DOA: A Degenerate Optimization Agent with Adaptive Pose Compensation Capability based on Deep Reinforcement Learning

Yanbin Li, Canran Xiao, Hongyang He, Shenghai Yuan, Ziruo Li, Jiajie Yu and Wenzheng Chi

I. Motivation & Rationality

Optimization methods are widely used in SLAM due to efficiency [3]–[5], [15]–[21]. These methods can fully utilize all available observation data, effectively handle the fusion of multi-sensor data and dynamic environmental changes, finally achieve high-precision localization and mapping by constructing a globally consistent optimal solution. SLAM based on optimization obtains the optimal robot poses and map point positions by minimizing the geometric constraints E between poses and map points or the errors between sensor measurements and model predictions:

$$E(x_t, m) = \sum_{i=1}^{n} ||z_t^{(i)} - h(x_t, m)||^2_{\sum_z^{-1}} + ||x_t - f(x_{t-1}, u_t)||^2_{\sum_o^{-1}} \tag{1}$$

where Σ_z and Σ_o represent the covariance matrices of observation and motion model, respectively.

SLAM may fail in degenerate environments due to the lack of environmental features. Suppose the motion direction is aligned with the x-axis. In the Jacobian matrix $J_h \approx \left[\frac{\partial h}{\partial x}, \frac{\partial h}{\partial y}, \frac{\partial h}{\partial \theta}\right]^T$ of the observation model, the component $\frac{\partial h}{\partial x}$ along the degenerate direction approaches zero. As shown in (2), the information matrix $I_z = J_h^T \Sigma_z^{-1} J_h$ of the observation model becomes rank-deficient in this direction. Consequently, the total information matrix $I = I_z + I_o$ which is shown in (3), has insufficient constraints in the x-direction, where I_o is the information matrix of motion model.

$$I_z \approx \begin{bmatrix} \lambda_x & 0 & 0 \\ 0 & \lambda_y & 0 \\ 0 & 0 & \lambda_\theta \end{bmatrix}, \quad \lambda_x \to 0 \tag{2}$$

$$I \approx \begin{bmatrix} (I_o)_x & 0 & 0 \\ 0 & \lambda_y + (I_o)_y & 0 \\ 0 & 0 & \lambda_\theta + (I_o)_\theta \end{bmatrix} \tag{3}$$

Since the pose covariance matrix Σ_x is equal to the inverse I^{-1} of information matrix, the uncertainty σ_x^2 of pose in x-direction is essentially determined by $(I_o)_x$:

$$\sigma_x^2 = I_x^{-1} \approx \frac{1}{(I_o)_x} \tag{4}$$

Due to the finite inherent noise $(I_o)_x$ of motion model, when the observation constraints are completely missing, the covariance approaches infinity as $\sigma_x^2 \to \infty$. At this point, the optimization algorithm cannot effectively update the pose in the x-direction, leading to the Hessian matrix $H = \frac{\partial^2 E}{\partial x^2}$ of the objective function shown in (5) being singular in the x-direction, where h is the second derivative.

$$H \approx \begin{bmatrix} 0 & 0 & 0 \\ 0 & h_{yy} & h_{y\theta} \\ 0 & h_{\theta y} & h_{\theta\theta} \end{bmatrix} \tag{5}$$

Furthermore, the iterative step length $\Delta x = -H^{-1}\nabla E$ of the optimization problem becomes unbounded in x-direction. The pose estimation error continues to accumulate, leading to a gradual decrease in the localization accuracy of SLAM.

The degenerate problem can be optimized through adaptive pose fusion. The motion model can provide a relatively accurate pose prediction in the short term to correct the failed observation pose. Specifically, the contribution of motion model and observation model is adjusted through the dynamic weight $a \in (0, 1)$ in our paper. The centroid $G_c \in \mathbb{R}^2$ and covariance Σ_c of the fused pose are:

$$\begin{aligned} G_c &= \Sigma_c[(1-a)\Sigma_z^{-1} G_z + a\Sigma_o^{-1} G_o], \\ \Sigma_c^{-1} &= (1-a)\Sigma_z^{-1} + a\Sigma_o^{-1} \end{aligned} \tag{6}$$

In the degenerate direction, Σ_z^{-1} approaches 0, so the fused covariance is $\Sigma_c^{-1} \approx a\Sigma_o^{-1}$. When the degree of degeneracy is high, increasing a enhances the contribution of the motion model to the pose optimization, thereby reducing the covariance σ_x^2 shown in (7) and improving the localization accuracy. SLAM based on optimization often uses a tight coupling method to achieve multi-sensor fusion and does not have the ability to dynamically allocate sensor weights, so it cannot achieve degeneracy optimization.

$$\sigma_x^2 = I_x^{-1} = \frac{1}{a(\Sigma_o^{-1})_x} \tag{7}$$

The dynamic sensor compensation problem possesses the core characteristics of Markov Decision Process (MDP). As introduced in our paper, we model this task using MDP, represented by the tuple $M = (S, A, p, r, \gamma)$. The state S encapsulates the position coordinates, which directly reflect the distribution characteristics of the current location and can describe the current state of the system. Moreover, the future state depends only on the current state and action, satisfying the Markov property. Subsequently, we employ deep reinforcement learning (DRL) to address the dynamic compensation problem. DRL does not rely on prior knowledge of the sensor noise distribution or the degree of environmental degeneracy. Instead, it learns adaptive policies through trial and error. The Degeneracy Optimization Agent (DOA) interacts dynamically with the SLAM environment and directly adjust the weight $a \in A$. We optimize the

TABLE I: Comparison of different methods.

Method	Multi-sensor Fusion?	Degeneracy Optimization?	Degeneracy Detection?	Detection Threshold Dependent?	Manual Labeling?	Parameters Dependent?
Orb-SLAM3 [1]	✗	✗	✗	✗	✗	✓
LOAM [2]	✗	✗	✗	✗	✗	✓
Dali-slam [3]	✓	✓	✓	✓	✗	✓
J. Zhang et al. [4]	✗	✓	✓	✓	✗	✓
Fast-livo2 [5]	✓	✓	✗	✗	✗	✓
Switch-slam [6]	✓	✓	✓	✓	✗	✓
P2d-do [7]	✗	✓	✓	✓	✗	✓
H. Zhou et al. [8]	✓	✓	✓	✓	✗	✓
X-icp [9]	✗	✓	✓	✓	✗	✓
GMapping [10]	✓	✗	✗	✗	✗	✓
Cartographer [11]	✓	✗	✗	✗	✗	✓
Hector [12]	✗	✗	✗	✗	✗	✓
J. Nubert et al. [13]	✗	✗	✓	✗	✓	✗
Li et al. [14]	✓	✓	✓	✗	✓	✗
Ours	✓	✓	✓	✗	✗	✗

policy network parameters θ of DOA using Proximal Policy Optimization (PPO):

$$\nabla_\theta J(\theta) = E_t \left[\nabla_\theta \log \pi_\theta(a_t|s_t) \cdot A_t\right] \tag{8}$$

where π is the current policy and E_t is the expected value of all timesteps. The advantage function $A_t = R_t + \gamma V_\phi(s_{t+1}) - V_\phi(s_t)$ measures the quality of an action using discount factor γ. The value network V_ϕ learns the state value by minimizing the temporal difference error:

$$L_V = E_t \left[(V_\phi(s_t) - (R_t + \gamma V_\phi(s_{t+1})))^2\right] \tag{9}$$

In PPO, the policy update magnitude is constrained to avoid policy oscillation during training. Its objective function is a clipped surrogate objective:

$$J(\theta) = E_t \left[\min\left(r_t(\theta)A_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t\right)\right] \tag{10}$$

Here, $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{old}(a_t|s_t)}$ is the importance sampling ratio, ϵ is the clipping threshold and $clip$ is the clipping function. This method enhances the stability of training while maintaining the ability to explore. The DOA training by PPO adjusts the sensor fusion weights by outputting a dynamic factor a that reflects the degree of degeneracy, thereby preventing the covariance of the fused position from diverging and improving the accuracy of localization:

$$\sigma_x^2 \leq \frac{1}{a_{\min}(\Sigma_o^{-1})} < \infty \tag{11}$$

II. What problem does our approach solve

As shown in Table. I, compared with SLAM using single sensors [1], [2], [4], [7], [9], [12], our method introduces a dynamic sensor compensation strategy based on the results of degeneracy detection. When lidar degeneracy occurs, odometry constraint information is introduced to improve the accuracy of pose optimization. Moreover, our method also includes a separate degeneracy optimization module. Traditional loop closure methods [1], [2] can enhance the probability of loop closure by detecting scene similarity, but these methods are not applicable in straight corridor scenarios. The separate degeneracy optimization module can improve SLAM accuracy in various scenes through real-time optimization.

Traditional degeneracy optimization methods [6]–[9], which heavily rely on the precision of degeneracy detection, require adjustment of degeneracy detection thresholds in different scenes, limiting the generalization ability of these optimization methods. Deep learning methods can be used to train models with degeneracy detection capabilities through supervised learning [13], [14]. However, these methods are highly dependent on the quality of the dataset, and the rules followed for manual labeling significantly affect the detection results. Moreover, degenerate data is difficult to collect and process. We use DRL to address the dependency of traditional supervised learning on labeling. DOA interacts with SLAM in real-time, captures degenerate features of the environment, and takes different optimization strategies based on the degree of degeneracy. It ultimately learns adaptive strategies through trial and error based on rewards. Experiments have proven that DOA has good generalization ability in various environments, without the need for parameter adjustment, achieving truly adaptive optimization.

III. Comparison with Cartographer

We conduct comparative experiments with Cartographer in the six environments set up in the paper (S1-S4 and R1-R2). The maps generated by both Cartographer and our

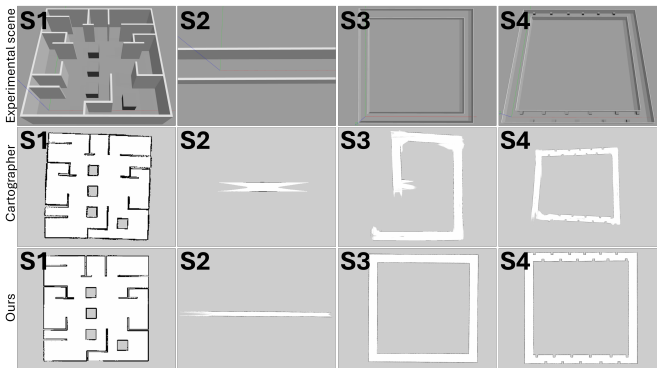

Fig. 1: Maps of Cartographer [11] and our method in simulations.

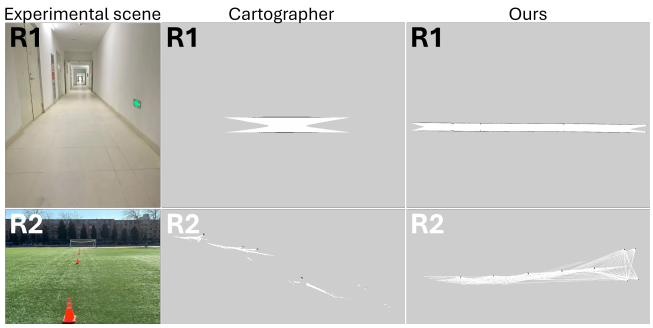

Fig. 2: Maps of Cartographer [11] and our method in real-world experiments.

method are shown in Fig. 1 and Fig. 2. In S1, both methods produce relatively accurate results. In S2, Cartographer exhibit motion stalling along the direction of movement in the long straight corridor, leading to incorrect corridor length. In S3, the loop closure in the map built by Cartographer fail to close correctly, resulting in significant misalignment. In S4, we enrich the environmental features by setting up obstacles. Despite improvements in Cartographer's loop closure performance, slight misalignments remain. In R1, Cartographer show motion stalling again, causing incorrect corridor length. In R2, the relative positions of landmarks in the map built by Cartographer are incorrect, with severe localization drift. In contrast, our method accurately build the map structures in all these environments, demonstrating robust optimization capabilities under degenerate conditions. In summary, Cartographer's performance in degenerate environments is subpar. Its loop closure mechanism fail to correct localization errors that arise in such environments. Moreover, Cartographer's heavy reliance on parameter tuning significantly undermines its generalization capability.

REFERENCES

[1] C. Campos, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[2] J. Zhang, "Loam: Lidar odometry and mapping in real-time." in *Robotics: Science and systems*, vol. 2, no. 9. Berkeley, CA, 2014, pp. 1–9.

[3] W. Wu, C. Chen, B. Yang, X. Zou, F. Liang, Y. Xu, and X. He, "Dali-slam: Degeneracy-aware lidar-inertial slam with novel distortion correction and accurate multi-constraint pose graph optimization," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 221, pp. 92–108, 2025.

[4] J. Zhang, M. Kaess, and S. Singh, "On degeneracy of optimization-based state estimation problems," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 809–816.

[5] C. Zheng, "Fast-livo2: Fast, direct lidarâinertialâvisual odometry," *IEEE Transactions on Robotics*, vol. 41, pp. 326–346, 2025.

[6] J. Lee, "Switch-slam: Switching-based lidar-inertial-visual slam for degenerate environments," *IEEE Robotics and Automation Letters*, 2024.

[7] W. Chen, "P2d-do: Degeneracy optimization for lidar slam with point-to-distribution detection factors," *IEEE Robotics and Automation Letters*, pp. 1–8, 2024.

[8] H. Zhou, "Lidar/uwb fusion based slam with anti-degeneration capability," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 1, pp. 820–830, 2020.

[9] T. Tuna, "X-icp: Localizability-aware lidar registration for robust localization in extreme environments," *IEEE Transactions on Robotics*, 2023.

[10] G. Grisetti, "Improved techniques for grid mapping with rao-blackwellized particle filters," *IEEE transactions on Robotics*, vol. 23, no. 1, pp. 34–46, 2007.

[11] W. Hess, "Real-time loop closure in 2d lidar slam," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 1271–1278.

[12] S. Kohlbrecher, "A flexible and scalable slam system with full 3d motion estimation," in *2011 IEEE international symposium on safety, security, and rescue robotics*. IEEE, 2011, pp. 155–160.

[13] J. Nubert, "Learning-based localizability estimation for robust lidar localization," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 17–24.

[14] Y. Li, "Anti-degeneracy scheme for lidar slam based on particle filter in geometry feature-less environments," *arXiv preprint arXiv:2502.11486*, 2025.

[15] G. Dubbelman and B. Browning, "Cop-slam: Closed-form online pose-chain optimization for visual slam," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1194–1213, 2015.

[16] C. Zheng, "Fast-livo: Fast and tightly-coupled sparse-direct lidar-inertial-visual odometry," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 4003–4009.

[17] J. Xu, "Intermittent vio-assisted lidar slam against degeneracy: Recognition and mitigation," *IEEE Transactions on Instrumentation and Measurement*, vol. 74, pp. 1–13, 2025.

[18] Q. H. Hoang and G.-W. Kim, "Imu augment tightly coupled lidar-visual-inertial odometry for agricultural environments," *IEEE Robotics and Automation Letters*, vol. 9, no. 10, pp. 8483–8490, 2024.

[19] K. Xu and S. Yuan, "Airslam: An efficient and illumination-robust point-line visual slam system," *IEEE Transactions on Robotics*, pp. 1–20, 2025.

[20] A. Adkins, "Obvi-slam: Long-term object-visual slam," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2909–2916, 2024.

[21] A. Kumar, "High-speed stereo visual slam for low-powered computing devices," *IEEE Robotics and Automation Letters*, vol. 9, no. 1, pp. 499–506, 2024.