Joint Localization based on Split Covariance Intersection on Lie Group

Liang Li, Ming Yang Member, IEEE,

Abstract—This paper presents a pose fusion method that accounts for the possible correlations among measurements. The proposed method can handle data fusion problems whose uncertainty has both independent part and dependent part. Different from the existing methods, the uncertainties of the various states or measurements are modeled on the Lie algebra and projected to the manifold through the exponential map, which is more precise than that modeled in the vector space. The dealing of the correlation is based on the theory of covariance intersection, where the independent and dependent parts are split to yield a more consistent result. In this paper, we provide a novel method for correlated pose fusion algorithm on the manifold. Theoretical derivation and analysis are detailed first, and then the experimental results are presented to support the proposed theory. The main contributions are threefold: (1) We provide a theoretical foundation for the split covariance intersection filter performed on the manifold, where the uncertainty is associated on the Lie algebra. (2) The proposed method gives an explicit fusion formalism on SE(3) and SE(2), which covers the most use cases in the field of robotics. (3) We present a localization framework that can work both for single robot and multi-robots systems, where not only the fusion with possible correlation is derived on the manifold, the state evolution and relative pose computation are also performed on the manifold. Experimental results validate its advantage over state-of-the-art methods.

Index Terms—Pose fusion, localization, Lie group, covariance intersection

I. INTRODUCTION

ATA fusion is a key technique in the field of robotics in many tasks, such as Simultaneous Localization And Mapping (SLAM) [1]–[3], object tracking [4]–[6], and robot controlling [7]–[9]. The basic problem of data fusion is to obtain a fused estimate given several measurements of the same state. An important use case of the fusion algorithm is the pose estimate for robots, as we have various manners of measuring the robots' movement, *i.e.*, inertial measurement unit (IMU), wheel encoder, visual odometer, LiDAR-based localization, GPS, relative pose estimation among robots via communication. The problem of how to yield a more consistent and robust pose estimate from all the available sources is an interesting research topic for several decades [10]–[14]. One noteworthy fact is that there are possible correlations among the various types of measurements, which is hard to measure beforehand. Because the estimate by different measurements may be based on the redundant information across different sensors or algorithms. Another fact is that the robot naturally moves on the Lie group of SE(3) or SE(2), and linearizing it in vector space causes an error for the fusion, composition, and relative pose measurement process [15], [16]. Developing an algorithm that takes the correlation into

consideration and models robot movement and measurement uncertainty on the Lie group is still to be investigated.

The filtering-based method takes the robot kinematics and estimation from measurement into consideration concurrently, where the procedure of pose evolution and measurement update proceed alternatively. Kalman filter [17] and its variants [18]-[20] are the most commonly used algorithms for filteringbased pose fusion. Kalman filter assumes Gaussian noise in the state evolution and measurement update, where no correlation is considered. Particle filter [21]–[23] utilizes the Monte Carlo method to sample the posterior probability by forming the filtering as a maximum a posteriori (MAP) problem. The benefit of the particle filter is that it does not require Gaussian distribution of the noise or approximation of the nonlinearity by Taylor expansion. So, it is usually more accurate and robust than the Kalman filter when there is enough number of particles. While on the other hand, the computational efficiency of the particle filter is usually much lower than that of the Kalman filter. Neither of these two types of fusion methods can take the possible correlation among measurements into consideration. To associate the correlation, covariance intersection filter [24] was utilized in the task of robot pose fusion [25]–[27], especially for the multi-robot systems. The idea of splitting dependent and independent parts of the measurements and tackling them separately was introduced in [28]-[30], which is called split covariance intersection filter.

One drawback of the methods mentioned above is that the uncertainties of the state or measurements are modeled in the vector space. While in reality, the robots' movement is on the special Euclidean Lie group SE(3), approximate the pose and uncertainty in vector space induces error for estimation. To address this problem, some studies [15], [16] estimate the pose directly on the Lie group. In these works, the authors model the uncertainty of the robot pose estimation on Lie algebra, then project it onto the Lie group when computing the pose. Results therein show the advantage of uncertainty modeled on Lie algebra over that in the vector space. Compared with [15], the method in [16] takes correlation into consideration for pose composition, inverse, and relative pose computation. Nevertheless [16] assumes the correlation is known, which is not the case in reality. Besides, it did not deduce the fusion strategy but only composition, inverse, and relative pose computation, as the derivation of fusion multi-measurements is not straightforward like that of composition, inverse, and relative pose computation.

To fill the gap of correlated pose fusion on the Lie group, we propose the split covariance intersection filter on the Lie group in this paper. The uncertainty is split into dependent and independent parts. First, we derive the filtering strategy of split covariance intersection, where the uncertainty is modeled on Lie algebra. Then we give the explicit formalism of pose estimation algorithm for a multi-agent system for the partial observation case, where the propagation of uncertainty and state is incorporated. The organization of the rest of this paper is as follows. In Section II, related works towards pose fusion for robots are reviewed. Section III give some preliminaries for the derivation of the split covariance intersection filter on the Lie group, which includes the covariance intersection theory and knowledge of Lie group. Section IV presents the split covariance intersection filter on the Lie group and its theoretical analysis in detail, which is the core part of this paper. Experimental results and comparison with state-of-theart methods are presented in Section V. Finally, Section VI concludes the paper.

II. RELATED WORKS

Given two or more measurements of the same variable where we know the level of uncertainty for every measurement, the fusion algorithm aims to estimate a more consistent value of the state that utilizes all the measurements. It is not always the more measurements to be fused, the better the result will be. As the measurement with large uncertainty may draw the fused result farther from the ground truth, though its effect decreases as its uncertainty increases. If we assume the system is linear and the uncertainty is Gaussian, then the fusion problem can be solved by Kalman filter [31]. While in reality, the system is usually nonlinear, e.g., robot pose estimation is nonlinear as rotation incorporated. Extended Kalman filter (EKF) [32] approximates the nonlinearity by using the firstorder Taylor expansion. Unscented Kalman filter (UKF) [33] approximates the nonlinearity with unscented transform, which can approximate the nonlinearity with the second order. Thus, it is more accurate than EKF. Recently, cubature rule was used to approximate the nonlinear integrals, which yields the cubature Kalman filter (CKF) [34]. CKF can approximate the nonlinearity with the fifth order, which means it is more accurate than EKF and UKF. The Kalman filter family is based on the state-space model with the predictive update using the system model and the observation update using the measurement model. Usually, the measurement is not the state directly, and these two procedures proceed alternatively.

To avoid the assumption of Gaussian noise of system and measurement, particle filter [35] models the fusion system in a probabilistic manner. It utilizes the Monte Carlo algorithm to compute the posterior distributions. The objective is to maximize the posterior probability given the prior knowledge and the likelihood of the measurements. Because it does not rely on any linearization or Gaussian assumption, the particle filter is more generalized than the Kalman filter family. Especially when the noise is not Gaussian, particle filter can perform better than the Kalman filter. While particle filter has the sampling degeneracy problem as the iteration goes on, various resampling algorithms [36]–[39] have been proposed to deal with this problem. Besides, the computation complexity of the particle filter is much higher than the Kalman filter, especially when the number of particles need to be drawn is large.

Kalman filter and particle filter assume the measurements are independent, while in reality, especially the multi-agent systems, this assumption does not hold. To incorporate the scheme of correlation, authors in [40] proposed the covariance intersection (CI) algorithm. CI regards the measurements to be fused as totally correlated and adjusts the correlation through a parameter. When there are independent parts, results by CI may be pessimistic. The same authors proposed the split covariance intersection in [30], where the dependent and independent parts of the source estimates are tackled separately. Consistency of CI and Split Covariance Intersection (SCI) has been proved in [29], [30], [40]. Its advantage over Kalman filter has also been presented therein, especially for the decentralized localization of multi-agent systems, the benefit is remarkable. All the methods mentioned so far are operated in the vector space, *i.e.*, the uncertainty of states and measurements are small perturbation around the vectorized parameter (translation and rotation, three parameters in 2D case, and six parameters in 3D case). While for the robot pose estimation problem, the state and measurement are nonlinear and forms Lie group naturally. There are offset between the linearized vector space and the real state on Lie group when implementing the algorithms directly.

In [15], [41], the authors addressed this problem by modeling the uncertainty on Lie algebra and projecting it to Lie group by exponential map. The authors derived propagation and fusion case therein and applied it for pose estimation with a nonlinear camera model. But [15], [41] did not consider the correlation between different source estimates, *i.e.*, it assume all the states are independent. To incorporate the correlation, authors in [16] characterized the uncertainty of jointly distributed poses in the lie algebra. Besides propagation, they also derived the form of inverse pose given a single measurement, and the relative pose given two measurements originate from the same point. Results on simulation data and real-world SLAM datasets validate the advantage of that method. But the method in [16] assumed the measurements are totally correlated and did not consider the partially correlated case. Here the partially correlated means the noise consists of two parts; The first part is independent component, which is not correlated by any other noise; The second part is the dependent component, which is jointly corrupted with the noise of other measurements. In contrast, totally correlated case only has the dependent component. Extending it to partially correlated case is not straightforward. Besides, they did not derive the form of fusion case, which is more complicated than that of propagation, inverse, and relative pose computation.

Authors in [42]–[44] proposed the Invariant Extended Kalman Filter (IEKF), which extend the fusion on Lie group to the dynamic system. For IEKF, the state and error are defined on the Lie group, which makes it independent from the true state that needs to be estimated. It is proved in [44] that IEKF is stable under any trajectory compared to the standard EKF which can be only stable around the ground-truth trajectory. This means that its is convergence guaranteed with arbitary initialization. In [45], it is further pointed out

that IEKF can be extended to handle nonlinear constraints by geometric equalities other than that of the Lie group. In [46], [47], the authors prove that the observability of the IEKF is consistent with the nonlinear system therein. It means that the IEKF-based algorithm can avoid inconsistency caused by the linearization points in the standard EKF. IEKF achieves better performance than the standard EKF in many roboticsrelated tasks, e.g., SLAM [48], [49], attitude estimation [43], robot state estimation [47]. But the IEKF does not consider the underlying correlation between different observation sources, which makes the estimation over-confident under the circumstances that the correlation really exists. This is one of the motivations for this paper. It is not difficult to verify that IEKF is a special case of the proposed method when we do not consider the underlying correlation and takes the firstorder term as approximation in the Lie algebra. In [50], [51], the authors make a further extension which assumes that the observation is also on the Lie group. In [52], the unscented transform is used to approximate the system dynamics on the Lie group, the results therein show that UKF-based algorithm is more robust than IEKF when the noise become bigger. In [53], the authors extend the multistate constraint Kalman filter to the Lie group based on the idea of Invariant EKF, and then applied it to the visual inertial odometry system. While the correlation was still not considered in these literatures.

To the best of the authors' knowledge, there is no algorithm that gives the formalism of partially correlated pose fusion whose uncertainty is modeled on Lie algebra so far, which is the main motivation of this paper. In the theoretical level, we give the full pipeline and derivation of correlation pose fusion scheme on Lie group, which can be regarded as the extension of SCI to manifold. In the application level, we give the form of system evolution and update for partial observation case. The proposed method is validated through a multirobot localization experiment, comparison, and discussion with respect to the baseline, and state-of-the-art methods are also presented.

III. PRELIMINARIES

A. Split Covariance Intersection Filter

First, we give a brief review of Kalman fusion scheme. If we have multiple estimate sources with white Gaussian noise for the same state (we take two sources for instance in this paper, cases of more than two is easily to extend), estimate sources here refers to algorithms or hardware that provide the estimation of robot pose. the Kalman filter fuse these sources using the following criteria:

$$\hat{\boldsymbol{\Sigma}}^{-1} = \boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1}$$

$$\hat{\mathbf{p}} = \hat{\boldsymbol{\Sigma}}(\boldsymbol{\Sigma}_1^{-1}\mathbf{p}_1 + \boldsymbol{\Sigma}_2^{-1}\mathbf{p}_2)$$
(1)

where \mathbf{p}_1 and \mathbf{p}_2 are the estimate sources, Σ_1 and Σ_2 are their covariance respectively. $\hat{\mathbf{p}}$, $\hat{\boldsymbol{\Sigma}}$ are the fused estimate and covariance. The intuition behind Kalman filter can be interpreted as minimization of Mahalanobis distance, as we assume Gaussian noise of the estimate sources. But in many applications, \mathbf{p}_1 and \mathbf{p}_2 do not measure the same state. For discrete fusion case, we usually fuse the result of previous step with the system model, which we call the predictive update. Then we fuse the predictive result with the measurement from the sensors or algorithms, which we call the measurement update. These two procedures proceed alternatively in an iterative manner. Given a state space model:

$$\mathbf{x}_{k} = \mathbf{F}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k} + \mathbf{w}_{k}$$

$$\mathbf{y}_{k} = \mathbf{H}\mathbf{x}_{k} + \mathbf{v}_{k}$$
(2)

where \mathbf{x}_k is the state at time k, \mathbf{u}_k is the known control input, \mathbf{y}_k is the measurement. **F**, **B**, and **H** are the state transform matrix, input matrix, and measurement matrix. \mathbf{w}_k and \mathbf{v}_k are the white Gaussian noise, whose covariance are \mathbf{Q}_k and \mathbf{R}_k . Then the predictive update is:

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_{k}$$

$$\mathbf{P}_{k|k-1} = \mathbf{F}\mathbf{P}_{k-1|k-1}\mathbf{F}^{T} + \mathbf{Q}_{k}$$
(3)

The measurement update is:

$$\mathbf{K}_{k} = \mathbf{P}_{k|k-1} \mathbf{H}^{T} \left(\mathbf{H} \mathbf{P}_{k|k-1} \mathbf{H}^{T} + \mathbf{R}_{k} \right)^{-1}$$
$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_{k} (\mathbf{y}_{k} - \mathbf{H} \hat{\mathbf{x}}_{k|k-1})$$
$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_{k} \mathbf{H}) \mathbf{P}_{k|k-1}$$
(4)

where \mathbf{K}_k is called the Kalman gain.

As mentioned in Section I, Kalman filter does not consider possible correlation between the estimate sources p_1 and p_2 . In [40], the authors proposed CI to take correlation into consideration. The fusion strategy of CI can be denoted as:

$$\hat{\boldsymbol{\Sigma}}^{-1} = \left(\frac{\boldsymbol{\Sigma}_1}{\omega}\right)^{-1} + \left(\frac{\boldsymbol{\Sigma}_2}{1-\omega}\right)^{-1}$$
$$\hat{\mathbf{p}} = \hat{\boldsymbol{\Sigma}} \left[\left(\frac{\boldsymbol{\Sigma}_1}{\omega}\right)^{-1} \mathbf{p}_1 + \left(\frac{\boldsymbol{\Sigma}_2}{1-\omega}\right)^{-1} \mathbf{p}_2 \right]$$
(5)

where $\omega \in [0, 1]$ is a coefficient to measure correlation level between the estimate sources. CI assumes the estimate sources are totally correlated, which may yield too conservative result when there are independent parts of the sources. An improved version of CI called SCI was proposed in [30], the fusion strategy of SCI is:

$$\begin{split} \boldsymbol{\Sigma}_{1} &= \frac{\boldsymbol{\Sigma}_{1,d}}{\omega} + \boldsymbol{\Sigma}_{1,i}, \boldsymbol{\Sigma}_{2} = \frac{\boldsymbol{\Sigma}_{2,d}}{1-\omega} + \boldsymbol{\Sigma}_{2,i} \\ \hat{\boldsymbol{\Sigma}}^{-1} &= \boldsymbol{\Sigma}_{1}^{-1} + \boldsymbol{\Sigma}_{2}^{-1}, \hat{\mathbf{p}} = \hat{\boldsymbol{\Sigma}} (\boldsymbol{\Sigma}_{1}^{-1} \mathbf{p}_{1} + \boldsymbol{\Sigma}_{2}^{-1} \mathbf{p}_{2}) \\ \hat{\boldsymbol{\Sigma}}_{i} &= \hat{\boldsymbol{\Sigma}} (\boldsymbol{\Sigma}_{1}^{-1} \boldsymbol{\Sigma}_{1,i} \boldsymbol{\Sigma}_{1}^{-1} + \boldsymbol{\Sigma}_{2}^{-1} \boldsymbol{\Sigma}_{2,i} \boldsymbol{\Sigma}_{2}^{-1}) \hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{\Sigma}}_{d} = \hat{\boldsymbol{\Sigma}} - \hat{\boldsymbol{\Sigma}}_{i} \end{split}$$
(6)

where Σ_d denotes the covariance of dependent part of **p**, and Σ_i denotes the covariance of independent part of **p**, and $\hat{\Sigma}_i$, $\hat{\Sigma}_d$ denote the corresponding estimated covariance. SCI is a general extension of CI, if $\Sigma_{1,i} = \Sigma_{2,i} = 0$, it reduces to CI. The partial observation case based on the state space model can be deduced from equation (2) to equation (4), and can also be found in [29], [30].

B. Lie Group Theory

In [15], the authors give a detailed derivation of the uncertianty modeld on Lie group. Inspired by that, we list some of the important formulas from [15] that will be used in the derivation in the following to make this paper self-contained. The robot's pose can be denoted as T that lies on the special Euclidean Lie group of SE(3) and can be expressed as a 4×4 matrix:

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix}$$
(7)

where **R** is a 3×3 rotation matrix that forms the special orthogonal Lie group of SO(3), **t** is a 3D vector denotes the translation. The Lie algebra of SO(3) is denoted as $\mathfrak{so}(3)$ can be represented as skew-symmetric matrix. Since the dimension of $\mathfrak{so}(3)$ is 3, we can define an operator \wedge that transform an element of \mathbb{R}^3 to an element of $\mathfrak{so}(3)$:

$$\phi^{\wedge} = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}^{\wedge} = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix}$$
(8)

A measurement of the pose **T** can be denoted as $\mathbf{x}_1 = \exp(\boldsymbol{\xi}_1^{\wedge}) \oplus \mathbf{T}_1$, where $\mathbf{T}_1 \in SE(3)$ is a noise free mean value, $\boldsymbol{\xi}_1 = [\boldsymbol{\rho}_1^T, \boldsymbol{\phi}_1^T]^T = \begin{bmatrix} \rho_{1,1} & \rho_{1,2} & \rho_{1,3} & \phi_{1,1} & \phi_{1,2} & \phi_{1,3} \end{bmatrix}^T \in \mathbb{R}^6$ is a noisy perturbation, and it is unbiased Gaussian with $\boldsymbol{\xi}_1 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_1)$. The operator \oplus is the group plus operation, *e.g.*, matrix multiplication for SE(3), similarly we can also define the group minus operation as \oplus . For convenience of representation, we omit it if it is matrix multiplication on SE(2) or SE(3) in the following of this paper. The operator \wedge casts the vector onto Lie algebra $\mathfrak{se}(3)$ which lies on the tangent space of SE(3):

$$\boldsymbol{\xi}_{1}^{\wedge} = \begin{bmatrix} 0 & -\phi_{1,3} & \phi_{1,2} & \rho_{1,1} \\ \phi_{1,3} & 0 & -\phi_{1,1} & \rho_{1,2} \\ -\phi_{1,2} & \phi_{1,1} & 0 & \rho_{1,3} \\ 0 & 0 & 0 & 0 \end{bmatrix}$$
(9)

where the left top 3×3 corner is Lie algebra $\phi_1^{\wedge} \in \mathfrak{so}(3)$ of SO(3) and is skew-symmetric. We can also define the inverse operation \vee that $(\boldsymbol{\xi}^{\wedge})^{\vee} = \boldsymbol{\xi}$. The Lie algebra and Lie group are related by exponential map and logarithm map, *i.e.*, $\exp(\boldsymbol{\xi}^{\wedge}) \in SE(3)$ and $\log(\mathbf{T}) \in \mathfrak{se}(3)$. The exponential map can be expressed as:

$$\exp(\boldsymbol{\xi}^{\wedge}) = \sum_{k=1}^{\infty} \frac{(\boldsymbol{\xi}^{\wedge})^k}{k!} = \mathbf{I} + \boldsymbol{\xi}^{\wedge} + \frac{(\boldsymbol{\xi}^{\wedge})^2}{2} + \dots$$
(10)

The logarithm map can be expanded in a similar way. The strategy here is that we model the uncertainty in tangent space with Gaussian distribution and then wrap it the Lie group through exponential map, the process can be seen in Fig.1. We can define the probability density function (PDF) of \mathbf{x}_1 over SE(3):

$$p(\mathbf{T}) = \mathcal{C}(\boldsymbol{\Sigma}_1) \exp\left(-\frac{1}{2} \left(\underbrace{[\log(\mathbf{T}\mathbf{T}_1^{-1})]^{\vee}}_{\boldsymbol{\epsilon}_{\mathbf{T}_1}}\right)^T \boldsymbol{\Sigma}_1^{-1} \boldsymbol{\epsilon}_{\mathbf{T}_1}\right) \quad (11)$$

where $C(\Sigma_1)$ is the normalization constant. Note that we project the geodesic from the mean to variant on Lie group onto its tangent space with vector representation, which is Gaussian. Another interesting relationship between the Lie group and Lie algebra is the Baker-Campbell-Hausdorff formula (BCH) [54]. Given two Lie algebra elements $\boldsymbol{\xi}_1^{\wedge}$ and $\boldsymbol{\xi}_2^{\wedge}$,



Fig. 1. An illustration of the uncertainty around point "A" in the Lie algebra, and then project the uncertainty to the Lie group through exponential map. Shadow on the plane denotes uncertainty in the vector space around "A". Then it is wrapped to the Lie group denoted in green blow, where red circle denotes the uncertainty. The outmost circle corresponds to the boundary of the shadow in the vector space. The more inside the circle is, the higher probability the uncertainty affects.

the Lie algebra of the exponential composition of them can be expressed as:

$$\begin{aligned} \boldsymbol{\xi}_{1,2}^{\wedge} &= \log(\exp(\boldsymbol{\xi}_{1}^{\wedge}) \exp(\boldsymbol{\xi}_{2}^{\wedge})) \\ &= \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n} \sum_{\substack{r_{i}+s_{i}>0\\i\in[1,n]}} \frac{(\boldsymbol{\xi}_{1}^{\wedge})^{r_{1}}(\boldsymbol{\xi}_{2}^{\wedge})^{s_{1}} \cdots (\boldsymbol{\xi}_{1}^{\wedge})^{r_{n}}(\boldsymbol{\xi}_{2}^{\wedge})^{s_{n}}}{r_{1}!s_{1}! \cdots r_{n}!s_{n}!} \\ &= \boldsymbol{\xi}_{1}^{\wedge} + \boldsymbol{\xi}_{2}^{\wedge} + \frac{1}{2}[\boldsymbol{\xi}_{1}^{\wedge}, \boldsymbol{\xi}_{2}^{\wedge}] \\ &+ \frac{1}{12}([\boldsymbol{\xi}_{1}^{\wedge}, [\boldsymbol{\xi}_{1}^{\wedge}, \boldsymbol{\xi}_{2}^{\wedge}]] + [\boldsymbol{\xi}_{2}^{\wedge}, [\boldsymbol{\xi}_{2}^{\wedge}, \boldsymbol{\xi}_{1}^{\wedge}]]) + \cdots \end{aligned}$$
(12)

where $[\cdot, \cdot]$ is the Lie bracket, and $[\boldsymbol{\xi}_1^{\wedge}, \boldsymbol{\xi}_2^{\wedge}] = \boldsymbol{\xi}_1^{\wedge} \boldsymbol{\xi}_2^{\wedge} - \boldsymbol{\xi}_2^{\wedge} \boldsymbol{\xi}_1^{\wedge}$ on $\mathfrak{se}(3), r_i, s_i \in \mathbb{N}$. According to [15], if we define the operator λ as:

$$\boldsymbol{\xi}_{i}^{\wedge} = \begin{bmatrix} \boldsymbol{\rho}_{i} \\ \boldsymbol{\phi}_{i} \end{bmatrix}^{\wedge} = \begin{bmatrix} \boldsymbol{\phi}_{i}^{\wedge} & \boldsymbol{\rho}_{i}^{\wedge} \\ \mathbf{0} & \boldsymbol{\phi}_{i}^{\wedge} \end{bmatrix}$$
(13)

where the operator \wedge is defined as equation (8) shows. Then the Lie bracket can be expressed as $[\boldsymbol{\xi}_1^{\wedge}, \boldsymbol{\xi}_2^{\wedge}] = (\boldsymbol{\xi}_1^{\wedge} \boldsymbol{\xi}_2)^{\wedge}$. So, the BCH formula is:

$$\boldsymbol{\xi}_{1,2}^{\wedge} = \boldsymbol{\xi}_{1}^{\wedge} + \boldsymbol{\xi}_{2}^{\wedge} + \frac{1}{2} (\boldsymbol{\xi}_{1}^{\wedge} \boldsymbol{\xi}_{2})^{\wedge} \\ + \frac{1}{12} ((\boldsymbol{\xi}_{1}^{\wedge} \boldsymbol{\xi}_{1}^{\wedge} \boldsymbol{\xi}_{2})^{\wedge} + (\boldsymbol{\xi}_{2}^{\wedge} \boldsymbol{\xi}_{2}^{\wedge} \boldsymbol{\xi}_{1})^{\wedge}) + \dots$$
(14)

Further, if we assume one of the variant is small and ignore the high-order of it, takes \lor operator, then we can get:

$$\boldsymbol{\xi}_{12} \approx \boldsymbol{\xi}_2 + \boldsymbol{\mathcal{J}}_2^{-1} \boldsymbol{\xi}_1 \approx \boldsymbol{\xi}_1 + \boldsymbol{\mathcal{J}}_1^{-1} \boldsymbol{\xi}_2 \tag{15}$$

where \mathcal{J}_i^{-1} is the inverse Jacobian related to $\boldsymbol{\xi}_i$:

$$\mathcal{J}_i^{-1} = \sum_{n=0}^{\infty} \frac{B_n}{n!} (\boldsymbol{\xi}_i^{\scriptscriptstyle \wedge})^n \tag{16}$$

and B_n are the Bernoulli numbers:

$$B_0 = 1, B_1 = -\frac{1}{2}, B_2 = \frac{1}{6}, B_3 = 0, \dots,$$
 (17)

It was pointed in [15] that \mathcal{J}_i is the left Jacobian of SE(3) with definition:

$$\mathcal{J}_i = \sum_{n=0}^{\infty} \frac{1}{(n+1)!} (\boldsymbol{\xi}_i^{\wedge})^n = \int_0^1 \exp(\boldsymbol{\xi}_i^{\wedge})^{\alpha} d\alpha \qquad (18)$$

We also introduce the adjoint operation as follows:

$$Ad_{\mathbf{T}}(\boldsymbol{\xi}^{\wedge}) := Ad_{\mathbf{T}}\boldsymbol{\xi}^{\wedge} = \log(\mathbf{T}\exp(\boldsymbol{\xi}^{\wedge})\mathbf{T}^{-1})$$
(19)

which can yield the following formation:

$$\mathbf{T}\exp(\boldsymbol{\xi}^{\wedge}) = \exp(Ad_{\mathbf{T}}\boldsymbol{\xi})\mathbf{T}$$
(20)

IV. METHODOLOGY

A. Problem Definition

Given two estimate sources of the ground-truth robot pose, the fusion algorithm aims to yield a better estimate. In this paper, we give the derivation of split covariance intersection case, as the totally correlated fusion can be seen as a special case of split covariance intersection with the independent part as zero. Different from [29], state here is computated on Lie group, and uncertainty of the estimate source is modeled on Lie algebra and projected to the Lie group. Different from [15], both the dependent and independent parts of the state are considered in the process of fusion. We assume the estimate sources are corrupted by white Gaussian noise, and denote them as:

$$\mathbf{x}_{1} = \{\mathbf{p}_{1}, \mathbf{\Sigma}_{1,i} + \mathbf{\Sigma}_{1,d}\}, \mathbf{x}_{2} = \{\mathbf{p}_{2}, \mathbf{\Sigma}_{2,i} + \mathbf{\Sigma}_{2,d}\}$$
(21)

where the expectations p_1 and p_2 are elements on the Lie group, e.g., SE(3), SE(2). The measurement $\mathbf{x}_1 =$ $\exp(\boldsymbol{\xi}_1^{\wedge})\mathbf{p}_1, \, \mathbf{x}_2 = \exp(\boldsymbol{\xi}_2^{\wedge})\mathbf{p}_2, \, \boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$ are the perturbation around their expectations, whose covariance are $\Sigma_{1,i} + \Sigma_{1,d}$ and $\Sigma_{2,i} + \Sigma_{2,d}$ respectively. $\Sigma_{1,i}$ and $\Sigma_{2,i}$ are the independent covariance, $\Sigma_{1,d}$ and $\Sigma_{2,d}$ are the dependent covariance. Here the uncertainty is modeled using a Gaussian distribution on the Lie algebra, and then projected on the Lie group by the exponential map. It is meaningful for both left and right multiplication, [15] indicates that left multiplication of the uncertainty has advantage over the right counterpart. The problem to be investigated in this paper is how to compute a better estimate $\hat{\mathbf{x}}$, specifically the fused expectation $\hat{\mathbf{p}}$, and the covariance on the Lie algebra Σ_i , Σ_d , with \mathbf{x}_1 and \mathbf{x}_2 . An illustration of the pose fusion problem can be seen in Fig. 2.

B. Fusion Strategy

As the uncertainty perturbs the measurement on Lie group, the uncertainties of different measurements located at different points need to be associated on the Lie group. First, the intuition behind the fusion needs to be clarified. The fused result can be regarded as a convex combination of the two estimate sources. Given two estimates and their covariance $\mathbf{x}_1 = {\mathbf{p}_1, \boldsymbol{\Sigma}_1}, \mathbf{x}_2 = {\mathbf{p}_2, \boldsymbol{\Sigma}_2}$, the fused result can be denoted as:

$$\hat{\mathbf{p}} = \mathbf{W}_1 \mathbf{p}_1 + \mathbf{W}_2 \mathbf{p}_2 \tag{22}$$

where $\mathbf{W}_1 > 0$, $\mathbf{W}_2 > 0$ are positive definite coefficients, and we also have $\mathbf{W}_1 + \mathbf{W}_2 = \mathbf{I}$. Then, these two coefficients need



Fig. 2. An illustration of the pose fusion problem with uncertainty modeled in Lie algebra. The two measurements are pose estimate from "i" to "j" denoted in orange and green, respectively. The fusion result is denoted in blue. The ellipses denote uncertainties in the Lie algebra, and they have both dependent and independent parts.

to be solved according to the optimization criteria. Usually, the result is obtained by minimizing Mahalanobis distance or determinant of the fused covariance. The covariance of the fused result can be defined as:

$$\Sigma = \mathbb{E}\left[(\hat{\mathbf{p}} - \bar{\mathbf{p}})(\hat{\mathbf{p}} - \bar{\mathbf{p}})^T \right] = \mathbb{E}\left[(\mathbf{W}_1(\mathbf{p}_1 - \bar{\mathbf{p}}) + \mathbf{W}_2(\mathbf{p}_2 - \bar{\mathbf{p}}))(\mathbf{W}_1(\mathbf{p}_1 - \bar{\mathbf{p}}) + \mathbf{W}_2(\mathbf{p}_2 - \bar{\mathbf{p}}))^T \right]$$
(23)

where $\bar{\mathbf{p}}$ is the ground truth of \mathbf{p} . If we assume \mathbf{x}_1 and \mathbf{x}_2 are independent, *i.e.*, $\mathbb{E}[(\mathbf{p}_1 - \bar{\mathbf{p}})(\mathbf{p}_2 - \bar{\mathbf{p}})^T] = 0$, then $\Sigma = \mathbf{W}_1 \Sigma_1 \mathbf{W}_1^T + \mathbf{W}_2 \Sigma_2 \mathbf{W}_2^T$. Substituting $\mathbf{W}_2 = \mathbf{I} - \mathbf{W}_1$, taking the derivative with respect to \mathbf{W}_1 and setting to zero will exactly lead to the result shown in equation (1). If the two estimates are dependent, *i.e.*, $\mathbb{E}[(\mathbf{p}_1 - \bar{\mathbf{p}})(\mathbf{p}_2 - \bar{\mathbf{p}})^T] = \Sigma_{12} \neq 0$, then $\Sigma = \mathbf{W}_1 \Sigma_1 \mathbf{W}_1^T + \mathbf{W}_2 \Sigma_2 \mathbf{W}_2^T + \mathbf{W}_1 \Sigma_{12} \mathbf{W}_2^T + \mathbf{W}_2 \Sigma_{21} \mathbf{W}_2^T$. Usually, the correlation Σ_{12} is unknown, ignoring it may cause inconsistent problem for the fusion. In [30], the authors proposed that Σ always lies within the intersection of Σ_1 and Σ_2 no matter what the choice of $\Sigma_{1,2}$ is. To prevent inconsistency of correlated fusion, [30] proposed to add a parameter in the fusion, which guarantees that uncertainty of the fused result covers the intersection of the uncertainties of the estimated sources. The fusion strategy of [30] is as equation (5) shows. This is based on the fact that

$$\begin{bmatrix} \boldsymbol{\Sigma}_1 & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_2 \end{bmatrix} \leqslant \begin{bmatrix} \boldsymbol{\omega} \boldsymbol{\Sigma}_1^{-1} & \boldsymbol{0} \\ \boldsymbol{0} & (1-\boldsymbol{\omega}) \boldsymbol{\Sigma}_2^{-1} \end{bmatrix}^{-1}$$
(24)

always holds no matter what $\Sigma_{1,2}$ is.

The previous fusion is derived in the vector space, for fusion on the manifold, the strategy is quite different. Influence of the uncertainty is not linearly imposed on the estimate. Following the idea of [15], error between the estimate source and the optimal estimate p^* can be expressed as:

$$\epsilon_{k} = \log(\mathbf{p}^{*}\mathbf{p}_{k}^{-1})^{\vee} = \log\left(\exp(\boldsymbol{\xi}^{\wedge})\hat{\mathbf{p}}\mathbf{p}_{k}^{-1}\right)^{\vee}$$
$$= \log\left(\exp(\boldsymbol{\xi}^{\wedge})\exp(\boldsymbol{\xi}_{k}^{\wedge})\right)^{\vee}$$
(25)

where $\boldsymbol{\xi} = \log(\mathbf{p}^* \hat{\mathbf{p}}^{-1})^{\vee}$ is the difference between the optimal estimate and our best guess so far, and $\boldsymbol{\xi}_k = \log(\hat{\mathbf{p}} \mathbf{p}_k^{-1})^{\vee}$ is the difference between our best guess so far and each individual estimate. Applying BCH formula in equation (12), the error can be expressed as:

$$\boldsymbol{\epsilon}_k \approx \boldsymbol{\xi}_k + \mathcal{J}_k^{-1} \boldsymbol{\xi} \tag{26}$$

According to equation (16), the inverse Jacobian is:

$$\mathcal{J}_k^{-1} = \sum_{n=0}^{\infty} \frac{B_n}{n!} (\boldsymbol{\xi}_k^{\scriptscriptstyle \wedge})^n \tag{27}$$

Here we assume $\boldsymbol{\xi}$ is small, *i.e.*, there is no big difference between the current best estimate and the optima. At the beginning, there may be a gap between the estimate and the optima if we do not have a good initialization. As the iteration goes on, the gap will become smaller and smaller. And in the last several iterations, this assumption will hold. This means that we still ignore the higher-order terms of $\boldsymbol{\xi}$ at the beginning. Though this approximation is not accurate enough for first several iterations, it still can minimize the error and drive the estimate towards the real optima.

We need to get an optimal solution of $\boldsymbol{\xi}$ to compensate the current best estimate proceeding to the real optimal estimate. As the error is an approximation, the result is not analytic nor optimal for one optimization step, this compensation procedure needs to be implemented iteratively until $\boldsymbol{\xi}$ is small enough. Because initialization is not close enough to the optimal value, and after every iteration, it moves closer to the optima. The criteria to get the optimal value at every step is to minimize Mahalanobis distance with respect to $\boldsymbol{\xi}$. We define the cost function using Mahalanobis distance as:

$$V = \frac{1}{2} \begin{bmatrix} \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \end{bmatrix}^T \begin{bmatrix} \boldsymbol{\Sigma}_1 & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_2 \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \end{bmatrix}$$
(28)

where $\Sigma_{2,1} = \Sigma_{1,2}^T$ is unknown. According to equation (21), $\Sigma_1 = \Sigma_{1,i} + \Sigma_{1,d}, \ \Sigma_2 = \Sigma_{2,i} + \Sigma_{2,d}$, where $\Sigma_{1,i}$ and $\Sigma_{2,i}$ denote the independent parts, $\Sigma_{1,d}$ and $\Sigma_{2,d}$ denote the dependent parts. Combining with equation (24), we have:

$$\begin{bmatrix} \boldsymbol{\Sigma}_{1} & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_{2} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_{1,i} & 0 \\ 0 & \boldsymbol{\Sigma}_{2,i} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\Sigma}_{1,d} & \boldsymbol{\Sigma}_{1,2} \\ \boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_{2,d} \end{bmatrix}$$

$$\leqslant \begin{bmatrix} \boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{1,d} & 0 \\ 0 & \boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d} \end{bmatrix}$$
(29)

Then, we model a more conservative cost function by substituting the covariance matrix with:

$$V = \frac{1}{2} \begin{bmatrix} \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \end{bmatrix}^T \begin{bmatrix} \boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{1,d} & 0 \\ \boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d} \end{bmatrix}^{-1} \begin{bmatrix} \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \end{bmatrix}$$
$$= \frac{1}{2} (\boldsymbol{\xi}_1 + \mathcal{J}_1^{-1} \boldsymbol{\xi})^T (\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{1,d})^{-1} (\boldsymbol{\xi}_1 + \mathcal{J}_1^{-1} \boldsymbol{\xi}) + \frac{1}{2} (\boldsymbol{\xi}_2 + \mathcal{J}_2^{-1} \boldsymbol{\xi})^T (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d})^{-1} (\boldsymbol{\xi}_2 + \mathcal{J}_2^{-1} \boldsymbol{\xi})$$
(30)

Taking the derivative of V with respect to $\boldsymbol{\xi}$ and set it to zero, we can get:

$$\boldsymbol{\xi} = -\left(\mathcal{J}_{1}^{-T}(\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega}\boldsymbol{\Sigma}_{1,d})^{-1}\mathcal{J}_{1}^{-1} + \mathcal{J}_{2}^{-T}(\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega}\boldsymbol{\Sigma}_{2,d})^{-1}\mathcal{J}_{2}^{-1}\right)^{-1}\left(\mathcal{J}_{1}^{-T}(\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega}\boldsymbol{\Sigma}_{1,d})^{-1}\boldsymbol{\xi}_{1} + \mathcal{J}_{2}^{-T}(\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega}\boldsymbol{\Sigma}_{2,d})^{-1}\boldsymbol{\xi}_{2}\right)$$
(31)

The updated covariance is:

$$\Sigma = \left(\mathcal{J}_{1}^{-T} (\Sigma_{1,i} + \frac{1}{\omega} \Sigma_{1,d})^{-1} \mathcal{J}_{1}^{-1} + \mathcal{J}_{2}^{-T} (\Sigma_{2,i} + \frac{1}{1-\omega} \Sigma_{2,d})^{-1} \mathcal{J}_{2}^{-1} \right)^{-1}$$
(32)

The choice of ω can be determined by optimizing the update with respect to different performance criteria, such as minimizing the trace or the determinant of Σ . As the cost function is convex to ω , lots of convex optimization algorithms can be used here, *e.g.*, Newton-Raphson, semidefinite and convex programming.

At the last iteration, we can compute the independent and dependent covariance of the fusion result. Let $\boldsymbol{\xi} = \boldsymbol{\xi}^i + \boldsymbol{\xi}^d$, where $\boldsymbol{\xi}^i$ and $\boldsymbol{\xi}^d$ denote the independent and correlated components of the uncertainty respectively. At every iteration, the uncertainty of $\boldsymbol{\xi}_k$ lies on the estimate source \mathbf{p}_k , as the previous estimate $\hat{\mathbf{p}}$ can be regarded as constant. So, we have:

$$\boldsymbol{\xi} = \boldsymbol{\xi}^{i} + \boldsymbol{\xi}^{d} = -\boldsymbol{\Sigma} \left(\mathcal{J}_{1}^{-T} (\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{1,d})^{-1} \boldsymbol{\xi}_{1} + \mathcal{J}_{2}^{-T} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d})^{-1} \boldsymbol{\xi}_{2} \right)$$
(33)

Squaring both sides, taking expectations, and collecting the independent terms, the updated independent covariance is:

$$\boldsymbol{\Sigma}_{i} = \boldsymbol{\Sigma} \left(\mathcal{J}_{1}^{-T} (\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{1,d})^{-1} \boldsymbol{\Sigma}_{1,i} (\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{1,d})^{-T} \mathcal{J}_{1}^{-1} + \mathcal{J}_{2}^{-T} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d})^{-1} \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d})^{-T} \mathcal{J}_{2}^{-1} \right) \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d})^{-T} \mathcal{J}_{2}^{-1}) \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d})^{-T} \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,i})^{-T} \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,i})^{-T} \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,i})^{-T} \boldsymbol{\Sigma}_{2,i})^{-T} \boldsymbol{\Sigma}_{2,i} (\boldsymbol{\Sigma}_{2,$$

The updated dependent covariance is $\Sigma_d = \Sigma - \Sigma_i$. We can update the estimate by:

$$\hat{\mathbf{p}} \leftarrow \exp(\boldsymbol{\xi}^{\wedge})\hat{\mathbf{p}}$$
 (35)

This is easily to extend to fusion case of various estimate

sources:

$$\boldsymbol{\xi} = -\left(\sum_{k=1}^{n} \mathcal{J}_{k}^{-T} (\boldsymbol{\Sigma}_{k,i} + \frac{1}{\omega_{k}} \boldsymbol{\Sigma}_{k,d})^{-1} \mathcal{J}_{k}^{-1}\right)^{-1} \cdot \left(\sum_{k=1}^{n} \mathcal{J}_{k}^{-T} (\boldsymbol{\Sigma}_{k,i} + \frac{1}{\omega_{k}} \boldsymbol{\Sigma}_{k,d})^{-1} \boldsymbol{\xi}_{k}\right) \\ \boldsymbol{\Sigma} = \left(\sum_{k=1}^{n} \mathcal{J}_{k}^{-T} (\boldsymbol{\Sigma}_{k,i} + \frac{1}{\omega_{k}} \boldsymbol{\Sigma}_{k,d})^{-1} \mathcal{J}_{k}^{-1}\right)^{-1} \\ \hat{\mathbf{p}} \leftarrow \exp(\boldsymbol{\xi}^{\wedge}) \hat{\mathbf{p}}$$
(36)
$$\boldsymbol{\Sigma}_{i} = \boldsymbol{\Sigma} \left(\sum_{k=1}^{n} \mathcal{J}_{k}^{-T} (\boldsymbol{\Sigma}_{k,i} + \frac{1}{\omega_{k}} \boldsymbol{\Sigma}_{k,d})^{-1} \boldsymbol{\Sigma}_{k,i} \right) \cdot (\boldsymbol{\Sigma}_{k,i} + \frac{1}{\omega_{k}} \boldsymbol{\Sigma}_{k,d})^{-T} \mathcal{J}_{k}^{-1}) \boldsymbol{\Sigma}$$
$$\omega_{k} \in [0, 1], \sum_{k=1}^{n} \omega_{k} = 1$$

The fusion strategy is summarized in Algorithm 1.

Algorithm 1: Split Covariance Intersection on Lie Group

Input: Esimate sources: \mathbf{p}_k **Output:** Fusion result: \mathbf{p}^*

- 1 Initialize: $\hat{\mathbf{p}}, \boldsymbol{\xi}_k = \log(\hat{\mathbf{p}}\mathbf{p}_k^{-1})^{\vee}$
- 2 Compute the updated covariance:

$$\boldsymbol{\Sigma} = \left(\sum_{k=1}^{n} \mathcal{J}_{k}^{-T} (\boldsymbol{\Sigma}_{k,i} + \frac{1}{\omega_{k}} \boldsymbol{\Sigma}_{k,d})^{-1} \mathcal{J}_{k}^{-1} \right)^{-1}$$

3 Minimize trace of $\boldsymbol{\Sigma}$ to get $\boldsymbol{\omega}$

- 4 while not converge do
- 5 Compute the compensation of the pose:

$$\begin{aligned} \mathbf{6} & \mathbf{\xi} = \mathbf{\Sigma} \left(\sum_{k=1}^{n} \mathcal{J}_{k}^{-T} (\mathbf{\Sigma}_{k,i} + \frac{1}{\omega_{k}} \mathbf{\Sigma}_{k,d})^{-1} \mathbf{\xi}_{k} \right) \\ \text{Update the estimate: } \hat{\mathbf{p}} \leftarrow \exp(\mathbf{\xi}^{\wedge}) \hat{\mathbf{p}} \end{aligned}$$

- 8 end
- 9 Update the independent covariance Σ_i as equation (34) 10 Compute the dependent covariance: $\Sigma_d = \Sigma - \Sigma_i$

C. Fusion for Partial Observation Case

We consider a more general case in terms of pose estimation for robots, *i.e.*, the two estimate sources do not reflect the same states. This is quite common in the field of robotics, for example, one estimate is 6D movement measurement from IMU, while the other is just 3D position from GPS. The state space model on Lie group is different from that in the vector space shown in equation (2), which can be expressed as:

$$\begin{aligned} \mathbf{X}_{k} = & \mathbf{X}_{k-1} \oplus \mathbf{u}_{k} \\ & \mathbf{Y}_{k} = \exp(\boldsymbol{\xi}^{h\wedge}) \mathbf{H}(\mathbf{X}_{k}) \end{aligned}$$
 (37)

where $\mathbf{u}_k = \exp(\boldsymbol{\xi}_k^{u\wedge}) \bar{\mathbf{u}}_k$ is the control input at time step k, with measurement $\bar{\mathbf{u}}_k$ and uncertainty $\boldsymbol{\xi}_k^u$ whose covariance is $\boldsymbol{\Sigma}^u$. **H** is the transition function which can be linear or nonlinear, uncertainty of it is denoted by $\boldsymbol{\xi}^h$ on the Lie algebra whose covariance is $\boldsymbol{\Sigma}_{2,i} + \boldsymbol{\Sigma}_{2,d}$. The measurement can be in the vector space, which can be regarded as a special case of

the Lie group. It also satisfies the above definition. Here we assume that the predictive process is not corrupted by correlated noise, as the measurements from IMU, wheel encoders are independent from each other at different timestamp, and are independent from the observation as well. The fusion strategy consists of two steps: predictive update and measurement update. The predictive update is based on the predictive model (first equation of equation (37)), which updates the estimate according to the control input. The measurement update is based on the measurement model (second equation of equation (37)), which updates the estimate when new measurement arrives. Next, the formula of these two steps will be derived in detail.

Predictive update Both the estimate and its uncertainty are propagated in the predictive update. Inspired by [16], formula of the predictive update can be expressed as:

$$\exp(\boldsymbol{\xi}_{k|k-1}^{\wedge})\mathbf{p}_{k|k-1} = \exp(\boldsymbol{\xi}_{k-1|k-1}^{\wedge})\mathbf{p}_{k-1|k-1}\exp(\boldsymbol{\xi}_{k}^{u\wedge})\bar{\mathbf{u}}_{k}$$
$$= \exp(\boldsymbol{\xi}_{k-1|k-1}^{\wedge})\exp(Ad_{\mathbf{p}_{k-1|k-1}}\boldsymbol{\xi}_{k}^{u\wedge})\mathbf{p}_{k-1|k-1}\bar{\mathbf{u}}_{k}$$
(38)

where the second equal sign holds because of equation (19). So, we can predict the estimate as:

$$\mathbf{p}_{k|k-1} = \mathbf{p}_{k-1|k-1}\bar{\mathbf{u}}_k \tag{39}$$

Then the uncertainty part is:

$$\exp(\boldsymbol{\xi}_{k|k-1}^{\wedge}) = \exp(\boldsymbol{\xi}_{k-1|k-1}^{\wedge})\exp(Ad_{\mathbf{p}_{k-1|k-1}}\boldsymbol{\xi}_{k}^{u\wedge}) \quad (40)$$

Using the BCH formula shown in equation (14), we can get:

$$\boldsymbol{\xi}_{k|k-1} = \boldsymbol{\xi}_{k-1|k-1} + \boldsymbol{\xi}_{k}^{u'} + \frac{1}{2} \boldsymbol{\xi}_{k-1|k-1}^{\lambda} \boldsymbol{\xi}_{k}^{u'} + \frac{1}{12} \boldsymbol{\xi}_{k-1|k-1}^{\lambda} \boldsymbol{\xi}_{k}^{u'} + \frac{1}{12} \boldsymbol{\xi}_{k}^{u'\lambda} \boldsymbol{\xi}_{k-1|k-1}^{u'\lambda} \boldsymbol{\xi}_{k-1|k-1}^{$$

where $\boldsymbol{\xi}_{k}^{u'} = Ad_{\mathbf{p}_{k-1|k-1}}\boldsymbol{\xi}_{k}^{u}$. To compute the covariance after prediction, we can derive it from the definition:

$$E[\boldsymbol{\xi}_{k|k-1}\boldsymbol{\xi}_{k|k-1}^{T}] \approx E \left[\underbrace{\boldsymbol{\xi}_{k-1|k-1}\boldsymbol{\xi}_{k-1|k-1}^{T} + \boldsymbol{\xi}_{k}^{u'}\boldsymbol{\xi}_{k}^{u'T}}_{\text{2nd Order Diagonal Terms}} + \underbrace{\boldsymbol{\xi}_{k-1|k-1}\boldsymbol{\xi}_{k}^{u'T} + \boldsymbol{\xi}_{k}^{u'}\boldsymbol{\xi}_{k-1|k-1}^{T}}_{\text{2nd Order Terms}} + \underbrace{\cdots}_{\text{Higher Order Terms}} \right]$$
(42)

As we assume the predictive noise is independent, expectation of all the cross terms are zeros. Ignoring the higher order terms, a first order estimate of the covariance is:

$$E[\boldsymbol{\xi}_{k|k-1}\boldsymbol{\xi}_{k|k-1}^{T}] \approx E\left[\boldsymbol{\xi}_{k-1|k-1}\boldsymbol{\xi}_{k-1|k-1}^{T}\right] + E\left[\boldsymbol{\xi}_{k}^{u'}\boldsymbol{\xi}_{k}^{u'T}\right]$$
(43)

which yields:

$$\boldsymbol{\Sigma}_{k|k-1} \approx \boldsymbol{\Sigma}_{k-1|k-1} + Ad_{\mathbf{p}_{k-1|k-1}} \boldsymbol{\Sigma}_{k}^{u} Ad_{\mathbf{p}_{k-1|k-1}}^{T} \qquad (44)$$

In equation (44), the frist term is the uncertainty last timestep, the second term is uncertainty from the control input propagating to the current prediction. Based on the independent assupption, we can also have:

$$\boldsymbol{\Sigma}_{k|k-1}^{i} = \boldsymbol{\Sigma}_{k-1|k-1}^{i} + Ad_{\mathbf{p}_{k-1|k-1}} \boldsymbol{\Sigma}_{k}^{u} Ad_{\mathbf{p}_{k-1|k-1}}^{T} \qquad (45)$$

The correlated uncertainty remains the same $\Sigma^d_{k|k-1}$ $\Sigma^d_{k-1|k-1}$.

Measurement update The estimate and its uncertainty are updated by fusing the predicted result and the measurement in this step. Different from that in the vector space, measurement update on the Lie group needs to be optimized iteratively. We needs to utilize the theorem presented in Section IV-B. Error computation for the predictive estimate is same, we only need to take a closer look at error computation for the measurement estimate:

$$\boldsymbol{\epsilon}_{2} = \log \left(\mathbf{H}(\mathbf{p}^{*}) \ominus \mathbf{y}_{k} \right)^{\vee} = \log \left(\mathbf{H}(\exp(\boldsymbol{\xi}^{\wedge}) \oplus \hat{\mathbf{p}}) \ominus \mathbf{y}_{k} \right)^{\vee}$$
$$= \log \left(\underbrace{\exp(\mathcal{J}_{\mathbf{H}}\boldsymbol{\xi}^{\wedge})}_{\exp(\boldsymbol{\xi}^{\prime})} \oplus \underbrace{(\mathbf{H}(\hat{\mathbf{p}}) \ominus \mathbf{y}_{k})}_{\exp(\boldsymbol{\xi}^{\wedge}_{2})} \right)^{\vee}$$
(46)

where the third equal sign is based on the form of left Jacobian in [55] (equation (45) therein), $\mathcal{J}_{\mathbf{H}}$ is the left Jacobian of function H, which can be expressed as:

$$\mathcal{J}_{\mathbf{H}} = \frac{D\mathbf{H}(\hat{\mathbf{p}})}{D\hat{\mathbf{p}}} = \frac{\partial \log(\mathbf{H}(\exp(\boldsymbol{\tau}) \oplus \hat{\mathbf{p}}) \ominus \mathbf{H}(\hat{\mathbf{p}}))}{\partial \boldsymbol{\tau}} \bigg|_{\boldsymbol{\tau}=0}$$
(47)

Note that the function H can be a function by which the measurement is still an element on the manifold (different from the original element or manifold). In this case, we compensate the error between the current and optimal estimates on the original manifold and transform it by the same function. Or the function H transforms the Lie group to the vector space, which can be regarded as a special case of the manifold. Then, H in equation (46) denotes the Jacobian computed from the Lie group. Applying the BCH formula twice gives the following form:

$$\boldsymbol{\epsilon}_2 \approx \boldsymbol{\xi}_2 + \boldsymbol{\mathcal{J}}_2^{-1} \boldsymbol{\xi}' = \boldsymbol{\xi}_2 + \boldsymbol{\mathcal{J}}_2^{-1} \boldsymbol{\mathcal{J}}_{\mathbf{H}} \boldsymbol{\xi}$$
(48)

Applying the formula in Section IV-B, the update criteria at one step can be given as:

$$\begin{split} \boldsymbol{\Sigma}_{1} &= \boldsymbol{\Sigma}_{k|k-1}^{i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{k|k-1}^{d} \\ \boldsymbol{\Sigma}_{2} &= \boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d} \\ \boldsymbol{\Sigma}_{k|k} &= \left(\mathcal{J}_{1}^{-T} \boldsymbol{\Sigma}_{1}^{-1} \mathcal{J}_{1}^{-1} + \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \mathcal{J}_{2}^{-1} \mathcal{J}_{\mathbf{H}} \right)^{-1} \quad (49) \\ \boldsymbol{\xi}_{1} &= \log(\mathbf{p}_{k|k} \mathbf{p}_{k|k-1}^{-1})^{\vee}, \boldsymbol{\xi}_{2} &= \log(\mathbf{H}(\mathbf{p}_{k|k}) \ominus \mathbf{y}_{k})^{\vee} \\ \boldsymbol{\xi} &= -\boldsymbol{\Sigma}_{k|k} \left(\mathcal{J}_{1}^{-T} \boldsymbol{\Sigma}_{1}^{-1} \boldsymbol{\xi}_{1} + \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \boldsymbol{\xi}_{2} \right) \\ \mathbf{p}_{k|k} \longleftarrow \exp(\boldsymbol{\xi}^{\wedge}) \mathbf{p}_{k|k} \end{split}$$

After the last iteration, the independent and correlated covariance of the fusion can be updated as:

$$\Sigma_{k|k}^{i} = \Sigma_{k|k} \left(\mathcal{J}_{1}^{-T} \Sigma_{1}^{-1} \Sigma_{k|k-1}^{i} \Sigma_{1}^{-T} \mathcal{J}_{1}^{-1} + \mathcal{J}_{H}^{T} \mathcal{J}_{2}^{-T} \Sigma_{2}^{-1} \Sigma_{2,i} \Sigma_{2}^{-T} \mathcal{J}_{2}^{-1} \mathcal{J}_{H} \right) \Sigma_{k|k}$$

$$\Sigma_{k|k}^{d} = \Sigma_{k|k} - \Sigma_{k|k}^{i}$$
(50)

It can be verified that the above fusion strategy for partial observation case is consistent with that of Kalman filter method (proved in the Appendix), what is different here is that the fusion is derived on the Lie group. The fusion strategy for the partial observation case is summarized in Algorithm 2.

Algorithm 2: SCI-LG for Partial Observation

Input: Control inputs \mathbf{u}_k , and measurements \mathbf{y}_k **Output:** Fusion result $\mathbf{p}_{k|k}$

```
1 Initialize: \mathbf{p}_0
```

2 Predictive update:

3 State propogation:
$$\mathbf{p}_{k|k-1} = \mathbf{p}_{k-1|k-1} \bar{\mathbf{u}}_k$$

Uncertainty prediction: $\Sigma_{k|k-1}^d = \Sigma_{k-1|k-1}^d$, 4

$$\boldsymbol{\Sigma}_{k|k-1}^{i} = \boldsymbol{\Sigma}_{k-1|k-1}^{i} + Ad_{\mathbf{p}_{k-1|k-1}} \boldsymbol{\Sigma}_{k}^{u} Ad_{\mathbf{p}_{k-1|k-1}}^{T}$$

5 Measurement update:

Initialize $\mathbf{p}_{k|k} = \mathbf{p}_{k|k-1}$ 6

Update the covariance $\Sigma_{k|k}$ as equation (49) 7

Minimize trace of $\Sigma_{k|k}$ with respect to ω 8

9 while not converge do

Compute the compensation
$$\boldsymbol{\xi}$$
 as equation (49)

Update estimate
$$\mathbf{p}_{k|k}$$
 as equation (49)

12 end

10 11

Update the independent covariance as equation (50) 13

14 Compute the correlated covariance as equation (50)

D. Theoretical Analysis

Finally, in this section, we analyze the consistency of the proposed method. Different from the consistency defined for purely independent or dependent sources fusion, the case for partially dependent fusion needs to consider both parts. In [29], the authors give the definition and proof of split covariance intersection filter in vector space. In this paper, we extend it to the Lie group. Following the framework in [29], we discuss the consistency in two cases: A-split consistency which is tightly consistent, and B-split consistency which is relatively loosely consistent.

A-split consistency: Given an estimate on the Lie group $\{\mathbf{p}, \boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_d\}$ (we can image the uncertainty also has two parts, *i.e.*, $\boldsymbol{\xi} = \boldsymbol{\xi}_i + \boldsymbol{\xi}_d$), it is A-split consistent if it satisfies:

$$\boldsymbol{\Sigma}_{d} \ge \boldsymbol{\Sigma}_{d}^{*} = E(\bar{\boldsymbol{\xi}}_{d} \bar{\boldsymbol{\xi}}_{d}^{T}) \quad \boldsymbol{\Sigma}_{i} \ge \boldsymbol{\Sigma}_{i}^{*} = E(\bar{\boldsymbol{\xi}}_{i} \bar{\boldsymbol{\xi}}_{i}^{T})$$
(51)

where $\bar{\xi}_i$, and $\bar{\xi}_d$ denote the real uncertainty of their estimate respectively.

B-split consistency: Given an estimate on the Lie group $\{\mathbf{p}, \boldsymbol{\Sigma}_i + \boldsymbol{\Sigma}_d\}$, it is A-split consistent if it satisfies:

$$\Sigma_d \ge \Sigma_d^* = E(\bar{\xi}_d \bar{\xi}_d^T) \quad \Sigma \ge \Sigma^* = E(\bar{\xi} \bar{\xi}^T)$$
 (52)

Theorem 1: If the two estimate sources are both A-split consistent, then the fusion result by the proposed method shown in Algorithm 1 is A-split consistent for any $\omega \in [0, 1]$. **Proof** 1: According to equation (33), we have:

$$\boldsymbol{\xi}_{i} = -\boldsymbol{\Sigma} \left(\underbrace{\mathcal{J}_{1}^{-T} (\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega} \boldsymbol{\Sigma}_{1,d})^{-1}}_{\mathbf{Q}_{1}} \boldsymbol{\xi}_{1,i} + \underbrace{\mathcal{J}_{2}^{-T} (\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega} \boldsymbol{\Sigma}_{2,d})^{-1}}_{\mathbf{Q}_{2}} \boldsymbol{\xi}_{2,i} \right)$$
(53)

The independent covariance after fusion is as equation (34) shows, then we have:

$$\begin{split} \boldsymbol{\Sigma}_{i} &- E[\bar{\boldsymbol{\xi}}_{i}\bar{\boldsymbol{\xi}}_{i}^{T}] = \boldsymbol{\Sigma} \left(\mathbf{Q}_{1}\boldsymbol{\Sigma}_{1,i}\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}\boldsymbol{\Sigma}_{2,i}\mathbf{Q}_{2}^{T} \right)\boldsymbol{\Sigma} - \\ \boldsymbol{\Sigma} \left(\mathbf{Q}_{1}E[\bar{\boldsymbol{\xi}}_{1,i}\bar{\boldsymbol{\xi}}_{1,i}^{T}]\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}E[\bar{\boldsymbol{\xi}}_{2,i}\bar{\boldsymbol{\xi}}_{2,i}^{T}]\mathbf{Q}_{2}^{T} \right)\boldsymbol{\Sigma} = \\ \boldsymbol{\Sigma} \left(\mathbf{Q}_{1}(\boldsymbol{\Sigma}_{1,i} - E[\bar{\boldsymbol{\xi}}_{1,i}\bar{\boldsymbol{\xi}}_{1,i}^{T}])\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}(\boldsymbol{\Sigma}_{2,i} - E[\bar{\boldsymbol{\xi}}_{2,i}\bar{\boldsymbol{\xi}}_{2,i}^{T}])\mathbf{Q}_{2}^{T} \right)\boldsymbol{\Sigma} \right. \end{split}$$
(54)

As the estimate sources are both A-split consistent, we have:

$$\boldsymbol{\Sigma}_{1,i} \ge E[\bar{\boldsymbol{\xi}}_{1,i}\bar{\boldsymbol{\xi}}_{1,i}^T] \quad \boldsymbol{\Sigma}_{2,i} \ge E[\bar{\boldsymbol{\xi}}_{2,i}\bar{\boldsymbol{\xi}}_{2,i}^T]$$
(55)

So, we can get:

$$\boldsymbol{\Sigma}_i - E[\bar{\boldsymbol{\xi}}_i \bar{\boldsymbol{\xi}}_i^T] \ge 0 \tag{56}$$

For the dependent part, because the estimate sources are both A-split consistent, then

$$\boldsymbol{\Sigma}_{1,d} \ge E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{1,d}^T] \quad \boldsymbol{\Sigma}_{2,d} \ge E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{2,d}^T] \tag{57}$$

which yields:

$$\begin{split} \Sigma_{d} - E[\bar{\boldsymbol{\xi}}_{d}\bar{\boldsymbol{\xi}}_{d}^{T}] &= \Sigma - \Sigma_{i} - \Sigma \left(\underbrace{\mathbf{Q}_{1}E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}]\mathbf{Q}_{1}^{T}}_{\mathbf{R}} \right) \\ \underbrace{\mathbf{Q}_{2}E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{2,d}^{T}]\mathbf{Q}_{2}^{T} + \mathbf{Q}_{1}E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{2,d}^{T}]\mathbf{Q}_{2}^{T} + \mathbf{Q}_{2}E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}]\mathbf{Q}_{1}^{T}}_{\mathbf{R}} \right) \\ \cdot \Sigma &= \Sigma \left(\mathbf{Q}_{1}\frac{1}{\omega}\Sigma_{1,d}\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}\frac{1}{1-\omega}\Sigma_{2,d}\mathbf{Q}_{2}^{T} - \mathbf{R} \right) \Sigma \\ \geq \Sigma \left(\mathbf{Q}_{1}\frac{1-\omega}{\omega}E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}]\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}\frac{\omega}{1-\omega}E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{2,d}^{T}]\mathbf{Q}_{2}^{T} - \mathbf{Q}_{1}E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{2,d}^{T}]\mathbf{Q}_{2}^{T} - \mathbf{Q}_{2}E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}]\mathbf{Q}_{1}^{T} \right) \Sigma \\ &= \Sigma E \left(\underbrace{\left[\sqrt{\frac{1-\omega}{\omega}}\mathbf{Q}_{1}\bar{\boldsymbol{\xi}}_{1,d} - \sqrt{\frac{\omega}{1-\omega}}\mathbf{Q}_{2}\bar{\boldsymbol{\xi}}_{2,d}}_{\mathbf{H}} \right] \mathbf{H}^{T} \right) \Sigma \geq 0 \\ \underbrace{\left[\sqrt{\frac{1-\omega}{\omega}}\mathbf{Q}_{1}\bar{\boldsymbol{\xi}}_{1,d} - \sqrt{\frac{\omega}{1-\omega}}\mathbf{Q}_{2}\bar{\boldsymbol{\xi}}_{2,d}}_{\mathbf{H}} \right] \mathbf{H}^{T} \right) \Sigma \geq 0 \\ \end{aligned} \right\}$$

$$(58)$$

Thus, A-split consistency of the fusion result based on the proposed method has been proved. \blacksquare

Theorem 2: If the two estimate sources are both B-split consistent, then the fusion result by the proposed method shown in Algorithm 1 is B-split consistent for any $\omega \in [0, 1]$.

Proof 2: According to Theorem 1, if we have $\Sigma_d \geq E[\bar{\xi}_{1,d}\bar{\xi}_{1,d}^T]$ and $\Sigma_{2,d} \geq E[\bar{\xi}_{2,d}\bar{\xi}_{2,d}^T]$, then $\Sigma_d - E[\bar{\xi}_d\bar{\xi}_d^T] \geq 0$. We only need to examine the other condition for the B-split consistency. As the estimate sources are both B-split consistent, we have:

$$\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_{1,i} + \boldsymbol{\Sigma}_{1,d} \ge E(\bar{\boldsymbol{\xi}}_1 \bar{\boldsymbol{\xi}}_1^T), \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_{2,i} + \boldsymbol{\Sigma}_{2,d} \ge E(\bar{\boldsymbol{\xi}}_2 \bar{\boldsymbol{\xi}}_2^T)$$
(59)

which yields:

$$\begin{split} \boldsymbol{\Sigma} - E[\bar{\boldsymbol{\xi}}\bar{\boldsymbol{\xi}}^{T}] &= \boldsymbol{\Sigma} \left(\mathbf{Q}_{1}(\boldsymbol{\Sigma}_{1i} + \frac{1}{\omega}\boldsymbol{\Sigma}_{1,d})\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}(\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega}\boldsymbol{\Sigma}_{2,d})\mathbf{Q}_{2}^{T} \right) \boldsymbol{\Sigma} - E[\bar{\boldsymbol{\xi}}\bar{\boldsymbol{\xi}}^{T}] \\ &= \boldsymbol{\Sigma} \left(\mathbf{Q}_{1}(\boldsymbol{\Sigma}_{1,i} + \frac{1}{\omega}\boldsymbol{\Sigma}_{1,d} - E[\bar{\boldsymbol{\xi}}_{1,i}\bar{\boldsymbol{\xi}}_{1,i}^{T}] - E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}])\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}(\boldsymbol{\Sigma}_{2,i} + \frac{1}{1-\omega}\boldsymbol{\Sigma}_{2,d} - E[\bar{\boldsymbol{\xi}}_{2,i}\bar{\boldsymbol{\xi}}_{2,i}^{T}] - E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{2,d}^{T}])\mathbf{Q}_{2}^{T} \\ &- \mathbf{Q}_{1}E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{2,d}^{T}]\mathbf{Q}_{2}^{T} - \mathbf{Q}_{2}E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}]\mathbf{Q}_{1}^{T})\boldsymbol{\Sigma} \\ &\geq \boldsymbol{\Sigma} \left(\mathbf{Q}_{1}\frac{1-\omega}{\omega}E[\bar{\boldsymbol{\xi}}_{1,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}]\mathbf{Q}_{1}^{T} + \mathbf{Q}_{2}\frac{\omega}{1-\omega}E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{2,d}^{T}]\mathbf{Q}_{2}^{T} - \mathbf{Q}_{2}E[\bar{\boldsymbol{\xi}}_{2,d}\bar{\boldsymbol{\xi}}_{1,d}^{T}]\mathbf{Q}_{1}^{T})\boldsymbol{\Sigma} \\ &= \boldsymbol{\Sigma}E \left(\underbrace{ \left(\sqrt{\frac{1-\omega}{\omega}}\mathbf{Q}_{1}\bar{\boldsymbol{\xi}}_{1,d} - \sqrt{\frac{\omega}{1-\omega}}\mathbf{Q}_{2}\bar{\boldsymbol{\xi}}_{2,d} \right)}_{\mathbf{H}} \mathbf{H}^{T} \right) \boldsymbol{\Sigma} \geq 0 \\ \end{aligned} \right)$$
(60)

Thus, B-split consistency of the fusion result based on the proposed method has been proved. \blacksquare

V. RESULTS

This section presents the experimental results based on the proposed method. We also compare it with some state-of-theart methods, after which discussion on the performance of different methods is given.

A. Experiment Setup

The experiment consists of three parts, and the first part presents a numerical study where we assume two fully observed states of the robot pose are available. These two estimates are corrupted by noises which both have independent and dependent parts, respectively. The estimate is on SE(3)Lie group which has 3D rotation and 3D translation. In this experiment, we also test how the hyper-parameters affect the precision of the fusion. Two hyper-parameters are tested herein, *i.e.*, the order of the inverse Jacobian \mathcal{J}^{-1} , and the number of iterations for optimization of the final fusion result. This gives a hint on the parameters we use for the second experiment. Besides hyper-parameter testing, we also compare the proposed method with state-of-the-art methods.

The second experiment is the joint localization simulation of the multi-robot system which consists of several unmanned aerial vehicles (UAV). Different from the ground vehicle systems [29] whose pose estimate is implemented on SE(2), motion of UAV is on SE(3). In this experiment, the estimate sources come from three kinds of measurements: absolute positioning measurement (APM), motion measurement (MM), and relative positioning measurement (RPM). The APM gives the absolute position of the vehicles; orientations of the vehicles cannot be obtained by APM. Thus, it is a partial observation as stated in Section IV-C. The frequency of this measurement is usually not very high, *e.g.*, 10Hz. Some typical examples of APM are the global positioning system (GPS), beacon-based localization system for indoor applications. MM can give a fully 6D motion estimate between two timestamps. Robot pose can be predicted based on it as described in Section IV-C. The frequency of it is usually high, e.g., 100Hz. A typical example of MM is the inertial measurement unit (IMU). RPM gives relative pose between two vehicles. It can provide the 3D position or 6D fully measurement according to the method and hardware used. A typical example of RPM is ad hoc networks through V2V communication. The framework of pose estimation is that when a new MM arrives, the robots' poses are predicted based on the received data. When a new APM or RPM arrives, the robots' poses are updated according to the measurement update criteria shown in Section IV-C. Some state-of-the-art or baseline methods are compared in this experiment, *i.e.*, Kalman filtering-based fusion in the vector space [56], covariance intersection (CI)-based fusion in the vector space [30], split covariance intersection (SCI)-based fusion in the vector space [29], fusion ignoring the correlation on the Lie group [15].

The third experiment is joint localization on the real platform which consists of six micro intelligent vehicles (MicroIV). There are also APM, MM and RPM in the system. State to be estimated is on SE(2), *i.e.*, we only estimate x, y position and orientation (yaw) of the vehicles. We also compare results of the baselines as that in the UAV simulation. Details of this real-world experiment can be found in Section V-D.

B. Numerical Experiment

We show the effects of hyper-parameters in this subsection. Ground truth of the simulation is set as:

$$\mathbf{p}_{gt} = \exp(\boldsymbol{\xi}_{gt}^{\wedge}), \quad \boldsymbol{\xi}_{gt} = \begin{bmatrix} 15 & 30 & 0 & 0 & \frac{\pi}{4} \end{bmatrix}$$

There are two estimates of the ground truth pose, each is corrupted by both independent and dependent noise modeled on the Lie algebra. Covariance of the noise are:

$$\begin{split} \boldsymbol{\Sigma}_{1,i} &= \alpha \begin{bmatrix} 3 & 2 & 0.01 & 0.01 & 0.01 & 0.1 \end{bmatrix} \\ \boldsymbol{\Sigma}_{1,d} &= \alpha \begin{bmatrix} 5 & 3 & 2 & 0.1 & 0.2 & 0.1 \end{bmatrix} \\ \boldsymbol{\Sigma}_{2,i} &= \alpha \begin{bmatrix} 2 & 1 & 0.1 & 0.01 & 0.01 & 0.1 \end{bmatrix} \\ \boldsymbol{\Sigma}_{2,d} &= \alpha \begin{bmatrix} 5 & 5 & 2 & 0.1 & 0.1 & 0.2 \end{bmatrix} \\ \boldsymbol{\Sigma}_{1,2} &= \alpha \begin{bmatrix} 2 & 1 & 0.5 & 0 & 0 & 0.1 \end{bmatrix} \end{split}$$

where α is the coefficient, in the simulation we set $\alpha = 3$. The uncertainty is drawn from $\boldsymbol{\xi}_{1,i} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_{1,i}) \in \mathbb{R}^6$, $\boldsymbol{\xi}_{2,i} \sim \mathcal{N}(0, \boldsymbol{\xi}_{2,i}) \in \mathbb{R}^6$, and $[\boldsymbol{\xi}_{1,d}, \boldsymbol{\xi}_{2,d}] \sim \mathcal{N}(0, \boldsymbol{\Sigma}_d) \in \mathbb{R}^{12}$, where the dependent covariance matrix is

$$\mathbf{\Sigma}_{d} = egin{bmatrix} \mathbf{\Sigma}_{1,d} & \mathbf{\Sigma}_{1,2} \ \mathbf{\Sigma}_{1,2}^T & \mathbf{\Sigma}_{2,d} \end{bmatrix}$$

In the experiment, we assume the correlated covariance $\Sigma_{1,2}$ is unknown. The two measurements are obtained by $\mathbf{p}_1 = \exp((\boldsymbol{\xi}_{1,i} + \boldsymbol{\xi}_{1,d})^{\wedge})\mathbf{p}_{gt}$, $\mathbf{p}_2 = \exp((\boldsymbol{\xi}_{2,i} + \boldsymbol{\xi}_{2,d})^{\wedge})\mathbf{p}_{gt}$. Two kinds of metrics are exploited to measure the performance of different methods. The first is Mahalanobis distance between the observations and the estimate as shown in equation (28).



Fig. 3. Test results of the proposed split covariance intersection on Lie group in terms of different orders in Inverse Jacobian.

The second metric is the Root mean square (RMS) error, which measures difference between the estimates and ground truth over M trials as:

$$\epsilon = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \varepsilon_m^T \varepsilon_m}$$
(61)

where ε_m is the error of the *m*-th estimate in vector space:

$$\boldsymbol{\varepsilon}_m = \log(\mathbf{p}_{gt}\bar{\mathbf{p}}_m)^{\vee} \tag{62}$$

First, we test the effect of order kept in the inverse Jacobian as shown in equation (16). The result is presented in Fig. 3. Here, we show the Mahalanobis distance and RMS under different orders of the inverse Jacobian, and we select seven different cases to present, i.e., order from one to six, and that of infinity. The order of infinity is same as the analytical expression derived in [15], which keeps all the terms in the expansion. It will not be elaborated here, please refer to equation (100) in [15] for a detailed derivation. For every test of the order, we run 1000 trials then compute the average Mahalanobis distance and RMS, and iteration for every trial is 20. As can be seen, both the Mahalanobis distance and RMS decrease as the number of terms in the inverse Jabobian increases. But after the case that the inverse Jacobian has four terms, the decrease is very small and can be neglected. This is similar to that for the fully independent pose fusion in [15]. Thus, without loss of generality in the following experiments, we can set the term to be N = 4.

Next, we test the effect of iteration number as shown in Algorithm 1. The result is shown in Fig. 4. Here only we present the Mahalanobis distance which is the objective function to be optimized. From the figure, we can see that the Mahalanobis distance decreases as the iteration goes on. The number of terms in the inverse Jacobian also affects the minimization of Mahalanobis distance as can be seen in the right of Fig. 4. The inverse Jacobian that has four terms performs better than that has two terms, but the difference



Fig. 4. Test result of the proposed split covariance intersection on Lie group in terms of Malahanobis distance as the iteration goes on.

between four and infinite terms is marginal, which is consistent with that shown in Fig. 3. It can be seen that the Mahalanobis distance almost does not change after ten iterations, which means the algorithm can converge within ten iterations.

Finally, we compare the proposed method with state-of-theart methods. The result is shown in Fig. 5. Here, we present box plot of different methods. We implemented 100 trials, and the ground truth for each trial is not the same, where we add a small random perturbation. For every trial, we also conduct the fusion for 1000 running and compute the average error, here the coefficient $\alpha = 3$. From the figure, it can be seen that modeling the uncertainty on the Lie group can reduce the error compared with that modeled in the vector space. One thing that needs to note here is that CI or SCI in the vector space performs not well as that of KF which does not consider correlation. This is because the correlation is on the Lie group in reality, modeling it in the vector space brings larger error than ignoring it, though the estimated covariance maybe better than KF. It can be seen that errors by the proposed method and that of the independent fusion [15] are at the same level, even though error by the proposed method is smaller than the method in [15] marginally. Although the improvement in terms of accuracy is marginal in one fusion step, the benefit of the proposed method is consistency compared to the independent fusion [15], as proved in Section IV-D. As a result, it can affect the accuracy in a time evolving system (see Section V-C). Because the proposed method considers the correlation on the Lie group, and yields a more consistent estimate. Here we compare the estimated uncertainty errors between the proposed method and the independent fusion in [15]. The covariance error is defined as follows:

$$\epsilon = \sqrt{tr((\boldsymbol{\Sigma} - \boldsymbol{\Sigma}_{mc})^T (\boldsymbol{\Sigma} - \boldsymbol{\Sigma}_{mc}))}$$
(63)

where Σ_{mc} denotes the covariance of the Monte Carlo simulation, and we use the covariance of all the fused results as the Monte Carlo simulation covariance. For every noise



Fig. 5. Box plots of the errors by different methods for 100 trials.



Fig. 6. Error of estimated uncertainty as the noise scaling α increases.

scaling, we run the tests for 10000 trials. The result is shown in Fig. 6, and here we increase the noise scaling α gradually to see how the covariance error changes. Increasing α will increase the overall noise level, and the correlation increases at the same time. As can be seen in the figure, the covariance error of the independent fusion increases linearly as the noise scaling increases. The covariance error of the proposed method is almost not affected by the correlation when the noise is small. It has some perturbations because Σ_{mc} also has some small offset to the real uncertainty at some points, and there are some errors for the estimated covariance.

To give a viusal understanding of how the proposed methdo is more consistent compared with the fusion algorithm in [15] that does not consider the correlation, a 2D fusion (set the third-dimensional components to be zeros) experiment is conducted over 1000 trials. The result is shown in Fig.7, here



Fig. 7. Fusion results of two correlated measurements of the same pose. The 1-sigma ellipses of estimated covariance by the two methods are shown in solid curves. Dash curves are boundaries of the fused results.

we show the measurements before fusion, estimated pose after fusion by the two methods, and their corresponding 1-sigma covariance ellipse. This experiment is conducted over 1000 trials to see the real covariance after fusion. From the result, we can see that ignoring the correlation between measurements makes the estimated covariance over-confident, as the green ellipse is obviously smaller than the real uncertainty. In contrast, covariance by the proposed method is much closer to the real uncertainty. It will affect next-step estimation for a dynamic system propagating through time.

C. Joint Localization Simulation

In this subsection, we present the simulation results of joint localization for UAVs. The designed system consists of six UAVs, where each UAV is equipped with APM and MM, and there are RPM between them by V2V communication. The movement of all the UAVs are in 3D space, *i.e.*, elements on special Euclidean group SE(3). Ground truth trajectories of all the vehicles are displayed in Fig.8. All the UAVs fly sixty seconds, frequency of the MM is 100Hz, frequency of the APM, and RPM are all 10Hz. We assume all the sensors on the UAVs are the same, *i.e.*, they have the same covariance for the same type of sensors. The covariances for the APM and RPM are similar to that of the numerical studies, and the only difference is that $\alpha = 1$ here. Variance of MM is 1×10^{-5} for linear velocity and 1×10^{-6} for the angular velocity.

The pose estimation for every UAV is decentralized, *i.e.*, every UAV estimates its own pose and position according to the AMP, MM on itself and RPM through communication between other UAVs. We fuse the available information using various methods, where motion estimate between two timestamps by MM, absolute poses at some timestamps, and relative poses among UAVs at some timestamps are all utilized. To achieve this, we design different timestamp functions that make the estimates or observations are obtained by certain



Fig. 8. Ground truth trajectories of all the six UAVs. Starting points are denoted by dots.

frequencies. For a clearer comparison, we only show the position errors by dead reckoning, invariant extended Kalman filter, and the proposed method in Fig.9. As results in lots of existing literatures (e.g. [44], [47], [49]) have verified fusion on the Lie group outperforms the counterpart in the vector space. The quantitative results by EKF, CI, and SCI in the vector space are presented in Table I. From Fig.9, we can see that errors by dead reckoning always increase as time elapses due to IMU drift, because it does not have the correction scheme by observation as that of the filtering-based methods. Position errors by the proposed method are smaller than that by invariant extended Kalman filter in all the six cases. Because the underlying correlation among different observations are modeled in the proposed algorithm. Table I shows the average errors in position and orientation for 100 trials as only conducting once is unfair due to the randomness. For the position error, it can be seen that errors of the proposed method is much smaller than that of the other methods. Errors of invariant EKF are smaller than that of SCI in the vector space except UAV 1. This indicates that estimating the system on Lie group can reduce more errors than modelling the split correlation in most cases, but it is not always guaranteed. Position errors of KF and CI are larger than that of SCI in vector space, which means ignoring the correlation or igonring the indepdent the part of covariance make the estimation less accurate compared with modelling the split correlation even though in the vector space. Errors of KF and CI are at the same level though the result of KF is better than CI marginally. This phenomenon means that ignoring independent or dependent covariance both contribute error of the estimation. Only splitting the correlation can make the estimte result more accurate and robust. For the orientation error, the different is not that obvious while errors of the proposed method are still a little smaller than others marginally. From the table, we can see that errors by the proposed method reduce about 70% compared with that of the independent fusion. If the correlation increases, this effect



Fig. 9. Mean absolute errors in position and orientation of the six UAVs by various methods.

 TABLE I

 Comparison of the absolute mean errors in position and orientation by the proposed method and baselines for all the six UAVs

UAV number	SCI on Lie group (Proposed)		Invariant EKF		KF		CI		SCI	
	Position	Orientation	Position	Orientation	Position	Orientation	Position	Orientation	Position	Orientation
UAV1	1.04m	0.049rad	4.04m	0.057rad	5.32m	0.075rad	5.51m	0.076rad	3.81m	0.074rad
UAV2	1.23m	0.039rad	4.52m	0.043rad	5.33m	0.039rad	5.73m	0.039rad	4.91m	0.039rad
UAV3	1.24m	0.035rad	4.42m	0.040rad	5.91m	0.034rad	6.36m	0.032rad	4.68m	0.034rad
UAV4	1.27m	0.050rad	5.20m	0.051rad	9.05m	0.107rad	9.78m	0.108rad	7.53m	0.106rad
UAV5	1.29m	0.048rad	5.40m	0.050rad	11.56m	0.092rad	11.66m	0.093rad	7.56m	0.092rad
UAV6	1.27m	0.046rad	6.04m	0.048rad	10.39m	0.105rad	11.26m	0.106rad	8.80m	0.104rad

can enhance further.

D. Results on the real platform

To verify the effectiveness of the proposed method, we conduct a localization experiment on the robotic platform in the real world. The system consists of six MicroIVs, and each of them is equipped with the same hardware configuration as can be seen in Fig. 10. The different modules can provide different positioning information:

• APM: The absolute positioning is achieved by a motion capture system, which consists of 12 external cameras fixed on the ceiling. And the camera we use is Flex13 produced by Natural Point for this purpose. On each vehicle, refelective markers are installed to be detected by the cameras. The positioning result by this system is accurate enough to serve as the ground truth (millimeter-level accuracy). We add some extra noise to the position-

ing result from the camera capture system to mimic the absolute positioning measurements in practise.

- MM: Each vehicle is installed with an odometer to measure movement of the vehicle at a high frequency.
- RPM: The relative positioning is achived by V2V communication, which is via XBee-Pro module. Relative position between the vehicles are calculated by the capture camera system and broadcast by the communication module.

Size of the test field is $6.6m \times 6.4m \times 2.6m$, which can cover multiple transportation scenarios, *e.g.*, road, intersection, parking. Details of the hardware configuration can be found in our previous work [57]. The relative positioning between vehicles is also fused, where the position of one vehicle is deduced from the estimated positions of other vehicles and the relative positionings between them. Since the RPMs are fused mutually, the absolute position of one specific vehicle deduced from the relative positioning source is dependent on the pre-



(c) Testing filed

Fig. 10. The platform for the real-world fusion experiment. The exterior, hardware configuration and the testing filed are shown in (a), (b) and (c) respectively.

vious estimates of the this vehicle. For example, the position of vehicle *i* deduced from the relative positioning with vehicle j at time k denoted as $\mathbf{p}_k^{i,j}$ is a combination of the estimate of vehicle $j \mathbf{p}_k^j$ and the relative measurement $rp_k^{i,j}$, whereas the estimate of \mathbf{p}_k^j also has relation with \mathbf{p}_k^i or \mathbf{p}_{k-1}^i . So, $p_k^{i,j}$ is correlated with p_k^i . Moreover, the positioning deduced from RPM is also based on absolute positioning which is already fused for the estimation. In this sense, the RPMs are partially correlated, and the RPM is also partially correlated with the APM. Though the exact value of the correlation is not easy to set or estimate beforehand, our method can handle it by tuning the parameter ω . A comparison of the position error by the proposed method and other baselines is presented in Fig. 11. Here the error box plots of all the six MicroIVs by the five methods are presented. From the figure, it is clear to see that errors by the proposed method is smaller than that of invariant EKF, which is consistent with the results of simulation on UAVs shown in Fig. 9. The quantitative results for all the six MicroIVs in both position and orientation by the proposed method and baselines are presented in Table II. From the table, we can see that the proposed algorithm performs best among all the five methods. As our method identifies the correlation of the noise among the different measurements, and also does not ignore the independent part of the noise.

E. Observability analysis

As that in [47], [58], we also analyze the observability of the proposed method. It is pointed out in [47] that the invariant EKF can handle the observability automatically in their robot estimation problem. It avoid the inconsistency that exists in the EKF in the vector space, which is brought by the unobservable



Fig. 11. Box plots of the positional error by different methods

states. Moreover, it does not need to perform a nonlinear observability analysis for determining the unobservable states. Inspired by [47], we also analyze the observability of the system in this paper. The measurement matrix is the Jacobian of the measurement model, *i.e.*, $\mathbf{C} = \mathcal{J}_{\mathbf{H}}$. In the case of UAV simulation, we have the measurement of the abosulte pose, thus we have $\mathbf{C} = [\mathbf{0}, \mathbf{I}]$ according to the definition of equation (47). Similarly, the system matrix is the Jocabian of the system model, *i.e.*, $\mathbf{A} = \mathcal{J}_1$. If we only keeps the first order, then it becomes $\mathbf{A} = \mathbf{I} + \frac{1}{2} \boldsymbol{\xi}^{\wedge} = \begin{bmatrix} \mathbf{I} + \phi^{\wedge} & \rho^{\wedge} \\ \mathbf{0} & \mathbf{I} + \phi^{\wedge} \end{bmatrix}$ according to equation (13) and equation (18).

$$\mathcal{O} = \begin{vmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \end{vmatrix} = \begin{vmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{I} + \phi^{\wedge} \\ \mathbf{0} & (\mathbf{I} + \phi^{\wedge})^2 \\ \vdots & \vdots \end{vmatrix}$$
(64)

From equation (64) it is clear that the first three colomns are all zeros, which indicates the absolute rotations are unobservable. Herein, we also do not need to carry out the nonlinear observability which is computationally expensive. So, for the UAV simulation, the proposed method has the same unobservable states as the nonlinear system, which can solve the problem of systemic inconsistency caused by the linearization point. Derivation of the micro IV is similar to this but on SE(2) which is not elaborated here.

VI. CONCLUSION

In this paper, we present a fusion strategy where the states are on the Lie group, and the uncertainties are modeled on its Lie algebra. The possible correlation between the different estimate sources are considered, whilst the independent parts of the noises are kept. Moreover, the correlation between estimate sources is assumed to be unknown. This is achieved by extending the split covariance intersection in the vector space to the Lie group. We model it as an optimization problem, in which the objective function is the Mahalanobis distance relaxed by a more conservative covariance matrix. The fusion

TABLE II Comparison of the absolute mean errors in position and orientation by the proposed method and baselines for all the six MicroIVs

MicroIV number	SCI on Lie group (Proposed)		Invariant EKF		KF		CI		SCI	
	Position	Orientation	Position	Orientation	Position	Orientation	Position	Orientation	Position	Orientation
MicroIV1	0.064m	0.010rad	0.075m	0.014rad	0.159m	0.026rad	0.137m	0.019rad	0.089m	0.019rad
MicroIV2	0.065m	0.011rad	0.078m	0.014rad	0.133m	0.018rad	0.086m	0.017rad	0.084m	0.018rad
MicroIV3	0.063m	0.014rad	0.070m	0.014rad	0.112m	0.025rad	0.082m	0.022rad	0.069m	0.022rad
MicroIV4	0.045m	0.015rad	0.049m	0.022rad	0.129m	0.033rad	0.108m	0.028rad	0.051m	0.023rad
MicroIV5	0.046m	0.016rad	0.047m	0.018rad	0.136m	0.039rad	0.114m	0.033rad	0.045m	0.024rad
MicroIV6	0.066m	0.015rad	0.078m	0.019rad	0.114m	0.035rad	0.087m	0.029rad	0.076m	0.021rad

strategy for a partial observation case is also derived on the Lie group. Experimental results with joint localization on multi-UAV systems demonstrate the advantages of the proposed method compared with state-of-the-art and baseline methods. For future work, we plan to use the proposed method in formation, platooning, and other vehicle cooperation tasks.

APPENDIX

Lemma 1: Matrix inversion lemma [59], given matrices with proper dimension, the following equation holds:

$$\left(\mathbf{A} + \mathbf{B}\mathbf{D}^{-1}\mathbf{C}\right)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{B}\left(\mathbf{D} + \mathbf{C}\mathbf{A}^{-1}\mathbf{B}\right)^{-1}\mathbf{C}\mathbf{A}^{-1}$$
(65)

Theorem 3: Fusion for partial observation case presented in Section IV-C is an extension of Kalman filtering strategy on the Lie group, which has the form:

$$\Sigma_{1} = \Sigma_{k|k-1}^{i} + \frac{1}{\omega} \Sigma_{k|k-1}^{d}$$

$$\Sigma_{2} = \Sigma_{2i} + \frac{1}{1-\omega} \Sigma_{2d}$$

$$\mathbf{K} = \mathcal{J}_{1} \Sigma_{1} \mathcal{J}_{1}^{T} \mathcal{J}_{\mathbf{H}}^{T} (\mathcal{J}_{\mathbf{H}} \mathcal{J}_{1} \Sigma_{1} \mathcal{J}_{1}^{T} \mathcal{J}_{\mathbf{H}}^{T} + \mathcal{J}_{2} \Sigma_{2} \mathcal{J}_{2}^{T})^{-1} \quad (66)$$

$$\Sigma_{k|k} = (\mathbf{I} - \mathbf{K} \mathcal{J}_{\mathbf{H}}) \mathcal{J}_{1} \Sigma_{1} \mathcal{J}_{1}^{T}$$

$$\mathbf{s} = \mathcal{J}_{2} \boldsymbol{\xi}_{2} - \mathcal{J}_{\mathbf{H}} \mathcal{J}_{1} \boldsymbol{\xi}_{1}$$

$$\boldsymbol{\xi} = \boldsymbol{\xi}_{1} + \mathbf{K} \mathbf{s}$$

Proof 3: According to Lemma 1, we have:

$$\Sigma_{k|k} = \left(\underbrace{\mathcal{J}_{1}^{-T}\Sigma_{1}^{-1}\mathcal{J}_{1}^{-1}}_{\mathbf{A}} + \underbrace{\mathcal{J}_{\mathbf{H}}^{T}}_{\mathbf{B}}\underbrace{\mathcal{J}_{2}^{-T}\Sigma_{2}^{-1}\mathcal{J}_{2}^{-1}}_{\mathbf{D}^{-1}}\underbrace{\mathcal{J}_{\mathbf{H}}}_{\mathbf{C}}\right)^{-1}$$
$$= \mathcal{J}_{1}\Sigma_{1}\mathcal{J}_{1}^{T} - \mathcal{J}_{1}\Sigma_{1}\mathcal{J}_{1}^{T}\mathcal{J}_{\mathbf{H}}^{T}(\mathcal{J}_{2}\Sigma_{2}\mathcal{J}_{2}^{T} + \mathcal{J}_{\mathbf{H}}\mathcal{J}_{1}\Sigma_{1}\mathcal{J}_{1}^{T}\mathcal{J}_{\mathbf{H}}^{T})^{-1}$$
$$\cdot \mathcal{J}_{\mathbf{H}}\mathcal{J}_{1}\Sigma_{1}\mathcal{J}_{1}^{T} = (\mathbf{I} - \mathbf{K}\mathcal{J}_{\mathbf{H}})\mathcal{J}_{1}\Sigma_{1}\mathcal{J}_{1}^{T}$$
(67)

where we call K the Kalman gain. For the compensation, we

have:

$$\begin{split} \boldsymbol{\xi} &= -\boldsymbol{\Sigma}_{k|k} \left(\mathcal{J}_{1}^{-T} \boldsymbol{\Sigma}_{1}^{-1} \boldsymbol{\xi}_{1} + \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \boldsymbol{\xi}_{2} \right) \\ &= -(\mathbf{I} - \mathbf{K} \mathcal{J}_{\mathbf{H}}) \mathcal{J}_{1} \boldsymbol{\Sigma}_{1} \mathcal{J}_{1}^{T} \left(\mathcal{J}_{1}^{-T} \boldsymbol{\Sigma}_{1}^{-1} \boldsymbol{\xi}_{1} + \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \boldsymbol{\xi}_{2} \right) \\ &= -(\mathbf{I} - \mathbf{K} \mathcal{J}_{\mathbf{H}}) \mathcal{J}_{1} \boldsymbol{\xi}_{1} - (\mathbf{I} - \mathbf{K} \mathcal{J}_{\mathbf{H}}) \mathcal{J}_{1} \boldsymbol{\Sigma}_{1} \mathcal{J}_{1}^{T} \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \boldsymbol{\xi}_{2} \\ &= -(\mathbf{I} - \mathbf{K} \mathcal{J}_{\mathbf{H}}) \mathcal{J}_{1} \boldsymbol{\xi}_{1} - \left(\mathcal{J}_{1}^{-T} \boldsymbol{\Sigma}_{1}^{-1} \mathcal{J}_{1}^{-1} + \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \mathcal{J}_{2}^{-1} \\ &\times \mathcal{J}_{\mathbf{H}} \right)^{-1} \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \boldsymbol{\xi}_{2} = -(\mathbf{I} - \mathbf{K} \mathcal{J}_{\mathbf{H}}) \mathcal{J}_{1} \boldsymbol{\xi}_{1} - \mathcal{J}_{1} \boldsymbol{\Sigma}_{1} \mathcal{J}_{1}^{T} \\ &\times \mathcal{J}_{\mathbf{H}}^{T} \left(\mathcal{J}_{\mathbf{H}}^{T} + \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \boldsymbol{\Sigma}_{2}^{-1} \mathcal{J}_{2}^{-1} \mathcal{J}_{\mathbf{H}} \mathcal{J}_{1} \boldsymbol{\Sigma}_{1} \mathcal{J}_{1}^{T} \mathcal{J}_{\mathbf{H}}^{T} \right)^{-1} \mathcal{J}_{\mathbf{H}}^{T} \mathcal{J}_{2}^{-T} \\ &\times \boldsymbol{\Sigma}_{2}^{-1} \boldsymbol{\xi}_{2} = -(\mathbf{I} - \mathbf{K} \mathcal{J}_{\mathbf{H}}) \mathcal{J}_{1} \boldsymbol{\xi}_{1} - \mathcal{J}_{1} \boldsymbol{\Sigma}_{1} \mathcal{J}_{1}^{T} \mathcal{J}_{\mathbf{H}}^{T} \left(\mathcal{J}_{\mathbf{H}} \mathcal{J}_{1} \boldsymbol{\Sigma}_{1} \right) \\ &\times \mathcal{J}_{1}^{T} \mathcal{J}_{\mathbf{H}}^{T} + \mathcal{J}_{2} \boldsymbol{\Sigma}_{2} \mathcal{J}_{2}^{T} \right)^{-1} \mathcal{J}_{2} \boldsymbol{\xi}_{2} \\ &= -(\mathcal{J}_{1} \boldsymbol{\xi}_{1} + \mathbf{Ks}) \end{split}$$

where s is called innovation.

REFERENCES

- C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. J. Leonard, "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Trans. on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.
- [2] M. Kaess, A. Ranganathan, and F. Dellaert, "isam: Incremental smoothing and mapping," *IEEE Trans. on Robotics*, vol. 24, no. 6, pp. 1365– 1378, 2008.
- [3] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: a versatile and accurate monocular slam system," *IEEE Trans. on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.
- [4] K. Qiu, T. Qin, W. Gao, and S. Shen, "Tracking 3-d motion of dynamic objects using monocular visual-inertial sensing," *IEEE Trans.* on Robotics, vol. 35, no. 9, pp. 799–816, 2019.
- [5] N. Bore, J. Ekekrantz, P. Jensfelt, and J. Folkesson, "Detection and tracking of general movable objects in large three-dimensional maps," *IEEE Trans. on Robotics*, vol. 35, no. 1, pp. 231–247, 2018.
- [6] A. Parikh, R. Kamalapurkar, and W. E. Dixon, "Target tracking in the presence of intermittent measurements via motion model learning," *IEEE Trans. on Robotics*, vol. 34, no. 3, pp. 805–819, 2018.
- [7] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Information-theoretic model predictive control: Theory and applications to autonomous driving," *IEEE Trans. on Robotics*, vol. 34, no. 6, pp. 1603–1622, 2018.
- [8] A. Giusti, J. Malzahn, N. G. Tsagarakis, and M. Althoff, "On the combined inverse-dynamics/passivity-based control of elastic-joint robots," *IEEE Trans. on Robotics*, vol. 34, no. 6, pp. 1461–1471, 2018.
- [9] Y. Chen and D. J. Braun, "Hardware-in-the-loop iterative optimal feedback control without model-based future prediction," *IEEE Trans.* on Robotics, vol. 35, no. 9, pp. 1419–1434, 2019.
- [10] M. Kam, X. Zhu, and P. Kalata, "Sensor fusion for mobile robot navigation," *Proc. IEEE*, vol. 85, no. 1, pp. 108–119, 1997.
- [11] O. Wijk and H. I. Christensen, "Triangulation-based fusion of sonar data with application in robot pose tracking," *IEEE Trans. on Robotics Autom.*, vol. 16, no. 6, pp. 740–752, 2000.
- [12] X. S. Zhou and S. I. Roumeliotis, "Robot-to-robot relative pose estimation from range measurements," *IEEE Trans. on Robotics*, vol. 24, no. 6, pp. 1379–1393, 2008.

- [13] D. Schleicher, L. M. Bergasa, M. Ocaña, R. Barea, and M. E. López, "Real-time hierarchical outdoor slam based on stereovision and gps fusion," *IEEE Trans. on Intell. Transp. Syst.*, vol. 10, no. 3, pp. 440–452, 2009.
- [14] M. Pfanne, M. Chalon, F. Stulp, and A. Albu-Schäffer, "Fusing joint measurements and visual features for in-hand object pose estimation," *IEEE Robotics Autom. Lett.*, vol. 3, no. 4, pp. 3497–3504, 2018.
- [15] T. D. Barfoot and P. T. Furgale, "Associating uncertainty with threedimensional poses for use in estimation problems," *IEEE Trans. on Robotics*, vol. 30, no. 3, pp. 679–693, 2014.
- [16] J. G. Mangelson, M. Ghaffari, R. Vasudevan, and R. M. Eustice, "Characterizing the uncertainty of jointly distributed poses in the lie algebra," arXiv preprint arXiv:1906.07795, 2019.
- [17] F. Janabi-Sharifi and M. Marey, "A kalman-filter-based method for pose estimation in visual servoing," *IEEE Trans. on Robotics*, vol. 26, no. 5, pp. 939–947, 2010.
- [18] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. on Robotics and Autom.*, pp. 3565–3572, IEEE, 2007.
- [19] S. Holmes, G. Klein, and D. W. Murray, "A square root unscented kalman filter for visual monoslam," in *Proc. IEEE Int. Conf. on Robotics* and Autom., pp. 3710–3716, IEEE, 2008.
- [20] J. Liu, B.-g. Cai, and J. Wang, "Cooperative localization of connected vehicles: Integrating gnss with dsrc using a robust cubature kalman filter," *IEEE Trans. on Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2111– 2125, 2016.
- [21] U. Thomas, S. Molkenstruck, R. Iser, and F. M. Wahl, "Multi sensor fusion in robot assembly using particle filters," in *Proc. IEEE Int. Conf.* on Robotics and Autom., pp. 3837–3843, IEEE, 2007.
- [22] M. Turan, Y. Almalioglu, H. Gilbert, H. Araujo, T. Cemgil, and M. Sitti, "Endosensorfusion: Particle filtering-based multi-sensory data fusion with switching state-space model for endoscopic capsule robots," in *Proc. IEEE Int. Conf. on Robotics and Autom.*, pp. 1–8, IEEE, 2018.
- [23] H. Kim, B. Liu, C. Y. Goh, S. Lee, and H. Myung, "Robust vehicle localization using entropy-weighted particle filter-based data fusion of vertical and road intensity information for a large scale urban area," *IEEE Robotics Autom. Lett.*, vol. 2, no. 3, pp. 1518–1524, 2017.
- [24] L. Chen, P. O. Arambel, and R. K. Mehra, "Estimation under unknown correlation: Covariance intersection revisited," *IEEE Trans. on Autom. Control.*, vol. 47, no. 11, pp. 1879–1882, 2002.
- [25] S. J. Julier and J. K. Uhlmann, "Using covariance intersection for slam," *Robotics Auton. Syst.*, vol. 55, no. 1, pp. 3–20, 2007.
- [26] L. C. Carrillo-Arce, E. D. Nerurkar, J. L. Gordillo, and S. I. Roumeliotis, "Decentralized multi-robot cooperative localization using covariance intersection," in *Proc. IEEE/RSJ Int. Conf. on Intell. Robots and Syst.*, pp. 1412–1417, IEEE, 2013.
- [27] B. Noack, J. Sijs, M. Reinhardt, and U. D. Hanebeck, "Decentralized data fusion with inverse covariance intersection," *Autom.*, vol. 79, pp. 35–41, 2017.
- [28] H. Li and F. Nashashibi, "Cooperative multi-vehicle localization using split covariance intersection filter," *IEEE Intell. Transp. Syst. Mag.*, vol. 5, no. 2, pp. 33–44, 2013.
- [29] H. Li, F. Nashashibi, and M. Yang, "Split covariance intersection filter: Theory and its application to vehicle localization," *IEEE Trans. on Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1860–1871, 2013.
- [30] S. Julier and J. K. Uhlmann, "General decentralized data fusion with covariance intersection," in *Handbook of multisensor data fusion*, pp. 339– 364, CRC Press, 2017.
- [31] R. Kalman, "On the general theory of control systems," *IRE Trans. on Autom. Control.*, vol. 4, no. 3, pp. 110–110, 1959.
- [32] V. Sastry, "Decomposition of the extended kalman filter," *IEEE Trans.* on Autom. Control., vol. 16, no. 3, pp. 260–261, 1971.
- [33] E. A. Wan and R. Van Der Merwe, "The unscented kalman filter for nonlinear estimation," in *Proc. IEEE Adaptive Syst. for Signal Processing, Commun., and Control Symp.*, pp. 153–158, IEEE, 2000.
- [34] I. Arasaratnam and S. Haykin, "Cubature kalman filters," *IEEE Trans.* on Autom. Control., vol. 54, no. 6, pp. 1254–1269, 2009.
- [35] P. Del Moral, "Non-linear filtering: interacting particle resolution," *Markov Process. Relat. Fields*, vol. 2, no. 4, pp. 555–581, 1996.
- [36] J. D. Hol, T. B. Schon, and F. Gustafsson, "On resampling algorithms for particle filters," in *Proc. IEEE Nonlinear Statistical Signal Processing Workshop*, pp. 79–82, IEEE, 2006.
- [37] G. Grisettiyz, C. Stachniss, and W. Burgard, "Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling," in *Proc. IEEE Int. Conf. on Robotics and Autom.*, pp. 2432– 2437, IEEE, 2005.

- [38] M. Bolic, P. M. Djuric, and S. Hong, "Resampling algorithms and architectures for distributed particle filters," *IEEE Trans. on Signal Process.*, vol. 53, no. 7, pp. 2442–2450, 2005.
- [39] X. Fu and Y. Jia, "An improvement on resampling algorithm of particle filters," *IEEE Trans. on Signal Process.*, vol. 58, no. 10, pp. 5414–5420, 2010.
- [40] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *Proc. American Control Conf.*, vol. 4, pp. 2369–2373, IEEE, 1997.
- [41] T. D. Barfoot, State estimation for robotics. Cambridge University Press, 2017.
- [42] S. Bonnable, P. Martin, and E. Salaün, "Invariant extended kalman filter: theory and application to a velocity-aided attitude estimation problem," in *Proc. IEEE Conf. Decision and Control*, pp. 1297–1304, IEEE, 2009.
- [43] A. Barrau and S. Bonnabel, "Intrinsic filtering on lie groups with applications to attitude estimation," *IEEE Trans. on Autom. Control.*, vol. 60, no. 2, pp. 436–449, 2014.
- [44] A. Barrau and S. Bonnabel, "The invariant extended kalman filter as a stable observer," *IEEE Trans. on Autom. Control.*, vol. 62, no. 4, pp. 1797–1812, 2017.
- [45] A. Barrau and S. Bonnabel, "Extended kalman filtering with nonlinear equality constraints: a geometric approach," *IEEE Trans. on Autom. Control.*, vol. 65, no. 6, pp. 2325–2338, 2020.
- [46] R. Hartley, M. Ghaffari, J. Grizzle, and R. Eustice, "Contact-aided invariant extended kalman filtering for legged robot state estimation," in *Proc. Robotics: Science and Systems*, (Pittsburgh, Pennsylvania), June 2018.
- [47] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended kalman filtering for robot state estimation," *The Int. J. Robotics Res.*, vol. 39, no. 4, pp. 402–430, 2020.
- [48] T. Zhang, K. Wu, J. Song, S. Huang, and G. Dissanayake, "Convergence and consistency analysis for a 3-d invariant-ekf slam," *IEEE Robotics Autom. Lett.*, vol. 2, no. 2, pp. 733–740, 2017.
- [49] K. Wu, T. Zhang, D. Su, S. Huang, and G. Dissanayake, "An invariantekf vins algorithm for improving consistency," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pp. 1578–1585, IEEE, 2017.
- [50] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, "Continuousdiscrete extended kalman filter on matrix lie groups using concentrated gaussian distributions," *J. Math. Imaging Vis.*, vol. 51, no. 1, pp. 209– 228, 2015.
- [51] G. Bourmaud, R. Mégret, A. Giremus, and Y. Berthoumieu, "From intrinsic optimization to iterated extended kalman filtering on lie groups," *J. Math. Imaging Vis.*, vol. 55, no. 3, pp. 284–303, 2016.
- [52] M. Brossard, S. Bonnabel, and J.-P. Condomines, "Unscented kalman filtering on lie groups," in *proc. IEEE/RSJ Int. Conf. Intell. Robots and Syst.*, pp. 2485–2491, IEEE, 2017.
- [53] S. Heo and C. G. Park, "Consistent ekf-based visual-inertial odometry on matrix lie group," *IEEE Sensors J.*, vol. 18, no. 9, pp. 3780–3788, 2018.
- [54] R. Gilmore, "Baker-campbell-hausdorff formulas," J. Math. Phys., vol. 15, no. 12, pp. 2090–2092, 1974.
- [55] J. Solà, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," arXiv preprint arXiv:1812.01537, 2018.
- [56] S. S. Kia, S. Rounds, and S. Martinez, "Cooperative localization for mobile agents: A recursive decentralized algorithm based on kalmanfilter decoupling," *IEEE Control. Syst. Mag.*, vol. 36, no. 2, pp. 86–101, 2016.
- [57] W. Tang, M. Yang, Z. Lv, Q. Qian, T. Su, B. Wang, and C. Wang, "Microiv: a cooperative driving hardware simulation platform for cooperative-its," *IEEE Trans. on Veh. Technol.*, vol. 67, no. 10, pp. 9173– 9182, 2018.
- [58] Y. Yang and G. Huang, "Observability analysis of aided ins with heterogeneous features of points, lines, and planes," *IEEE Trans. on Robotics*, vol. 35, no. 6, pp. 1399–1418, 2019.
- [59] D. J. Tylavsky and G. R. Sohie, "Generalization of the matrix inversion lemma," *Proc. IEEE*, vol. 74, no. 7, pp. 1050–1052, 1986.