Convergence and Consistency Analysis for A Lie Group Based 3D EKF SLAM (Full Version)

Teng Zhang, Kanzhi Wu, Jingwei Song,
Shoudong Huang and Gamini Dissanayake*

Abstract

In this paper, we investigate the convergence and consistency properties of a Lie group based 3D Extended Kalman Filter (EKF) Simultaneous Localization and Mapping (SLAM) algorithm. Each element of the adopted Lie group consists of one robot pose and positions of all the observed landmarks. Basic convergence properties of the new EKF SLAM algorithm are proven without having to resort to the restrictive assumption that Jacobians of the motion and observation models need to be evaluated at the ground truth, as in [1]. It is also shown that output of the Lie group based EKF SLAM algorithm is invariant under any *stochastic* rigid body transformation in contrast to the traditional EKF SLAM algorithm that is only invariant under deterministic rigid body transformations. Implications of this much stronger property on the consistency of the estimator are discussed. Monte Carlo simulation results demonstrate that the Lie group based EKF SLAM algorithm outperforms the traditional EKF and the "First Estimates Jacobian" EKF in different scenarios.

1 INTRODUCTION

Extended Kalman filter (EKF) has been used extensively in solving SLAM problem in the past. The main advantage of the EKF SLAM is that its state vector is relatively compact, consisting of the current robot pose and all landmark locations in contrast to optimization based algorithms (e.g., [2]) where all the previous robot poses are included in the state vector. However, possible estimator inconsistency in the traditional EKF based point feature SLAM was recognized as early as in 2001 [3]. This issue was discussed in detail later in [1][4]. Inconsistency here refers to the fact that the algorithm underestimates the uncertainty of the estimate leading to an overconfident result. This is clearly a limitation of EKF SLAM and the optimization based algorithms in recent implementations have been widely used due to their good performance and the efficiency of the modern sparse solvers [5][6]. This paper introduces a Lie group based

*The authors are with Center for Autonomous Systems, University of Technology Sydney, Australia {Teng.Zhang, Kanzhi.Wu, Jingwei.Song, Shoudong.Huang, Gamini.Dissanayake}@uts.edu.au

3D EKF SLAM with consistency properties that provides an alternative solution, particularly when computational resources are limited.

Some research to enhance the consistency of EKF SLAM is reported in the literature. Robocentric EKF SLAM [7] estimates the location of landmarks in the robot local coordinate frame. As a result landmark positions to be estimated keep changing although landmarks are stationary in a fixed global coordinate frame. However, it has been shown that robot-centric formulations lead to better performance in terms of estimator consistency. Guerreiro et al. [8] also reported a globally asymptotically stable filter for SLAM problem formulated in the local coordinate system. Although this algorithm is shown to produce consistent estimates, it is verified using only a small-scale experimental dataset.

It was shown in [9] that the inconsistency in EKF SLAM is closely related to the partial observability of SLAM problem [10][11]. This insight resulted in a number of EKF SLAM algorithms which significantly improve consistency, such as the "First Estimates Jacobian" EKF SLAM [9], observability-constrained EKF SLAM [12][13]. Recently, in [14], a 2D invariant EKF SLAM algorithm is proposed that is demonstrated to be consistent in simulations. However, most of the above mentioned algorithms are 2D and no theoretical proof on the convergence is provided in all the above mentioned research.

Recently, Lie group representation for three-dimensional orientation/pose has become popular in SLAM and 3D reconstruction in computer vision [6][15]. Advantages of using Lie group formulation for vehicle state estimations are highlighted in recent research [2][16]. It has been shown that better convergence and accuracy can be achieved for both filter based algorithms [17] and the optimization based algorithms [18], when Lie group representation is used for the state vector. The invariant EKF SLAM in [14] also intrinsically uses the Lie group representation, although it is only formulated for two-dimensional scenarios.

Motivated by the uncertainty representation in [14], in this paper a Lie group based 3D EKF SLAM algorithm[1] is introduced. Instead of only representing the space of robot orientation using the Lie group $\mathbb{SO}(3)$, a Lie group $\mathscr{G}(N)$ which consists of both robot pose and landmarks is used as the state space. For this new SLAM algorithm, it is important to analyze convergence properties in addition to consistency. A number of authors have addressed the behaviour of EKF SLAM to examine the convergence properties and derive bounds for the uncertainty of the estimate. In 2001, Dissanayake et. al. [19] proved three essential convergence properties of the algorithm under the assumption of linear motion and observation models, with theoretical achievable lower bounds on the covariance matrix given. In 2006, Mourikis and Roumeliotis [20] provided an analytical upper bound of the map uncertainty based on the observation noise level, the process noise level, and the size of the map. In 2007, Huang and Dissanayake [1] extended the proof of the convergence properties and the achievable lower bounds on covariance matrix in [19] to nonlinear case, but under a restrictive assumption that the Jacobians are evaluated at the ground truth.

In this paper, a convergence analysis for the proposed Lie group based EKF SLAM

[1]The main contributions of this paper are the theoretical proofs of convergence and consistency properties.

that does not require this assumption is presented. Furthermore, it is shown that output of the filter is invariant under any *stochastic* rigid body transformation in contrast to the traditional EKF SLAM algorithm (T-EKF) that is only invariant under rigid body transformations, and an explanation for the importance of this distinction is provided. Monte Carlo simulations are used to demonstrate that the Lie group based EKF SLAM algorithm significantly outperforms both the traditional EKF SLAM algorithm and "First Estimates Jacobian" EKF SLAM algorithm, especially when the noise level is high.

This paper is organized as follows[2]. The motion model and observation model of SLAM problem in 3D is recalled in Section 2. In Section 3, the general EKF framework is introduced and Right Lie Group based EKF SLAM algorithm (R-EKF) is described. Section 4 proves some convergence results of R-EKF SLAM. In Section 5, the consistency of R-EKF is analyzed, it is proven that the output of R-EKF is invariant under stochastic rigid body transformation. In Section 6, the performance of R-EKF is demonstrated through Monte Carlo simulations. Finally, Section 7 outlines the main conclusions of this work.

2 Problem Statement

The EKF SLAM algorithms focus on estimating the current robot pose and the positions of all the observed landmarks with the given motion model and the observation model. In this work, SLAM problem in 3D scenarios is investigated and the state to be estimated is explicitly denoted by

$$\mathbf{X} = \left(\mathbf{R}, \mathbf{p}, \mathbf{f}^1, \cdots, \mathbf{f}^N\right), \tag{1}$$

where $\mathbf{R} \in \mathbb{SO}(3)$ and $\mathbf{p} \in \mathbb{R}^3$ are the robot orientation and robot position, $\mathbf{f}^i \in \mathbb{R}^3$ ($i = 1, \cdots, N$) is the coordinate of the landmark i, all described in the fixed world coordinate frame.

A general motion model for moving robot and static landmarks in 3D scenarios can be represented by

$$\begin{aligned} \mathbf{X}_{n+1} &= f(\mathbf{X}_n, \mathbf{u}_n, \boldsymbol{\varepsilon}_n) \\ &= \left(\mathbf{R}_n \exp(\mathbf{w}_n + \boldsymbol{\varepsilon}_\mathbf{w}), \mathbf{p}_n + \mathbf{R}_n(\mathbf{v}_n + \boldsymbol{\varepsilon}_\mathbf{v}), \mathbf{f}_n^1, \cdots, \mathbf{f}_n^N\right), \end{aligned} \tag{2}$$

where $\mathbf{u}_n = \begin{bmatrix} \mathbf{w}_n^\mathsf{T} & \mathbf{v}_n^\mathsf{T} \end{bmatrix}^\mathsf{T} \in \mathbb{R}^6$ is the odometry, being $\mathbf{w}_n \in \mathbb{R}^3$ and $\mathbf{v}_n \in \mathbb{R}^3$ the angular increment and linear translation from time n to time $n+1$, $\exp(\mathbf{w}_n) \in \mathbb{SO}(3)$ is the rotation corresponding to $\mathbf{w}_n$ (exp is the exponential mapping for $\mathbb{SO}(3)$ defined in (17)) and $\boldsymbol{\varepsilon}_n = \begin{bmatrix} \boldsymbol{\varepsilon}_\mathbf{w}^\mathsf{T} & \boldsymbol{\varepsilon}_\mathbf{v}^\mathsf{T} \end{bmatrix}^\mathsf{T} \sim \mathcal{N}(\mathbf{0}, \Phi_n)$ is the odometry noise at time n.

As the robot is likely to observe different sets of landmarks in each time step, the notation O_{n+1} is used to represent the set that indicates the landmarks observed at time

[2]**Notations:** Throughout this paper bold lower-case and upper-case letters are reserved for vectors and matrices, respectively. $\mathbf{S}_1 \preceq \mathbf{S}_2$ means $\mathbf{S}_2 - \mathbf{S}_1$ is a positive semi-definite matrix. The function $\mathrm{S}(\cdot)$ is the skew symmetric operator that transforms a 3-dimensional vector into a skew symmetric matrix: $\mathrm{S}(\mathbf{x})\mathbf{y} = \mathbf{x} \times \mathbf{y}$ for $\mathbf{x}, \mathbf{y} \in \mathbb{R}^3$, where the notation $\times$ refers to the cross product.

$n+1$. Also by assuming a 3D sensor which provides the coordinate of landmark i in $n+1$-th robot frame, observation model is given as follows

$$\mathbf{z}_{n+1} = h_{n+1}(\mathbf{X}_{n+1}, \xi_{n+1}), \tag{3}$$

where $h_{n+1}(\mathbf{X}_{n+1}, \xi_{n+1})$ is a column vector obtained by stacking all entries $h^i(\mathbf{X}_{n+1}, \xi^i_{n+1}) = \mathbf{R}^\intercal_{n+1}(\mathbf{f}^i_{n+1} - \mathbf{p}_{n+1}) + \xi^i_{n+1} \in \mathbb{R}^3$ for all $i \in \mathrm{O}_{n+1}$, $\xi_{n+1} \sim \mathcal{N}(\mathbf{0}, \Psi_{n+1})$ is the observation noise vector obtained by stacking all entries $\xi^i_{n+1} \sim \mathcal{N}(\mathbf{0}, \Psi^i_{n+1})$ $(i \in \mathrm{O}_{n+1})$. The covariance matrix Ψ_{n+1} of observation noise is a block diagonal matrix consisting of all Ψ^i_{n+1} $(i \in \mathrm{O}_{n+1})$.

3 Lie Group based EKF SLAM Algorithm

In this section, the general EKF framework inspired by [21] as an extension of standard EKF is briefly introduced. Then a Lie group based EKF SLAM algorithm (R-EKF) under this framework is provided.

3.1 General EKF framework

Algorithm 1: General EKF framework

Input: $\hat{\mathbf{X}}_n$, $\mathbf{P}_n$, $\mathbf{u}_n$, $\mathbf{Z}_{n+1}$;
Output: $\hat{\mathbf{X}}_{n+1}$, $\mathbf{P}_{n+1}$;
Propagation:
$\hat{\mathbf{X}}_{n+1|n} \leftarrow f(\mathbf{X}_n, \mathbf{u}_n, \mathbf{0})$, $\mathbf{P}_{n+1|n} \leftarrow \mathbf{F}_n\mathbf{P}_n\mathbf{F}_n^\intercal + \mathbf{G}_n\Phi_n\mathbf{G}_n^\intercal$;
Update:
$\mathbf{S} \leftarrow \mathbf{H}_{n+1}\mathbf{P}_{n+1|n}\mathbf{H}_{n+1}^\intercal + \Psi_{n+1}$, $\mathbf{K} \leftarrow \mathbf{P}_{n+1|n}\mathbf{H}_{n+1}^\intercal\mathbf{S}^{-1}$;
$\mathbf{y} \leftarrow h_{n+1}(\hat{\mathbf{X}}_{n+1|n}, \mathbf{0}) - \mathbf{z}_{n+1}$;
$\mathbf{X}_{n+1} \leftarrow \mathbf{X}_{n+1|n} \oplus \mathbf{K}\mathbf{y}$, $\mathbf{P}_{n+1} \leftarrow (\mathbf{I} - \mathbf{K}\mathbf{H}_{n+1})\mathbf{P}_{n+1|n}$;
(Optional: $\mathbf{P}_{n+1} \leftarrow \mathbf{J}_{n+1}\mathbf{P}_{n+1}\mathbf{J}_{n+1}^\intercal$);

In the general EKF framework, the state space is formulated as a $\mathfrak{N}$ dimensional manifold. The uncertainty of $\mathbf{X}$ is described by $\mathbf{X} = \hat{\mathbf{X}} \oplus \mathbf{e}$, where $\mathbf{e} \in \mathbb{R}^{\mathfrak{N}}$ is a white Gaussian noise vector with covariance $\mathbf{P}$ and $\hat{\mathbf{X}}$ in this manifold is the mean estimate of $\mathbf{X}$. In simple terms, the notation $\oplus$ has to be designed as a smooth mapping such that $\mathbf{X} = \mathbf{X} \oplus \mathbf{0}$ and there exists the inverse mapping $\ominus$ of $\oplus$: $\mathbf{e} = \mathbf{X} \ominus \hat{\mathbf{X}}$. The general EKF framework summarized in Alg. 1 follows the framework of the standard EKF via the first-order approximation except for the uncertainty representation and the optional step, where the Jacobians are given as: $\mathbf{F}_n = \frac{\partial(f(\hat{\mathbf{X}}_n \oplus \mathbf{e}, \mathbf{u}_n, \mathbf{0}) \ominus f(\hat{\mathbf{X}}_n, \mathbf{u}_n, \mathbf{0}))}{\partial \mathbf{e}}|_{\mathbf{e}=\mathbf{0}}$, $\mathbf{G}_n = \frac{\partial(f(\hat{\mathbf{X}}_n, \mathbf{u}_n, \varepsilon) \ominus f(\hat{\mathbf{X}}_n, \mathbf{u}_n, \mathbf{0}))}{\partial \varepsilon}|_{\varepsilon=\mathbf{0}}$ and $\mathbf{H}_{n+1} = \frac{\partial h_{n+1}(\hat{\mathbf{X}}_{n+1|n} \oplus \mathbf{e}, \mathbf{0})}{\partial \mathbf{e}}|_{\mathbf{e}=\mathbf{0}}$[3].

[3] $\mathbf{J}_{n+1} = \frac{\partial((\hat{\mathbf{X}}_{n+1|n} \oplus \mathbf{K}\mathbf{y}) \ominus (\hat{\mathbf{X}}_{n+1|n} \oplus \mathbf{e}))}{\partial \mathbf{e}}|_{\mathbf{e}=\mathbf{K}\mathbf{y}}$. In the following algorithm, the optional step is turned off for consistency issue, see Appendix 4.

3.2 R-EKF SLAM algorithm

The R-EKF SLAM algorithm formulates the state space of SLAM system in Lie group $\mathscr{G}(N)$ that contains the robot pose and N observed landmarks. The background knowledge about Lie group $\mathscr{G}(N)$ is provided in Appendix .1.

3.2.1 The choice of $\oplus$

Using exponential mapping exp on $\mathscr{G}(N)$, the special addition $\oplus$ in R-EKF is chosen such that $\mathbf{X} = \hat{\mathbf{X}} \oplus \mathbf{e} := \exp(\mathbf{e})\hat{\mathbf{X}}$, where exp is the exponential mapping[4], $\mathbf{X} \in \mathscr{G}(N)$ is the actual pose and landmarks, $\hat{\mathbf{X}} \in \mathscr{G}(N)$ is the *mean* estimate and the noise $\mathbf{e} = \begin{bmatrix} \mathbf{e}_\theta^\mathsf{T} & \mathbf{e}_p^\mathsf{T} & (\mathbf{e}^1)^\mathsf{T} & \cdots & (\mathbf{e}^N)^\mathsf{T} \end{bmatrix}^\mathsf{T} \in \mathbf{R}^{3N+6}$ follows the Gaussian distribution $\mathscr{N}(\mathbf{0}, \mathbf{P})$.

3.2.2 Jacobian matrices

The Jacobians in propagation are[5]

$$\begin{aligned} \mathbf{F}_n &= \mathbf{I}_{3N+6}, \\ \mathbf{G}_n &= \mathrm{ad}_{\hat{\mathbf{X}}_n} \mathbf{B}_n, \end{aligned} \tag{4}$$

where $\mathbf{B}_n = \begin{bmatrix} -\mathrm{J}_r(-\mathbf{w}_n) & \mathbf{0}_{3,3} \\ -\mathrm{S}(\mathbf{v}_n)\mathrm{J}_r(-\mathbf{w}_n) & \mathbf{I}_3 \\ \mathbf{0}_{3N,3} & \mathbf{0}_{3N,3} \end{bmatrix}$. Adjoint ad and right Jacobian J_r are given in Appendix A. The Jacobian matrix $\mathbf{H}_{n+1}$ of observation model in update is obtained by stacking all matrices $\mathbf{H}_{n+1}^i$ for all $i \in \mathrm{O}_{n+1}$, where

$$\mathbf{H}_{n+1}^i = \begin{bmatrix} \mathbf{0}_{3,3} & \hat{\mathbf{R}}_{n+1|n}^\mathsf{T} & \mathbf{0}_{3,3(i-1)} & -\hat{\mathbf{R}}_{n+1|n}^\mathsf{T} & \mathbf{0}_{3,3(N-i)} \end{bmatrix}. \tag{5}$$

3.2.3 Landmark initialization

This subsection provides the method to augment the state $\mathbf{X} \in \mathscr{G}(N)$ and adjust the covariance matrix $\mathbf{P}$ when the robot observes a new landmark with the observation $\mathbf{z} \in \mathbf{R}^3$. For brevity, the mathematical derivation is ignored here and the process for augment is summarized in Alg. 2, where

$$\mathbf{M}_N := \begin{bmatrix} \mathbf{0}_{3,3} & \mathbf{I}_3 & \mathbf{0}_{3,3N} \end{bmatrix}^\mathsf{T} \tag{6}$$

and Ψ is the covariance matrix representing the noise level in the new landmark observation.

4 Convergence Analysis of R-EKF SLAM algorithm

This section proves some convergence results for R-EKF. The obvious first result is:

[4]The exponential mapping is an overloaded function for Lie group and hence we also denote exp as the exponential mapping for $\mathscr{G}(N)$.

[5]For more details, refer to Appendix 1 and 6–8.

Algorithm 2: R-EKF SLAM: Landmark Initialization

Input: $\hat{\mathbf{X}}$, $\mathbf{P}$, $\mathbf{z}$;
Output: $\hat{\mathbf{X}}_{new}$, $\mathbf{P}_{new}$;
Process:
$\hat{\mathbf{f}}^{N+1} = \hat{\mathbf{p}} + \hat{\mathbf{R}}\mathbf{z} \in \mathbb{R}^3$
$\hat{\mathbf{X}}_{new} \leftarrow (\hat{\mathbf{X}}, \hat{\mathbf{f}}^{N+1}) \in \mathscr{G}(N+1)$;
$\mathbf{P}_{new} \leftarrow \begin{bmatrix} \mathbf{P} & \mathbf{P}\mathbf{M}_N \\ \mathbf{M}_N^\intercal \mathbf{P} & \hat{\mathbf{R}}\Psi\hat{\mathbf{R}}^\intercal + \mathbf{M}_N^\intercal \mathbf{P}\mathbf{M}_N \end{bmatrix}$.

Theorem 1 *In R-EKF SLAM algorithm, the covariance matrix has the property:* $\mathbf{P}_n \preceq \mathbf{P}_{n+1|n}$ *and* $\mathbf{P}_{n+1} \preceq \mathbf{P}_{n+1|n}$[6].

Proof 1 *The covariance matrix "increases" in the propagation:* $\mathbf{P}_n \preceq \mathbf{F}_n\mathbf{P}_n\mathbf{F}_n^\intercal + \mathbf{G}_n\Phi_n\mathbf{G}_n^\intercal = \mathbf{P}_{n+1|n}$ *due to* $\mathbf{F}_n = \mathbf{I}$. *The covariance matrix "decreases" in the update:* $\mathbf{P}_{n+1} = (\mathbf{I} - \mathbf{K}\mathbf{H}_{n+1})\mathbf{P}_{n+1|n} = \mathbf{P}_{n+1|n} - \mathbf{P}_{n+1|n}^\intercal(\mathbf{H}_{n+1}^\intercal \mathbf{S}^{-1}\mathbf{H}_{n+1})\mathbf{P}_{n+1|n} \preceq \mathbf{P}_{n+1|n}$ *due to* $\mathbf{S}$ *is symmetric.*

The general expression for the covariance matrices evolution of R-EKF cannot be easily obtained. Therefore, two representative scenarios are considered: (i) the robot is stationary, and (ii) the robot then moves one step. The convergence results of R-EKF SLAM algorithm in the two scenarios are presented and proven, under the condition that Jacobians are evaluated at the latest estimate. Hence the results are significant extension to similar theorems in [1] where Jacobians evaluated at the true state are assumed to be available.

The general setting analyzed in the following subsections is as follows. The robot starts at point A with the initial condition $(\hat{\mathbf{X}}_0, \mathbf{P})$, where $\mathbf{P}$ is covariance matrix and $\hat{\mathbf{X}}_0 = (\hat{\mathbf{R}}, \hat{\mathbf{p}}, \hat{\mathbf{f}}^1, \cdots, \hat{\mathbf{f}}^N)$ (N landmarks have been observed). The covariance matrix of odometry noise is Φ and the covariance matrix of one landmark observation noise is Ψ. In the following sections, $\mathbf{M}_N$ is given in (6), $\mathbf{L} := \mathbf{P}\mathbf{M}_N$ and $\mathbf{W} := \mathbf{M}_N^\intercal \mathbf{P}\mathbf{M}_N$.

4.1 Scenario A: Robot stationary

Theorem 2 *If the robot remains stationary at point A and does not observe any of the previously seen landmarks but observes a new landmark for k times, the mean estimate of robot pose and the N landmarks and their related uncertainty remain unchanged. The covariance matrix of the state when the new landmark is integrated becomes* $\mathbf{P}_k = \begin{bmatrix} \mathbf{P} & \mathbf{L} \\ \mathbf{L}^\intercal & \frac{\hat{\mathbf{R}}\Psi\hat{\mathbf{R}}^\intercal}{k} + \mathbf{W} \end{bmatrix}$. *When* $k \to \infty$, *the covariance matrix becomes*

$$\mathbf{P}_\infty^A = \begin{bmatrix} \mathbf{P} & \mathbf{L} \\ \mathbf{L}^\intercal & \mathbf{W} \end{bmatrix}. \tag{7}$$

Proof 2 *See Appendix .2.*

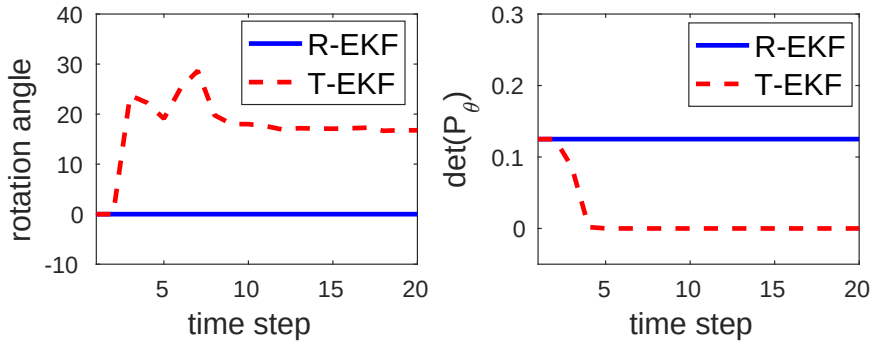

Figure 1: **Illustration of Theorem 2**. In this case, robot is stationary and always only observes the "new" landmark. Left: The error (unit: degree) in robot orientation w.r.t the ground truth as estimated by R-EKF and T-EKF[7]. Right: $\det(\mathbf{P}_\theta)$ estimated by R-EKF and T-EKF.

We illustrate the results of Theorem 2 using the following scenario. The simulated robot remains stationary and obtains accurate odometry, (i.e, $\mathbf{w} = \mathbf{v} = \mathbf{0}$ and $\Phi = \mathbf{0}$) and always observes the "new" landmark ($\Psi \neq 0$). The "new" landmark is always being observed and the standard deviation of observation noise is set as 5% of robot-to-landmark distance along each axis. The initial covariance matrix $\mathbf{P}_\theta \in \mathbb{R}^{3\times 3}$ of robot orientation is set as $\frac{1}{2}\mathbf{I}_3$. Fig. 1 presents results of a simulation of this scenario. Clearly observations to previously unseen landmarks do not convey any new information on the location of the robot. The rotation angle relative to the initial orientation and $\det(\mathbf{P}_\theta)$ from R-EKF correctly infers that the robot remains stationary and the orientation uncertainty remains unchanged. In contrast, T-EKF updates the robot orientation and furthermore predicts that the orientation uncertainty decreases as observations continue to be made, both of which are clearly erroneous and therefore leads to estimator inconsistency.

Theorem 2 can be easily extended into a multiple landmark scenario.

Corollary 1 *If the robot is stationary at point A and only observes m new landmark k times, the estimate of pose from R-EKF does not change while the covariance matrix of the estimate becomes* $\mathbf{P}_k = \begin{bmatrix} \mathbf{P} & \mathbf{L} & \mathbf{L} & \cdots & \mathbf{L} \\ \mathbf{L}^\intercal & \mathbf{Q}_k & \mathbf{W} & \cdots & \mathbf{W} \\ \mathbf{L}^\intercal & \mathbf{W} & \mathbf{Q}_k & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \mathbf{W} \\ \mathbf{L}^\intercal & \mathbf{W} & \cdots & \mathbf{W} & \mathbf{Q}_k \end{bmatrix}$, *where* $\mathbf{Q}_k = \frac{\hat{\mathbf{R}}\Psi\hat{\mathbf{R}}^\intercal}{k} + \mathbf{W}$.

[6]Here it is assumed that no new landmark is observed and thus the dimension of covariance does not change.

[7]T-EKF is the extension of 2D traditional EKF SLAM algorithm into 3D case based on the general EKF framework.

When $k \to \infty$, the covariance matrix becomes

$$\mathbf{P}_{\infty}^{A} = \begin{bmatrix} \mathbf{P} & \mathbf{L} & \mathbf{L} & \cdots & \mathbf{L} \\ \mathbf{L}^{\intercal} & \mathbf{W} & \mathbf{W} & \cdots & \mathbf{W} \\ \mathbf{L}^{\intercal} & \mathbf{W} & \mathbf{W} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \mathbf{W} \\ \mathbf{L}^{\intercal} & \mathbf{W} & \cdots & \mathbf{W} & \mathbf{W} \end{bmatrix}. \tag{8}$$

4.2 Scenario B: Robot takes a step after a stationary period

Consider the condition that the robot moves one step after being stationary for a period of time while observing new landmarks.

Theorem 3 *Assume $\Psi = \phi \mathbf{I}_3$ ($\phi \in \mathbb{R}^+$). If the robot remains stationary at point A, does not observe any of the previously seen landmarks but observes m new landmarks for $k = \infty$ times and then take a step to B using control action $\mathbf{u} = \begin{bmatrix} \mathbf{w}^{\intercal} & \mathbf{v}^{\intercal} \end{bmatrix}^{\intercal}$ and observes the same set of landmarks l times, then the covariance matrix from R-EKF becomes $\mathbf{P}_l^B = \mathbf{P}_{\infty}^A + \bar{\mathbf{P}}_l^B$, where $\mathbf{P}_{\infty}^A$ is given in (8), $\bar{\Psi} = \phi \mathbf{I}_{3m}$ and*

$$\bar{\mathbf{P}}_l^B = \mathrm{ad}_{\hat{\mathbf{X}}_\mathrm{A}} \mathbf{E} (\tilde{\Phi}^{-1} + l \tilde{\mathbf{H}}^{\intercal} \bar{\Psi}^{-1} \tilde{\mathbf{H}})^{-1} \mathbf{E}^{\intercal} \mathrm{ad}_{\hat{\mathbf{X}}_\mathrm{A}}^{\intercal}, \tag{9}$$

where $\bar{\Psi} = \phi \mathbf{I}_{3m}$ and the control noise covariance matrix is Φ. In (9), $\hat{\mathbf{X}}_A$ is the estimated state at the point A, $\tilde{\Phi} = \mathbf{B} \Phi \mathbf{B}^{\intercal}$ is a positive definite matrix and

$$\begin{aligned} \mathbf{B} &= \begin{bmatrix} -\mathrm{J_r}(-\mathbf{w}) & \mathbf{0}_{3,3} \\ -\mathrm{S}(\mathbf{v}) \mathrm{J_r}(-\mathbf{w}) & \mathbf{I}_3 \end{bmatrix}, \\ \mathbf{E} &= \begin{bmatrix} \mathbf{I}_6 \\ \mathbf{0}_{3(N+m),6} \end{bmatrix}, \ \tilde{\mathbf{H}} = \mathbf{H} \mathrm{ad}_{\hat{\mathbf{X}}_\mathrm{A}} \mathbf{E}, \end{aligned} \tag{10}$$

where $\mathbf{H}$ is obtained by stacking all matrices $\mathbf{H}^i = \begin{bmatrix} \mathbf{0}_{3,3} & \mathbf{I}_3 & \mathbf{0}_{3,3(N+i-1)} & -\mathbf{I}_3 & \mathbf{0}_{3,3(m-i)} \end{bmatrix}$. When l tends to infinity, the covariance matrix becomes $\mathbf{P}_{\infty}^B = \mathbf{P}_{\infty}^A$ under the condition that there are three landmarks non-coplanar with the robot position.

Proof 3 *See Appendix .3.*

We illustrate the results of Theorem 3 using the following scenario. Initially the robot is stationary at point A and continually observes ten previously unseen landmarks. It moves one step to point B after 200 such observations and then remains stationary for 200 more time steps while observing the same set of landmarks. The initial covariance matrix of robot pose is set as non-zero. In Fig. 2, $\log(\det(\mathbf{P}_r))$ describes the robot pose uncertainty, where $\mathbf{P}_r \in \mathbb{R}^{6\times6}$ is the covariance matrix of the robot pose. In Fig. 2, the pose uncertainty from R-EKF remains unchanged and increases at time 200 when robot moves one step due to odometry noise as expected. Further landmark observations at point B while remaining stationary gradually reduce the pose uncertainty. In contrast, pose uncertainty from T-EKF falls below the initial value indicating incorrect injection of information, leading to an overconfident estimate of uncertainty.

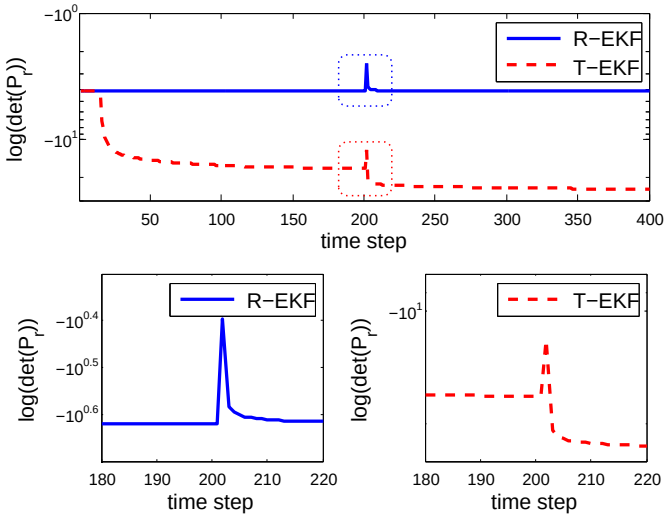

Figure 2: **Illustration of Theorem 3**. The y-axis is $\log(\det(\mathbf{P}_r))$ that represents the pose uncertainty. $\mathbf{P}_r$ is the covariance matrix of robot pose. Robot remains stationary from time 1 to time 200, moves one step at time 200 and then remains stationary.

5 Consistency Analysis

As seen in the previous section, R-EKF SLAM algorithm meets the expectation that observing new landmarks does not help in reducing the robot pose uncertainty [4][8], while T-EKF contradicts this. This section further analyzes the behavior of R-EKF and T-EKF in the context of consistency.

5.1 Unobservability of SLAM formulation

This subsection first reviews the observability of SLAM formulation (1)–(3), which is strongly related to the consistency issues of various SLAM estimation algorithms. The earliest concept of observability for nonlinear systems is proposed in [22]. From the viewpoint of nonlinear systems, SLAM formulation is not locally observable [22], as understood in [12][23]. Intuitively, it is impossible to globally localize a robot and the landmarks by only using the relative measurements (odometry and observations). As the motion model and observation model in SLAM formulation involve noises, the observability property of SLAM formulation in terms of stochastic systems is discussed in the following.

Definition 1 *For SLAM problem formulation (1)–(3), a stochastic rigid body transformation $\mathscr{T}_{\mathbf{g}}$ is*

$$\begin{aligned}\mathscr{T}_{\mathbf{g}}(\mathbf{X}) = (&\exp(\Theta_1)\bar{\mathbf{R}}\mathbf{R}, \exp(\Theta_1)\bar{\mathbf{R}}\mathbf{p} + \bar{\mathbf{T}} + \Theta_2, \\ &\exp(\Theta_1)\bar{\mathbf{R}}\mathbf{f}^1 + \bar{\mathbf{T}} + \Theta_2, \cdots, \exp(\Theta_1)\bar{\mathbf{R}}\mathbf{f}^N + \bar{\mathbf{T}} + \Theta_2),\end{aligned} \tag{11}$$

where $\mathbf{X}$ *is given in (1),* $\mathbf{g} = (\bar{\mathbf{R}}, \bar{\mathbf{T}}, \Theta)$, $\bar{\mathbf{R}} \in \mathbb{SO}(3), \bar{\mathbf{T}} \in \mathbb{R}^3$ *and* $\Theta = \begin{bmatrix} \Theta_1^\intercal & \Theta_2^\intercal \end{bmatrix}^\intercal \in \mathbb{R}^6$ *is white Gaussian noise with covariance* $\bar{\Sigma}$.

It can be easily verified that **any stochastic rigid body transformation $\mathscr{T}_{\mathbf{g}}$ does *not* affect the output (observations) of the system** (1)–(3), that is, for any initial conditions, $\mathbf{X}_0$ and $\mathbf{Y}_0 := \mathscr{T}_{\mathbf{g}}(\mathbf{X}_0)$, we have $h_n(\mathbf{X}_n, \xi_n) = h_n(\mathbf{Y}_n, \xi_n)$ for all $n \geq 0$, where $\mathbf{X}_k = f(\mathbf{X}_{k-1}, \mathbf{u}_{k-1}, \varepsilon_{k-1})$ and $\mathbf{Y}_k = f(\mathbf{Y}_{k-1}, \mathbf{u}_{k-1}, \varepsilon_{k-1})$ $(k = 1, \cdots, n-1)$. Therefore, SLAM formulation (1)–(3) is *unobservable* in terms of stochastic rigid body transformation. More specifically, SLAM formulation lacks enough information to verify whether a stochastic rigid body transformation is applied or not by only using the input and output of the system.

5.2 A consistency criterion for SLAM estimator

The unobservability in terms of stochastic rigid body transformation is a fundamental property of SLAM formulation. Therefore a consistent estimator should maintain this property. Thus the criterion that **the output of the estimator is invariant under any stochastic rigid body transformation** can be used to examine the consistency of any estimator used for solving SLAM. Violation of this criterion implies that the spurious information is generated such that it becomes possible to verify whether a stochastic rigid body transformation is applied or not by only using the input and output of the filter. This will clearly lead to an inconsistent estimate.

In the following, the consistency criterion for the general EKF framework estimator of SLAM formulation will be mathematically described. Let the estimate at time 0 be $(\hat{\mathbf{X}}_0, \mathbf{P}_0)$. After applying a stochastic rigid body transformation $\mathscr{T}_{\mathbf{g}}$ defined in Def. 1, via the first-order approximation, the estimate becomes $(\hat{\mathbf{Y}}_0, \mathbf{P}y_0)$, where $\hat{\mathbf{Y}}_0 = \mathscr{T}_{\mathbf{g}}(\hat{\mathbf{X}}_0)$ and $\mathbf{P}y_0 = \bar{\mathbf{Q}}_1 \mathbf{P}_0 \bar{\mathbf{Q}}_1^\intercal + \bar{\mathbf{Q}}_2 \bar{\Sigma} \bar{\mathbf{Q}}_2^\intercal$, where

$$\begin{aligned} \bar{\mathbf{Q}}_1 &= \left. \frac{\partial (\mathscr{T}_{\hat{\mathbf{g}}}(\hat{\mathbf{X}}_0 \oplus \mathbf{e}) \ominus \mathscr{T}_{\hat{\mathbf{g}}}(\hat{\mathbf{X}}_0))}{\partial \mathbf{e}} \right|_{\mathbf{e}=\mathbf{0}}, \\ \bar{\mathbf{Q}}_2 &= \left. \frac{\partial (\mathscr{T}_{\mathbf{g}}(\hat{\mathbf{X}}_0) \ominus \mathscr{T}_{\hat{\mathbf{g}}}(\hat{\mathbf{X}}_0))}{\partial \Theta} \right|_{\Theta=\mathbf{0}}, \end{aligned} \tag{12}$$

and $\hat{\mathbf{g}} = (\bar{\mathbf{R}}, \bar{\mathbf{T}}, \mathbf{0})$. Given the same odometry and observations from time 0 to n, $\hat{\mathbf{X}}_n$ and $\hat{\mathbf{Y}}_n$ are the *mean* estimate of the filter at time n from the estimate $(\hat{\mathbf{X}}_0, \mathbf{P}_0)$ and $(\hat{\mathbf{Y}}_0, \mathbf{P}y_0)$ at time 0, respectively. The invariance refers to $h_n(\hat{\mathbf{X}}_n, \mathbf{0}) = h_n(\hat{\mathbf{Y}}_n, \mathbf{0})$ for all $n \geq 0$ and $\mathscr{T}_{\mathbf{g}}$. The following theorem states that R-EKF conforms to this criterion.

Theorem 4 *R-EKF follows the proposed criterion, i.e., its output is invariant under any stochastic rigid body transformation.*

Proof 4 *See Appendix .4.*

As shown in the following, both R-EKF and T-EKF are invariant under deterministic rigid body transformation.

Theorem 5 *The outputs of T-EKF and R-EKF for SLAM formulation are both invariant under deterministic rigid body transformation.*

Proof 5 *See Appendix .5.*

In Section 6.1 we further illustrate the two theorems above using examples.

As it has been demonstrated that T-EKF can produce inconsistent estimate, we argue that the invariance under deterministic rigid body transformation is not adequate to maintain consistency.

Remark 1 *In [9] [12], a framework for designing an observability constrained filter is proposed. In summary, the core of the observability constrained filter is to calculate the Jacobian matrices* $\mathbf{F}_i$ *and* $\mathbf{H}_i$ *(*$i \geq 0$*) at some selected points (instead of the latest estimate) such that the right nullspace of the local observability matrix* $\mathbf{M}_{0:\infty} := \begin{bmatrix} \mathbf{H}_0^\intercal & (\mathbf{H}_1\mathbf{F}_0)^\intercal & (\mathbf{H}_2\mathbf{F}_1)^\intercal & \dots \end{bmatrix}^\intercal$ *is of rank 3, corresponding to 3 DOFs of the rigid body transformation in 2D. In this way, the output of the observability constraint filter is invariant under stochastic rigid body transformation.*

Remark 2 *In [14] the observability analysis is performed on the linearized error-state model from the viewpoint of information matrix. Our insight is in a different viewpoint that an estimator should mimic the unobservability (to stochastic rigid transformation) of the original system, which makes our analysis more intuitive and general.*

6 Simulation Results

In order to validate the theoretical results, we first give a simple example to illustrate the invariance property of R-EKF and T-EKF, which is discussed in Section 6.1. Then in Section 6.2, we perform Monte Carlo simulations and compare R-EKF to T-EKF and the First Estimates Jacobian EKF SLAM algorithm (FEJ-EKF)[8] under various conditions for performance evaluation.

6.1 Case study for invariance property

In this subsection, we illustrate the invariance property of R-EKF and T-EKF via a simple scenario. The trajectory $(\mathbf{R}_n, \mathbf{p}_n) = (Euler(\mathbf{a}_n), \mathbf{b}_n)$ of robot pose is used for generating observations and odometry, where $Euler(\cdot)$ refers to the Euler angles function, $\mathbf{a}_n = \begin{bmatrix} 0.1n & -0.2n & 0.3n \end{bmatrix}^\intercal$ and $\mathbf{b}_n = \begin{bmatrix} 5.1\cos(0.1n) & 4\sin(0.2n) & 2.1\sin(0.2n) \end{bmatrix}^\intercal$. The standard deviation of odometry noise and observation noise is set as 10% of the absolute value of the ground truth along each direction. We assume that robot can always see one landmark located at $\begin{bmatrix} -62 & -43 & 2.76 \end{bmatrix}^\intercal$.

For comparison, we set several different initial estimates $(\mathbf{X}_0, \mathbf{P}_0)$, $(\mathbf{Y}_0, \mathbf{P}_0)$ and $(\mathbf{X}_0, \bar{\mathbf{P}}_0)$, which correspond to non-transformation, deterministic rigid body transformation and stochastic rigid body transformation at time 0, respectively. The detailed

[8] The original FEJ-EKF firstly proposed in [9] is in 2D. For comparison, we extend FEJ-EKF into 3D. For details of T-EKF and FEJ-EKF, refer to Appendix 9 and 10.

values are $\mathbf{X}_0 = (Euler(\mathbf{a}_0), \mathbf{b}_0)$ and $\mathbf{Y}_0 = \mathscr{T}_{\mathbf{g}}(\mathbf{X}_0)$, $\mathbf{P}_0 = \mathbf{0}$, $\bar{\mathbf{P}}_0 = \frac{1}{2}\mathbf{I}_6$, $\mathbf{g} = (\exp(\mathbf{a}), \mathbf{b}, \mathbf{0})$, where $\mathbf{a} = \mathbf{b} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^\intercal$.

To briefly show the invariance properties of R-EKF and T-EKF, the output at time 100 is presented in Table 1. One can see that although the output of T-EKF is invariant under deterministic rigid body transformation ($\hat{h}_{100}|_{(\mathbf{X}_0,\mathbf{P}_0)} = \hat{h}_{100}|_{(\mathbf{Y}_0,\mathbf{P}_0)}$), but it does not hold the invariant property under stochastic rigid body transformation ($\hat{h}_{100}|_{(\mathbf{X}_0,\mathbf{P}_0)} \neq \hat{h}_{100}|_{(\mathbf{X}_0,\bar{\mathbf{P}}_0)}$). R-EKF indeed has the invariant property under deterministic/stochastic rigid body transformation.

Table 1: **Output of Filters at time 100**[9]

R-EKF	$\hat{h}_{100,x}$	$\hat{h}_{100,y}$	$\hat{h}_{100,z}$
$(\mathbf{X}_0, \mathbf{P}_0)$	33.9730	-95.0159	94.9821
$(\mathbf{Y}_0, \mathbf{P}_0)$	33.9730	-95.0159	94.9821
$(\mathbf{X}_0, \bar{\mathbf{P}}_0)$	33.9730	-95.0159	94.9821
T-EKF	$\hat{h}_{100,x}$	$\hat{h}_{100,y}$	$\hat{h}_{100,z}$
$(\mathbf{X}_0, \mathbf{P}_0)$	33.9754	-95.0439	94.9554
$(\mathbf{Y}_0, \mathbf{P}_0)$	33.9754	-95.0439	94.9554
$(\mathbf{X}_0, \bar{\mathbf{P}}_0)$	32.9893	-93.9909	92.1173

6.2 Performance evaluation

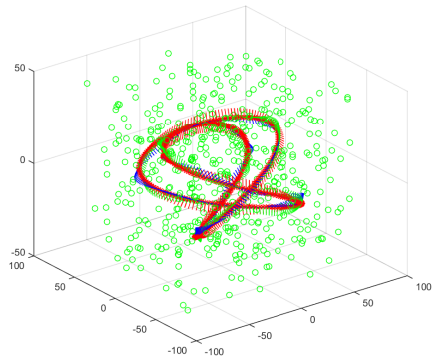

Figure 3: The robot trajectory and the landmark positions used in simulation. The green dots are landmarks and the colorful one represents the robot trajectory (R/G/B segment represent X/Y/Z axis of the orientation).

In this subsection, we investigate the performance of R-EKF, T-EKF and FEJ-EKF via numerical simulations under conditions of different noise level. The root mean square (RMS) error and the average normalized estimation error squared (NEES) are

[9]The notation $[\hat{h}_{100,x}, \hat{h}_{100,y}, \hat{h}_{100,z}]^\intercal$ are x,y,z components of the output $\hat{h}_{100}$.

used to evaluate accuracy and consistency. We consider that a robot moves in a trajectory shown in Fig. 3, which allows sufficient 6-DOFs motion. In this environment, 500 landmarks are randomly generated around the specified robot trajectory. The observations and odometry with noises are randomly generated by this specific trajectory and the simulated robot always observes the landmarks in the sensor range (less than 20m and 120°FoV). In every simulation, the number of steps is 400 (about 8 loops). For each condition (different noise level), 50 Monte Carlo simulations are performed.

The simulation results are summarized in Table 2 where σ_{od}[10] is the odometry noise level and σ_{ob} is the observation noise level. From Table 2, R-EKF is the best performing filter in terms of accuracy (RMS) and consistency (NEES) while FEJ-EKF also significantly outperforms T-EKF.

It is not difficult to understand the better performance of R-EKF relative to T-EKF due to the invariance property. In fact, FEJ-EKF also has the invariance property under stochastic rigid body transformation. However, the Jacobian matrices in FEJ-EKF are not evaluated at the latest estimate and hence some loss of information cannot be fully avoided. In comparison, R-EKF can always safely employ the latest estimate in Jacobians, which is a reason why R-EKF outperforms FEJ-EKF, especially in the cases of high noise level.

Table 2: **Performance Evaluation**

$\sigma_{od} = 5\%$, $\sigma_{ob} = 5\%$	R-EKF	FEJ-EKF	T-EKF
RMS of position(m)	**0.89412**	0.91898	1.15868
RMS of orientation(rad)	**0.02007**	0.02084	0.02517
NEES of orientation	**1.06112**	1.16085	2.13923
NEES of pose	**1.92385**	2.65565	4.24515
$\sigma_{od} = 10\%$, $\sigma_{ob} = 5\%$	R-EKF	FEJ-EKF	T-EKF
RMS of position(m)	**1.18966**	1.47779	1.90292
RMS of orientation(rad)	**0.02575**	0.03155	0.03821
NEES of orientation	**1.03224**	1.82509	4.67173
NEES of pose	**2.29364**	2.95015	4.41364
$\sigma_{od} = 15\%$, $\sigma_{ob} = 10\%$	R-EKF	FEJ-EKF	T-EKF
RMS of position(m)	**2.72075**	3.49021	5.09975
RMS of orientation(rad)	**0.05008**	0.06879	0.10301
NEES of orientation	**1.39223**	3.73554	16.49889
NEES of pose	**5.97046**	8.51823	13.89302
$\sigma_{od} = 20\%$, $\sigma_{ob} = 10\%$	R-EKF	FEJ-EKF	T-EKF
RMS of position(m)	**3.31427**	5.26981	7.69863
RMS of orientation(rad)	**0.05992**	0.11357	0.15742
NEES of orientation	**1.59500**	10.88764	33.25359
NEES of pose	**7.67372**	23.41042	47.07094

[10]For example, $\sigma_{od} = 5\%$ means the standard deviation of odometry noise is 5% of the absolute value of the ground truth along each direction.

7 Conclusion

In this work, the convergence properties and consistency of a Lie group EKF based SLAM algorithm are analyzed. Based on the general EKF framework, we introduce a Lie group EKF based SLAM algorithm (R-EKF). For convergence, several theorems with proofs are provided for two fundamental cases. For consistency, we propose a criterion for the general EKF framework based filter, i.e., the output is invariant under stochastic rigid body transformation. It is proved that R-EKF naturally satisfies this criterion but the traditional EKF SLAM (T-EKF) does not. In addition, we prove that T-EKF holds the invariant property under deterministic rigid body transformation but has inconsistency issues.

Although optimization based SLAM algorithms are becoming popular recently due to the high quality performance and the efficiency of the modern sparse solvers [6], they may not always be the best to be used in practice. Thus, the Lie group based EKF SLAM algorithm introduced in this paper not only provides insights into the theoretical understanding on the filter based SLAM algorithm convergence and consistency, but also serves as an alternative in the scenarios when the optimization based SLAM algorithm is not the best choice.

Future work includes extensively comparing the performance of R-EKF SLAM algorithm with optimization based SLAM algorithms to identify situations under which R-EKF SLAM is sufficient, as well as extending the R-EKF SLAM to the case of SLAM using IMU-vision integration.

.1 Lie Group $\mathscr{G}(N)$

The notation $\mathscr{G}(N)$ is a Lie group, defined as

$$\mathbf{G} = \{\left(\mathbf{R}, \mathbf{p}, \mathbf{f}^1, \cdots, \mathbf{f}^N\right) | \mathbf{R} \in \mathbb{SO}(3), \mathbf{p} \text{ and } \mathbf{f}^i \in \mathbb{R}^3\}. \tag{13}$$

The associated multiplication operation of $\mathscr{G}(N)$ is

$$\begin{aligned} &\mathbf{X}_1\mathbf{X}_2 \in \mathscr{G}(N) \\ &= \left(\mathbf{R}_1\mathbf{R}_2, \mathbf{R}_1\mathbf{p}_2 + \mathbf{p}_1, \mathbf{R}_1\mathbf{f}_2^1 + \mathbf{f}_1^1, \cdots, \mathbf{R}_1\mathbf{f}_2^N + \mathbf{f}_1^N\right), \end{aligned} \tag{14}$$

where $\mathbf{X}_i = \left(\mathbf{R}_i, \mathbf{p}_i, \mathbf{f}_i^1, \cdots, \mathbf{f}_i^N\right) \in \mathscr{G}(N)$ for $i = 1, 2$. The associated Lie algebra of $\mathscr{G}(N)$ is homomorphic to $\mathbb{R}^{3N+6}$. In addition, the inverse of $\mathbf{X} \in \mathscr{G}(N)$ is

$$\mathbf{X}^{-1} = (\mathbf{R}^\intercal, -\mathbf{R}^\intercal\mathbf{p}, -\mathbf{R}^\intercal\mathbf{f}^1, \cdots, -\mathbf{R}^\intercal\mathbf{f}^N) \in \mathscr{G}(N) \tag{15}$$

The exponential mapping exp is represented as

$$\begin{aligned} &\exp(\mathbf{e}) \in \mathscr{G}(N) \\ &= \left(\exp(\mathbf{e}_\theta), \mathrm{J}_r(-\mathbf{e}_\theta)\mathbf{e}_p, \mathrm{J}_r(-\mathbf{e}_\theta)\mathbf{e}^1, \cdots, \mathrm{J}_r(-\mathbf{e}_\theta)\mathbf{e}^N\right) \end{aligned} \tag{16}$$

for $\mathbf{e} = \begin{bmatrix} \mathbf{e}_\theta^\intercal & \mathbf{e}_p^\intercal & (\mathbf{e}^1)^\intercal & \cdots & (\mathbf{e}^N)^\intercal \end{bmatrix}^\intercal \in \mathbb{R}^{3N+6}$, where $\mathbf{e}_\theta$, $\mathbf{e}_p$ and $\mathbf{e}^i \in \mathbb{R}^3$ ($i = 1, \cdots, N$), the notation exp in the right side of (16) and the mapping J_r are given:

$$\exp(\mathbf{y}) = \mathbf{I}_3 + \frac{\sin(\|\mathbf{y}\|)}{\|\mathbf{y}\|}\mathrm{S}(\mathbf{y}) + \frac{1 - \cos(\|\mathbf{y}\|)}{\|\mathbf{y}\|^2}\mathrm{S}^2(\mathbf{y}) \tag{17}$$

$$\mathrm{J}_r(\mathbf{y}) = \mathbf{I}_3 - \frac{1-\cos(\|\mathbf{y}\|)}{\|\mathbf{y}\|^2}\mathrm{S}(\mathbf{y}) + \frac{\|\mathbf{y}\| - \sin(\|\mathbf{y}\|)}{\|\mathbf{y}\|^3}\mathrm{S}^2(\mathbf{y}) \tag{18}$$

for $\mathbf{y} \in \mathbb{R}^3$. The adjoint $\mathrm{ad}_{\mathbf{X}}$ for $\mathbf{X} = \left(\mathbf{R}, \mathbf{p}, \mathbf{f}^1, \cdots, \mathbf{f}^N\right) \in \mathscr{G}(N)$, is computed as

$$\mathrm{ad}_{\mathbf{X}} = \begin{bmatrix} \mathbf{R} & \mathbf{0}_{3,3} & \cdots & \cdots & \mathbf{0}_{3,3} \\ \mathrm{S}(\mathbf{p})\mathbf{R} & \mathbf{R} & \ddots & & \vdots \\ \mathrm{S}(\mathbf{f}^1)\mathbf{R} & \mathbf{0}_{3,3} & \mathbf{R} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \mathbf{0}_{3,3} \\ \mathrm{S}(\mathbf{f}^N)\mathbf{R} & \mathbf{0}_{3,3} & \cdots & \mathbf{0}_{3,3} & \mathbf{R} \end{bmatrix} \tag{19}$$

.2 Proof of Theorem 2

In the following, we use mathematical induction to prove this theorem. Note that at the beginning the estimate is $(\hat{\mathbf{X}}, \mathbf{P})$ where $\hat{\mathbf{X}} = \left(\hat{\mathbf{R}}, \hat{\mathbf{p}}, \hat{\mathbf{f}}^1, \cdots, \hat{\mathbf{f}}^N\right)$. After the first observation, the mean estimate of state and covariance matrix are augmented as below via the method shown in Alg. 2: $\hat{\mathbf{X}}_1 = \left(\hat{\mathbf{R}}, \hat{\mathbf{p}}, \hat{\mathbf{f}}^1, \cdots, \hat{\mathbf{f}}^N, \hat{\mathbf{f}}^{N+1}\right)$ and $\mathbf{P}_1 = \begin{bmatrix} \mathbf{P} & \mathbf{L} \\ \mathbf{L}^\intercal & \hat{\mathbf{R}}\Psi\hat{\mathbf{R}}^\intercal + \mathbf{W} \end{bmatrix}$. Obviously, after one observation, the mean estimate of robot pose and the previous "landmarks" does not change and the covariance matrix follows the proposed form. We now assume that after k times observations, the estimate becomes $\hat{\mathbf{X}}_k = \left(\hat{\mathbf{R}}, \hat{\mathbf{p}}, \hat{\mathbf{f}}^1, \cdots, \hat{\mathbf{f}}^N, \hat{\mathbf{f}}_k^{N+1}\right)$ and $\mathbf{P}_k = \begin{bmatrix} \mathbf{P} & \mathbf{L} \\ \mathbf{L}^\intercal & \frac{\hat{\mathbf{R}}\Psi\hat{\mathbf{R}}^\intercal}{k} + \mathbf{W} \end{bmatrix}$. Now we discuss the case after k times observations of next propagation and update. Because the robot is always perfectly stationary, after propagation at time k, the mean estimate is $\hat{\mathbf{X}}_{k+1|k} = \hat{\mathbf{X}}_k$ and covariance matrix becomes $\mathbf{P}_{k+1|k} = \mathbf{P}_k$. According to Alg. 1, we have $\mathbf{S} = \mathbf{H}\mathbf{P}_{k+1|k}\mathbf{H}^\intercal + \Psi = \frac{k+1}{k}\Psi$ and

$$\mathbf{K} = \mathbf{P}_{k+1|k}\mathbf{H}^\intercal\mathbf{S}^{-1} = \begin{bmatrix} \mathbf{0}_{3,(3N+6)} & -\frac{1}{k+1}\hat{\mathbf{R}}^\intercal \end{bmatrix}^\intercal, \tag{20}$$

where $\mathbf{H} = \begin{bmatrix} \mathbf{0}_{3,3} & \hat{\mathbf{R}}^\intercal & \mathbf{0}_{3,3N} & -\hat{\mathbf{R}}^\intercal \end{bmatrix}$. Then it is easy to see that all elements from the vector $\mathbf{K}\mathbf{y}$ are zero except the last 3 elements, and hence the estimate of robot pose and the old landmarks after $k+1$ times observations are **the same** as that in the time step k. The covariance matrix at time $k+1$ is $\mathbf{P}_{k+1} = (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}_{k+1|k} = \begin{bmatrix} \mathbf{P} & \mathbf{L} \\ \mathbf{L}^\intercal & \frac{\hat{\mathbf{R}}\Psi\hat{\mathbf{R}}^\intercal}{k+1} + \mathbf{W} \end{bmatrix}$. When k converges to infinity, we have (7).

.3 Proof of Theorem 3

After l times observations at point B, the information matrix Ω_l^B (the inverse of $\mathbf{P}_l^B$) becomes $\Omega_l^B = \Omega_0^B + \sum_{j=1}^{l}\mathbf{H}_j^\intercal\bar{\Psi}^{-1}\mathbf{H}_j$, where $\mathbf{H}_j$ is obtained by stacking all matrices $\mathbf{H}_j^i = \hat{\mathbf{R}}_j^\intercal\mathbf{H}^i$ $(i = 1, \cdots, m)$, and $\hat{\mathbf{R}}_j$ is the estimated orientation after j times observations at point B. Note that $\bar{\Psi}$ is a spheric matrix, we have $\mathbf{H}_j^\intercal\bar{\Psi}^{-1}\mathbf{H}_j = \mathbf{H}^\intercal\bar{\Psi}^{-1}\mathbf{H}$ $(j = 1, \cdots, l)$. Therefore, the information matrix is $\Omega_l^B = \Omega_0^B + l\mathbf{H}^\intercal\bar{\Psi}^{-1}\mathbf{H}$. Via the

matrix inversion lemma in [1], the covariance matrix after l observations at point B is

$$\mathbf{P}_B^l = (\Omega_l^B)^{-1} = \mathbf{P}_B^0 - \mathbf{P}_B^0\mathbf{H}^\intercal(\frac{\bar{\Psi}}{l} + \mathbf{H}\mathbf{P}_B^0\mathbf{H}^\intercal)^{-1}\mathbf{H}\mathbf{P}_B^0. \tag{21}$$

By using result in Theorem 1 and the Jacobian matrices in (4), we have

$$\mathbf{P}_B^0 = \mathbf{P}_A^\infty + \Delta\mathbf{P}, \tag{22}$$

where $\mathbf{P}_A^\infty$ (given in (8)) is the covariance matrix before moving to the point B, $\Delta\mathbf{P} = \mathrm{ad}_{\hat{\mathbf{X}}_A}\mathbf{E}\tilde{\Phi}\mathbf{E}^\intercal\mathrm{ad}_{\hat{\mathbf{X}}_A}^\intercal$ can be regarded as the incremental uncertainty caused by the odometry noise, and $\tilde{\Phi} = \mathbf{B}\Phi\mathbf{B}^\intercal$ is a positive definite matrix. Note that $\mathbf{H}\mathbf{P}_A^\infty = \mathbf{0}$, we substitute (22) into (21):

$$\begin{aligned}
\mathbf{P}_B^l &= \mathbf{P}_A^\infty + \Delta\mathbf{P} - \Delta\mathbf{P}\mathbf{H}^\intercal(\frac{\bar{\Psi}}{l} + \mathbf{H}\Delta\mathbf{P}\mathbf{H}^\intercal)^{-1}\mathbf{H}\Delta\mathbf{P} \\
&= \mathbf{P}_A^\infty + \mathrm{ad}_{\hat{\mathbf{X}}_A}\mathbf{E}(\tilde{\Phi} - \tilde{\Phi}\tilde{\mathbf{H}}^\intercal(\frac{\bar{\Psi}}{l} + \tilde{\mathbf{H}}\tilde{\Phi}\tilde{\mathbf{H}}^\intercal)^{-1}\tilde{\mathbf{H}}\tilde{\Phi})\mathbf{E}^\intercal\mathrm{ad}_{\hat{\mathbf{X}}_A}^\intercal \\
&= \mathbf{P}_A^\infty + \mathrm{ad}_{\hat{\mathbf{X}}_A}\mathbf{E}(\tilde{\Phi}^{-1} + l\tilde{\mathbf{H}}^\intercal\bar{\Psi}^{-1}\tilde{\mathbf{H}})^{-1}\mathbf{E}^\intercal\mathrm{ad}_{\hat{\mathbf{X}}_A}^\intercal \\
&= \mathbf{P}_A^\infty + \bar{\mathbf{P}}_B^l.
\end{aligned} \tag{23}$$

Furthermore, $\tilde{\mathbf{H}}^\intercal\bar{\Psi}^{-1}\tilde{\mathbf{H}} = \begin{bmatrix} \mathbf{S}_1 & \mathbf{S}_2 \\ \mathbf{S}_2^\intercal & m\Psi^{-1} \end{bmatrix}$ where $\mathbf{S}_1 = \sum_{i=1}^m \mathrm{S}^\intercal(\tilde{\mathbf{f}}_i)\Psi^{-1}\mathrm{S}(\tilde{\mathbf{f}}_i)$, $\mathbf{S}_2 = (\sum_{i=1}^m \mathrm{S}(\tilde{\mathbf{f}}_i))^\intercal\Psi^{-1}$ and $\tilde{\mathbf{f}}_i = \hat{\mathbf{R}}^\intercal(\hat{\mathbf{p}} - \hat{\mathbf{f}}_i)$ $(i = 1, \cdots, m)$. Generally speaking, $\tilde{\mathbf{H}}^\intercal\Psi^{-1}\tilde{\mathbf{H}}$ is full rank when $m > 3$ and there are three landmarks that are non-coplanar with the robot position. Under this condition, it is easy to see that $\mathbf{P}_B^l \to \mathbf{P}_A^\infty$ when $l \to \infty$.

.4 Proof of Theorem 4

In Therem 5, we will prove that R-EKF has invariant property under deterministic rigid body transformation. Here we only need to prove the invariant property under stochastic rigid body transformation $\mathscr{T}_\mathbf{g}$ $(\mathbf{g} = (\mathbf{I}_3, \mathbf{0}, \Theta))$ for all $\bar{\Sigma}$ where $\bar{\Sigma}$ is the covariance matrix of noise Θ. Consider the estimate at time 0 is $(\hat{\mathbf{X}}_0, \mathbf{P}_0)$ in R-EKF. If the stochastic rigid body transformation $\mathscr{T}_\mathbf{g}$ is applied, the estimate becomes $(\hat{\mathbf{X}}_0, \mathbf{P}_0 + \Delta\mathbf{P})$ where $\Delta\mathbf{P} = \mathbf{C}\bar{\Sigma}\mathbf{C}^\intercal$ and

$$\mathbf{C} = \frac{\partial(\mathscr{T}_\mathbf{g}(\hat{\mathbf{X}}_0) \ominus (\hat{\mathbf{X}}_0))}{\partial\Theta}|_\mathbf{0} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3,3} \\ \mathbf{0}_{3,3} & \mathbf{I}_3 \\ \vdots & \vdots \\ \mathbf{0}_{3,3} & \mathbf{I}_3 \end{bmatrix}. \tag{24}$$

After propagation, the estimate becomes $(\hat{\mathbf{X}}_{1|0}, \mathbf{P}_{1|0} + \Delta\mathbf{P})$ due to $\mathbf{F}_n = \mathbf{I}$ given in (4). Note that $\mathbf{H}_1\Delta\mathbf{P} = \mathbf{0}$, it is easy to get the posterior estimate $(\hat{\mathbf{X}}_1, \mathbf{P}_1 + \Delta\mathbf{P})$. By mathematical induction, we can conclude that the output of R-EKF is invariant under stochastic rigid body transformation.

If the optional step is turned on, the posterior estimation becomes $(\hat{\mathbf{X}}_1, \mathbf{J}(\mathbf{P}_1 + \Delta\mathbf{P})\mathbf{J}^\intercal)$. At the next step, the invariance property cannot be guaranteed.

.5 Proof of Theorem 5

Assume the estimate at time 0 is $(\hat{\mathbf{X}}_0, \mathbf{P}_0)$ in terms of the general EKF framework. After one step propagation via the odometry $\mathbf{u}_0$, the estimate becomes $(\hat{\mathbf{X}}_{1|0}, \mathbf{P}_{1|0})$. Then after obtaining observations $\mathbf{z}_1$, the estimate becomes $(\mathbf{X}_1, \mathbf{P}_1)$. On the other hand, in T-EKF and R-EKF, there exists a matrix $\mathbf{Q}_{\mathscr{T}}$ for any rigid body transformation $\mathscr{T}$ such that

$$\mathscr{T}(\mathbf{X} \oplus \mathbf{Q}_{\mathscr{T}}^{-1}\mathbf{e}) = \mathscr{T}(\mathbf{X}) \oplus \mathbf{e} \quad \forall\, \mathbf{X}. \tag{25}$$

Remark 3 *When* $\mathscr{T} = (\bar{\mathbf{R}}, \bar{\mathbf{T}}, \mathbf{0})$, $\mathbf{Q}_{\mathscr{T}} = ad_{\mathbf{U}}$ *in R-EKF, where* $\mathbf{U} = (\bar{\mathbf{R}}, \bar{\mathbf{T}}, \mathbf{0}, \cdots, \mathbf{0}) \in \mathscr{G}(N)$. *In T-EKF,* $\mathbf{Q}_{\mathscr{T}} = \begin{bmatrix} \bar{\mathbf{R}} & \mathbf{0}_{3,3N+3} \\ \mathbf{0}_{3N+3,3} & \mathbf{I}_{3N+3} \end{bmatrix}$.

Therefore, if a deterministic rigid body transformation $\mathscr{T}$ is applied at time 0, the estimate becomes $(\hat{\mathbf{Y}}_0, \mathbf{P}y_0)$, where $\hat{\mathbf{Y}}_0 = \mathscr{T}(\hat{\mathbf{X}}_0)$ and $\mathbf{P}y_0 = \mathbf{Q}_{\mathscr{T}}\mathbf{P}_0\mathbf{Q}_{\mathscr{T}}^{\mathsf{T}}$. Now we calculate the new Jacobians $\mathbf{F}y_0$ and $\mathbf{G}y_0$ in propagation

$$\begin{aligned}
\mathbf{F}y_0 =& \left.\frac{\partial(f(\hat{\mathbf{Y}}_0 \oplus \mathbf{e}, \mathbf{u}_0, \mathbf{0}) \ominus f(\hat{\mathbf{Y}}_0, \mathbf{u}_0, \mathbf{0}))}{\partial \mathbf{e}}\right|_{\mathbf{0}} \\
=& \left.\frac{\partial(f(\mathscr{T}(\hat{\mathbf{X}}_0) \oplus \mathbf{e}, \mathbf{u}_0, \mathbf{0}) \ominus f(\mathscr{T}(\hat{\mathbf{X}}_0), \mathbf{u}_0, \mathbf{0}))}{\partial \mathbf{e}}\right|_{\mathbf{0}} \\
\overset{(25)}{=}& \left.\frac{\partial(f(\mathscr{T}(\hat{\mathbf{X}}_0 \oplus \mathbf{Q}_{\mathscr{T}}^{-1}\mathbf{e}), \mathbf{u}_0, \mathbf{0}) \ominus f(\mathscr{T}(\hat{\mathbf{X}}_0), \mathbf{u}_0, \mathbf{0}))}{\partial \mathbf{e}}\right|_{\mathbf{0}} \\
=& \left.\frac{\partial(\mathscr{T}(f(\hat{\mathbf{X}}_0 \oplus \mathbf{Q}_{\mathscr{T}}^{-1}\mathbf{e}), \mathbf{u}_0, \mathbf{0}) \ominus \mathscr{T}(f(\hat{\mathbf{X}}_0), \mathbf{u}_0, \mathbf{0}))}{\partial \mathbf{e}}\right|_{\mathbf{0}} \\
=& \mathbf{Q}_{\mathscr{T}}\mathbf{F}_0\mathbf{Q}_{\mathscr{T}}^{-1}.
\end{aligned} \tag{26}$$

Similarly, we have $\mathbf{G}y_0 = \mathbf{Q}_{\mathscr{T}}\mathbf{G}_0$. Hence, after one step propagation the estimate becomes $(\hat{\mathbf{Y}}_{1|0}, \mathbf{P}y_{1|0})$, where $\hat{\mathbf{Y}}_{1|0} = f(\hat{\mathbf{Y}}_0, \mathbf{u}_0, \mathbf{0}) = \mathscr{T}(\hat{\mathbf{X}}_{1|0})$ and $\mathbf{P}y_{1|0} = \mathbf{F}y_0\mathbf{P}y_0\mathbf{F}y_0^{\mathsf{T}} + \mathbf{G}y_0\Phi_0\mathbf{G}y_0^{\mathsf{T}} = \mathbf{Q}_{\mathscr{T}}\mathbf{P}_{1|0}\mathbf{Q}_{\mathscr{T}}^{\mathsf{T}}$. The new Jacobians in update becomes $\mathbf{H}y_1 = \mathbf{H}_1\mathbf{Q}_{\mathscr{T}}^{-1}$. Then it is easy to obtain $\mathbf{K}_y = \mathbf{Q}_{\mathscr{T}}\mathbf{K}$, resulting in $\hat{\mathbf{Y}}_1 = \hat{\mathbf{Y}}_{1|0} \oplus \mathbf{K}_y\mathbf{y} = \mathscr{T}(\hat{\mathbf{X}}_{1|0}) \oplus \mathbf{Q}_{\mathscr{T}}\mathbf{K}\mathbf{y} = \mathscr{T}(\hat{\mathbf{X}}_{1|0} \oplus \mathbf{K}\mathbf{y}) = \mathscr{T}(\hat{\mathbf{X}}_1)$. The covariance matrix after update becomes $\mathbf{P}y_1 = (\mathbf{I} - \mathbf{K}_y\mathbf{H}y_1)\mathbf{P}y_{1|0} = \mathbf{Q}_{\mathscr{T}}\mathbf{P}_1\mathbf{Q}_{\mathscr{T}}^{\mathsf{T}}$. In all, $\hat{\mathbf{Y}}_1 = \mathscr{T}(\hat{\mathbf{X}}_1)$ and $\mathbf{P}y_1 = \mathbf{Q}_{\mathscr{T}}\mathbf{P}_1\mathbf{Q}_{\mathscr{T}}^{\mathsf{T}}$. By mathematical induction, we can see the outputs of T-EKF and R-EKF are invariant under deterministic rigid body transformation.

.6 R-EKF Reformulation

The motion model can be formulated in Lie group $\mathscr{G}(N)$ as follows:

$$\mathbf{X}_{n+1} = f(\mathbf{X}_n, \mathbf{u}_n, \varepsilon_n) = \mathbf{X}_n\mathbf{U}_n(\varepsilon_n) \tag{27}$$

where $\mathbf{U}_n(\varepsilon_n) = (\exp(\mathbf{w}_n + \varepsilon_{\mathbf{w}}), \mathbf{v}_n + \varepsilon_{\mathbf{v}}, \mathbf{0}, \cdots, \mathbf{0}) \in \mathscr{G}(N)$.

.7 R-EKF: Uncertainty Representation

In R-EKF, the special addition is defined via the exponential for Lie group $\mathscr{G}(N)$ as

$$\mathbf{X} = \hat{\mathbf{X}} \oplus \mathbf{e} = \exp(\mathbf{e})\hat{\mathbf{X}}, \tag{28}$$

where $\mathbf{X} \in \mathscr{G}(N)$ is the actual state, $\hat{\mathbf{X}} \in \mathscr{G}(N)$ and $\mathbf{e} \in \mathbb{R}^{3N+6}$ is the white Gaussian noise. Therefore, the special subtraction $\ominus$ corresponding to $\oplus$ is

$$\mathbf{e} = \mathbf{X} \ominus \hat{\mathbf{X}} := (\exp(\mathbf{e})\hat{\mathbf{X}}) \ominus \hat{\mathbf{X}} = \log(\mathbf{X}\hat{\mathbf{X}}^{-1}) \tag{29}$$

where the inverse operation is given in (15) and the logarithm mapping log is defined as

$$\log(\mathbf{X}) = \begin{bmatrix} \mathbf{x}_\theta \\ \mathbf{J}_r^{-1}(-\mathbf{x}_\theta)\mathbf{p} \\ \mathbf{J}_r^{-1}(-\mathbf{x}_\theta)\mathbf{f}^1 \\ \vdots \\ \mathbf{J}_r^{-1}(-\mathbf{x}_\theta)\mathbf{f}^N \end{bmatrix} \tag{30}$$

where $\mathbf{x}_\theta = \log(\mathbf{R}) \in \mathbb{R}^3$, $\mathbf{J}_r$ is given in (18) and log (overloaded function) here represents the logarithm mapping for $\mathbb{SO}(3)$.

.8 The Derivation of Jacobian in R-EKF

Here we calculate the Jacobians in R-EKF.

$$\begin{aligned}
\mathbf{F}_n &= \frac{\partial(f(\hat{\mathbf{X}}_n \oplus \mathbf{e}, \mathbf{u}_n, \mathbf{0}) \ominus f(\hat{\mathbf{X}}_n, \mathbf{u}_n, \mathbf{0}))}{\partial \mathbf{e}}|_{\mathbf{e}=\mathbf{0}} \\
&= \frac{\partial((\hat{\mathbf{X}}_n \oplus \mathbf{e})\mathbf{U}_n(\mathbf{0})) \ominus (\hat{\mathbf{X}}_n\mathbf{U}_n(\mathbf{0}))}{\partial \mathbf{e}}|_{\mathbf{0}} \\
&= \frac{\partial(\exp(\mathbf{e})\hat{\mathbf{X}}_n\mathbf{U}_n(\mathbf{0})) \ominus (\hat{\mathbf{X}}_n\mathbf{U}_n(\mathbf{0}))}{\partial \mathbf{e}}|_{\mathbf{0}} \\
&= \frac{\partial \log((\exp(\mathbf{e})\hat{\mathbf{X}}_n\mathbf{U}_n(\mathbf{0}))(\hat{\mathbf{X}}_n\mathbf{U}_n(\mathbf{0}))^{-1})}{\partial \mathbf{e}}|_{\mathbf{0}} \\
&= \frac{\partial \log(\exp(\mathbf{e}))}{\partial \mathbf{e}}|_{\mathbf{0}} \\
&= \mathbf{I}_{3N+6}
\end{aligned} \tag{31}$$

$$
\begin{aligned}
\mathbf{G}_n &= \frac{\partial(f(\hat{\mathbf{X}}_n,\mathbf{u}_n,\boldsymbol{\varepsilon})\ominus f(\hat{\mathbf{X}}_n,\mathbf{u}_n,\mathbf{0}))}{\partial\boldsymbol{\varepsilon}}|_{\varepsilon=\mathbf{0}} \\
&= \frac{\partial((\hat{\mathbf{X}}_n\mathbf{U}_n(\boldsymbol{\varepsilon}))\ominus(\hat{\mathbf{X}}_n\mathbf{U}_n(\mathbf{0})))}{\partial\boldsymbol{\varepsilon}}|_{\varepsilon=\mathbf{0}} \\
&= \frac{\partial(\log(\hat{\mathbf{X}}_n\mathbf{U}_n(\boldsymbol{\varepsilon})\mathbf{U}_n^{-1}(\mathbf{0})\hat{\mathbf{X}}_n^{-1}))}{\partial\boldsymbol{\varepsilon}}|_{\varepsilon=\mathbf{0}} \\
&= \frac{\partial(\log(\hat{\mathbf{X}}_n\exp(\mathbf{B}_n\boldsymbol{\varepsilon}+O(\|\boldsymbol{\varepsilon}\|))\hat{\mathbf{X}}_n^{-1}))}{\partial\boldsymbol{\varepsilon}}|_{\varepsilon=\mathbf{0}} \\
&= \frac{\partial(\log(\exp(\mathrm{ad}_{\hat{\mathbf{X}}_n}\mathbf{B}_n\boldsymbol{\varepsilon})))}{\partial\boldsymbol{\varepsilon}}|_{\varepsilon=\mathbf{0}} \\
&= \mathrm{ad}_{\hat{\mathbf{X}}_n}\mathbf{B}_n
\end{aligned}
\tag{32}
$$

where the 4-th equality is obtained by $\mathbf{U}_n(\boldsymbol{\varepsilon})\mathbf{U}_n^{-1}(\mathbf{0}) = \exp(\mathbf{B}_n\boldsymbol{\varepsilon}+O(\|\boldsymbol{\varepsilon}\|))$ and the 5-th equability comes from the adjoint definition in Lie group.

Note that $\mathbf{H}_{n+1} = \frac{\partial h_{n+1}(\hat{\mathbf{X}}_{n+1|n}\oplus\mathbf{e},\mathbf{0})}{\partial\mathbf{e}}|_{\mathbf{e}=\mathbf{0}}$ and $\mathbf{h}_{n+1}$ is obtained by stacking all $\mathbf{h}_{n+1}^i$ ($i\in O_{i+1}$). Hence, $\mathbf{H}_{n+1}$ can be obtained by stacking all $\mathbf{H}_{n+1}^i = \frac{\partial h_{n+1}(\hat{\mathbf{X}}_{n+1|n}\oplus\mathbf{e},\mathbf{0})}{\partial\mathbf{e}}|_{\mathbf{e}=\mathbf{0}}$ for all $i\in O_{i+1}$, where

$$
\begin{aligned}
\mathbf{H}_{n+1}^i &= \frac{\partial h_{n+1}^i(\hat{\mathbf{X}}_{n+1|n}\oplus\mathbf{e},\mathbf{0})}{\partial\mathbf{e}}|_{\mathbf{e}=\mathbf{0}} \\
&= \frac{\partial(\exp(\mathbf{e}_\theta)\mathbf{R}_{n+1|n})^\mathsf{T}((\hat{\mathbf{p}}_{n+1|n}+\mathrm{J}_r(-\mathbf{e}_\theta)\mathbf{e}_p)-(\hat{\mathbf{f}}_{n+1|n}^i+\mathrm{J}_r(-\mathbf{e}_\theta)\mathbf{e}^i)))}{\partial\mathbf{e}}|_{\mathbf{e}=\mathbf{0}} \\
&= \begin{bmatrix} \mathbf{0}_{3,3} & \hat{\mathbf{R}}_{n+1|n}^\mathsf{T} & \mathbf{0}_{3,3(i-1)} & -\hat{\mathbf{R}}_{n+1|n}^\mathsf{T} & \mathbf{0}_{3,3(N-i)} \end{bmatrix}.
\end{aligned}
\tag{33}
$$

.9 T-EKF

The special addition $\oplus$ in T-EKF is defined as

$$
\mathbf{X} = \hat{\mathbf{X}}\oplus\mathbf{e} = (\exp(\mathbf{e}_\theta)\mathbf{R}, \mathbf{p}+\mathbf{e}_p, \mathbf{f}^1+\mathbf{e}^1, \cdots, \mathbf{f}^N+\mathbf{e}^N), \tag{34}
$$

where $\mathbf{e} = \begin{bmatrix} \mathbf{e}_\theta \\ \mathbf{e}_p \\ \mathbf{e}^1 \\ \vdots \\ \mathbf{e}^N \end{bmatrix} \in \mathbb{R}^{3N+6}$. The Jacobians in T-EKF are provided as the following:

$$
\mathbf{F}_n = \begin{bmatrix} \mathbf{I}_{3N} & \mathbf{0}_{3,3} & \mathbf{0}_{3,3N} \\ -\mathrm{S}(\hat{\mathbf{R}}_n\mathbf{v}_n) & \mathbf{I}_3 & \mathbf{0}_{3,3N} \\ \mathbf{0}_{3N,3} & \mathbf{0}_{3N,3} & \mathbf{I}_{3N} \end{bmatrix} \tag{35}
$$

$$
\mathbf{G}_n = \begin{bmatrix} -\hat{\mathbf{R}}_n\mathrm{J}_r(-\omega_n) & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3,3} & \hat{\mathbf{R}}_n \\ \mathbf{0}_{3N,3} & \mathbf{0}_{3N,3} \end{bmatrix} \tag{36}
$$

$$\mathbf{H}_{n+1}^{i} = \begin{bmatrix} \mathbf{T} & \hat{\mathbf{R}}_{n+1|n}^{\intercal} & \mathbf{0}_{3,(3i-3)} & -\hat{\mathbf{R}}_{n+1|n}^{\intercal} & \mathbf{0}_{3,(3N-3i)} \end{bmatrix} \tag{37}$$

where $\mathbf{T} = -\hat{\mathbf{R}}_{n+1|n}^{\intercal}\mathrm{S}(\hat{\mathbf{f}}_{n+1|n}^{i} - \hat{\mathbf{p}}_{n+1|n})$.

Algorithm 3: T-EKF SLAM: Landmark Initialization

Input: $\hat{\mathbf{X}}$, $\mathbf{P}$, $\mathbf{z}$;
Output: $\hat{\mathbf{X}}_{new}$, $\mathbf{P}_{new}$;
Process:
$\hat{\mathbf{f}}^{N+1} = \hat{\mathbf{p}} + \hat{\mathbf{R}}\mathbf{z} \in \mathbb{R}^3$
$\hat{\mathbf{X}}_{new} \leftarrow (\hat{\mathbf{X}}, \hat{\mathbf{f}}^{N+1}) \in \mathscr{G}(N+1)$;
$\mathbf{P}_{new} \leftarrow \begin{bmatrix} \mathbf{P} & \mathbf{P}\mathbf{M}_N \\ \mathbf{M}_N^{\intercal}\mathbf{P} & \hat{\mathbf{R}}\Psi\hat{\mathbf{R}}^{\intercal} + \mathbf{M}_N^{\intercal}\mathbf{P}\mathbf{M}_N \end{bmatrix}$,
where $\mathbf{M}_N := \begin{bmatrix} -\mathrm{S}(\hat{\mathbf{R}}\mathbf{z}) & \mathbf{I}_3 & \mathbf{0}_{3,3N} \end{bmatrix}^{\intercal}$.

.10 FEJ-EKF

The differences between FEJ-EKF and T-EKF are the Jacobians $\mathbf{F}_n$ and $\mathbf{H}_{n+1}^{i}$. In FEJ-EKF,

$$\mathbf{F}_n = \begin{bmatrix} \mathbf{I}_{3N} & \mathbf{0}_{3,3} & \mathbf{0}_{3,3N} \\ -\mathrm{S}(\mathbf{p}_{n+1|n} - \mathbf{p}_{n|n-1}) & \mathbf{I}_3 & \mathbf{0}_{3,3N} \\ \mathbf{0}_{3N,3} & \mathbf{0}_{3N,3} & \mathbf{I}_{3N} \end{bmatrix} \tag{38}$$

and

$$\mathbf{H}_{n+1}^{i} = \begin{bmatrix} \mathbf{T} & \hat{\mathbf{R}}_{n+1|n}^{\intercal} & \mathbf{0}_{3,(3i-3)} & -\hat{\mathbf{R}}_{n+1|n}^{\intercal} & \mathbf{0}_{3,(3N-3i)} \end{bmatrix} \tag{39}$$

where $\mathbf{T} = -\hat{\mathbf{R}}_{n+1|n}^{\intercal}\mathrm{S}(\hat{\mathbf{f}}_{\mathbf{TIME}(i)}^{i} - \hat{\mathbf{p}}_{n+1|n})$ and $\mathbf{TIME}(i)$ is the time when the robot firstly observes landmark i.

.11 Details of Simulation Results

The RMS and NEES indicators used in Section 6.2 are

$$\begin{aligned} \text{RMS of position} &= \frac{\sum_{n=1}^{T} \|\mathbf{p}_n - \hat{\mathbf{p}}_n\|}{T} \\ \text{RMS of orientation} &= \frac{\sum_{n=1}^{T} \arccos\left(\frac{\mathrm{tr}(\mathbf{R}_n\hat{\mathbf{R}}_n^{\intercal})-1}{2}\right)}{T} \\ \text{NEES of orientation} &= \frac{\sum_{n=1}^{T} \mathbf{e}_{\theta,n}^{\intercal}\mathbf{P}_{\theta,n}\mathbf{e}_{\theta,n}}{3T} \\ \text{NEES of pose} &= \frac{\sum_{n=1}^{T} \mathbf{e}_{r,n}^{\intercal}\mathbf{P}_{r,n}\mathbf{e}_{\theta,n}}{6T} \end{aligned} \tag{40}$$

where T is the number of simulated steps and $\mathbf{e}_{\theta,n} = \log(\mathbf{R}_n\hat{\mathbf{R}}_n^{\intercal})$ is the error vector between the estimated orientation and actual estimated orientation at time n, $\mathbf{P}_{\theta,n}$ represents the covariance matrix regarding orientation at time n and $\mathbf{P}_{r,n}$ represents the co-

variance matrix regarding robot pose at time n. In addition, $\mathbf{e}_{r,n} = \begin{bmatrix} \mathbf{e}_{r,n} \\ \mathbf{J}_r^{-1}(\mathbf{e}_{r,n})(\mathbf{p}_n - \hat{\mathbf{p}}_n) \end{bmatrix}$ in REKF and $\mathbf{e}_{r,n} = \begin{bmatrix} \mathbf{e}_{r,n} \\ \mathbf{p}_n - \hat{\mathbf{p}}_n \end{bmatrix}$ in T-EKF and FEJ-EKF. For more simulation results, see Fig. 4 and 5.

References

[1] S. Huang and G. Dissanayake, "Convergence and consistency analysis for extended kalman filter based slam," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 1036–1049, Oct 2007.

[2] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual–inertial odometry," *IEEE Transactions on Robotics*, vol. PP, no. 99, pp. 1–21, 2016.

[3] S. J. Julier and J. K. Uhlmann, "A counter example to the theory of simultaneous localization and map building," in *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, vol. 4, 2001, pp. 4238–4243 vol.4.

[4] T. Bailey, J. Nieto, J. Guivant, M. Stevens, and E. Nebot, "Consistency of the ekf-slam algorithm," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2006, pp. 3562–3568.

[5] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "isam2: Incremental smoothing and mapping with fluid relinearization and incremental variable reordering," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 3281–3288.

[6] R. Kmmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, May 2011, pp. 3607–3613.

[7] J. A. Castellanos, J. Neira, and J. D. Tardós, "Limits to the consistency of ekf-based slam," 2004.

[8] P. Lourenço, B. J. Guerreiro, P. Batista, P. Oliveira, and C. Silvestre, "Simultaneous localization and mapping for aerial vehicles: a 3-d sensor-based gas filter," *Autonomous Robots*, vol. 40, no. 5, pp. 881–902, 2016.

[9] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Analysis and improvement of the consistency of extended kalman filter based slam," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*, May 2008, pp. 473–479.

[10] J. Andrade-Cetto and A. Sanfeliu, "The effects of partial observability in slam," in *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 1, April 2004, pp. 397–402 Vol.1.

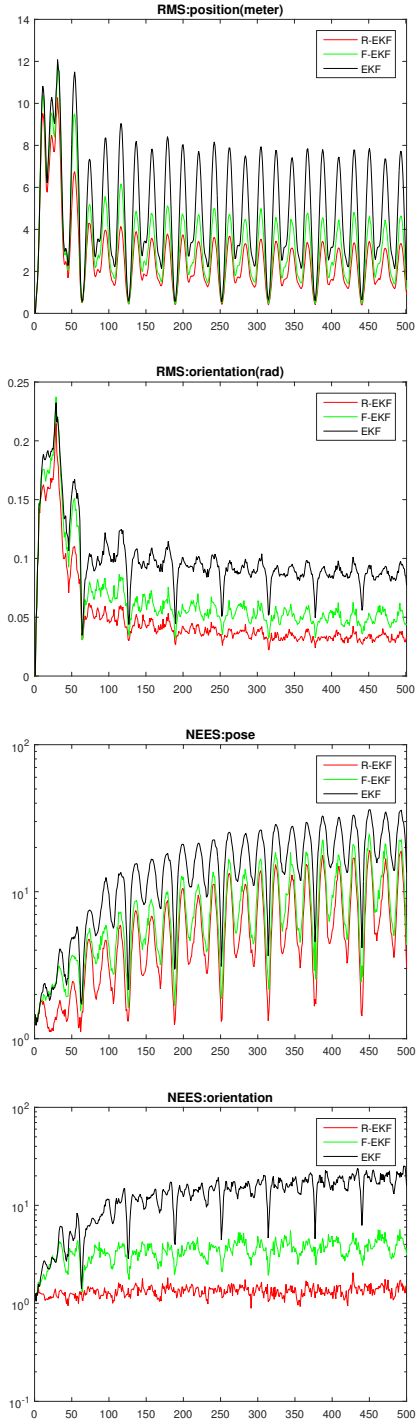

Figure 4: RMS and NEES of R-EKF, FEJ-EKF and T-EKF in the noise level $\sigma_{od} = 15\%$ and $\sigma_{ob} = 10\%$

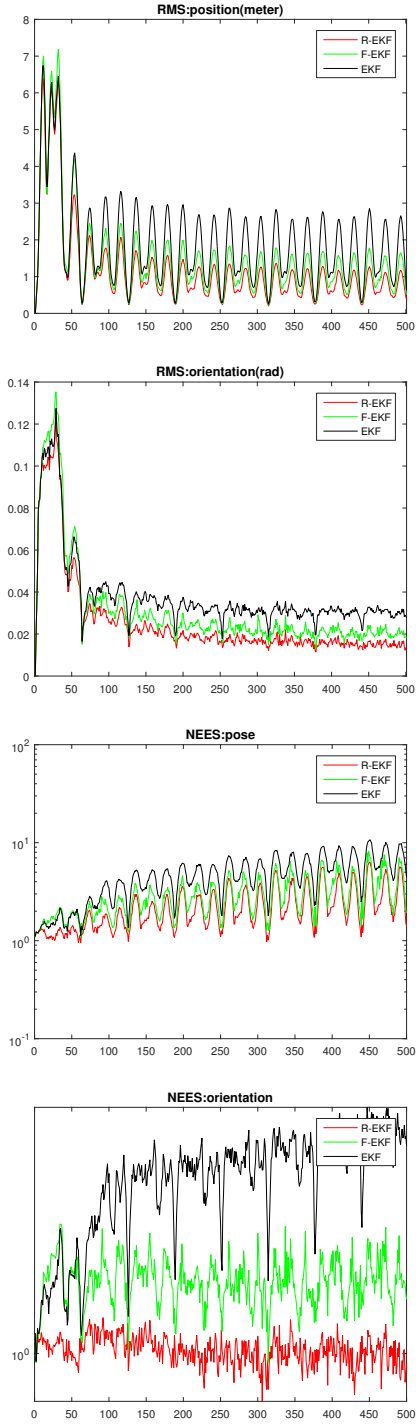

Figure 5: RMS and NEES of R-EKF, FEJ-EKF and T-EKF in the noise level $\sigma_{od} = 10\%$ and $\sigma_{ob} = 5\%$

[11] K. W. Lee, W. S. Wijesoma, and J. I. Guzman, "On the observability and observability analysis of slam," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2006, pp. 3569–3574.

[12] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based rules for designing consistent ekf slam estimators," *The International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, 2010.

[13] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Consistency analysis and improvement of vision-aided inertial navigation," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 158–176, Feb 2014.

[14] A. Barrau and S. Bonnabel, "An ekf-slam algorithm with consistency properties," *arXiv preprint arXiv:1510.06263*, 2015.

[15] J. Hu, D. Q. Zhang, H. Yu, and C. W. Chen, "High resolution free-view interpolation of planar structure," in *2014 IEEE International Conference on Multimedia and Expo (ICME)*, July 2014, pp. 1–6.

[16] T. D. Barfoot and P. T. Furgale, "Associating uncertainty with three-dimensional poses for use in estimation problems," *IEEE Transactions on Robotics*, vol. 30, no. 3, pp. 679–693, June 2014.

[17] R. Mahony, T. Hamel, and J. M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," *IEEE Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, June 2008.

[18] C. Hertzberg, R. Wagner, U. Frese, and L. Schrder, "Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds," *Information Fusion*, vol. 14, no. 1, pp. 57 – 77, 2013.

[19] G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, Jun 2001.

[20] A. Mourikis and S. Roumeliotis, "Analytical characterization of the accuracy of slam without absolute orientation measurements," in *Proceedings of Robotics: Science and Systems*, Philadelphia, USA, August 2006.

[21] G. Bourmaud, R. Mgret, A. Giremus, and Y. Berthoumieu, "Discrete extended kalman filter on lie groups," in *21st European Signal Processing Conference (EUSIPCO 2013)*, Sept 2013, pp. 1–5.

[22] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, vol. 22, no. 5, pp. 728–740, Oct 1977.

[23] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Camera-imu-based localization: Observability analysis and consistency improvement," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 182–201, 2014.