods with the sensitivity and conditioning of the least-squares minimization problem. The sensitivity of a problem to small changes in the input is referred to as the conditioning of the problem. A problem is called ill-conditioned if small changes in the input lead to big changes in the output. Otherwise, the problem is called well-conditioned [95].

Following the derivation [95], [96], the Euclidian condition number of matrix \mathbf{A} can be defined as:

$$\text{cond}(\mathbf{A}) = \frac{\sigma_{\max}}{\sigma_{\min}} = \|\mathbf{A}\|_2 \|\mathbf{A}^+\|_2. \quad (6)$$

Here, \mathbf{A}^+ is the pseudo-inverse of \mathbf{A} . The pseudo-inverse can be calculated using the SVD of \mathbf{A} as detailed in Section III-C

and (15). Furthermore, σ_{\max} and σ_{\min} refer to the maximum and minimum eigenvalue of \mathbf{A} , respectively. Similarly, eigenvalues can be retrieved using SVD as detailed in (15).

Previously, Zhang et al. [19] provided in-depth technical details of their degeneracy detection factor with the sensitivity analysis of a square linear system $\mathbf{A} \mathbf{x} = \mathbf{b}$. However, in reality, the ICP formulation does not use this equality and instead solves a least-squares minimization problem as described in Section III and (5). The conditioning of a least-squares problem $\mathbf{A} \mathbf{x} \approx \mathbf{b}$ depends not only on the matrix \mathbf{A} but also on the right-hand-side vector \mathbf{b} , and thus $\text{cond}(\mathbf{A})$ alone does not suffice to characterize sensitivity. As seen in Fig. 3, a constraint vector \mathbf{b} near $\text{span}(\mathbf{A})$ would result in a smaller residual, thereby, perturbations on \mathbf{b} would induce lesser variations in \mathbf{x} . It can be inferred that as $\theta \rightarrow (\pi/2)$ the least-squares solution becomes singular. This behavior can be characterized with $\cos \theta = (\|\mathbf{A} \mathbf{x}\| / \|\mathbf{b}\|)$.

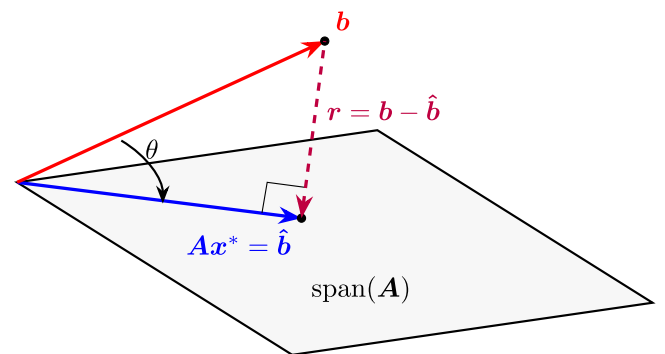


FIGURE 3. Geometric interpretation of the least-squares minimization problem. Here, θ is the angle between the approximated constraints $\hat{\mathbf{b}}$ and the original constraints \mathbf{b} . The minimization of (3) minimizes residual the r .

1) PERTURBATION ON \mathbf{b}

Here, an analysis of the sensitivity of \mathbf{x} to a change in the right-hand-side vector \mathbf{b} is provided. For brevity, the 2-norm notation is shown as $\|\cdot\|$ in the following. Given a perturbation $\delta \mathbf{b}$ and a resulting change in \mathbf{x} , $\delta \mathbf{x}$ the normal equation (5) can

be rewritten as

$$\begin{aligned} A^T A (x + \Delta x) &= A^T (b + \Delta b) \\ A^T \overline{Ax} + A^T A \Delta x &= A^T \overline{b} + A^T \Delta b. \end{aligned} \quad (7)$$

Canceling the original components of the normal equation [see (5)] leaves

$$\begin{aligned} A^T A \Delta x &= A^T \Delta b \\ \Delta x &= \underbrace{(A^T A)^{-1}}_{A^+} A^T \Delta b \\ \Delta x &= A^+ \Delta b. \end{aligned} \quad (8)$$

As this analysis investigates the amplitude of the change, the 2-norm of both sides is taken, which results in

$$\|\Delta x\| = \|A^+\| \|\Delta b\|. \quad (9)$$

To retrieve the relative amplitude of the change, both sides are normalized by $\|x\|$. This results in an inequality bound instead of Eq. Con. as a consequence of the Cauchy–Schwarz inequality, $\text{cond}(A) = \|A\| \|A^+\| \geq 1$ and $\|A^+\| \approx (1/\sigma_{\min})$. The inequality results in

$$\begin{aligned} \frac{\|\Delta x\|}{\|x\|} &\leq \|A^+\| \frac{\|\Delta b\|}{\|x\|} \\ \frac{\|\Delta x\|}{\|x\|} &\leq \frac{\text{cond}(A) \|\overline{b}\| \|\Delta b\|}{\|A\| \|\overline{b}\| \|x\|} \\ \frac{\|\Delta x\|}{\|x\|} &\leq \text{cond}(A) \underbrace{\frac{\|\overline{b}\|}{\|Ax\|}}_{(\cos \theta)^{-1}} \frac{\|\Delta b\|}{\|\overline{b}\|}. \end{aligned} \quad (10)$$

Equation (10) shows that the sensitivity of Δx is affected by $\text{cond}(A)$ and how well b is approximated with Ax . Interestingly, when $\cos \theta \approx 1$ the effective condition number is approximately $\text{cond}(A)$ however, if $\cos \theta \approx 0$ the effective condition number would be much larger than $\text{cond}(A)$. This finding implies that, despite perfect conditioning, a least-squares optimization could be ill-defined if b has noise and the residual is large.

2) PERTURBATION ON A

To fully understand the sensitivity, the perturbation of A should also be considered. Similar to the perturbation of b , the normal equation is performed after adding a perturbation of ΔA to A

$$(A + \Delta A)^T (A + \Delta A) (x + \Delta x) = (A + \Delta A)^T b. \quad (11)$$

Expanding both sides of the above equation results to

$$\begin{aligned} A^T \overline{Ax} + A^T A \Delta x + A^T (\Delta A) x + (\Delta A)^T A x \\ + \underbrace{A^T \Delta A \Delta x + (\Delta A)^T A \Delta x + (\Delta A)^T \Delta A x}_{\approx 0} \\ + \underbrace{(\Delta A)^T \Delta A \Delta x}_{\approx 0} \\ = A^T \overline{b} + (\Delta A)^T b. \end{aligned} \quad (12)$$

Higher order terms are neglected to simplify the calculations, and the components of the original normal equation [see (5)] are canceled out. The above equation is rewritten after the simplifications as follows:

$$\begin{aligned} A^T A \Delta x + A^T (\Delta A) x + (\Delta A)^T A x &\approx (\Delta A)^T b \\ A^T A \Delta x + A^T (\Delta A) x &\approx (\Delta A)^T \underbrace{(b - Ax)}_r \\ A^T A \Delta x &\approx (\Delta A)^T r - A^T (\Delta A) x \\ \Delta x &\approx \underbrace{(A^T A)^{-1}}_{A^+} (\Delta A)^T r - \underbrace{(A^T A)^{-1}}_{A^+} A^T (\Delta A) x \\ \Delta x &\approx (A^T A)^{-1} (\Delta A)^T r + A^+ (\Delta A) x. \end{aligned} \quad (13)$$

In order to complete the sensitivity analysis, the relative changes are required. Hence, applying the 2-norm and rearranging the variables as

$$\begin{aligned} \frac{\|\Delta x\|}{\|x\|} &\leq \frac{\text{cond}(A)^2}{\|A\|^2} \|\Delta A\| \frac{\|r\|}{\|x\|} + \frac{\text{cond}(A)}{\|A\|} \|\Delta A\| \\ \frac{\|\Delta x\|}{\|x\|} &\leq \text{cond}(A)^2 \frac{\|r\|}{\|Ax\|} \frac{\|(\Delta A)\|}{\|A\|} + \text{cond}(A) \frac{\|\Delta A\|}{\|A\|} \\ \frac{\|\Delta x\|}{\|x\|} &\leq \left(\text{cond}(A)^2 \frac{\|r\|}{\underbrace{\|Ax\|}_{\tan \theta}} + \text{cond}(A) \right) \frac{\|\Delta A\|}{\|A\|}. \end{aligned} \quad (14)$$

Here, the fact that $\|(\Delta A)^T\| = \|(\Delta A)\|$ and the definition of condition number [see (6)] is used to bring the equation into an interpretable format. With the conclusion of (14), it becomes clear how the sensitivity of the least-squares problem is modeled against perturbations to A . It is easy to see that two terms affect the propagation of the perturbation to x , namely, the condition number of A and a second term proportional to the residual r , scaled with **square** of the condition number of A . This second term is particularly important as any remaining residual r will be scaled by $\text{cond}(A)^2$, which is important for ill-conditioned problems. As a result of the above explanations, the effective conditioning of the least-squares optimization problem differs from the conditioning of A . This finding shows the importance of mitigating the optimization degeneracy at the time of or before optimization so that the residual term does not interfere with the quality of the minimization.

C. SOLVING THE OPTIMIZATION

In Section III-A, the sensitivity of the normal equations [see (5)] are analyzed for perturbations on A and b ; however, the retrieval of the optimal optimization variables is not discussed. Thus, this section introduces how the optimized variables that minimize (3) can be calculated with SVD. Equation 5 can be solved with efficient factorization methods such as the Cholesky factorization, given that $A^T A$ is not ill-posed (near-singular). However, since this work

focuses on LiDAR degeneracy and, subsequently, an ill-conditioned optimization, using Cholesky factorization is not advantageous. Instead, SVD, a factorization method that produces two orthonormal bases next to the diagonalized original matrix, is used to solve this equation, especially when the optimization problem is ill-conditioned, as the SVD solution exists for all (nonsquare) matrices. For a matrix A of size $m \times n$, the decomposition is defined as

$$A = U\Sigma V^\top, \quad A^+ = V\Sigma^{-1}U^\top \quad (15)$$

with U a $m \times m$ rotation matrix, Σ a diagonal $m \times n$ scaling and projection matrix that contains the singular values of the matrix $A^\top A$, and finally, V the second rotation matrix, with size $n \times n$. Moreover, A^+ is the pseudo-inverse of A and in the full-rank case of A , $A^+ \approx A^{-1}$. For symmetric and positive (semi)-definite A , SVD becomes identical to eigenvalue-decomposition $A = V\Sigma V^\top$, and subsequently

$$A^\top A = (V\Sigma V^\top)^\top (V\Sigma V^\top) = V\Sigma^2 V^\top \quad (16)$$

where Σ^2 denotes a square diagonal matrix that has the squared singular values of A as entries. The right singular vectors V of A are equivalent to the eigenvectors of $A^\top A$, while the singular values of A are the square root of the eigenvalues of $A^\top A$.

After the SVD decomposition, the resulting orthonormal bases and the diagonal scaling matrix allow for the pseudo-inverse calculation

$$(A^\top A)^+ = V\Sigma^{-2}V^\top. \quad (17)$$

Applying (17) and (15) to (5) results in

$$\begin{aligned} x^* &= (A^\top A)^{-1} A^\top b, \\ x^* &= (V\Sigma^{-2}V^\top) A^\top b \\ x^* &= (V\Sigma^{-2}V^\top) (V\Sigma U^\top) b \\ x^* &= V\Sigma^{-1}U^\top b. \end{aligned} \quad (18)$$

As SVD is based on the pseudo-inversion described in (17), the optimal x^* can be found regardless of the conditioning of A . However, this optimal x^* would not be optimal along the ill-conditioned directions of the optimization, as these directions will remain unobservable.

On a similar note, SVD is particularly effective against ill-conditioned or nearly rank-deficient problems, as the decomposition allows for omitted near-zero singular values. As a result, the least-squares solution is less sensitive to perturbations in the data [95]. This property is leveraged by one of the methods, TSVD, detailed in Section IV-A4.

D. LiDAR DEGENERACY DETECTION

When robots are deployed in geometrically featureless and challenging environments, point cloud registration can become ill-conditioned. Hence, a method to detect the degenerate directions of the optimization is required. As this work

focuses on degeneracy mitigation, all works discussed here use the degeneracy detection method described in X-ICP [13] to ensure a fair comparison (for all methods that require localizability information). This localizability detection process is shown as the localizability detection block in Fig. 2. For more details on LiDAR-based degeneracy detection, the readers are referred to X-ICP (see [13], Section V).

Following the derivation from [13], the optimization Hessian is divided into submatrices to identify the eigenvectors corresponding to the rotational and translational parts r and t of the optimization variables x :

$$A = \begin{bmatrix} A_{rr} & A_{rt} \\ A_{tr} & A_{tt} \end{bmatrix}_{6 \times 6}. \quad (19)$$

The factorized Hessians $A_{rr} \in \mathbb{R}^{3 \times 3}$ and $A_{tt} \in \mathbb{R}^{3 \times 3}$ correspond to rotational and translational information, respectively. Furthermore, A_{rr} and A_{tt} are utilized for degeneracy detection, as it is not trivial to treat t and r jointly due to differences in scale and physical meaning. For the corresponding submatrices, the eigenvalue-decomposition can be written as

$$A_{tt} = V_t \Sigma_t V_t^\top, \quad A_{rr} = V_r \Sigma_r V_r^\top. \quad (20)$$

The matrices $V_t \in \mathbb{R}^{3 \times 3}$ and $V_r \in \mathbb{R}^{3 \times 3}$ are the eigenvectors in matrix form, and $\Sigma_t \in \{\text{diag}(\sigma_{tt}) : \sigma_{tt} \in \mathbb{R}^3 \geq 0\}$ and $\Sigma_r \in \{\text{diag}(\sigma_{rr}) : \sigma_{rr} \in \mathbb{R}^3 \geq 0\}$ are the scaling matrices for A_{tt} and A_{rr} , respectively. Here, σ_{tt} and σ_{rr} are the eigenvalues of A_{tt} and A_{rr} . For the remainder of the work, the null space approximation of V_t and V_r is assumed to be known through this analysis in combination with their associated eigenvalues from Σ_t and Σ_r . If an eigenvector, $v_j \in \{V_t, V_r\}$, is within the null space of V_t or V_r , it is denoted as *degenerate* (nonlocalizable), otherwise *nondegenerate* (localizable). In addition, the mapping $V \Leftrightarrow \{V_r, V_t\}$ is taken into account since a subset of methods described later in Section IV expect degenerate eigenvectors in $\mathbb{R}^{3 \times 3}$, while others expect them in $\mathbb{R}^{6 \times 6}$. Moreover, the separate detection of translational and rotational degeneracy allows careful consideration of the scale differences of the entries in the Hessian A . As a result of this consideration, the cross-terms A_{tr} and A_{rt} are omitted. However, since all methods that require degeneracy detection information have access to the same degeneracy information, the effect of this omission is alleviated.

IV. METHODOLOGY AND IMPLEMENTATION

In this section, the theoretical and implementation details of the methods compared in Section V are laid out. Motivated by the discussion in Sections I and II, the methods are classified into two categories, namely, i) active degeneracy mitigation methods and ii) passive degeneracy mitigation methods, as shown in Fig. 4. The main difference between these two classes of methods is the utilization of the available degeneracy detection. Passive methods do not utilize this available information, while active methods utilize this

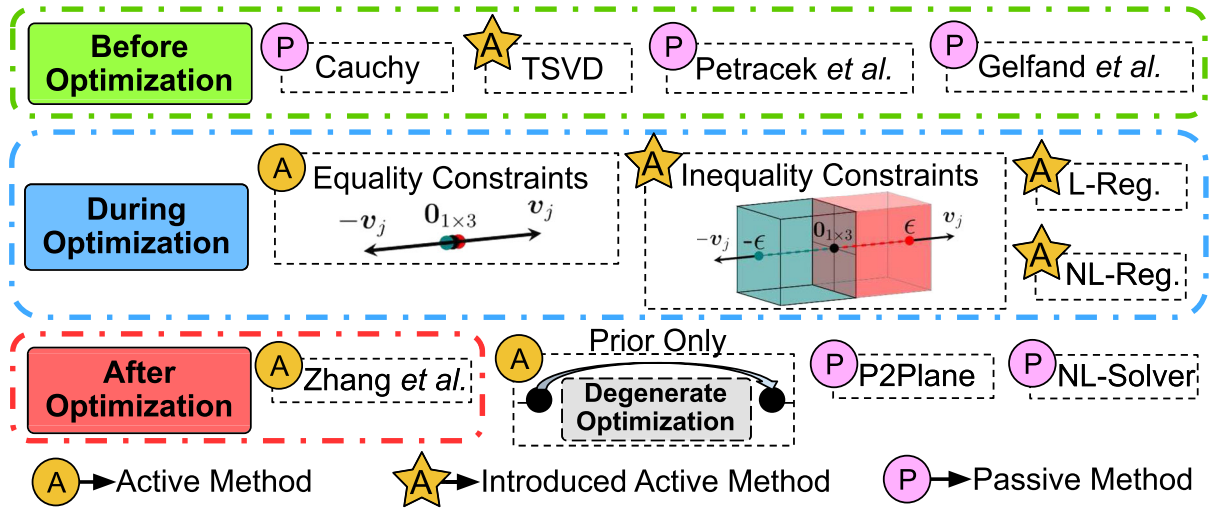


FIGURE 4. Overview of all the investigated methods, highlighting at which stage these methods affect the point cloud registration process. The letters P and A mark passive and active degeneracy mitigation methods, respectively. The \star symbol indicates that the method is investigated in the context of degenerate point cloud registration for the first time in this work. Zhang et al. refers to work in [19], Petracek et al. [14], and Gelfand et al. to [15].

information in different ways to mitigate the adverse effects of optimization degeneracy.

A. ACTIVE DEGENERACY MITIGATION

The active degeneracy mitigation class contains the methods that utilize the detected degeneracy information to alter the optimization cost function or the optimization variables. Often, these methods require accurate information on degeneracy detection to perform optimally. In addition, these methods perform additional computations to mitigate the effects of degeneracy or circumvent degeneracy by avoiding registration. In addition, active methods can be separated into further classes based on when the method affects the ICP, as shown in Fig. 4.

1) EQUALITY CONSTRAINTS

Eq. Con., as recently discussed in X-ICP [13], utilize constrained optimization techniques to add additional Eq. Con. along the degenerate directions, thus preventing optimization of the solution along these directions during iterative refinement. In the formulation of the Eq. Con., the degenerate eigenvectors $v_j \in \{V_t, V_r\}$ of size 3×1 each for translation and rotation will be used as constraint directions. Here, where $V_{r,t} \in \mathbb{R}^{3 \times 3}$ are the eigenvector matrices for rotational and translational parts, respectively. Given the eigenvectors, the constraint formulation is

$$\begin{aligned} v_j \cdot (t - t_0) &= 0, & \text{if } v_j \in V_t \\ v_j \cdot (r - r_0) &= 0, & \text{if } v_j \in V_r \end{aligned} \quad (21)$$

where t_0 and r_0 are the translation and rotation constraint values, respectively. In this work, Eq. Con. values are set to $t_0 = r_0 = 0$ to prevent motion updates along the degenerate directions. In practice, these constraint values can be set to any predetermined value to *steer* the optimization to the

correct minima to improve convergence. The determination of this value is an active research topic and is not covered in this work. As a consequence, the values of the constraints are set to 0, which means that optimization is limited to having updates on the linear combination of optimization variables x in the degenerate directions $v_j \in \{V_t, V_r\}$.

The ability to control the direction and amplitude of the motion updates is a key property of Eq. Con., directly benefiting the effectiveness against degenerate optimization problems. The underlying effect of these additional constraints is previously highlighted in Section III-B.

To utilize these constraints in the unconstrained cost function shown in (3), the equations are augmented with zero vectors to be applied to a 6-DoF optimization problem. The resulting augmented constraints are

$$\begin{aligned} [0_{1 \times 3}, v_j] \cdot x &= 0, & \text{if } v_j \in V_t \\ [v_j, 0_{1 \times 3}] \cdot x &= 0, & \text{if } v_j \in V_r. \end{aligned} \quad (22)$$

The constraints are then stacked into a matrix of form $Cx = d$ and integrated into the ICP problem formulation, via

$$\begin{bmatrix} 0_{m_r \times 3} & v_j \\ \vdots & \vdots \\ v_j & 0_{m_t \times 3} \end{bmatrix} x = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} \quad (23)$$

$C_{(m_r+m_t) \times 6} \qquad d_{(m_r+m_t) \times 1}$

where the number of Eq. Con. is denoted as m_t translational and m_r rotational ones, resulting in a total number of $c = m_t + m_r$ constraints.

The unconstrained ICP cost function shown in (3) can then be rewritten as a constrained optimization

$$\begin{aligned} \min_{x \in \mathbb{R}^6} & \quad \left\| Ax - b \right\|_2 \\ \text{s.t.} & \quad Cx - d = 0. \end{aligned} \quad (24)$$

Introducing Lagrangian multipliers can convert an equality-constrained optimization problem into an unconstrained optimization problem [13]. The Lagrangian of the minimization problem of (24) is defined as

$$\mathbb{L}(\mathbf{x}, \boldsymbol{\lambda}) = \mathbf{x}^\top \mathbf{A}^\top \mathbf{A} \mathbf{x} - 2\mathbf{b}^\top \mathbf{A} \mathbf{x} + \boldsymbol{\lambda}^\top (\mathbf{C} \mathbf{x} - \mathbf{d}) + \mathbf{b}^\top \mathbf{b} \quad (25)$$

where $\boldsymbol{\lambda} \in \mathbb{R}^c$ are the Lagrangian multipliers. Here, c indicates the number of constraints in the optimization. Interestingly, the effect of the constraints can be inferred by the magnitude of the Lagrangian multiplier the constraint is paired with. As the effect of the constraint gets larger, the magnitude of the Lagrangian multiplier increases accordingly. Similarly, if the magnitude of the Lagrangian multiplier is close to zero, the constraint has a minimal effect on the optimization. The effect of the constraints can be seen by comparing (4) and (25), with the only difference being the added constraint terms. Reformatting the Lagrangian into matrix form reveals the familiar least-squares formulation

$$\min_{\mathbf{x}' \in \mathbb{R}^{6+c}} \left(\underbrace{\begin{bmatrix} 2\mathbf{A}^\top \mathbf{A} & \mathbf{C}^\top \\ \mathbf{C} & \mathbf{0} \end{bmatrix}}_{\mathbf{A}'} \underbrace{\begin{bmatrix} \mathbf{x}^* \\ \boldsymbol{\lambda}^* \end{bmatrix}}_{\mathbf{x}'} - \underbrace{\begin{bmatrix} 2\mathbf{A}^\top \mathbf{b} \\ \mathbf{d} \end{bmatrix}}_{\mathbf{b}'} \right). \quad (26)$$

Here, $\mathbf{C} \in \mathbb{R}^{c \times 6}$ is the constraint matrix, relating the optimization variables \mathbf{x} to the constraints $\mathbf{d} \in \mathbb{R}^{c \times 1}$. Furthermore, the augmented optimization vector \mathbf{x}' constitutes the optimization variables and the Lagrangian multipliers. This augmented cost function can be solved using SVD, similar to the standard point-to-plane ICP cost function in (3). Theoretically, the activation of the degeneracy constraints can be investigated through the value of the Lagrangian multipliers $\boldsymbol{\lambda}$. The higher the amplitude of the multiplier, the more the constraint affects the optimization [97]. This notion suggests that the redundancy of the degeneracy mitigation constraints can be analyzed through $\boldsymbol{\lambda}$. In the context of this work, this method adds hard constraints to the optimization and, through the generation of the Lagrangian [see (25)], affects the error characteristics of the optimization. The relative error on \mathbf{b} and the residual \mathbf{r} are bounded by the added constraints, also bounding the sensitivity of the least-squares optimization problem to perturbations [see (14) and (10)].

2) INEQUALITY CONSTRAINTS

Although Eq. Con. provide a straightforward way to deal with degenerate directions, the system's systematic errors, such as eigenvector calculation errors, sensor noise, and the iterative nature of ICP, might require relaxation to escape local minima. In this scenario, the inequality constraints (Ineq. Con.) method provides a simple yet powerful alternative to Eq. Con. with a potentially larger convergence basin

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^6} & \quad \left\| \mathbf{A} \mathbf{x} - \mathbf{b} \right\|_2 \\ \text{s.t.} & \quad -\boldsymbol{\epsilon} \leq \mathbf{L} \mathbf{x} \leq \boldsymbol{\epsilon}. \end{aligned} \quad (27)$$

Here, the constraint matrix $\mathbf{L}_{6 \times 6}$ is generated as a row-major matrix based on the optimization's degenerate directions, with rows indicating the degenerate directions. As discussed in Section III-D, an X-ICP [13]-based degeneracy detection method retrieves the degenerate directions. The upper and lower bounds of the Ineq. Con. can differ, but for simplicity, both are set to the same magnitude vector, $\boldsymbol{\epsilon}$ of dimension 6×1 . Importantly, to account for the difference in scale between translation and rotation components, the amplitude of the rotational and translational components is set to $(\epsilon/2)$ and ϵ , resulting in $\boldsymbol{\epsilon} = [(\epsilon/2), (\epsilon/2), (\epsilon/2), \epsilon, \epsilon, \epsilon]^\top$. Here, ϵ is selected after heuristic tuning on the Ulmberg tunnel dataset, and set to $\epsilon = 0.0014$.

It is not trivial to solve inequality-constrained optimization problems as it is for Eq. Con. Thus, to find the solution, the above formulation is converted to a quadratic program (QP)

$$\begin{aligned} \min_{\mathbf{x} \in \mathbb{R}^6} & \quad \frac{1}{2} \mathbf{x}^\top \mathbf{F} \mathbf{x} + \mathbf{f}^\top \mathbf{x}, \\ \text{s.t.} & \quad -\boldsymbol{\epsilon} \leq \mathbf{L} \mathbf{x} \leq \boldsymbol{\epsilon}. \end{aligned} \quad (28)$$

Here, $\mathbf{F} = 2\mathbf{A}^\top \mathbf{A}$ and $\mathbf{f} = -2\mathbf{A}^\top \mathbf{b}$. The QP problem formulation suits the cases where \mathbf{F} might be ill-posed and can handle different types of constraints (equality and inequality at the same time); however, for the sake of clear comparison, only Ineq. Con. are used in the investigation presented in this work. The open-source QP library² QPmad [98] is utilized to solve this QP problem. QPmad internally employs the Goldfarb and Idnani dual primal method [99]. The minimum of the objective function, subject to the current active constraint set, is calculated at every iteration [98]. The additional Ineq. Con. can be removed from the active constraint set if the constraint is no longer active, which relaxes the constrained optimization. The state of the Ineq. Con. can be inferred from its dual variables (Lagrangian multipliers). If the constraints are not activated, the bounds of the Ineq. Con. $\boldsymbol{\epsilon}$ are larger in magnitude than the motion update in that direction. Similar to Eq. Con., this method adds additional degeneracy-aware constraints to the optimization to provide bounds for the relative error and the residual \mathbf{r} . Subsequently, the effect of the perturbations on the optimization variables is reduced, as discussed in Section III-B.

3) SOLUTION REMAPPING

Solution remapping denotes the optimization degeneracy mitigation method proposed by Zhang et al. [19], shown in Fig. 4 as Zhang et al. method. This method is referred to as Zhang et al. from now on. In this method, a solution projection matrix is constructed from the eigenvalue decomposition of the optimization Hessian, which is used to project the degenerate optimization solution \mathbf{x} only in well-constrained directions. First, this method generates an augmented eigenvector matrix \mathbf{V}_{aug} based on the degeneracy detection by setting the corresponding eigenvectors in \mathbf{V}

²<https://github.com/asherikov/qpmad>

to $\mathbf{0}$. Using the introduced definitions in Section III, the reprojection matrix is calculated as

$$\mathbf{S} = \mathbf{V}_{\text{aug}}^{-1} \mathbf{V}. \quad (29)$$

After minimizing (3), the reprojection matrix is applied to the solution \mathbf{x}^* via

$$\mathbf{x}_{\text{aug}}^* = \mathbf{S} \mathbf{x}^*. \quad (30)$$

This algorithm relies on the notion that the optimization degrades only in the degenerate directions, and identification of the null-space of the Hessian is sufficient to reproject the solution on the well-constrained directions of the optimization. As in this work, the null space of the Hessian is assumed to be available; only the reconstruction of the projection matrix is required. In the context of this work, Zhang et al. does not interfere with $\text{cond}(\mathbf{A})$ or the residual \mathbf{r} but projects the solution \mathbf{x} to the well-constrained directions of the optimization. Consequently, the noise in the input can be amplified during the least-squares optimization (see Section III-B). Interested readers are referred to the original work [19] for more detailed derivation and explanations.

4) TRUNCATED SVD

TSVD is a linear algebra method to handle ill-conditioned matrices with high-condition numbers to effectively reduce their rank and improve their computational stability [100], [101]. The TSVD of a symmetric and semi-positive definite matrix involves truncating the smallest eigenvalues while retaining only the largest eigenvalues. As a result, producing a k -rank approximation of a matrix where k refers to the top- k retained eigenvalues of the matrix. TSVD requires the eigenvalue decomposition for the Hessian defined in (3), which is given as

$$\mathbf{A} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^{\top}. \quad (31)$$

Next, the diagonal scaling matrix $\mathbf{\Sigma}$ is truncated by overwriting the eigenvalues $\sigma_i \forall i \in \{1, \dots, 6\}$ in the diagonal entries, based on whether the corresponding eigenvector of the eigenvalue aligns with a degenerate direction. If an eigenvector is along a degenerate direction, then $\sigma_i^{tr} = 0$, and if not, then $\sigma_i^{tr} = (1.0/\sigma_i)$. Here, the truncated diagonal entries are denoted with $\sigma_i^{tr} \forall i \in \{1, \dots, 6\}$. Setting $\sigma_i^{tr} = 0$ effectively creates a rank deficiency which LAPACK SVD routines.³

The truncated Hessian can be directly used to find the solution \mathbf{x}^*

$$\begin{aligned} \mathbf{x}^* &= \mathbf{A}_{tr}^{-1} \mathbf{b} \\ \mathbf{A}_{tr}^{-1} &= \mathbf{V} \mathbf{\Sigma}_{tr} \mathbf{U}^{\top}. \end{aligned} \quad (32)$$

The $\mathbf{\Sigma}_{tr}$ the truncated diagonal scaling matrix is defined as follows:

$$\mathbf{\Sigma}_{tr} = \begin{bmatrix} \sigma_1^{tr} & & & & & \\ & \ddots & & & & \\ & & 0 & & & \\ & & & \ddots & & \\ & & & & 0 & \\ & & & & & \sigma_6^{tr} \end{bmatrix}. \quad (33)$$

³<https://netlib.org/lapack/explore-html-3.6.1/d1/d7e/>

As $\mathbf{\Sigma}_{tr}$ is constructed through the truncated eigenvalues, it omits the under-constrained directions from the decomposition. By truncating the eigenvalues, the constraints to mitigate the motion along the degenerate directions are directly integrated into the Hessian. By design, \mathbf{A}_{tr} is rank-deficient; however, as SVD exists for all matrices, (15)–(18) can be used to acquire the solution.

a) ON EFFECTIVENESS OF TSVD

The Eckart–Young theorem [101] provides detailed insights for the TSVD method. According to the theorem, the best least-squares approximation of \mathbf{A} by a rank k is \mathbf{A}_{tr} , and the truncation approximation 2-norm error is defined as the sum of the truncated eigenvalues

$$\|\mathbf{A} - \mathbf{A}_{tr}\|_F = \sqrt{\sum_{i=k+1}^6 \sigma_i^2}. \quad (34)$$

Here, F indicates the Frobenius norm. In addition to this observation, a closer look at (32) shows the increased sensitivity to noise in \mathbf{b} for small eigenvalues. By truncating these small eigenvalues, the noise amplification effect is mitigated by TSVD. An example of noise in \mathbf{b} would be errors in surface normal calculation and correspondence search. The effect of perturbation on the quality of the output of the least-squares minimization problem is discussed in Section III-B; in this context, by truncating the near-singular eigenvalues, TSVD changes the condition number $\text{cond}(\mathbf{A})$, the noise in \mathbf{b} , the residual \mathbf{r} , and the relative error.

5) TIKHONOV REGULARIZED LINEAR LEAST-SQUARES

Regularization, commonly employed in optimization techniques such as LM, offers an easy way to escape singularities and improve optimization characteristics. Although the characteristics and properties of Tikhonov regularization are well-established in the linear algebra community [102], the benefits in the field of degeneracy-aware point cloud registration have not yet been investigated. Motivated by this idea, linear optimization with Tikhonov regularization (L-Reg.), shown in Fig. 4 as L-Reg., is employed for a subspace of the linear least-squares minimization based on the null-space approximation provided by the localizability information. The previously defined cost function in (3) can be rewritten as a regularized linear least-squares problem as follows:

$$\begin{aligned} \min_x \|\mathbf{A} \mathbf{x} - \mathbf{b}\|_2^2 + \lambda \|\mathbf{L} \mathbf{x}\|_2^2 \\ = \min_x \left\| \begin{pmatrix} \mathbf{A} \\ \sqrt{\lambda} \mathbf{L} \end{pmatrix} \mathbf{x} - \begin{pmatrix} \mathbf{b} \\ \mathbf{0} \end{pmatrix} \right\|_2^2. \end{aligned} \quad (35)$$

Here, the scalar λ is the linear regularization parameter, and $\mathbf{L} \in \mathbb{R}^{c \times 6}$ is the constraint matrix, with c denoting the number of degenerate directions. The number of degenerate directions is provided by the localizability detection block discussed in Section III-D. Following a similar derivation

to (5), the normal equations of the regularized minimization are

$$\left(A^\top A + \lambda L^\top L \right) x = A^\top b. \quad (36)$$

Immediately, λ can be identified as the regularization scaling parameter to the additional cost matrix $L^\top L$ induced by the regularization term. This term provides the penalization needed to minimize the projection of the solution onto the null-space approximation provided to the optimization [by improving the conditioning of the problem (see Section III)]. The regularization parameter $\lambda > 0$ is not known a priori and has to be determined based on the problem data. In this work, λ is selected after heuristic tuning on the Ulmberg Tunnel dataset similar to Ineq. Con. method, and set to $\lambda = 440$. The constraint matrix L is set as a row-major matrix consisting of the degenerate directions of the optimization, $v_j \in \{\{V_t, V_r\} \in \mathbb{R}^{3 \times 3}\}$. Equation (36) can be solved with SVD as described in Section III.

a) UNDERSTANDING THE TIKHONOV REGULARIZATION

It is important to understand the effect of regularization on the underlying optimization. An analysis for the general case of $L \neq I$ requires the solution of the generalized SVD of (A, L) . Hence, for the brevity of understanding of the effects, the constraint matrix is set as $L = I$ for the following explanation. Similar to (18), given the SVD of A , the steps to obtain the solution x_λ are as follows:

$$\left(V \Sigma^\top \underbrace{U^\top U}_{=I} \Sigma V^\top + \lambda \underbrace{I}_{=V V^\top} \right) x_\lambda = V \Sigma^\top U^\top b. \quad (37)$$

Similar terms are bundled together and rearranged, which reveals how the regularization affects the Hessian. Multiplying and setting $y = V^\top x_\lambda$, the equation can be simplified to

$$\left(\Sigma^\top \Sigma + \lambda I \right) y = \Sigma^\top U^\top b. \quad (38)$$

Since V is orthogonal, the 2-norm of y is the same as x_λ . The solution to the regularized problem can then be derived using the dyadic decomposition

$$x_\lambda = \sum_{i=1}^6 z_i \frac{(u_i^\top b)}{\sigma_i} v_i \quad z_i = \frac{\sigma_i^2}{\sigma_i^2 + \lambda}. \quad (39)$$

Here, z is the filter coefficient damping the eigenvalues $\sigma_i < \lambda$ of A and subsequently eigenvalues $\sigma_i^2 < \lambda$ of $A^\top A$. Hence, the regularization mainly affects the degenerate directions and effectively changes the Hessian's eigenvalues. Finally, as shown in (39), this method alters the eigenvalues of the optimization Hessian and subsequently improves the condition number $\text{cond}(A)$ which in return improves the sensitivity characteristics of the least-squares minimization problem as shown in (10) and (14).

6) NONLINEAR OPTIMIZATION WITH TIKHONOV REGULARIZATION

This section introduces the nonlinear optimization-based regularization technique, nonlinear optimization with Tikhonov

TABLE 1. Nonlinear optimizer configuration for the NL-Reg. method.

	Value	Default
Parameter Tolerance	10^{-3}	10^{-8}
Function Tolerance	10^{-3}	10^{-6}
Gradient Tolerance	10^{-6}	10^{-10}

regularization (NL-Reg.), shown in Fig. 4 as NL-Reg. Previous works such as [103] and [28] utilize nonlinear cost functions to improve the ICP behavior. In particular, Nashed et al. [28] proposed to incorporate the null space of the optimization as a regularization term for the nonlinear cost function in the context of factor graph optimization. Motivated by this, Tikhonov regularization is employed for point cloud registration with the nonlinear cost function as

$$E_{\text{NL}}(x) = \sum_{i=1}^N \left\| \underbrace{((R p_i + t) - q_i) \cdot n_i}_{e_i} \right\|_2 + \lambda_D \|Lx\|_2^2 \quad (40)$$

where $x = [r^\top, t^\top]^\top \in \mathbb{R}^6$ are the optimization variables. Moreover, λ_D is the degeneracy regularization parameter, and L is the row-major constraint matrix consisting of the degenerate directions of the optimization. Similar to previous methods, the value of λ_D is selected after heuristic tuning on the Ulmberg tunnel dataset, and set to $\lambda_D = 675$. The well-known LM algorithm is used for the minimization of the cost function in (40). The Jacobian, gradient and Hessian of cost $E(x)_{\text{NL}}$ with respect to x are defined as

$$\begin{aligned} J_{\text{NL}} &= \frac{\partial e}{\partial x} \\ \nabla E_{\text{NL}}(x) &= 2J_{\text{NL}}^\top e_i + 2\lambda_D L^\top Lx \\ H_{\text{NL}} &= \frac{\partial^2 E_{\text{NL}}(x)}{\partial x^2} = 2J_{\text{NL}}^\top J_{\text{NL}} + 2\lambda_D L^\top L. \end{aligned} \quad (41)$$

Utilizing these definitions, the update rule of the LM algorithm is derived as follows:

$$x_{\text{new}} = x - (H_{\text{NL}} + \lambda_s I)^{-1} \nabla E_{\text{NL}}(x). \quad (42)$$

Here, λ_s refers to the smoothness parameter of the LM algorithm, ensuring the stability of the matrix inversion. As discussed in the Ceres documentation, the magnitude of the state correction might lead to a nonconvergent algorithm. Hence, a trust region constraint is added to the solution. The reader is referred to the documentation of Ceres for more implementation details on robust nonlinear least-squares. For clarity, the altered optimization termination tolerances are provided in Table 1.

As this method utilizes the LM algorithm to solve the nonlinear cost function (40), the sensitivity analysis in Section III-B does not apply directly. Nevertheless, NL-Reg. method improves the conditioning of the Hessian H_{NL} by adding a degeneracy-aware regularization with λ_D . Besides,

the LM algorithm is inherently more robust against ill-conditioned problems due to the regularization of the damping parameter λ .

7) EXTERNAL PRIOR

In this approach, if optimization ill-conditioning is detected, the initial guess $T_{ML,init}$ is directly used for point cloud registration. Consequently, this operation renders point cloud registration ineffective by skipping the optimization process altogether. This method relies on external odometry estimates during degeneracy to propagate the solution; hence, it is denoted and shown in Fig. 4 as *Prior Only*. As this method completely skips the point cloud registration step in the event of optimization degeneracy, the least-squares optimization step of point cloud registration is skipped entirely. Since there are no pose updates from point cloud registration, the accuracy of this method depends on the accuracy of the external pose prior while LiDAR degeneracy is present.

B. PASSIVE DEGENERACY MITIGATION

Passive degeneracy mitigation refers to the methods that are not *actively* utilizing the detected degeneracy information to affect optimization. Instead, these methods use this information passively or rely on algorithms that are supposedly less affected by disturbances and degeneracy.

1) GLOBALLY OPTIMAL POINT CLOUD REGISTRATION

In Quatro [52], the authors define *degeneracy* in a globally optimal point cloud registration context as the *absence of inlier correspondences* in the matching process. This definition differs from the LiDAR degeneracy definition used in this work (see Section III), which focuses on the inherent lack of geometric information the environment provides along specific directions. Quatro [52] is a robust globally optimal point cloud registration method estimating 4-DoF transformation while requiring less than three correspondences. Fast global registration (FGR) [51] is a well-known global registration method that performs a single-stage objective minimization to find the transform between two overlapping surfaces. Section V-A2 highlights that these globally optimal point cloud registration frameworks are not able to compensate for the lack of geometric constraints provided by the environment; inlier point correspondences also do not constrain the optimization along symmetric and self-similar directions of motion.

2) M-ESTIMATORS

To analyze the efficacy of robust outlier rejection methods in point cloud registration, this work investigates the effectiveness of the Cauchy robust cost function, which is an M-Estimator, in combination with the median of absolute deviation (MAD) scale estimator [61]. In the remainder of this work, this method is referred to as *Cauchy* for simplicity and is shown in Fig. 4 with the same name. A nonconvex robust cost function can be used in an iterative reweighted

least squares (IRLS) [104] fashion, adapting the point cloud registration cost function defined in (3) as

$$\min_{x \in \mathbb{R}^6} \rho \left(\underbrace{\|Ax - b\|_2}_e \right). \quad (43)$$

Babin et al. [61] found the Cauchy robust cost function $\rho(\cdot)$ to be effective against the presence of outliers, it is selected as the cost function studied in this analysis. The corresponding function and the weight $w(\cdot)$ are formulated as follows:

$$\rho(e) = \frac{\kappa^2}{2} \log \left(1 + \left(\frac{e}{\kappa} \right)^2 \right), \quad w(e) = \frac{1}{1 + (e/\kappa)^2}. \quad (44)$$

Here, e refers to the error term induced by each correspondence pair in the iterative process of ICP, and κ refers to the function tuning parameter. Here, the weight function $w(\cdot)$ is the decomposed weight of the nonconvex $\rho(\cdot)$ cost function. For details of the MAD scale estimator and how to solve the IRLS problem, the readers are referred to [61]. As suggested in [61] the M-Estimator parameter $\kappa = 1$ has been selected. Finally, using the weight calculated in (44) the cost function (3) is rewritten as

$$\min_{x \in \mathbb{R}^6} w(e) \|Ax - b\|_2. \quad (45)$$

In the context of this work, the Cauchy method improves the sensitivity of the least-squares problem by reducing the noise levels both in b and A and, subsequently, mitigates the adverse effects of noise amplification in ill-conditioned cases, however, do not alter the condition number of A .

3) GEOMETRICALLY STABLE SAMPLING

As one of the seminal works on using sampling to improve the conditioning of ICP, Gelfand et al. [15] proposed pairwise information sampling from correspondence pairs (p, n) . In this method, point and surface normal pairs are sampled from the set of correspondences until a number of samples are collected or the conditioning threshold is met. Specifically, sampling is performed subsequently for all eigenvectors v_k , $k \in \{1, \dots, 6\}$

$$f_i = \left(\left[p_i \times n_i^\top, n_i^\top \right] \cdot v_k \right)^2, \quad i \in \{1 \dots N\}$$

$$\hat{\sigma}_k \approx \sum_{i=0}^N f_i. \quad (46)$$

This approximation of the eigenvalues by summation of information allows for a graceful adjustment of the Hessian condition number. Specifically, the criterion is to resample the correspondences to construct a Hessian with a condition number as close to 1 as possible. This method provides an effective means of improving the condition number $\text{cond}(A)$ by resampling the information from the input data and consequently preventing the oversensitivity of the least-squares solution to perturbations as discussed in Section III-B. This method is depicted as Gelfand et al. in Fig. 4, and the

open-source implementation in *libpointmatcher* [92] is used to replicate in this work.

4) REDUNDANCY MINIMIZING SAMPLING (RMS)

As discussed in Section II, Petracek et al. [14] proposed rms as a means of minimizing redundancy in the input point cloud and, while, maximizing the entropy of the data in the same point cloud to retain information. Specifically, this method maximizes the entropy of information while keeping the Hessian well-conditioned by sampling points from the input point cloud in all directions of the optimization. The authors in this work introduce a heuristic called *gradient flow*, which quantifies the uniqueness of a point with the flow of the geometric gradient. The gradient flow of each point p within a spherical neighborhood with radius λ_p as

$$\begin{aligned} \mathcal{N}_p &= \{j \mid \|j - p\|_2 < \lambda_p, j \neq p, j \in \mathcal{P}\} \\ \Delta_p &= \frac{1}{|\mathcal{N}_p|} \sum_{j \in \mathcal{N}_p} (j - p). \end{aligned} \quad (47)$$

The gradient flow-based data entropy maximization is defined as

$$\hat{\mathcal{P}} = \underset{\Omega \in \{\Theta \mid \Theta \subseteq \mathcal{P}_v, \Theta \neq \emptyset\}}{\arg \max} H_{\Delta}(\Omega). \quad (48)$$

Here, \mathcal{P}_v corresponds to the voxelized point cloud set and Ω to a subset of \mathcal{P}_v . Moreover, this maximization is conditioned on the normalized relative entropy rate to encourage early termination. Although this method does not balance the rotation-space observability, the authors specify that this is only relevant in high degrees of rotation. The readers are referred to the original work for a more detailed explanation and derivation of this method. Similar to Gelfand et al., this method provides a means of sampling the input point cloud to maximize information gain and improve the condition number $\text{cond}(\mathbf{A})$ while minimizing the information redundancy to decrease possible noise in \mathbf{b} and \mathbf{A} . As a result, this method mitigates the amplification effect of a sensitive least-squares optimization problem, as discussed in Section III-B. In the context of this work, rms is considered a passive degeneracy mitigation method, as it alters the input point cloud instead of actively using the detected degeneracy information. This method is depicted as Petracek et al. in Fig. 4 and in the rest of this work. The open-source implementation of the original work⁴ is used to realize this work.

5) POINT-TO-PLANE ICP

Point-to-plane refers to the ICP formulation using the point-to-plane [54] cost function as previously detailed in Section III. This method is referred to as P2Plane in the remainder of this work. point-to-plane ICP (P2Plane), is a passive baseline method and is shown in Fig. 4 with the same name. The results of the baseline method show the adverse effects of LiDAR degeneracy on point cloud registration with linear solvers and LiDAR pose estimation

if no degeneracy-aware actions are taken to mitigate the effects of optimization ill-conditioning. The optimization cost function of this method is defined in (3). As this method does not change the sensitivity properties of the problem, it serves as a baseline in the evaluation.

6) NONLINEAR POINT-TO-PLANE ICP

This baseline method, nonlinear solver (NL-Solver), is introduced to understand the effect of the solver. Specifically, this method uses the cost function introduced in (40) and (42) using $\lambda_D = 0$. Similar to the NL-Reg. method, this baseline uses Ceres to implement the cost function and uses the LM algorithm with default regularization. The optimization termination criteria for this method are determined by the parameters defined in Table 1. This method is shown as NL-Solver in Fig. 4. Compared to the P2Plane baseline method, NL-Solver uses the gradient-based LM solver. Finally, NL-Solver does not interfere with the conditioning of $\text{cond}(\mathbf{A})$ with degeneracy-aware regularization; however, it uses the damping parameter λ to improve stability during the update step of the LM algorithm.

C. IMPLEMENTATION DETAILS

All methods described in this section are implemented in the *libpointmatcher* [92] open-source registration library and are integrated into the *Open3d SLAM* [10] framework as the scan-to-submap registration module. *Open3d SLAM* runs at the LiDAR rate of 10 Hz, and utilizes external odometry pose estimates as the registration prior $T_{\text{ML}, \text{init}}$, enabling scan-to-submap nearest-neighbor search to identify the point correspondences. The surface normal of a point is calculated using the k-nearest-neighbor search and principal component analysis, where the neighborhood is set as $k = 10$. The LiDAR scans are cut off at a 70 m maximum distance and processed without downsampling. Furthermore, all methods use the trimmed outlier filter to replicate a realistic system. In addition, the maximum number of iterations for the point-to-plane ICP is set to 30 iterations. The ICP algorithm terminates early if the norm of the pose update drops to 0.01 m for translation or if the angular distance between updates is less than 0.001 rad. These values are based on the default values used in *libpointmatcher*.

Moreover, as discussed in Section II-B, a localizability module similar to X-ICP [13] is used as a localizability source for all methods requiring localizability information. Specifically, the X-ICP is simplified to do binary localizability classification, namely, localizable and nonlocalizable, to prevent giving an unintentional advantage to more complex methods. This method requires three localizability parameters, $\kappa_1, \kappa_2, \kappa_3$, and 1 filtering parameter κ_f . The localizability parameters are set based on the basin of convergence of the employed SLAM framework and the ICP algorithm. Furthermore, these parameters are set as $\kappa_1 \geq \kappa_2 > \kappa_3$. For all experiments in this work, the parameters are kept the same and set as $\kappa_1 = 300, \kappa_2 = 150, \text{ and } \kappa_3 = 45$. These values are set only once according to the voxelization and filtering

⁴<https://github.com/ctu-mrs/rms>

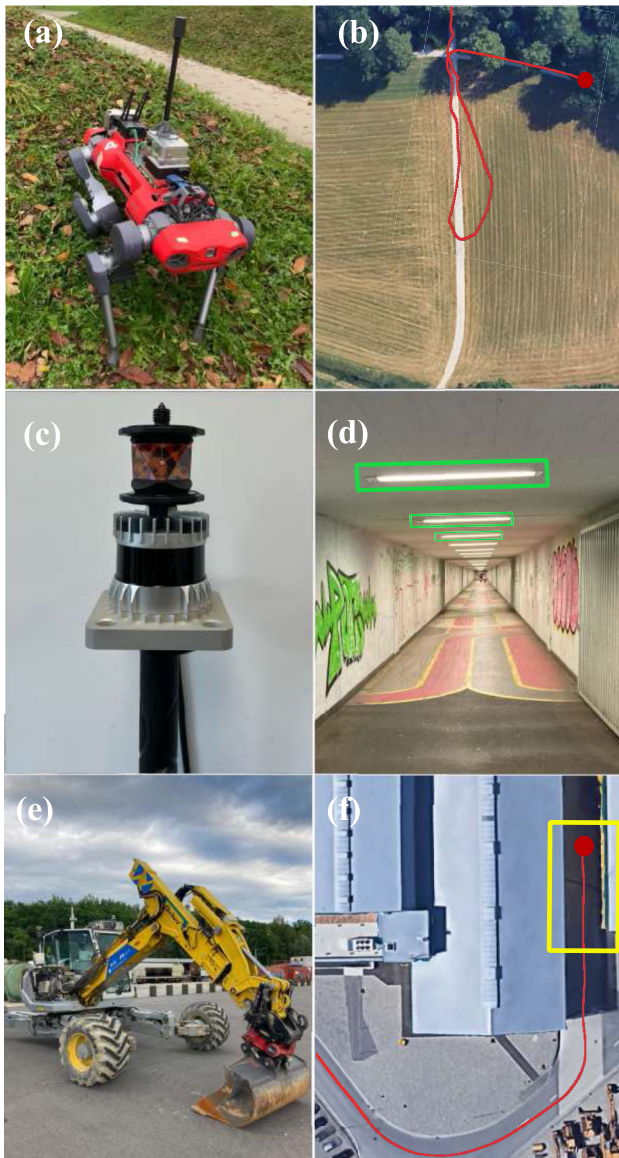


FIGURE 5. Illustration of three real-world deployments with environmental LiDAR degeneracy. (a) ANYmal robot in the deployed environment. (b) Robot's GPS trajectory (in red) over a satellite image of the deployed environment is shown. The shown open field is subject to 3-axis LiDAR degeneracy. (c) Handheld sensor payload of the Ulmberg bicycle tunnel [33] experiment is shown. (d) Degenerate bicycle tunnel is shown. In addition, the sunken LED light housings are highlighted in green, as these housings are the only identifiable geometric features along the tunnel direction. (e) HEAP [105] walking excavator. (f) Construction environment for the excavator experiment is shown with the trajectory of HEAP overlaid as a red line. The LiDAR degenerate narrow-passage (highlighted in yellow) at the start (red dot) poses a significant challenge to LiDAR-SLAM solutions.

characteristics of *Open3d SLAM*. Moreover, the filtering parameter is set as $\kappa_f = 80$ for all experiments except the HEAP excavator experiment discussed in Section V-D. For that experiment, it is set to $\kappa_f = 60$ due to an older and noisy LiDAR variant Ouster OS0 Rev-1.

The parameter dependent methods L-Reg., NL-Reg., and Ineq. Con. are tuned based on performance in the Ulmberg bicycle tunnel experiment (see Section V-C). In addition, the rms method by Petracek et al. [14] and Gelfand et al. also have a single tunable parameter each. The tunable parameter of rms λ is set to 0.0042 for experiments discussed in Sections V-C and V-D. For the remaining experiments, the value is set to 0.0036. These values are obtained after testing rms with different λ values and empirically selecting the most successful ones. For the method of [15], when the number of points in a scan exceeds 1000, the maximum number of sampled points is set to 60% of the total points. The limit for scans with fewer points is set to 90% of the available points. All evaluations were performed on a desktop computer with an *Intel i9 13900K CPU*, and to guarantee repeatability and deterministic execution, all operations are performed in a single-threaded fashion.

V. EXPERIMENTS AND EVALUATION

In this section, the different degeneracy mitigation methods are analyzed in depth using simulation studies and real-world field experiments to identify the advantages and weaknesses of each method.

First, a simulated iterative degenerate point cloud registration experiment is performed in Section V-A to determine the feasibility and provide a deeper understanding of each method. Later, a simulated global point cloud registration study is discussed in Section V-A2, showing the impact of global point cloud registration methods in mitigating LiDAR degeneracy. Finally, in Section V-A1, the compared methods are tested on simulated data with *Open3D-SLAM* [10] in the loop.

In addition, the methods are tested in different real-world robotic field deployments as shown in Fig. 5, that include natural open fields (see Section V-B), urban tunnels (see Section V-C), and construction sites (see Section V-D). The real-world tests are also conducted using a diverse combination of robotic platforms and sensor payloads, namely, i) the ANYmal legged robot [106] equipped with a Velodyne VLP-16 LiDAR (see Section V-B), ii) the HEAP [105] excavator equipped with an Ouster OS0-128 LiDAR (see Section V-D), and iii) the ENWIDE hand-held sensor pack⁵ [33] equipped with an Ouster OS1-128 LiDAR (see Section V-C).

Each experiment introduces a unique subset of real-world challenges for the existing point cloud registration frameworks, allowing for a detailed analysis of the compared methods. Importantly, some of these unique challenges render the possibility of recovering the true point cloud map of the environment infeasible. For example, in the Ulmberg tunnel experiment (see Section V-C), the LiDAR degeneracy continues for more than 80% of the dataset, making it infeasible for LiDAR-based solutions to provide a reasonable map. In this kind of scenario, the performance is mainly

⁵<https://projects.asl.ethz.ch/datasets/enwide>

TABLE 2. Challenge summary of the experiments presented in Section V. ✓ indicates positivity and ✗ indicates negativity.

	Good Motion Prior	Severe Degeneracy	Long Degeneracy Duration	Is recovery feasible? ^a
Static Sim. (Section V-A-1)	✗	✓	-	✗
Global Sim. (Section V-A-2)	✗	✓	-	✗
Dynamic Sim. (Section V-A-3)	✓	✓	✓	✓
Ulmer Tunnel (Section V-C)	✓	✗	✓	✗
ANYmal Forest (Section V-B)	✗	✗	✗	✓
HEAP Excavator (Section V-D)	✓	✓	✗	✓

^aHere, recovery feasibility indicates the availability of sufficient information in the environment.

driven by better usage of the external motion prior to mitigate the effects of LiDAR degeneracy. In contrast, the experiments discussed in Sections V-B and V-D either have mild LiDAR degeneracy where some geometric information is still present or have a short duration of LiDAR degeneracy, which the scan-to-submap registration can recover from. In accordance with the challenges mentioned above, the level of degeneracy and expected behavior of the investigated methods are summarized in Table 2.

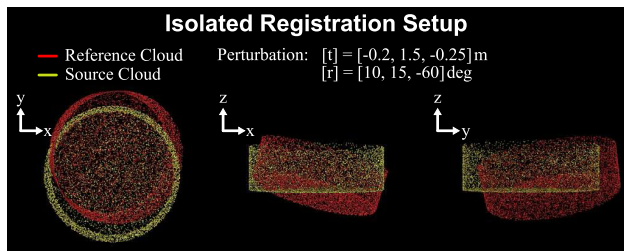


FIGURE 6. Setup of the static simulation experiment is shown. The red reference point cloud is shifted by r and t relative to the yellow source point cloud. The registration goal is to find the best-aligning transformation between the two point clouds.

A. SIMULATION STUDIES

A set of simulated experiments is conducted to understand the applicability and effectiveness of each method. First, an iterative point cloud registration experiment is conducted in Section V-A1 to isolate the point cloud registration step from the mapping framework. The registration step includes two point clouds with known perturbation and initial transformation guesses. Second, Section V-A2 analyzes the feasibility of globally optimal point cloud registration methods using the same setup. Finally, a simulated walking ANYmal [106] robot experiment is conducted with *Open3d-SLAM* in the loop to perform LiDAR-SLAM during a simulated robotic application.

1) STATIC SIMULATION — ITERATIVE POINT CLOUD REGISTRATION

In this experiment, each method is provided with source and reference point clouds, shown in Fig. 6, sampled from a simulated, perfectly cylindrical, and symmetric mesh. The reference point cloud is perturbed by a known transformation, and an identity initial guess associates these point clouds.

The purpose of this experiment is twofold: first, to analyze the performance of each method in a rotation-only (rotation around the Z-axis), 1-DoF, degenerate setting, and second, to inspect the evolution of the solutions per iteration of the ICP algorithm. The Prior Only method is omitted, as this experiment focuses on the registration task, and the Prior Only method skips the registration step. Similar to other experiments, the methods are employed for an ICP registration task with a maximum iteration count of 30. The results are shown in Fig. 7.

First, as seen from the top left plot in Fig. 7, the nonlinear optimization-based methods NL-Reg. and NL-Solver have higher computational costs compared to the other methods. In addition, sampling-based methods took more time in the beginning, possibly due to a lower number of correspondences. Other methods are comparable in computational complexity and suited for real-time deployment. Second, the ICP residual is minimized for all methods to a similar magnitude after a spike in the beginning (top middle) for all methods except Cauchy due to a rapid increase in correspondence inliers (top right). However, this minimized cost does not necessarily reflect convergence to the actual transformation; it only indicates a successful residual minimization. In particular, this can be seen in the plot of the Frobenius distance as many methods that have reduced the ICP residual, in fact, have increased the Frobenius distance to the true transform. The reason is that every rotation along the axis of the degenerate rotation direction is in fact a local-minima for the optimization. However, all active degeneracy mitigation methods correctly utilize degeneracy information and prevent further increases in rotational error while decreasing the translation error (center middle). Interestingly, the difference in internal dynamics of the active degeneracy mitigation methods is revealed by comparing the total motion residual in the degenerate direction (rotation around Z, center right). As expected, Eq. Con. does not produce any motion residual as the degeneracy constraints are utilized as Eq. Con. Among the soft-constrained methods, NL-Reg. allows some motion until iteration 10 while Ineq. Con. allows a linear *creep* motion as the optimization can move along the degenerate direction in every registration step. In contrast, L-Reg. allows for motion in the degenerate direction, possibly due to a lower regularization parameter. Finally, TSVD and Zhang et al. perform very similarly since these methods both manipulate the eigenvectors of the optimization through eigenvalues, either with k -rank approximation or solution projection.

2) STATIC SIMULATION—GLOBAL POINT CLOUD REGISTRATION

In this experiment, the global point cloud registration methods Quatro [52] and FGR [51], discussed in Section IV-B1, are employed for a similar task as in Section V-A1. Moreover, similar to Section V-A1, a 1-axis rotational degenerate point cloud is used, as shown in Fig. 8, but with a different perturbation from the reference point cloud

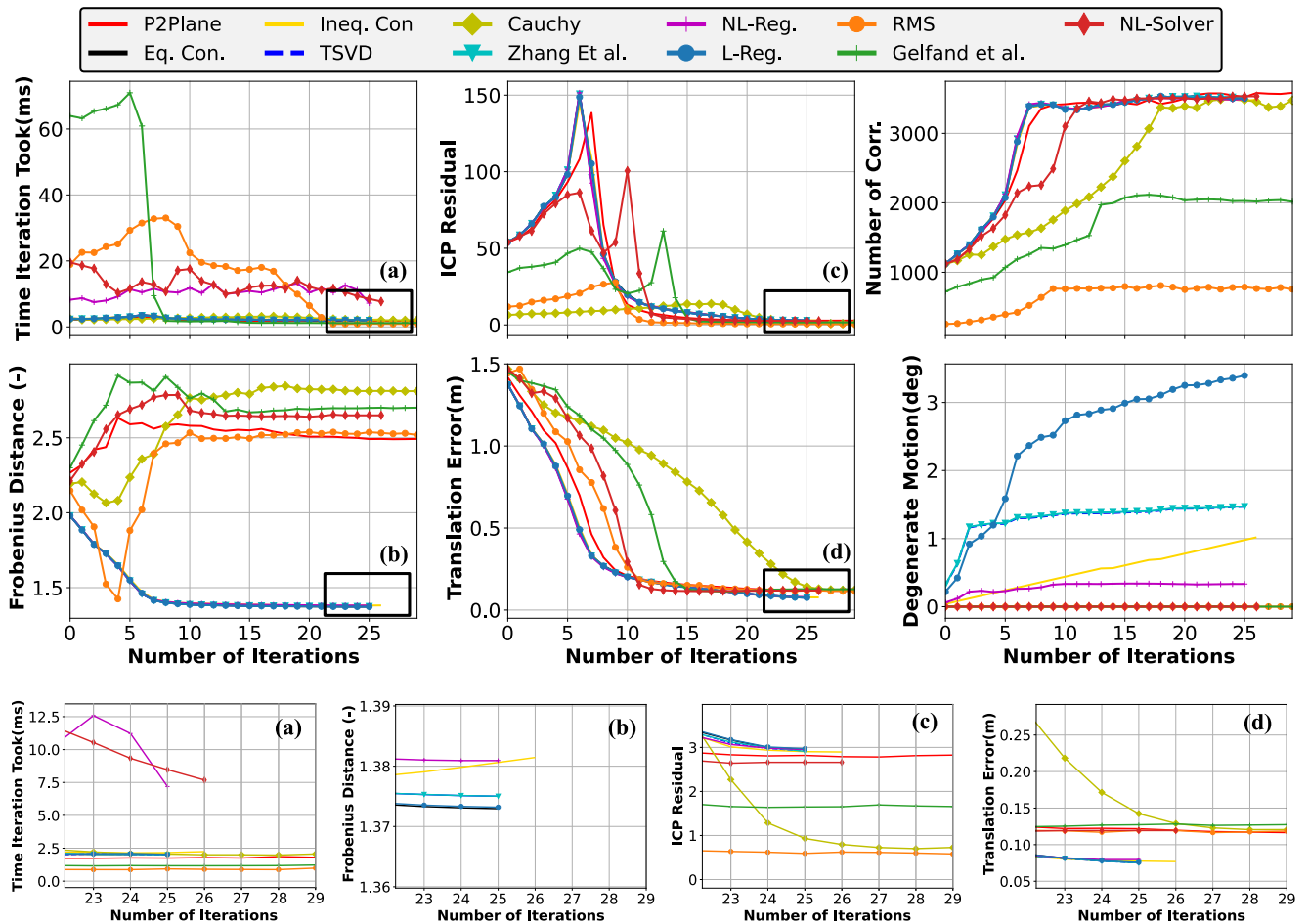


FIGURE 7. Analytics of the degenerate point cloud registration for the iterative registration experiment (see Section V-A1). Top left: Computational time for each iteration. Top middle: The residual cost of the ICP formulation at the time of each iteration. Top right: The number of correspondences at the time of each iteration. Center left: The Frobenius distance to the true transform (the y-axis is unitless). Center middle: The translation errors of the registration. Center right: The motion residual along the degenerate direction during the registration process. The sub-figures indicated with (a)–(d) provide a close-up view of their respective main figures for ease of comparison of methods.

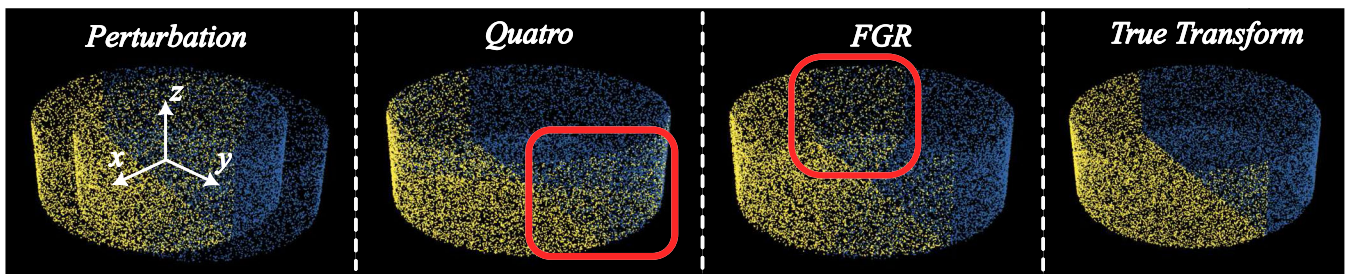


FIGURE 8. Results of Quatro [52] and FGR [51] are shown with the resulting transformation. The mismatch of the source and reference clouds is highlighted in red. The point clouds are colored to visualize the final transformation, with the ideal outcome depicted on the right. The perturbation is $t = [0.2, 1.5, -0.25]$ m for translation and $r = [0^\circ, 0^\circ, -60.00^\circ]$ for rotation.

for rotation around the Y - and X -axes, since Quatro can only estimate the 4-DoF transforms. Multiple configurations have been employed for both methods to ensure objective comparison, and the best performing configuration is used in the generation of Fig. 8. The purpose of this experiment is to highlight the fact that the degenerate point cloud registration problem is not a correspondence outlier problem but, in fact,

an absence of geometric information within correspondences, i.e., absence of true inliers problem. Specifically, this experiment underlines the importance of degeneracy detection and mitigation and shows that in the absence of inliers, even globally optimal point cloud registration methods are unable to retrieve the true transformation. As this is an absence of data problem, the expected output from these methods is not

the recovery of the true transform but, in fact, mitigation of further adverse effects of the degeneracy.

As shown in Fig. 8, both methods recovered translation very well and, as expected, but failed to retrieve the rotation around the degenerate axis (i.e., rotation around the Z-axis) correctly, as illustrated by the mismatched colors of the point clouds. This result is expected since the rotation around the degenerate axis is unobservable, and termination of the optimization after convergence of all well-conditioned directions is considered a successful optimization. This experiment supports the notion that global registration methods are not *drop-in* solutions to the problem of degenerate point cloud registration, as for the LiDAR point cloud this problem is inherently ill-defined, and there are many minima for the optimization solution. Many global registration methods are robust to outliers and can recover the true minima from noisy data *given the presence of inliers*. The assumption of the presence of the inliers no longer holds for the problem of degenerate point cloud registration, and as a result, these methods are unable to provide additional benefits to the other methods described in Section IV.

3) ANYmal SIMULATION

In this experiment, a simulated ANYmal robot equipped with a Velodyne VLP-16 LiDAR is simulated to navigate within a 3-axis LiDAR degenerate environment as shown in Fig. 9. The robot traverses around a rectangular pillar on a planar surface, challenged with degeneracy in translation on the ground plane and rotation around the ground plane normal. During the experiment, the body velocities of the robot reach 0.85 m s^{-1} and 40° s^{-1} , adding an additional layer of difficulty.

To leverage the simulated environment and analyze the robustness of each algorithm, a uniformly distributed noise is added to the otherwise perfect external pose prior, which is used as an initial guess for the point cloud registration process in all methods. The noise is sampled from normal distributions $\mathcal{N}(\mu_t, \sigma_t^2)$ and $\mathcal{N}(\mu_r, \sigma_r^2)$ where the distribution variables are $\mu_t = 0 \text{ cm}$, $\sigma_t = 0.05 \text{ m}$, $\mu_r = 0 \text{ rad}$, and $\sigma_r = 0.01 \text{ rad}$. The mapping results for each method and the ground truth point cloud of the pillar are shown in Fig. 9. To highlight the degradation of the mapping quality due to LiDAR degeneracy, the ground plane is cut from the results of each method. As seen in Fig. 9-A, the sparsity of the LiDAR scan and the size of the environment create another layer of complexity since only a limited number of LiDAR scanlines fall onto the pillar. Uniquely, in this experiment, the robot starts in a spot where the LiDAR is already degenerate; hence, degenerate scan-to-submap registration can be thought of as degenerate scan-to-scan registration at the beginning of the sequence. This further challenges the optimization process as the sensitivity to outliers increases since the number of correspondences is limited at this stage.

The mapping results shown in Fig. 9 indicate a good performance of Eq. Con. and Prior Only methods. As expected, a misaligned map was generated by the baseline methods

TABLE 3. APE and RPE (per 1-m distance traveled) metrics are provided for the dynamic ANYmal simulation experiment where the best is in bold and the second best is underlined.

	APE		RPE	
	Translation $\mu(\sigma)[m]$	Rotation $\mu(\sigma)[\text{deg}]$	Translation $\mu(\sigma)[m]$	Rotation $\mu(\sigma)[\text{deg}]$
Eq. Con.	<u>0.083(0.055)</u>	0.374(0.206)	<u>0.022(0.018)</u>	0.362(0.241)
InEq. Con.	0.651(0.665)	1.900(1.610)	0.046(0.032)	0.494(0.286)
Zhang <i>et al.</i>	0.658(0.347)	<u>0.521(0.325)</u>	0.041(0.040)	0.428(0.329)
TSVD	2.05(1.23)	1.224(0.642)	0.039(0.035)	0.439(0.289)
P2Plane	2.485(1.299)	8.31(635)	0.155(0.209)	0.940(1.451)
Prior Only	0.018(0.009)	0.583(0.201)	0.017(0.007)	0.77(0.295)
NL-Reg.	1.169(0.918)	3.811(2.502)	0.046(0.055)	0.471(0.886)
NL-Solver	2.832(1.71)	10.43(4.696)	0.133(0.185)	1.140(3.255)
L-Reg.	1.105(1.114)	3.127(2.434)	0.043(0.042)	0.586(0.801)
Cauchy	1.210(0.802)	4.627(4.196)	0.077(0.132)	0.732(1.757)
RMS	2.863(1.423)	9.855(4.01)	0.13(0.197)	1.042(2.347)
Gelfand <i>et al.</i>	4.273(3.24)	14.30(10.24)	0.173(0.222)	1.495(3.60)

P2Plane, NL-Solver, and the Cauchy method. This is due to the severe LiDAR degeneracy introduced by the experiment setup, where the robot starts and stops the simulated experiment in the presence of multidirectional LiDAR degeneracy. In addition, the sampling methods rms method of [14] and Gelfand et al. also fail to register the point clouds correctly. The rms method is described as not resilient against high-rate rotations [14], providing a reason for the observed behavior. Besides, the limited number of features in the environment affects the efficacy of the sampling methods. Moreover, the TSVD, L-Reg., and Ineq. Con. methods generate a similar map, showing a moderate amount of drift in rotation and translation. The drift of L-Reg. and Ineq. Con. can be attained to the usage of parameters tuned for the Ulmberg tunnel experiment. The environment-based tuning might improve the performance of these methods. Performance of TSVD might indicate the importance of mitigating the degeneracy during or after the optimization as this is the only active degeneracy mitigation method acting before the optimization. Finally, the NL-Reg. method performs worse than the other active degeneracy mitigation methods but better than the passive counterparts; similar reasoning with parameter selection might explain the performance.

The quantitative comparison of methods is presented in Table 3 as relative and absolute error metrics [107] and calculated using the EVO evaluation package.⁶ The metrics show a comparable performance between Eq. Con., Prior Only and Zhang et al. with a lead from Eq. Con. method. Since the Prior Only method utilizes the noisy pose prior in the presence of degeneracy, the observed errors originate from the numerical errors of registration and mapping, and the added motion prior noise. Interestingly, none of the tunable methods allowing for constraint relaxation, Ineq. Con., L-Reg., and NL-Reg. perform comparably to the best-performing method.

⁶<https://github.com/MichaelGrupp/evo>

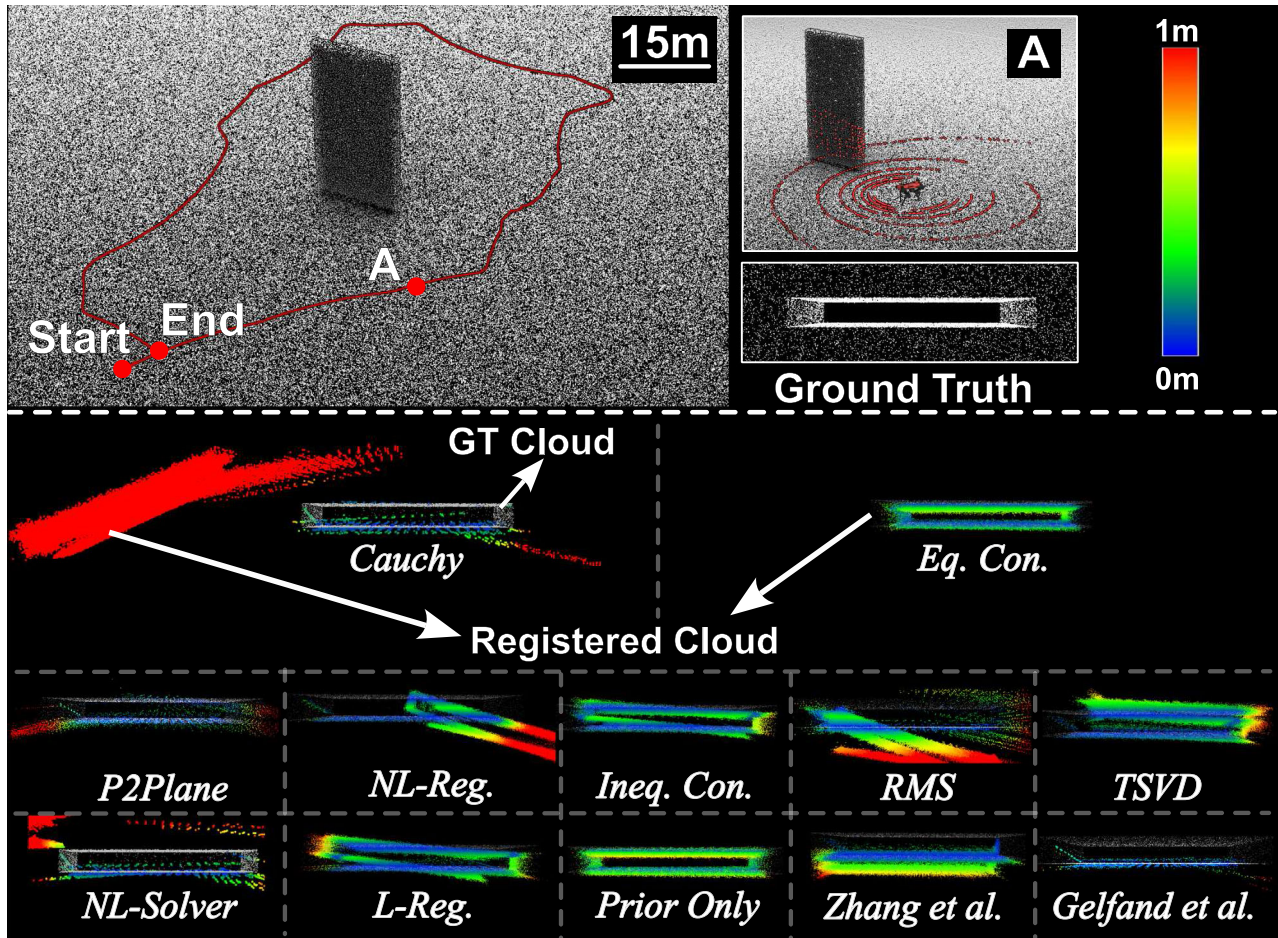


FIGURE 9. Mapping results of the dynamic ANYmal simulation experiment with *Open3d SLAM* in the loop are shown. Top: The robot trajectory is overlaid with the simulated environment ground truth map. Bottom: The registered maps (with the ground removed) for each method are shown next to an error color bar. Points in the maps are colored according to the point-to-point distances to the ground truth map of the environment. Results of *Cauchy* and *Eq. Con.* methods are shown from a far-away perspective to illustrate the scale of the registration error. For visual clarity, the remaining methods are cropped around the pillar.

Intuitively, constraint relaxation indicates reliance more on the point cloud registration rather than the external pose prior, and since the experiment consists of multiple snippets of severe degeneracy, relying on the registration does not result in better performance. Instead, methods that do not allow for constraint relaxation perform better. On the other hand, passive degeneracy mitigation methods and baselines show LiDAR slip and incorrectly registered maps.

B. ANYmal FOREST EXPERIMENT: OPEN FIELD DEGENERACY

During this experiment, the ANYmal robot, shown in Fig. 5(a), starts next to a forest and navigates to an open field, does rapid rotations, and returns to the same passage it previously observed after walking for 107 m. In the open field, shown in Fig. 5(b), the optimization is expected to have 3-axes degeneracy. That is, XY translations along the ground plane and rotation around the normal of this plane. However, as the robot is close to a forest region, the tree canopy provides unstructured yet sufficient geometric

information for the optimization to converge. As a result, LiDAR degeneracy only occurs when the LiDAR loses line of sight to the forest canopy during the experiment. In this experiment, the ANYmal's kinematic leg odometry estimator (TSIF) [108] propagates the previous scan-to-submap registered pose along the degenerate direction. The ground truth trajectory is acquired through Global Navigation Satellite Systems (GNSS) positioning and used to analyze the global consistency of the estimated robot translation, as the GNSS measurements only provide position information. Moreover, the ground truth point cloud of the region of interest is collected by the hand-held Leica BLK2GO [109] scanner. Interestingly, even this commercial scanner failed to register in the degenerate open field; hence, only the points around the revisited area are kept for error metrics computation.

The mapping results of this experiment and the ground truth map of the revisited are provided in Fig. 1. Qualitatively, the methods *NL-Reg.*, *Ineq. Con.*, *Eq. Con.*, *TSVD*, and *NL-Solver* produced a good map, inferred by inspecting the

TABLE 4. Error metrics and ICP-loop computational time for the ANYmal forest experiment, best in bold and the second best is underlined. ATE, Absolute translation error; RTE, relative translation error.

	RTE $\mu(\sigma)[m]$	ATE $\mu(\sigma)[m]$	Computational Cost $\mu(\sigma)[ms]$
Eq. Con.	0.049(0.048)	0.490(0.403)	9.67(3.71)
Ineq. Con.	0.045(0.031)	0.405(0.302)	11.92(4.29)
Zhang <i>et al.</i>	0.083(0.094)	0.372(0.224)	10.45(4.76)
TSVD	0.068(0.066)	0.542(0.423)	10.43(4.83)
P2plane	0.212(0.281)	2.62(1.99)	11.45(6.53)
Prior Only	<u>0.044</u> (0.045)	0.662(0.436)	9.02(5.39)
NL-Reg	0.039 (<u>0.037</u>)	<u>0.364</u> (0.290)	59.86(30.20)
NL-Solver	0.048(0.047)	0.342 (<u>0.263</u>)	45.32(19.93)
L-Reg	0.062(0.086)	0.843(0.517)	11.91(4.93)
Cauchy	0.051(0.086)	0.882(0.738)	<u>8.82</u> (3.98)
RMS	0.362(0.415)	3.70(3.48)	8.28 (<u>4.22</u>)
Gelfand <i>et al.</i>	0.258(0.269)	2.61(2.44)	15.20(6.42)

man-made bench present in the environment. All these methods, except NL-Solver, perform active degeneracy mitigation and act on degeneracy before (TSVD) or during the optimization (NL-Reg., Ineq. Con., and Eq. Con.). NL-Solver performs well without awareness of degeneracy, which supports the notion that the degeneracy in this experiment is mild and can be recovered from. Moreover, the sampling methods rms and Gelfand et al. did not register the bench correctly. The main reason is the presence of large amounts of leaves and vegetation, which hinder the correct calculations of surface normals due to their unstructured nature. It is important to note that the methods NL-Reg. and Ineq. Con. have been fine-tuned for the Ulmberg bicycle tunnel experiment, yet have also performed adequately in this experiment. Out of the methods that do not require parameter tuning, Eq. Con. and TSVD perform better, possibly thanks to the degeneracy mitigation before or during the optimization. As the P2plane and Cauchy methods are blind to LiDAR degeneracy, the map generated by these methods is broken, as seen by the duplication. The Prior Only method relies on the odometry prior, and since the leg odometry estimator does not perform well on soft terrain, the generated map is also misaligned. Differently, Zhang et al. and L-Reg. methods, despite actively mitigating degeneracy, show LiDAR drift in the generated maps. The good performance of Zhang et al. might be explained by the difference in the way the method addresses degeneracy, since this is the only method to act on optimization degeneracy after optimization. This method performs among the best until the $t = 180$ -s mark, where the payload of the robot blocks the view of the unique features, leading to LiDAR drift. Moreover, the performance of L-Reg. can be explained by a sensitivity to parameter tuning, as the performance of this method relies on the value of the degeneracy regularization parameter. RTE per 2-m traversed distance and absolute translation error (ATE)

metrics are provided along with the ICP computational cost in Table 4. Quantitatively, the best method in terms of RTE and ATE mean error is NL-Reg. followed by Ineq. Con. in RTE and Zhang et al. in ATE. It should be noted that the good performance of Zhang et al. is shown in statistical metrics, but the misaligned map shown in Fig. 1 indicates a suboptimal estimation in rotation as the provided metrics only measure the translation errors. To ensure an in-depth understanding of the performance differences of the compared results, the RTE and ATE errors of each method are also provided throughout the robot trajectory as shown in Fig. 10. In both plots, at time $t = 100$ s, multiple methods' performance considerably changes. This point in time corresponds to the 3-axes degeneracy, including the rotation around the ground plane normal. The ATE performance of Cauchy and Zhang et al. is better than that of the other methods until the degeneracy at time $t = 100$ s is reached. This suggests that these methods might be sensitive to the severity of the degeneracy.

Regarding computational cost, all methods, except the best performing NL-Reg. and NL-Solver methods, behave similar to the baseline and Prior Only taking the lead, as this method skips the registration step. However, NL-Reg. and NL-Solver are approximately five times more computationally expensive as the underlying nonlinear optimization takes more time to solve despite the same degeneracy constraints provided as with other methods. Moreover, the fastest method is rms by [14]; this result is expected as rms samples, partially to reduce the redundancy in the point cloud. However, ATE and RTE suggest that this method might not be suitable for unstructured natural environments where rapid rotations are performed. The RTE error plot also suggests unstable behavior without LiDAR degeneracy, which supports the challenges of the environment in this method.

C. ULMBERG BICYCLE TUNNEL EXPERIMENT: ONE DIRECTIONAL DEGENERACY

One of the common degenerate cases for LiDAR-SLAM systems is the one-directional degeneracy, typical in corridor or tunnel-like structures. Ulmberg bicycle tunnel dataset [68], depicted in Fig. 5(d), represents an example of such an environment. The handheld sensor payload used in this experiment, shown in Fig. 5(c), is equipped with an Ouster OS0-128 LiDAR. Importantly, as the handheld payload does not have kinematic measurements that can be used to provide a pose prior, the intensity-based odometry estimates from COIN-LIO [68] are used to propagate the previous scan-to-submap registered pose during optimization degeneracy for all methods. COIN-LIO can successfully localize in such environments since it utilizes the visual cues of the environment in the intensity spectrum and does not rely on the geometric features of the degraded environment.

1) PARAMETER TUNING

The methods Ineq. Con., L-Reg., and NL-Reg. are sensitive to parameter selection. To address this problem, the performance of these methods is analyzed using the data

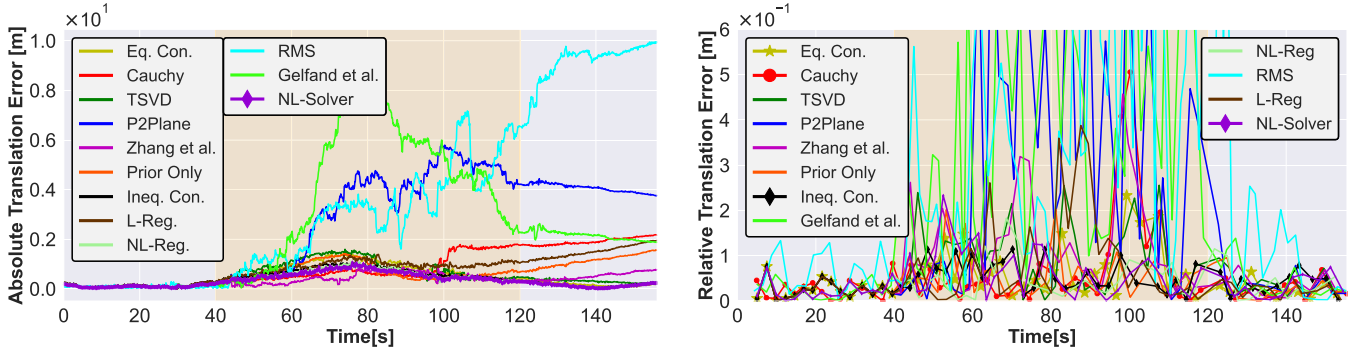


FIGURE 10. ATE (left) and RTE (right) over time of each method for the duration of the ANYmal forest experiment. The approximate LiDAR degenerate region is highlighted in orange. For relative translation error (RTE), the per 1-m traveled distance error increases on average in the degenerate region.

from this experiment. Fig. 11 shows the effect of the upper and lower bound selection for the Ineq. Con. method on the RTE and ATE metrics. The RTE metric is the measure of local consistency and stability of the method; on the other hand, the ATE metric measures the absolute error and is the measure of consistency over longer horizons. When the bounds are numerically close to 0, the RTE error increases, indicating local errors; however, the decreasing ATE suggests that the constraint’s increased tightness helps prevent large-scale drift. Consequently, the bound parameter is set to $\epsilon = 0.0014$, indicating a lower bound of $-\epsilon$ and an upper bound of ϵ . This value is selected as it falls to the saddle region where the error metrics are, on average, low.

Similarly, the effect of the regularization parameter for the methods L-Reg. and NL-Reg. are provided in Fig. 11, respectively, in relation to the RTE and ATE errors. Interestingly, as the regularization weight increases, both methods reach a minimum error region for the RTE error; however, further increments increase the RTE error. On the other hand, increments in the regularization weight generally decrease the ATE error. Based on this analysis, $\lambda = 440$ for L-Reg. and $\lambda_R = 675$ for NL-Reg. are selected and used for all the experiments in this work.

Intuitively, the impact of the regularization is positively related to the amplitude of the regularization weight. Consequently, as the weight increases, these methods approximate the behavior of hard-constrained methods. Since the strength of regularization-based methods originates from the delicate balance between degeneracy-aware regularization and the point-to-plane cost function, as a consequence, the first regularization parameter that results in low RTE and ATE is selected.

2) GLOBAL POINT CLOUD REGISTRATION

A standalone registration analysis is performed to evaluate the feasibility of global registration methods in real-world LiDAR degenerate scans. The registration setup consists of two point clouds extracted from the experiment data while maintaining approximately 70% overlap between the scans. As shown in the upper row of Fig. 12, the source

point cloud is transformed and provided as input along with the reference cloud to the global registration methods, Quatro [52] and FGR [51]. Compared to the experiment described in Section V-A2, where only a rotational direction is degenerate, in this experiment, only a translational direction is degenerate. As seen in the bottom row of Fig. 12, both methods were unable to find the true global minima in the absence of distinguishable geometric features. However, both methods successfully retrieved the correct transform for all well-constrained axes, illustrating a correct usage of the method. Furthermore, for this problem setup, the empty spots in the scan act as false global minima, which the methods cannot distinguish from the true global minima, which can be inferred from the roof light structures illustrated in the upper-right part of Fig. 12. This degenerate scan-to-scan point cloud registration experiment furthermore strengthens the notion that, in the absence of observable inliers, global point cloud registration methods also suffer from the problem of LiDAR degeneracy and might not bring additional benefits compared to other approaches.

3) SLAM IN THE LOOP REGISTRATION

For each method, the point cloud map of the twice-observed (upon return) stairs region is provided in Fig. 13. Moreover, the ground truth map of the nondegenerate section, collected again with the vision-aided Leica BLK2GO scanner [109], and the real-life picture of the environment are provided in Fig. 13. The mapping results show the effectiveness of active degeneracy mitigation methods, as all methods of this category generate a point cloud map with minimal or no drift.

Importantly, Ineq. Con., L-Reg., and NL-Reg. methods are tuned for this environment; hence, these methods are expected to perform well. The P2Plane and Prior Only methods show map duplication along the direction of the tunnel, and the Cauchy method shows the most drift among all methods. Moreover, both sampling methods, the rms and Gelfand et al., improve their performance compared to the baseline P2Plane; however, they perform worse compared to the nonlinear baseline NL-Solver. The underlying reason is the severity and duration of the degeneracy in this experiment

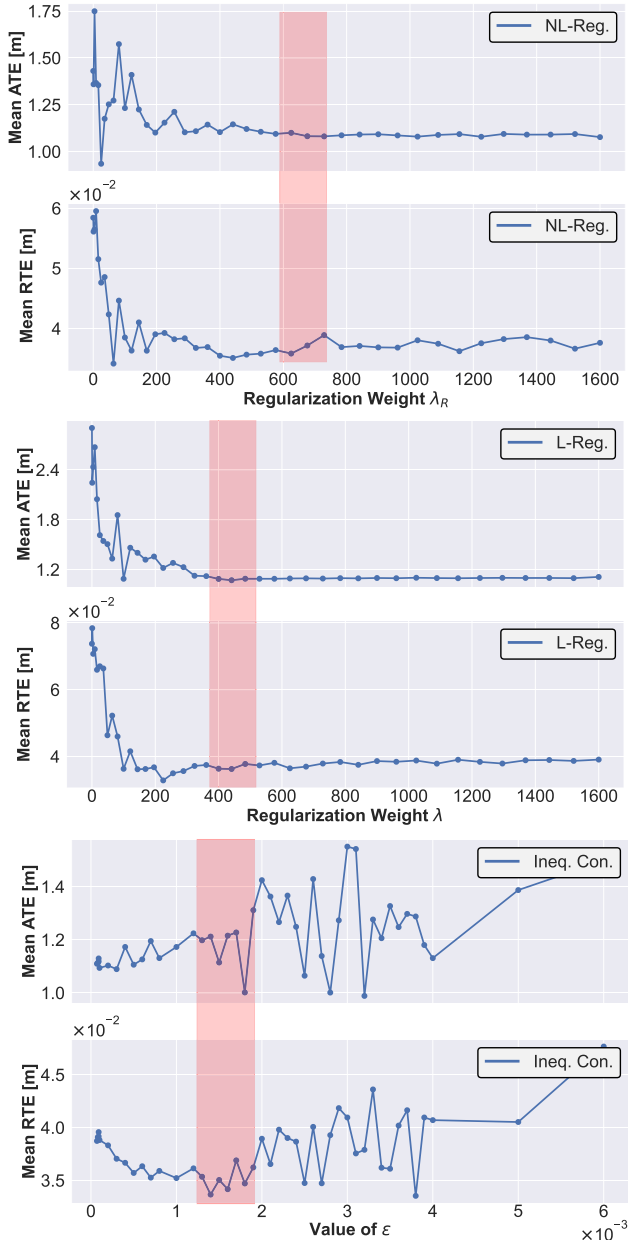


FIGURE 11. Effect of parameter tuning is illustrated for NL-Reg. (top), L-Reg. (middle), and Ineq. Con. (bottom) methods for the Ulmberg bicycle tunnel experiment. Selected regularization weight and ϵ values with low errors are highlighted in red for each method.

which, limits the performance of sampling-based methods. The performance difference between the two baseline methods might suggest the effectiveness of the nonlinear solver in mitigating the adverse effect of an ill-conditioned Hessian. The change in RTE per 1-m traversed distance and ATE are shown in Fig. 14. As shown in Fig. 14, the active degeneracy mitigation methods have a lower error variation compared to the baseline P2Plane and Cauchy methods. In comparison, the Prior Only method has a bad estimate peak around $t = 180$ s, which is sufficient to generate a misaligned map. Interestingly, a comparison of Ineq. Con.

TABLE 5. Error metrics and ICP registration computational time for the Ulmberg bicycle tunnel experiment are shown (best in bold), and the second best is underlined.

	RTE $\mu(\sigma)[m]$	ATE $\mu(\sigma)[m]$	Computational Cost $\mu(\sigma)[ms]$
Eq. Con.	0.039(0.020)	1.10(1.04)	18.95(3.18)
Ineq. Con.	0.033 (0.026)	1.21(0.97)	19.13(2.98)
Zhang <i>et al.</i>	0.038(<u>0.021</u>)	1.125(<u>0.983</u>)	18.90(3.03)
TSVD	0.039(0.021)	1.11(1.01)	18.95(3.00)
P2plane	0.073(0.082)	2.90(1.78)	16.96(3.50)
Prior Only	0.056(0.107)	1.54(0.947)	10.24 (9.28)
NL-Reg	<u>0.035</u> (0.022)	1.097 (1.031)	80.14(29.04)
NL-Solver	0.058(0.063)	1.43(0.846)	64.70(17.05)
L-Reg	0.036(0.024)	<u>1.099</u> (1.030)	22.17(4.18)
Cauchy	0.050(0.046)	1.76(1.090)	18.33(3.52)
RMS	0.09(0.1)	1.62(0.922)	<u>10.942</u> (1.06)
Gelfand <i>et al.</i>	0.073(0.075)	1.97(1.28)	24.321(4.54)

and NL-Reg. reveals that despite both methods performing well, the error variation in Ineq. Con. is higher, resulting in a duplication of the map at the end of the trajectory. This result might indicate that stability and robustness are preferable to better local accuracy for large-scale mapping.

On the other hand, as seen in Fig. 14, all methods accumulate errors over the trajectory of the platform. The source of this error is the aleatoric uncertainty of the data, and the epistemic uncertainty of *Open3D-SLAM* [10] as the map curves throughout the trajectory due to the high incident angle of the observed points.

It is interesting to observe that between $t = 40$ s and $t = 80$ s, the Cauchy method outperforms all other methods until $t = 90$ s, where it starts accumulating errors due to misregistration of the scans. This indicates that robust norms are powerful tools that might benefit accuracy; however, they still suffer from the absence of inliers. In addition, the sampling methods rms, Gelfand et al. and the nonlinear baseline NL-Solver all outperform the active degeneracy methods until the severe degeneracy between $t = 120$ s and $t = 150$ s takes its effect. The intuition behind this behavior is possibly twofold: 1) active methods depend on motion prior quality and 2) sampling methods actively select scarce unique features.

Furthermore, the mean and standard deviation of these metrics are provided along with the ICP computational cost of the ICP registration in Table 5. In a comparison of all methods, Ineq. Con. methods produce the best mean error in RTE while Eq. Con. has the least variation. Among the not fine-tuned methods, Zhang et al. performs the best in terms of mean RTE error while Eq. Con. method has the least ATE error. Moreover, NL-Reg. performs the best in terms of ATE error, and Prior Only has the least variation. The performance is unsurprising as NL-Reg. is fine-tuned to this environment. However, the computational cost of this method makes it less

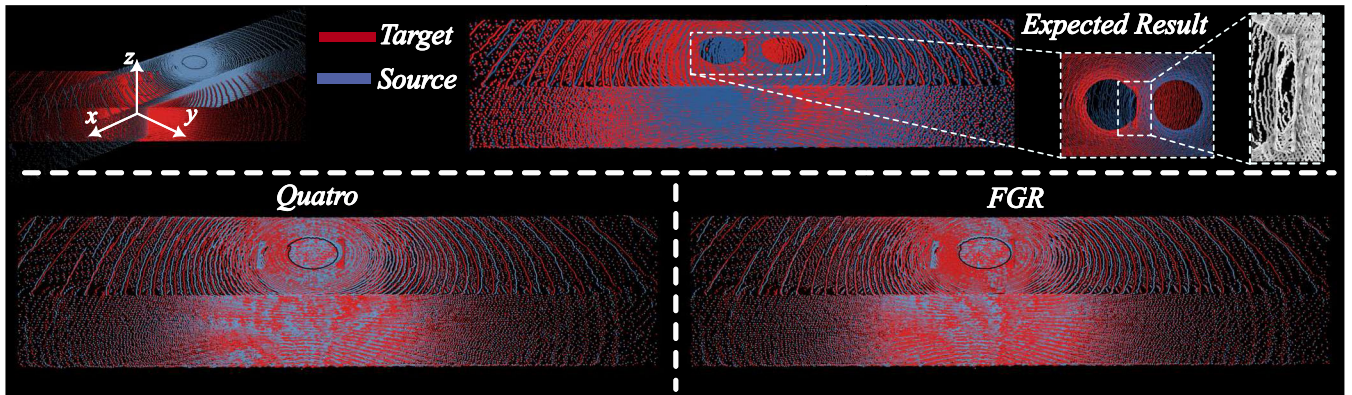


FIGURE 12. Top-left: The point cloud registration setup is provided with the perturbed reference cloud. Top-right: The expected registration result is shown alongside the highlighted light housing of the tunnel; for better visualization of the edges, the points are in shaded gray. Bottom: The registration results from the Quatro [52], [59] and FGR [51] methods are shown. The perturbation is $t = [-1, -1, -0.5]$ m for translation and $r = [0^\circ, 0^\circ, -30.00^\circ]$ for rotation.

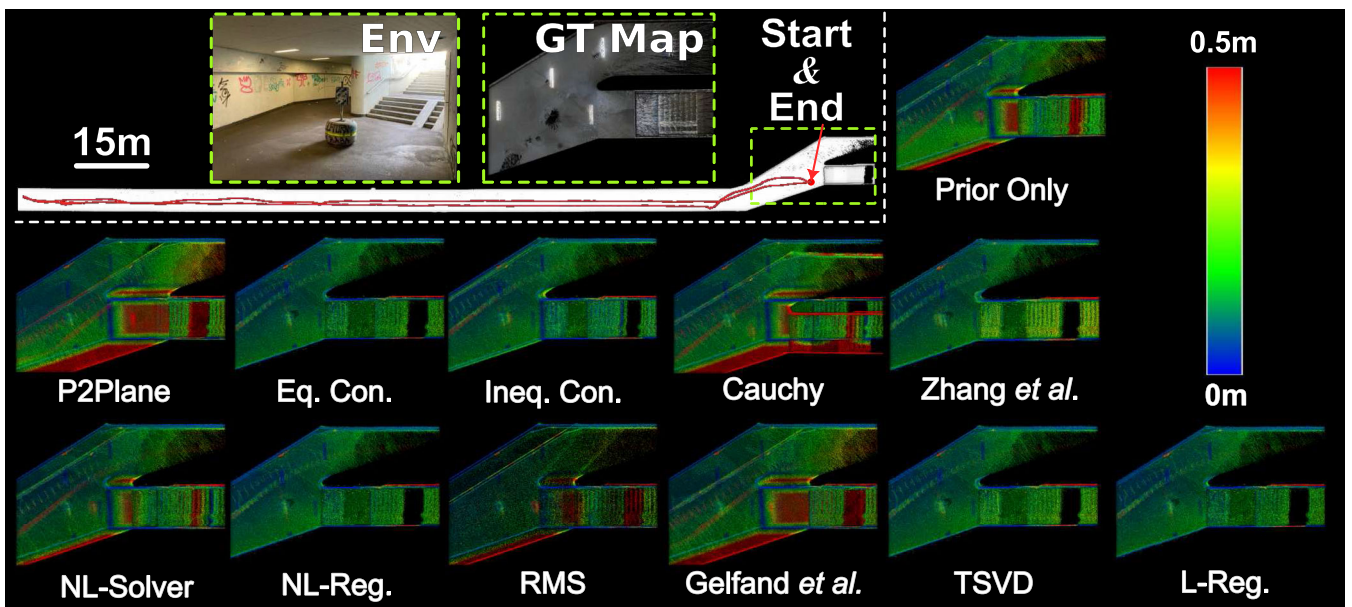


FIGURE 13. Mapping results from the Ulmberg bicycle tunnel experiment are shown. Top-left: The ground truth map of the tunnel and the real-world image of the region of interest are shown next to the point cloud map of the Eq. Con. method. Rest: The registered maps for each method are shown next to an error color bar. Points in the maps are colored according to the point-to-point distances to the ground truth map of the environment.

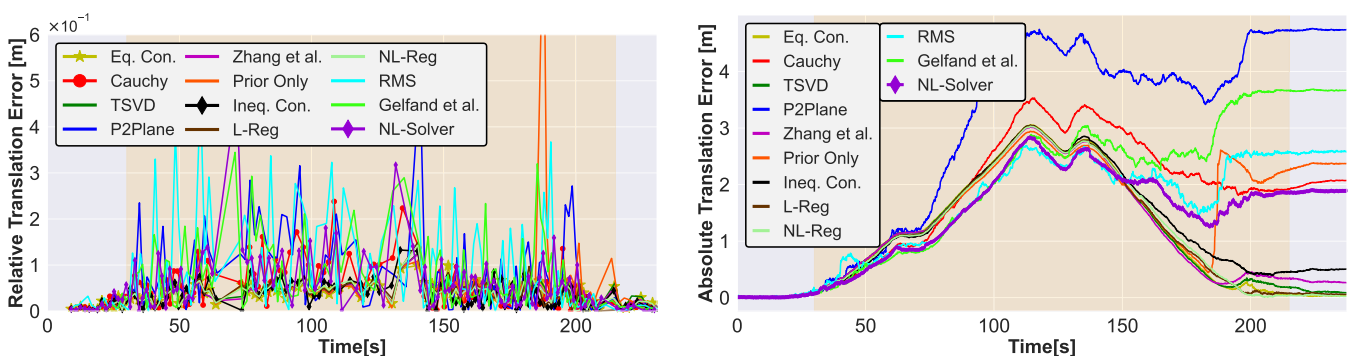


FIGURE 14. RTE (left) and ATE (right) for each method is provided over the duration of the experiment. The LiDAR degenerate region is highlighted in orange. For RTE, the active degeneracy mitigation methods consistently prevent higher errors. For ATE, the common increase in error at the middle of the trajectory is mainly due to the bending of the map originating from the errors in surface normal extraction.

applicable to compute-limited platforms. In addition, the rms sampling method results in the most stable computational cost variation, thanks to its sampling strategy to reduce redundancy in the point cloud.

D. HEAP EXCAVATOR EXPERIMENT: SHORT BURST DEGENERACY

In this experiment, a construction site in Oberglatt, Switzerland, is traversed by a total of 170 m by the HEAP [105] autonomous excavator. As previously shown in [87], the environment contains a GNSS-denied narrow passage between two buildings. HEAP and the construction environment are shown in Fig. 5. The purpose of this experiment is to analyze the performance of the methods against short and severe LiDAR degeneracy while providing a good registration initial guess through COIN-LIO [68]. As shown in Fig. 5(f), only the between-building section of the environment is LiDAR degenerate. However, the global consistency of the LiDAR map is still compromised if the degeneracy is not mitigated. The mapping result of this experiment is shown in Fig. 15. The results indicate that all active degeneracy prevention methods perform comparably, given a good initial guess, regardless of the severity of the degeneracy. The accuracy of the registration initial guess can be inferred from the performance of the Prior Only method. Since the map from the Prior Only method is consistent with the real-world geometry, the odometry from COIN-LIO is inferred to be good. On the other hand, the P2Plane, Cauchy, and NL-Solver methods show LiDAR slip alongside the degenerate direction as these methods do not actively use the degeneracy information in the registration process. Finally, rms and Gelfand et al. sampling methods perform marginally better than the rest of the methods, which are good up to the accuracy of the motion prior. This improvement is due to the lines and corners present in the environment, which the sampling methods can actively identify and utilize.

VI. PRACTICAL MATTERS AND LESSONS LEARNED

In this section, the authors provide insights and lessons learned on robust point cloud registration with a focus on field deployments in LiDAR degenerate environments.

A. BEST PRACTICES FOR REGISTRATION IN FIELD ROBOTICS

Often, field robotics implies a functional set of modules interacting with each other. These modules should be customized with respect to each other for the best performance, such as communication delays, sensor calibrations, and computation. From these modules, the availability of an external odometry module (e.g., visual-inertial odometry [68], kinematic-inertial odometry [108], and radar odometry [110]) is crucial, as the authors found that reliable external odometry prior is crucial to ensure the robustness of the robot pose estimate through a subsequent degenerate point cloud registration process. This has been illustrated in the experiments presented in Section V-D, as all degeneracy-aware methods performed

equally given a good motion prior. Multimodal tightly coupled degeneracy-aware solutions could mitigate this need. However, this is a topic for future research. Moreover, hardware and, subsequently, the deployed sensor setup play a crucial role. For example, a limited field of view of a LiDAR can create LiDAR degeneracy in one environment, while a different sensor placement successfully localizes in the same environment. This phenomenon is realized in Section V-B as the VLP-16 LiDAR of ANYmal was partially blocked. Similarly, the HEAP excavator experiment described in Section V-D showed that a large field of view LiDAR, such as the Ouster OS0-128, is sufficient to create a degeneracy-aware solution to traverse in a self-similar corridor between two buildings. This is inferred through the performance of the sampling methods Gelfand et al. and rms.

B. IMPORTANCE OF THE SOFTWARE SETUP

Different than the robot hardware configuration, the SLAM backend and degeneracy-aware registration framework are similarly important. Degeneracy mitigation alone is ineffective unless coupled with the correct degeneracy detection, as this facilitates degeneracy-aware point cloud registration. The detection method should be robust and generalizable to changes in environments. Furthermore, excellent and reliable spatial feature extraction (see surface normals) is necessary for correct degeneracy detection. Although this work does not focus on selecting the LiDAR degeneracy detection method, it is vital to mitigating its adverse effects. Solutions such as robust norms, statistical filters, and semantic filters (e.g., rms and Gelfand et al.) can help mitigate noise in feature extraction. This is observed in Section V-C and shown in Fig. 14 where Cauchy, rms, and Gelfand et al. performed better than other methods until errors generated from LiDAR degeneracy accumulated past a certain range.

Besides the data and detection quality, the mapping backend plays a critical role in robust scan-to-submap point cloud registration, as the submap acts as the reference point cloud and the system's understanding of the world. As discussed in Section V-C and Fig. 14, all active degeneracy prevention methods drift on the global scale, as common epistemic errors originating from the mapping back-end affect all the methods. Finally, the computational complexity of the system must be real-time-capable, ideally matching the rate of the LiDAR sensor with minimal latency, to ensure usability for downstream tasks such as locomotion and navigation. Throughout Section V, the computational cost of NL-Reg. is highlighted as, while accurate, this method would have to skip measurements in computationally limited systems.

C. HOW TO MITIGATE LiDAR DEGENERACY

Experiments in various real-world scenarios showed that active degeneracy mitigation methods perform more accurately and reliably than passive degeneracy methods in the presence of LiDAR degeneracy. Among the active degeneracy mitigation methods, Eq. Con., TSVD, and NL-Reg. demonstrate robust performance in all experiments, as shown

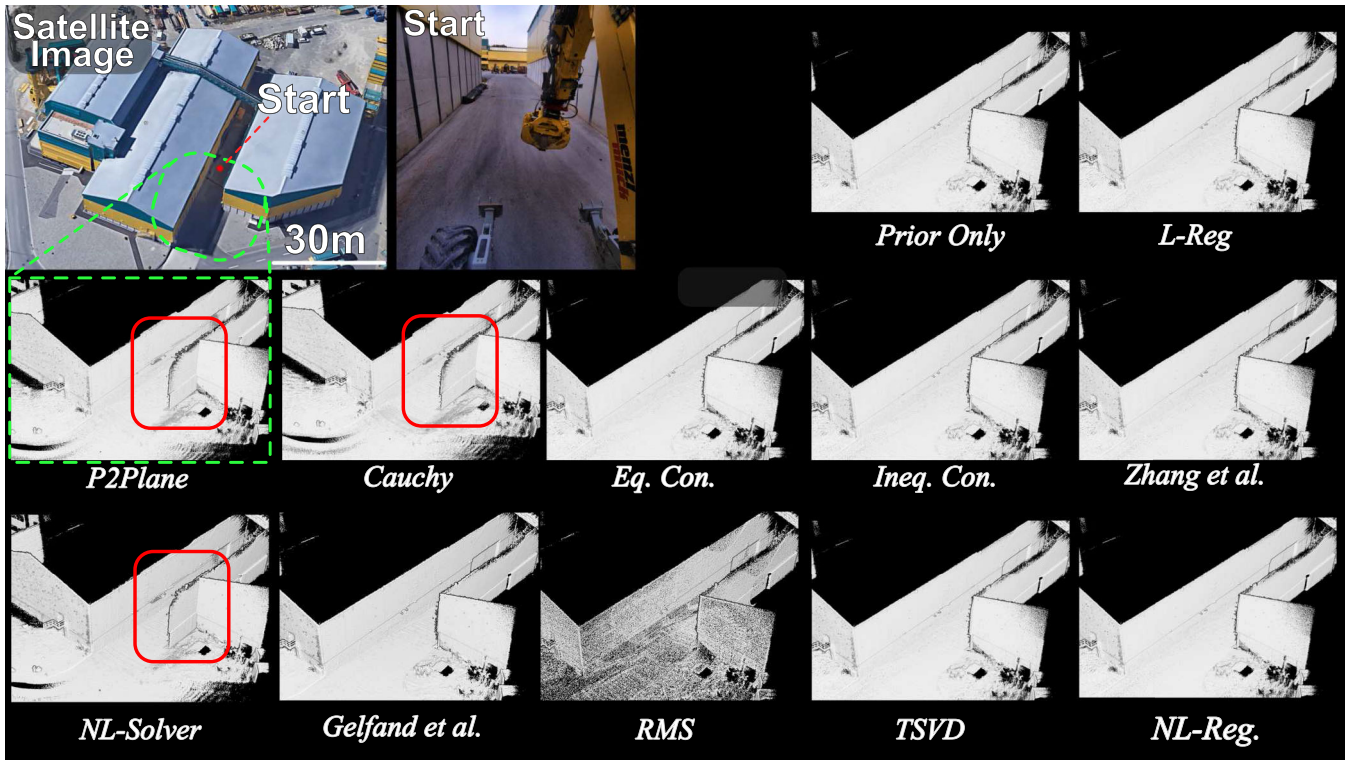


FIGURE 15. Mapping results of the excavator narrow-passage experiment are shown. Top-left: The experiment site’s satellite imagery is shown alongside HEAP’s start location. Rest: The maps from each method are shown. The adverse effect of the LiDAR degeneracy is outlined in red for the P2Plane, NL-Solver, and Cauchy methods.

in Table 6, and show good mapping performance with minimal final ATE error as shown in the results in Section V. As a common factor among these methods, the constraints or the regularization terms are added *before* or *during* the optimization but not after. Furthermore, the heuristic parameter dependent L-Reg. and Ineq. Con. methods are found to be more sensitive to parameter tuning than NL-Reg. method. For example, the best parameters identified for the Ulmberg tunnel experiment do not necessarily perform similarly in other environments, as shown in Table 6. It is crucial to note that the conclusions drawn for these tunable methods are subject to the empirical parameter selection, as described in Section V-C1. An automatic and optimal selection of these parameters is subject to future research. Between different constraint types, it is hard to designate one method as better than the other. Eq. Con. method is more consistent in all environments while Ineq. Con. can be more accurate, albeit with lower estimation consistency. The TSVD method provides a good alternative solution due to the ease of implementation, minimal computation overhead, and overall consistent performance across all environments with minimal error, as seen in Table 6.

Importantly, the overall performance of the Prior Only method indicates that when the motion prior is good (ANYmal simulation experiment) or degeneracy is short in duration (HEAP excavator experiment), skipping the point cloud registration entirely can be a viable solution. Using

TABLE 6. Performance summary of the compared degeneracy mitigation methods. ✓ indicates no visible drift, ● indicates minimal drift, and ✗ indicates a broken point cloud map.

	ANYmal Forest	Ulmberg Tunnel	HEAP Excavator	ANYmal Simulation
<u>Hard-Constrained</u>				
Eq. Con.	●	✓	✓	✓
Zhang <i>et al.</i>	✗	●	✓	●
TSVD	●	●	✓	●
<u>Soft-Constrained</u>				
Ineq. Con.	✓	●	✓	●
NL-Reg	✓	✓	✓	✗
L-Reg	✗	✓	✓	●
<u>Unconstrained</u>				
P2plane	✗	✗	✗	✗
NL-Solver	✓	✗	✗	✗
Cauchy	✗	✗	✗	✗
Prior Only	✗	✗	✓	✓
RMS	✗	✗	✓	✗
Gelfand <i>et al.</i>	✗	✗	✓	✗

Prior Only can also avoid additional computations while also removing the adverse effects of LiDAR degeneracy from the final solution. However, it should be considered that skipping the entirety of the registration would also mean not doing

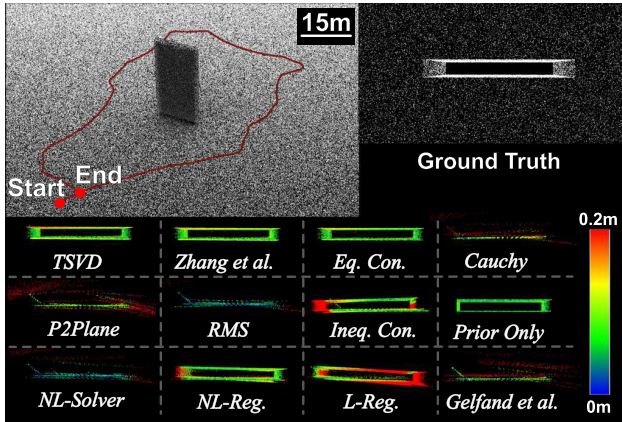


FIGURE 16. Top: The ground truth point cloud and trajectory of the ANYmal simulation experiment. Rest: The mapping results of the experiment (see Section V-A3) with ground truth poses as motion prior. The registered maps (with the ground removed) for each method are shown next to an error color bar. Points in the maps are colored according to the point-to-point distances to the ground truth map of the environment. The scale of the color bar is adjusted to make errors visually distinguishable.

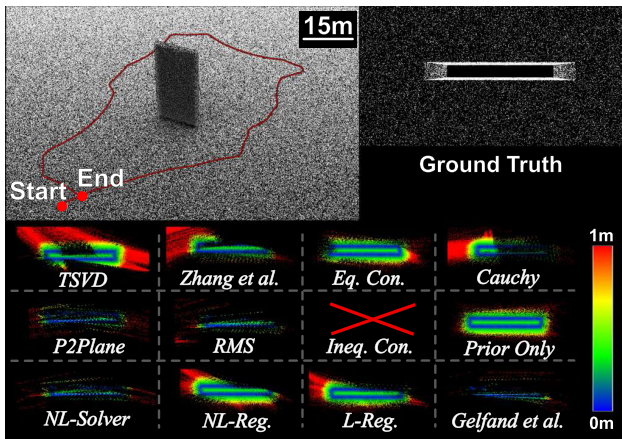


FIGURE 17. Top: The ground truth point cloud and trajectory of the ANYmal simulation experiment, repeated for clarity. Rest: The mapping results of the repetition of the dynamic ANYmal simulation experiment with heavily noised motion prior. The point cloud maps for each method are shown next to an error color bar. Points in the maps are colored according to the point-to-point distances to the ground truth map of the environment.

registration in the well-constrained directions, hence losing the LiDAR accuracy. Moreover, the sensor-fusion methods described in Section II can be used if the robotic platform has additional sensing modalities such as a camera and a radar. This effectively makes the Prior Only much more effective, yet this solution would only offload the problem to other modalities instead of addressing the core problem of optimization instability.

D. IMPORTANCE OF MOTION PRIOR QUALITY

Accuracy of the motion prior is an important element for the successful mitigation of LiDAR degeneracy. To highlight its importance, an additional experiment is performed as

a variation of the ANYmal simulation experiment (see Section V-A3). First, instead of adding noise to the simulated robot pose, this perfect pose is used as the motion prior. The results in Fig. 16 show that all active methods perform similarly, with soft-constrained methods (Ineq. Con., NL-Reg., and L-Reg.) being marginally worse. This observation implies that in this severely degenerate scene, degenerate point cloud registration can degrade an otherwise perfect pose prior estimate. It should be noted that here, the point clouds are voxelized at 0.1 m, which implies that due to the effect of voxelization, the point-to-point error is not expected to be exactly zero.

Second, to highlight the adverse effect of noisy motion prior, increased noise is added to the ground truth robot poses compared to before Section V-A3. This noise is sampled from the normal distributions $\mathcal{N}(\mu_t, \sigma_t^2)$ and $\mathcal{N}(\mu_r, \sigma_r^2)$ where the distribution variables are $\mu_t = 0.05$ m, $\sigma_t = 0.05$ m, $\mu_r = 0$ rad, and $\sigma_r = 0.02$ rad. The results of this variation are shown in Fig. 17. It can be seen through the performance of the Prior Only method that the effect of noise on the quality of the registration is detrimental. Almost all the methods except Eq. Con. diverged, which can be seen through the misaligned point cloud sections of the wall in red. Moreover, Ineq. Con. method also diverged before registering sufficient points on the wall of the environment and, as a result, not visualized. As motion prior is a key element for all the downstream tasks, such as feature extraction, correspondence search, and degeneracy detection, a thorough analysis is required to draw any conclusions about which method might be more robust to variations in the motion prior.

E. CONSIDERATIONS AND RECOMMENDATIONS

Regardless of the performance of LiDAR degeneracy mitigation methods, some additional aspects, such as computational complexity and ease of use, are equally important to consider. In terms of computation, all methods except NL-Reg. are close to each other and capable of performing the optimization at the LiDAR sensor rate. As the NL-Reg. method uses nonlinear optimization techniques; it takes more computing power, it can fall behind the LiDAR rate, and consequently drop measurements in exchange for increased accuracy and robustness. Another consideration to take into account is the need for parameter tuning. The methods NL-Reg., L-Reg., and Ineq. Con. require parameter tuning. As discussed in Section V-C1, the performance of these methods changes according to the set parameters. There are applications, such as continuous inspection and surveillance, in which a robot would need to operate in the same environment multiple times. For such applications, the ability to fine-tune and overfit to an environment can be a benefit instead of a drawback. Similarly, in some applications, such as the generation of metric-semantic digital twins, the accuracy and robustness of the registration can have priority over ease of use, parameter tuning, or computational requirements.

The summary provided in Table 6 shows that Eq. Con., TSVD, and Ineq. Con. did not fail in any of the experi-

ments. Meanwhile, the P2Plane and Cauchy methods always generated an incorrect map. Despite failing in the ANYmal dynamic simulation experiment, NL-Reg. method generated accurate results. The method of Zhang et al. performed comparably with misregistration in the ANYmal simulation and the ANYmal forest experiments.

As a recommendation, if more computational resources and the ability to fine-tune are available, NL-Reg. method is a good option. If deployed on a computationally limited system, with fine-tuning availability Ineq. Con. method has great potential since this method (thanks to the formulation of the QP problem) can also accept Eq. Con. On the other hand, if fine-tuning is not an option, Eq. Con. method performed reliably throughout the experiments with good accuracy; however, the performance of this method is tied to the accuracy of the motion prior. If the freedom to choose an optimization is desired, Zhang et al. and the TSVD method showed the same reliability in exchange for accuracy but without the addition of hard-constraints to the optimization problem (e.g., no need for QP formulation, Lagrangian, Ceres, and sampling). As a sampling approach, rms [14] provided a speed-up to the framework throughout all experiments and performed the best in the urban setting. Although this method did not perform well in other experiments, if the environment is geometrically feature-rich, it could be a strong candidate for many frameworks to mitigate the effect of mild degeneracy while providing a substantial speed-up.

The results and the recommendations done in this work are to provide an objective view of the problem of degenerate point cloud registration. However, the reader should beware of the merits of the system used in this work (*Open3D-SLAM* [10]), which uses a voxel-based mapping backend, external pose prior, and scan-to-submap registration as described in Section IV-C.

VII. CONCLUSION AND FUTURE WORK

This work analyzed the efficacy of degeneracy mitigation methods in various real-world scenarios and simulated setups. New methods such as Ineq. Con., TSVD, and L-Reg. are proposed and introduced to the field of degeneracy-aware point cloud registration. In addition, previously introduced methods such as Eq. Con. [13], Solution Remapping [19], Cauchy robust norm [61], rms [14], and Gelfand et al. [15] are compared.

As discussed in Section V and summarized in Table 6, active degenerate mitigation is crucial and beneficial for severe LiDAR degeneracy, in mild cases of LiDAR degeneracy, passive methods, such as sampling methods, can provide the necessary robustness. As a result, selecting the LiDAR degeneracy mitigation method is a problem that depends, at least, on the severity of the LiDAR degeneracy and the availability of a good motion prior.

Among the active degeneracy mitigation methods, solutions such as TSVD, Eq. Con., and Ineq. Con. perform more consistently. Compared to methods such as Zhang et al., L-Reg., and NL-Reg. methods. On the other hand, among the

passive degeneracy methods rms [14] performed the best in the mild degeneracy case while providing a speed-up. The robust norm method, Cauchy, improved the performance of the SLAM system in the absence of LiDAR degeneracy as previously highlighted [61]; however, it consistently failed among all datasets when LiDAR degeneracy was present. The positive effect of nonlinear optimization, in exchange for the additional computational cost, is shown through the comparison between the P2Plane and NL-Solver baseline methods.

As the focus of this work is on robust point cloud registration, multimodal sensor-fusion solutions are not covered in this work; however, in practice, sensor-fusion is often employed in robotic deployments complimentary modalities can resolve the necessity of the LiDAR degeneracy mitigation. The fusion of ill-conditioned poses in sensor-fusion frameworks is still an open-resource question. Finally, globally optimal point cloud registration methods are studied for the degenerate point cloud registration problem. As shown in Fig. 12 and Section V-A2, global registration methods are not compatible drop-in solutions for LiDAR degeneracy, as the assumption of the existence of informative inliers is often violated. These results supported the claim that LiDAR degeneracy is not an outlier removal problem (e.g., the Cauchy method did not perform well) but rather an absence of an informative inliers problem.

An interesting future direction is to utilize active and passive methods together while acknowledging the strengths and weaknesses of each category. An example would be to combine NL-Reg. and rms methods to compensate for the computational load of NL-Reg. with the speed-up provided by rms.

Similarly, detailed analysis of the effect of fine-tuning methods L-Reg., NL-Reg., and Ineq. Con. and ways to automate the selection of these parameters is still an open research question. These methods have the potential to perform better if the manual environment-specific parameter selection requirement is removed or mitigated.

As an outlook, ICP is still a dominant solution in point-cloud registration, and many researchers focus on addressing the weaknesses that arise from optimization degradation. The problem originates from the lack of sufficient inlier information provided to the least-squares optimization through steps such as surface-normal extraction and K -nearest neighbors. A one-fits-all solution does not seem to be possible without quantifying the errors in these steps. As a result, the future steps in ICP-based robust point-cloud registration in degenerate cases might lie in metric uncertainty estimation for each step or reduction of the importance of these steps with novel research (see new loss formulation, a new optimization technique, and data-driven correspondence search method).

ACKNOWLEDGMENT

This work was done as an outside activity and not in Shehryar Khattak's capacity as an employee of the Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, USA.

REFERENCES

- [1] K. Ebadi et al., "Present and future of SLAM in extreme underground environments," *IEEE Trans. Robot.*, vol. 40, no. 1, pp. 936–959, Jan. 2022.
- [2] I. A. Kazerouni, L. Fitzgerald, G. Dooly, and D. Toal, "A survey of state-of-the-art on visual SLAM," *Expert Syst. Appl.*, vol. 205, Nov. 2022, Art. no. 117734.
- [3] S. Khattak, C. Papachristos, and K. Alexis, "Keyframe-based thermal-inertial odometry," *J. Field Robot.*, vol. 37, no. 4, pp. 552–579, Dec. 2019.
- [4] M. Nissov, S. Khattak, J. A. Edlund, C. Padgett, K. Alexis, and P. Spieler, "ROAMER: Robust offroad autonomy using multimodal state estimation with radar velocity integration," in *Proc. IEEE Aerosp. Conf.*, Mar. 2024, pp. 1–10.
- [5] H. Bavle, J. L. Sanchez-Lopez, C. Cimarelli, A. Tourani, and H. Voos, "From SLAM to situational awareness: Challenges and survey," *Sensors*, vol. 23, no. 10, p. 4849, May 2023.
- [6] C. Cadena et al., "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. Robot.*, vol. 32, no. 6, pp. 1309–1332, Dec. 2016.
- [7] A. Devadas Nair, J. Kindle, P. Levchev, and D. Scaramuzza, "Hilti SLAM challenge 2023: Benchmarking single + multi-session SLAM across sensor constellations in construction," 2024, *arXiv:2404.09765*.
- [8] J. Li, P. Shi, Q. Hu, and Y. Zhang, "QGORE: Quadratic-time guaranteed outlier removal for point cloud registration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 9, pp. 11136–11151, Sep. 2023.
- [9] S. Khattak, H. Nguyen, F. Mascari, T. Dang, and K. Alexis, "Complementary multi-modal sensor fusion for resilient robot pose estimation in subterranean environments," in *Proc. Int. Conf. Unmanned Aircr. Syst. (ICUAS)*, Sep. 2020, pp. 1024–1029.
- [10] E. Jelavic, J. Nubert, and M. Hutter, "Open3D SLAM: Point cloud based mapping and localization for education," in *Proc. Robotic Perception Mapping: Emerg. Techn. ICRA Workshop*, May 2022, p. 24.
- [11] A. Censi, "An accurate closed-form estimate of ICP's covariance," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 3167–3172.
- [12] M. Brossard, S. Bonnabel, and A. Barrau, "A new approach to 3D ICP covariance estimation," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 744–751, Apr. 2020.
- [13] T. Tuna, J. Nubert, Y. Nava, S. Khattak, and M. Hutter, "X-ICP: Localizability-aware LiDAR registration for robust localization in extreme environments," *IEEE Trans. Robot.*, vol. 40, no. 6, pp. 452–471, Nov. 2024.
- [14] P. Petracek, K. Alexis, and M. Saska, "RMS: Redundancy-minimizing point cloud sampling for real-time pose estimation," *IEEE Robot. Autom. Lett.*, vol. 9, no. 6, pp. 5230–5237, Jun. 2024.
- [15] N. Gelfand, L. Ikemoto, S. Rusinkiewicz, and M. Levoy, "Geometrically stable sampling for the ICP algorithm," in *Proc. 4th Int. Conf. 3-D Digit. Imag. Modeling (3DIM)*, Oct. 2003, pp. 260–267.
- [16] J. Hatleskog and K. Alexis, "Probabilistic degeneracy detection for point-to-plane error minimization," 2024, *arXiv:2410.10784*.
- [17] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "KISS-ICP: In defense of point-to-point ICP—Simple, accurate, and robust registration if done the right way," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 1029–1036, Feb. 2023.
- [18] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "FAST-LIO₂: Fast direct LiDAR-inertial odometry," *IEEE Trans. Robot.*, vol. 38, no. 4, pp. 2053–2073, Aug. 2022.
- [19] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Stockholm, Sweden, May 2016, pp. 809–816.
- [20] E. Westman and M. Kaess, "Degeneracy-aware imaging sonar simultaneous localization and mapping," *IEEE J. Ocean. Eng.*, vol. 45, no. 4, pp. 1280–1294, Oct. 2020.
- [21] S. Nobili, G. Tinchev, and M. Fallon, "Predicting alignment risk to prevent localization failure," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 1003–1010.
- [22] M. Ramezani, G. Tinchev, E. Iuganov, and M. Fallon, "Online LiDAR-SLAM for legged robots with robust registration and deep-learned loop closure," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 4158–4164.
- [23] A. Tagliabue et al., "LION: LiDAR-inertial observability-aware navigator for vision-denied environments," in *Proc. Int. Symp. Exp. Robot.*, 2020, pp. 380–390.
- [24] Y. Liu, J. Wang, and Y. Huang, "A localizability estimation method for mobile robots based on 3D point cloud feature," in *Proc. IEEE Int. Conf. Real-time Comput. Robot. (RCAR)*, Jul. 2021, pp. 1035–1041.
- [25] J. Nubert, E. Walther, S. Khattak, and M. Hutter, "Learning-based localizability estimation for robust LiDAR localization," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 17–24.
- [26] W. Zhen, S. Zeng, and S. Soberer, "Robust localization and localizability estimation with a rotating laser scanner," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 6240–6245.
- [27] W. Zhen and S. Scherer, "Estimating the localizability in tunnel-like environments using LiDAR and UWB," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 4903–4908.
- [28] S. B. Nashed, J. Jin Park, R. Webster, and J. W. Durham, "Robust rank deficient SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 6603–6608.
- [29] S. Zhao et al., "SubT-MRS dataset: Pushing SLAM towards all-weather environments," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 22647–22657.
- [30] Z. Chen et al., "Heterogeneous LiDAR dataset for benchmarking robust localization in diverse degenerate scenarios," 2024, *arXiv:2409.04961*.
- [31] M. Nissov, N. Khedekar, and K. Alexis, "Degradation resilient LiDAR-radar-inertial odometry," 2024, *arXiv:2403.05332*.
- [32] M. Tranzatto et al., "CERBERUS in the DARPA subterranean challenge," *Sci. Robot.*, vol. 7, no. 66, p. 9742, May 2022.
- [33] P. Pfreundschuh, H. Oleynikova, C. Cadena, R. Siegwart, and O. Andersson. (Mar. 2024). *Envide Dataset*. [Online]. Available: <https://projects.asl.ethz.ch/datasets/envide>
- [34] S. Ji et al., "A Point-to-distribution degeneracy detection factor for LiDAR SLAM using local geometric models," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2024, pp. 12283–12289.
- [35] Z. Jin and C. Jiang, "Range-aided LiDAR-inertial multi-vehicle localization and mapping in degenerate environments," *IEEE Trans. Intell. Vehicles*, pp. 1–13, 2024, doi: [10.1109/TV.2024.3360286](https://doi.org/10.1109/TV.2024.3360286).
- [36] K. Ebadi, M. Palieri, S. Wood, C. Padgett, and A.-A. Agha-Mohammadi, "DARE-SLAM: Degeneracy-aware and resilient loop closing in perceptually-degraded environments," *J. Intell. Robotic Syst.*, vol. 102, no. 1, pp. 1–25, Apr. 2021.
- [37] Z. Chen et al., "RELEAD: Resilient localization with enhanced LiDAR odometry in adverse environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2024, pp. 3999–4005.
- [38] F. Pomerleau, F. Colas, and R. Siegwart, "A review of point cloud registration algorithms for mobile robotics," *Found. Trends Robot.*, vol. 4, no. 1, pp. 1–104, 2015.
- [39] A. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," in *Robotics: Science and Systems*, vol. 2. Seattle, WA, USA: Robots.ox.ac.uk, 2009, p. 435.
- [40] P. Babin, P. Dandurand, V. Kubelka, P. Giguère, and F. Pomerleau, "Large-scale 3D mapping of subarctic forests," in *Field and Service Robotics*. Cham, Switzerland: Springer, 2021, pp. 261–275.
- [41] A. Censi, "An ICP variant using a point-to-line metric," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2008, pp. 19–25.
- [42] S. Rusinkiewicz, "A symmetric objective function for ICP," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–7, Aug. 2019.
- [43] J. Zhang and S. Singh, "LOAM: LiDAR odometry and mapping in real-time," *Robot., Sci. Syst.*, vol. 2, no. 9, pp. 1–9, 2014.
- [44] J. Behley and C. Stachniss, "Efficient surfel-based SLAM using 3D laser range data in urban environments," in *Proc. Robot. Sci. Syst.*, Jun. 2018, p. 59.
- [45] P. Biber and W. Straßer, "The normal distributions transform: A new approach to laser scan matching," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Jun. 2003, pp. 2743–2748.
- [46] A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 12, pp. 2262–2275, Dec. 2010.
- [47] F. A. Maken, F. Ramos, and L. Ott, "Estimating motion uncertainty with Bayesian ICP," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 8602–8608.
- [48] A. De Maio and S. Lacroix, "Deep Bayesian ICP covariance estimation," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 6519–6525.
- [49] X. Bai et al., "PointDSC: Robust point cloud registration using deep spatial consistency," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 15859–15869.

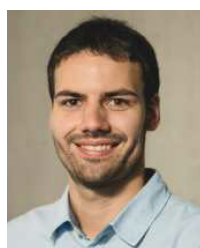
- [50] J. Nubert, S. Khattak, and M. Hutter, "Self-supervised learning of LiDAR odometry for robotic applications," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2021, pp. 9601–9607.
- [51] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2016, pp. 766–782.
- [52] H. Lim et al., "A single correspondence is enough: Robust global registration to avoid degeneracy in urban environments," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 8010–8017.
- [53] P. J. Besl and N. D. McKay, "Method for registration of 3-D shapes," *Proc. SPIE*, vol. 1611, pp. 586–606, Apr. 1992.
- [54] K.-L. Low, "Linear least-squares optimization for point-to-plane ICP surface registration," vol. 4. Chapel Hill, NC, USA: Chapel Hill, Univ. North Carolina, 2004, pp. 1–3.
- [55] F. Pomerleau, "Applied registration for robotics: Methodology and tools for ICP-like algorithms," Ph.D. dissertation, Dept. Mech. Process Eng., Swiss Federal Inst. Technol., ETH Zürich, Zürich, Switzerland, 2013, doi: [10.3929/ethz-a-009932416](https://doi.org/10.3929/ethz-a-009932416).
- [56] H. Yang, J. Shi, and L. Carlone, "TEASER: Fast and certifiable point cloud registration," *IEEE Trans. Robot.*, vol. 37, no. 2, pp. 314–333, Apr. 2020.
- [57] H. Yang and L. Carlone, "A quaternion-based certifiably optimal solution to the wahba problem with outliers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1665–1674.
- [58] L. Carlone, D. M. Rosen, G. Calafiore, J. J. Leonard, and F. Dellaert, "Lagrangian duality in 3D SLAM: Verification techniques and optimal solutions," *IEEE Int. Conf. Intell. Robots Syst. (IROS)*, vol. 2015, no. 1, pp. 1608–1615, Sep. 2015.
- [59] H. Lim, B. Kim, D. Kim, E. Mason Lee, and H. Myung, "Quatro++: Robust global registration exploiting ground segmentation for loop closing in LiDAR SLAM," *Int. J. Robot. Res.*, vol. 43, no. 5, pp. 685–715, Apr. 2024.
- [60] J. Zhang, Y. Yao, and B. Deng, "Fast and robust iterative closest point," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3450–3466, Jul. 2022.
- [61] P. Babin, P. Giguere, and F. Pomerleau, "Analysis of robust functions for registration algorithms," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 1451–1457.
- [62] N. Chebrou, T. Läbe, O. Vysotska, J. Behley, and C. Stachniss, "Adaptive robust kernels for non-linear least squares problems," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 2240–2247, Apr. 2021.
- [63] H. Yang, P. Antonante, V. Tzoumas, and L. Carlone, "Graduated non-convexity for robust spatial perception: From non-minimal solvers to global outlier rejection," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1127–1134, Apr. 2020.
- [64] J. Briales and J. González-Jiménez, "Convex global 3D registration with Lagrangian duality," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, vol. 2017, pp. 4960–4969, Jul. 2017.
- [65] B. Jian and B. C. Vemuri, "Robust point set registration using Gaussian mixture models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1633–1645, Aug. 2011.
- [66] T.-H. Kwok, "DNSS: Dual-Normal-Space sampling for 3-D ICP registration," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 1, pp. 241–252, Jan. 2019.
- [67] T.-H. Kwok and K. Tang, "Improvements to the iterative closest point algorithm for shape registration in manufacturing," *J. Manuf. Sci. Eng.*, vol. 138, no. 1, Jan. 2016, Art. no. 011014.
- [68] P. Pfreundschuh, H. Oleynikova, C. Cadena, R. Siegwart, and O. Andersson, "COIN-LIO: Complementary intensity-augmented LiDAR inertial odometry," 2023, [arXiv:2310.01235](https://arxiv.org/abs/2310.01235).
- [69] S. Bonnabel, M. Barczyk, and F. Goulette, "On the covariance of ICP-based scan-matching techniques," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2016, pp. 5498–5503.
- [70] W. Talbot et al., "Principled ICP covariance modelling in perceptually degraded environments for the EELS mission concept," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2023, pp. 10763–10770.
- [71] A. Censi, "On achievable accuracy for range-finder localization," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 4170–4175.
- [72] K. Huang et al., "LOG-LIO2: A LiDAR-inertial odometry with efficient uncertainty analysis," *IEEE Robot. Autom. Lett.*, vol. 9, no. 10, pp. 8226–8233, Oct. 2024.
- [73] Y. Gao et al., "A prediction method of localizability based on deep learning," *IEEE Access*, vol. 8, pp. 110103–110115, 2020.
- [74] 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, p. 935, Mar. 2018.
- [75] M. L. R. Arévalo, "On the uncertainty in active slam: Representation, propagation and monotonicity," Ph.D. dissertation, Informática e Ingeniería de Sistemas, Universidad de Zaragoza, Zaragoza, Spain, 2018. [Online]. Available: <https://zaguan.unizar.es/record/75422>
- [76] A. Hinduja, B.-J. Ho, and M. Kaess, "Degeneracy-aware factors with applications to underwater SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 1293–1299.
- [77] F. Han, H. Zheng, W. Huang, R. Xiong, Y. Wang, and Y. Jiao, "DAMS-LIO: A degeneration-aware and modular sensor-fusion LiDAR-inertial odometry," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 2745–2751.
- [78] X. Bai, W. Wen, and L.-T. Hsu, "Degeneration-aware outlier mitigation for visual inertial integrated navigation system in urban canyons," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–15, 2021.
- [79] J. Lee et al., "Switch-SLAM: Switching-based LiDAR-inertial-visual SLAM for degenerate environments," *IEEE Robot. Autom. Lett.*, vol. 9, no. 8, pp. 7270–7277, Aug. 2024.
- [80] J. Zou, L. Shao, H. Tang, H. Chen, H. Bao, and X. Pan, "Lmapping: Tightly-coupled LiDAR-inertial odometry and mapping for degraded environments," *Intell. Service Robot.*, vol. 16, no. 5, pp. 583–597, Nov. 2023.
- [81] T. Wen, Y. Fang, B. Lu, X. Zhang, and C. Tang, "LIVER: A tightly coupled LiDAR-Inertial-Visual state estimator with high robustness for underground environments," *IEEE Robot. Autom. Lett.*, vol. 9, no. 3, pp. 2399–2406, Mar. 2024.
- [82] H. Lim, D. Kim, B. Kim, and H. Myung, "AdaLIO: Robust adaptive LiDAR-inertial odometry in degenerate indoor environments," in *Proc. 20th Int. Conf. Ubiquitous Robots (UR)*, Jun. 2023, pp. 48–53.
- [83] S. Ferrari, L. D. Giammarino, L. Brizi, and G. Grisetti, "MAD-ICP: It is all about matching data—Robust and informed LiDAR odometry," *IEEE Robot. Autom. Lett.*, vol. 9, no. 11, pp. 9175–9182, Sep. 2024.
- [84] S. Floery, "Constrained matching of point clouds and surfaces," Ph.D. thesis, Institut für Diskrete Mathematik und Geometrie, Technische Universität Wien, Vienna, Austria, 2010. [Online]. Available: <http://hdl.handle.net/20.500.12708/11766>
- [85] C. Olsson and A. Eriksson, "Solving quadratically constrained geometrical problems using Lagrangian duality," in *Proc. 19th Int. Conf. Pattern Recognit.*, Dec. 2008, pp. 1–5.
- [86] 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. Autom.*, May 2013, pp. 62–69.
- [87] J. Nubert, S. Khattak, and M. Hutter, "Graph-based multi-sensor fusion for consistent localization of autonomous construction robots," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 10048–10054.
- [88] J. Nubert et al., "Holistic fusion: Task- and setup-agnostic robot localization and state estimation with factor graphs," 2025, [arXiv:2504.06479](https://arxiv.org/abs/2504.06479).
- [89] S. Zhao, H. Zhang, P. Wang, L. Nogueira, and S. Scherer, "Super odometry: IMU-centric LiDAR-visual-inertial estimator for challenging environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 8729–8736.
- [90] S. Zhao et al., "SuperLoc: The key to robust LiDAR-inertial localization lies in predicting alignment risks," 2024, [arXiv:2412.02901](https://arxiv.org/abs/2412.02901).
- [91] P. Dellenbach, J.-E. Deschaud, B. Jacquet, and F. Goulette, "CT-ICP: Real-time elastic LiDAR odometry with loop closure," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2022, pp. 5580–5586.
- [92] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP variants on real-world data sets," *Auto. Robots*, vol. 34, no. 3, pp. 133–148, Apr. 2013.
- [93] S. Agarwal, K. Mierle, and T. C. S. Team. (Oct. 2024). *Ceres Solver*. [Online]. Available: <https://github.com/ceres-solver/ceres-solver>
- [94] M. E. Wall, A. Rechtsteiner, and L. M. Rocha, "Singular value decomposition and principal component analysis," in *A Practical Approach to Microarray Data Analysis*. Norwell, MA, USA: Springer, 2003, pp. 91–109.
- [95] M. T. Heath, *Scientific Computing*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2018, doi: [10.1137/1.9781611975581](https://doi.org/10.1137/1.9781611975581). [Online]. Available: <https://epubs.siam.org/doi/abs/10.1137/1.9781611975581>

- [96] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed., Philadelphia, PA, USA: Johns Hopkins Univ. Press, 2013, doi: [10.1137/1.9781421407944](https://doi.org/10.1137/1.9781421407944).
- [97] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. New York, NY, USA: Academic, 2014.
- [98] C. Schreppel and J. Brembeck, "A QP solver implementation for embedded systems applied to control allocation," *Computation*, vol. 8, no. 4, p. 88, Oct. 2020.
- [99] D. Goldfarb and A. Idrani, "Dual and primal-dual methods for solving strictly convex quadratic programs," in *Proc. Numer. Anal. 3rd IIMAS Workshop Held Cocoyoc*. Cham, Switzerland: Springer, Jan. 1982, pp. 226–239.
- [100] P. C. Hansen, "Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank," *SIAM J. Sci. Stat. Comput.*, vol. 11, no. 3, pp. 503–518, May 1990.
- [101] J. Draisma, G. Ottaviani, and A. Tocino, "Best rank-k approximations for tensors: Generalizing Eckart–Young," *Res. Math. Sci.*, vol. 5, no. 2, p. 27, Jun. 2018.
- [102] G. H. Golub, P. C. Hansen, and D. P. O’Leary, "Tikhonov regularization and total least squares," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 1, pp. 185–194, 1999.
- [103] A. W. Fitzgibbon, "Robust registration of 2D and 3D point sets," *Image Vis. Comput.*, vol. 21, nos. 13–14, pp. 1145–1153, Dec. 2003.
- [104] P. Bergström and O. Edlund, "Robust registration of point sets using iteratively reweighted least squares," *Comput. Optim. Appl.*, vol. 58, no. 3, pp. 543–561, Jul. 2014.
- [105] D. Jud et al., "HEAP—The autonomous walking excavator," *Autom. Construction*, vol. 129, Sep. 2021, Art. no. 103783.
- [106] M. Hutter et al., "ANYmal—toward legged robots for harsh environments," *Adv. Robot.*, vol. 31, no. 17, pp. 918–931, Sep. 2017.
- [107] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vilamoura-Algarve, Portugal, Oct. 2012, pp. 573–580.
- [108] M. Bloesch, M. Burri, H. Sommer, R. Siegwart, and M. Hutter, "The two-state implicit filter recursive estimation for mobile robots," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, pp. 573–580, Jan. 2018.
- [109] G. Del Duca and C. Machado, "Assessing the quality of the leica BLK2GO mobile laser scanner versus the focus 3D S120 static terrestrial laser scanner for a preliminary study of garden digital surveying," *Heritage*, vol. 6, no. 2, pp. 1007–1027, Jan. 2023.
- [110] M. Nissov, J. A. Edlund, P. Spieler, C. Padgett, K. Alexis, and S. Khattak, "Robust high-speed state estimation for off-road navigation using radar velocity factors," *IEEE Robot. Autom. Lett.*, vol. 9, no. 12, pp. 11146–11153, Dec. 2024.



TURCAN TUNA (Graduate Student Member, IEEE) received the double major B.Sc. degree (Hons.) in mechanical engineering and control and automation engineering from Istanbul Technical University, Istanbul, Türkiye, in 2019, and the M.Sc. degree in robotics, systems, and control from ETH Zürich, Zürich, Switzerland, in 2022, where he is currently pursuing the Ph.D. degree with the Robotic Systems Laboratory.

His research interests include developing and deploying robust localization, perception, and mapping frameworks on robotic systems.



JULIAN NUBERT (Student Member, IEEE) received the M.Sc. degree in robotics, systems, and control from ETH Zürich, Zürich, Switzerland, in 2020, where he is currently pursuing the Ph.D. degree with the Robotic Systems Laboratory.

He is also affiliated with the Max Planck Institute through the MPI ETH Center for Learning Systems, Stuttgart, Germany. His research interests lie in the field of robust robot perception, and how it can be used for the deployment of mobile

robotic systems.

Mr. Nubert received the ETH Silver Medal and was awarded the Willi-Studer-Price for his achievements during the master’s studies.



PATRICK PFREUNDSCHUH (Student Member, IEEE) received the M.Sc. degree in robotics, systems, and control from ETH Zürich, Zürich, Switzerland, in 2020, where he is currently pursuing the Ph.D. degree with the Autonomous Systems Laboratory.

In his research, he focuses on robust LiDAR odometry in challenging environments.

Mr. Pfreundschuh worked as a member of the team CERBERUS, which won the DARPA SubT Challenge in 2021.



CESAR CADENA (Member, IEEE) received the Ph.D. degree in computer science from the University of Zaragoza, Zaragoza, Spain, in 2011.

He is a Senior Scientist with ETH Zürich, Zürich, Switzerland. His research interests are in providing machines the ability to understand and interact with our ever-changing world. His work covers robotic scene understanding, both geometry and semantics, covering semantic mapping, data association, and place recognition tasks, simultaneous localization and mapping, as well as persistent mapping in dynamic environments and robot navigation.



SHEHRYAR KHATTAK (Member, IEEE) received the B.S. degree in mechanical engineering from Ghulam Ishaq Khan Institute of Engineering Sciences and Technology (GIKI), Topi, Khyber Pakhtunkhwa, Pakistan, in 2009, the M.S. degree in aerospace engineering from Korea Advanced Institute of Science & Technology (KAIST), Daejeon, South Korea, in 2012, and the M.S. and Ph.D. degrees in computer science from the University of Nevada at Reno, Reno, NV, USA,

in 2017 and 2019, respectively.

He is currently a Robotics Technologist with the Perception Systems Group, NASA’s Jet Propulsion Laboratory (JPL), Pasadena, CA, USA. Currently, he is the Principal Investigator (PI) for the Multi-Robot Autonomous Intelligent Search and Rescue Project, JPL. He previously served as the Perception Lead for JPL’s Team in the DARPA RACER Project and for team CERBERUS, winners of the DARPA Subterranean Challenge. Before joining JPL, he was a Post-Doctoral Researcher at ETH Zürich, Zürich, Switzerland. His work focuses on enabling resilient robot autonomy in complex environments through multisensor information fusion.



MARCO HUTTER (Member, IEEE) received the M.Sc. and Ph.D. degrees in design, actuation, and control of legged robots from ETH Zürich, Zürich, Switzerland, in 2009 and 2013, respectively.

He is currently an Associate Professor for robotic systems with ETH Zürich. He was a PI of the NCCRs Robotics and Digital Fabrication, a PI in various EU projects and international challenges, and a Co-Founder of several ETH startups such as ANYbotics AG, Zürich. His research interests are in the development of novel machines and actuation concepts together with the underlying control, planning, and machine learning algorithms for locomotion and manipulation.

Dr. Hutter was a recipient of an ERC Starting Grant.

• • •