HAL open science

Invariant Smoothing for Localization: Including the IMU Biases

Paul Chauchat, Silvère Bonnabel, Axel Barrau

To cite this version:

Paul Chauchat, Silvère Bonnabel, Axel Barrau. Invariant Smoothing for Localization: Including the IMU Biases. 2024 IEEE 63th Annual Conference on Decision and Control (CDC), Dec 2024, Milan, Italy. hal-04212340v3

HAL Id: hal-04212340

https://hal.science/hal-04212340v3

Submitted on 3 Sep 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Invariant Smoothing for Localization: Including the IMU Biases

Paul Chauchat, Axel Barrau, Silvère Bonnabel

***Abstract*—In this article we investigate smoothing (i.e., optimisation-based) state estimation techniques for robot localization using an IMU aided by other localization sensors. We more particularly focus on Invariant Smoothing (IS), a variant based on the use of Lie groups within the invariant filtering framework. We bring the recently introduced Two Frames Group (TFG) to bear on the problem of Invariant Smoothing, to better take into account the IMU biases, as compared to the state-of-the-art in localization and navigation. Experiments based on the KITTI dataset show the proposed framework compares favorably to the state-of-the-art smoothing methods in terms of robustness in some challenging situations.**

I. INTRODUCTION

Lie group embeddings have become standard tools in navigation and mobile robotics over the last decade, see e.g., [14,31,37]. Extended Kalman filters or observers based on the classical groups $SE(3)$ and $SE(2)$ have been introduced and successfully used, see [6,7,26]. Novel Lie groups were then introduced and proved to come with similar properties to accomodate more complex dynamics, see [1,2]. They came with a new class of estimators, under the name of Invariant Extended Kalman Filter (IEKF) [2,3]. The theoretical properties include convergence guarantees [2], consistency properties [3], and have led to applications in various fields, [3,9,22,25,28,30,32,38] and in the industry [3]. Observer design also benefited from the introduction of those new groups, see e.g., [24,36], and they are also studied in their own right [33].

The success of the IEKF for models from inertial navigation relies on the introduction of the Lie group of double spatial direct isometries $SE_2(3)$, or extended poses, in [1,2], and [8] for a more thorough exposition. However, inertial measurement unit (IMU) biases could not be properly included into this Lie group structure, although they are unknown and need be estimated. Hence, they were treated as additional parameters leading to an "imperfect" invariant framework [1,32]. However, the recently introduced Two Frame Group (TFG) structure [4] additionally allows properly including the accelerometer bias.

If we turn to state estimation, smoothing has become the most popular technique in robotics for simultaneous localisation and mapping (SLAM) and visual odometry, thanks to its reducing the consequences of linearisation errors [20]. It is now often used in inertial navigation too [27,35]. Leveraging the framework of Invariant filtering for smoothing, a new estimation algorithm was recently proposed, namely Invariant Smoothing (IS) [12], see also [35], which delivers "physically consistent" estimates [13], although no convergence guarantee was derived. While recent SLAM and attitude estimation methods with convergence guarantees were proposed, they do not consider biased acceleration measurements [18,23,40].

This paper provides all the tools required to use the TFG structure in a new framework, namely smoothing, in order to tackle localization using biased IMUs. It completes [4], providing formulas for the adjoint map in particular. To evaluate the proposed framework, experiments were conducted based on data from the KITTI dataset [21], focusing in particular on the difficult problem of localization with no prior information, known as "in-flight alignment" problem in inertial navigation when the sensors used are the IMU and GNSS [10,29,39].

The paper is organized as follows. The main principles of Invariant Filtering and Smoothing are recalled in Section II. The considered application to localization is presented in Section III, as well as the presentation of the TFG and its properties for the considered problem. The proposed invariant smoother is detailed in Section IV, and the differences to the existing state-of-the-art smoothers are explained. Experimental results and comparisons are presented in Section V and show the proposed IS based on the TFG favorably compares to state-of-the-art smoothing schemes [17,20], and to the former "imperfect" IS based on $SE_2(3)$.

II. SMOOTHING ON LIE GROUPS

We first briefly recall the invariant filtering framework [2, 3]. The reader is referred to [31] for a general presentation. We consider a state $\chi \in G$, with G a Lie group of dimension q. Its Lie algebra $\mathfrak{g}$ is identified with $\mathbb{R}^q$. Thus we consider its exponential map to be defined as $\exp : \mathbb{R}^q \to G$. We denote its local inverse by log. We recall the notion of adjoint operator matrix of $\chi \in G$, $\mathbf{Ad}_\chi$, which satisfies

$$\forall \chi \in G, \xi \in \mathbb{R}^q,\ \chi^{-1}\exp(\xi)\chi = \exp(\mathbf{Ad}_\chi \xi) \tag{1}$$

Automorphisms are bijective maps $\phi : G \to G$ satisfying

$$\phi(\chi\eta) = \phi(\chi)\phi(\eta) \quad \text{for } \chi, \eta \in G. \tag{2}$$

The Lie group Lie algebra correspondance, see [5], ensures for *any* automorphism ϕ there is $\mathbf{M} \in \mathbb{R}^{q\times q}$ so that

$$\forall (\chi, \xi) \in G \times \mathbb{R}^q,\ \phi(\chi\exp(\xi)) = \phi(\chi)\exp(\mathbf{M}\xi), \tag{3}$$

which is closely related to the *log-linearity* property of [2]. The operator $\nu \mapsto \chi^{-1}\nu\chi$ is easily checked to be a group automorphism, and we see indeed from (1) that $\mathbf{M} = \mathbf{Ad}_\chi$. We define random variables on Lie groups through the exponential, following [3,7,8,14]. The probability distribution $\chi \sim \mathcal{N}_L(\bar{\chi}, \mathbf{P})$ for the random variable $\chi \in G$ is defined as

$$\chi = \bar{\chi}\exp(\xi),\ \xi \sim \mathcal{N}(\mathbf{0}, \mathbf{P}), \tag{4}$$

P. Chauchat is with Aix-Marseille Univ, CNRS, LIS, France (paul.chauchat@lis-lab.fr). A Barrau is with OFFROAD. S. Bonnabel is with MINES Paris PSL, PSL Research University, France (silvere.bonnabel@mines-paristech.fr).

where $\bar{\chi} \in G$, and $\mathbf{P}$ is a covariance matrix. In the following, we consider a discrete-time trajectory $(\chi_i)_{0 \leq i \leq n}$, denoted $(\chi_i)_i$ for brevity, of the following system

$$\chi_0 \sim \mathcal{N}_L(\bar{\chi}, \mathbf{P}_0), \quad \chi_{i+1} = f_i(\chi_i)\exp(\mathbf{w}_i), \quad y_k = h_k(\chi_{I_k}) + \mathbf{n}_k \tag{5}$$

where f_i, h_k are the dynamics and observation functions respectively, $\mathbf{P}_0 \in \mathbb{R}^{q \times q}$ the initial state error covariance, $\mathbf{w}_i, \mathbf{n}_k$ are white noises of covariance $\mathbf{Q}_i$ and $\mathbf{N}_k$, and χ_{I_k} denotes a subset of the states which are involved in the measurements at t_k.

A. Smoothing on Lie groups

We first briefly recall the Invariant Smoothing (IS) framework introduced in [12]. Departing from a system of the form (5), the goal of smoothing is to find

$$(\chi_i)_i^* = \underset{(\chi_i)_i}{\operatorname{argmax}}\, \mathbb{P}((\chi_i)_i | y_0, \ldots, y_n) \tag{6}$$

i.e., the maximum a posteriori (MAP) estimate of the trajectory. It is usually found through Gauss-Newton or Levenberg-Marquardt algorithms. First we devise a cost function associated to Problem (6) as the negative log likelihood

$$\mathscr{C} = -\log\big(\mathbb{P}((\chi_i)_i | y_0, \ldots, y_n)\big)$$

that we seek to minimize. Given a current guess of the trajectory's states, $(\hat{\chi}_i)_i$, the cost function $\mathscr{C}$ is linearised and then the resulting linear problem is solved exactly, yielding a novel estimate, and so on until convergence. Since χ_i belongs to a Lie group, linearisation in IS is carried out as

$$\forall 0 \leq i \leq n \quad \chi_i = \hat{\chi}_i \exp(\xi_i). \tag{7}$$

where $(\xi_i)_i$ are the searched parameters that minimize the linearized cost. IS linearises the cost $\mathscr{C}$ as [12]

$$\begin{aligned} \tilde{\mathscr{C}} = & \|\mathbf{p}_0 + \xi_0\|^2_{\widetilde{\mathbf{P}}_0} \\ & + \sum_i \|\hat{\mathbf{u}}_i - \mathbf{F}_i \xi_i + \xi_{i+1}\|^2_{\mathbf{Q}_i} + \sum_k \|\hat{\mathbf{n}}_k + \mathbf{H}_k \Xi\|^2_{\mathbf{N}_k} \end{aligned} \tag{8}$$

where we used the notation $\|\mathbf{Z}\|^2_{\mathbf{P}} = \mathbf{Z}^\top \mathbf{P}^{-1} \mathbf{Z}$, and where Ξ is the concatenation of $(\xi_i)_i$. (8) relies on the Baker-Campbell-Haussdorff formula [31] $\log(\exp(a)\exp(b)) = BCH(a,b)$. $\widetilde{\mathbf{P}}_0 = \mathbf{J}_0^{-1}\mathbf{P}_0\mathbf{J}_0^{-T}$, where $\mathbf{J}_0$ is the left Jacobian of the Lie group G [14,31], satisfying $BCH(\mathbf{p}_0, \xi) = \mathbf{p}_0 + \mathbf{J}_0\xi + o(\|\xi\|^2)$, $\mathbf{p}_0 = \log(\bar{\chi}_0^{-1}\hat{\chi}_0)$ with a prior $\bar{\chi}_0$, $\hat{\mathbf{u}}_i = \log(f_i(\hat{\chi}_i)^{-1}\hat{\chi}_{i+1})$, $\hat{\mathbf{n}}_k = \mathbf{y}_k - h_k(\hat{\chi}_{I_k})$, and $\mathbf{F}_i, \mathbf{H}_k$ are the (Lie group) Jacobians of f_i and h_k respectively. $\mathbf{H}_k$ was padded with zero blocks for the indices not contained in I_k, i.e. not involved in measurement y_k. The principle of smoothing algorithms is to solve the linearized problem (8) in closed form, and to update the trajectory substituting the optimal ξ_i in (7). The problem is then relinearised at this new estimate until convergence.

B. Group affine Dynamics and Invariant Smoothing

In the invariant framework, f_i is assumed to be group affine. These dynamics were introduced in continuous time in [2], and in discrete time in [5]. The main idea is that they extend the notion of linear dynamics (i.e. defined by affine maps) from vector spaces to Lie groups.

Definition 1. *Group affine dynamics are defined as*

$$\chi_{i+1} = f_i(\chi_i) = \Lambda_i \phi(\chi_i) \Upsilon_i. \tag{9}$$

with $\Lambda_i, \Upsilon_i \in G$, and ϕ an automorphism, i.e., satisfies (2).

Group affine dynamics include a large class of systems of engineering interest revolving around navigation and robotics, as shown in e.g. [2,5,28,35]. They come with the *log-linear property*, originally introduced and proved in [2].

Proposition 1 (from [5], discrete-time log-linear property)**.** *Let $\chi_{i+1} := f_i(\chi_i)$ for group affine dynamics* (9)*. We have*

$$f_i(\chi_i \exp(\xi)) = \chi_{i+1} \exp(\mathbf{F}_i \xi) \tag{10}$$

with $\mathbf{F}_i = \mathbf{Ad}_{\Upsilon_i^{-1}}\mathbf{M}$ a linear operator, and $\mathbf{M}$ from (3)*.*

Log-linearity ensures strong properties of invariant smoothing [13], since dynamics' linearized approximation in (8) becomes exact. Moreover, they are easily shown to possess a general preintegration property, see [5,8], extending that of [20]. However the IMU equations are group affine only when sensor biases are neglected, relying on the Lie group $SE_2(3)$ [1,2,35]. The recently introduced two-frames group (TFG) structure partially overcame this limitation, and suggested a new way to account for IMU biases in a principled manner, while unifying most group affine systems discovered so far.

III. Localization with an IMU using the TFG

This section introduces the localization problem to which we want to apply invariant smoothing using the TFG. We start by considering the accelerometer biases only.

A. Considered simplified problem

Consider a mobile body equipped with an inertial measurement unit (IMU) providing gyroscope and accelerometer measurements, and a GNSS (e.g., GPS) receiver providing position measurements $\mathbf{y}_i$. We neglect, for now, the gyroscope bias. A simple discretization of the continuous-time equations [20] yields the discrete-time dynamics:

$$\begin{cases} \mathbf{R}_{i+1} = \mathbf{R}_i \exp_m[\Delta t(\omega_i)_\times] \\ \mathbf{v}_{i+1} = \mathbf{v}_i + \Delta t\ (\mathbf{g} + \mathbf{R}_i(\mathbf{a}_i - \mathbf{b}_i^a)) \\ \mathbf{p}_{i+1} = \mathbf{p}_i + \Delta t\ \mathbf{v}_i \\ \mathbf{b}_{i+1}^a = \mathbf{b}_i^a \end{cases}, \tag{11}$$

with observation $\mathbf{y}_i = \mathbf{p}$. In the above Δt is a time step, $\mathbf{R}_i \in G = SO(3)$ denotes the transformation at time step i that maps the frame attached to the IMU (body) to the earth-fixed frame, $\mathbf{p}_i \in \mathbb{R}^3$ denotes the position of the body in space, $\mathbf{v}_i \in \mathbb{R}^3$ denotes its velocity, $\mathbf{g}$ is the earth gravity vector, $\mathbf{a}_i, \omega_i \in \mathbb{R}^3$ the accelerometer and gyroscope signals, $\mathbf{b}_i^a$ the accelerometer bias, $\exp_m$ denotes the matrix exponential, and for any vector $\beta \in \mathbb{R}^3$, the quantity $(\beta)_\times$ denotes the skew-symmetric matrix such that $(\beta)_\times \gamma = \beta \times \gamma$ for any $\gamma \in \mathbb{R}^3$.

B. Making the system group affine

To fall into the formalism of Section II, one needs to endow the state variables with a Lie group structure. Previously to

the TFG theory, it was known that in the absence of IMU biases, inertial navigation is group affine [2], but how to deal with the biases was unclear, although there have been recent propositions [19,34]. apart from the latter, most works implementing the invariant framework for inertial navigation - including the authors' - have treated IMU biases linearly, that is, completed the group structure of $SE_2(3)$, introduced in [2], with a linear structure regarding body variables (biases). The considered group composition law thus writes

$$\begin{pmatrix}\mathbf{R}_1\\ \mathbf{v}_1\\ \mathbf{p}_1\\ \mathbf{b}_1^a\end{pmatrix} \bullet \begin{pmatrix}\mathbf{R}_2\\ \mathbf{v}_2\\ \mathbf{p}_2\\ \mathbf{b}_2^a\end{pmatrix} = \begin{pmatrix}\mathbf{R}_1\mathbf{R}_2\\ \mathbf{v}_1+\mathbf{R}_1\mathbf{v}_2\\ \mathbf{p}_1+\mathbf{R}_1\mathbf{p}_2\\ \mathbf{b}_1^a+\mathbf{b}_2^a\end{pmatrix} \quad \text{(Imperfect IEKF law)} \tag{12}$$

This group law gave rise to the "Imperfect IEKF", see [1], leading to "imperfect invariant smoothing (IS)" [11,35], and led to practical successes [3,15,22,36,38]. However, none of these approaches, including [19,34], allowed the biased IMU equations to be group affine.

In this regard, the two-frames group structure proposed in [4] was a leap forward. Indeed, it turns out that this structure, which unifies a large number of estimation problems related to navigation, can be cast into the invariant filtering framework using the TFG group law. Following [4], the state space can be cast as a two-frame group (TFG). The TFG group composition law defines a way to combine the state variables which is defined as follows

$$\begin{pmatrix}\mathbf{R}_1\\ \mathbf{v}_1\\ \mathbf{p}_1\\ \mathbf{b}_1^a\end{pmatrix} \bullet \begin{pmatrix}\mathbf{R}_2\\ \mathbf{v}_2\\ \mathbf{p}_2\\ \mathbf{b}_2^a\end{pmatrix} = \begin{pmatrix}\mathbf{R}_1\mathbf{R}_2\\ \mathbf{v}_1+\mathbf{R}_1\mathbf{v}_2\\ \mathbf{p}_1+\mathbf{R}_1\mathbf{p}_2\\ \mathbf{b}_2^a+\mathbf{R}_2^\top\mathbf{b}_1^a\end{pmatrix} \quad \text{(TFG law)} \tag{13}$$

This defines the alternative two-frames group (TFG) law. Its identity element is $(\mathbf{I},\mathbf{0},\mathbf{0},\mathbf{0})$ and the inverse is given by: $\left(\mathbf{R},\mathbf{v},\mathbf{p},\mathbf{b}^a\right)^{-1} = \left(\mathbf{R}^\top, -\mathbf{R}^\top\mathbf{v}, -\mathbf{R}^\top\mathbf{p}, -\mathbf{R}\mathbf{b}^a\right)$.

C. Theoretical results

The TFG structure enables a more principled treatment of sensors' biases, since it leads to group affine properties.

Proposition 2 (from [4])**.** *The IMU equations in 3D with accelerometer bias are group affine, in the sense of the TFG group structure, whenever $\omega_i = \mathbf{0}$, that is, the orientation of the robot remains unchanged (but it may be arbitrary), while changes in acceleration and velocity are allowed.*

IV. Application to biased IMU based localization

Now that we have recalled the TFG structure, we would like to leverage it for smoothing based localization, which has never been done before. In practice, another bias needs to be considered when using an IMU, that of the gyroscope. By accounting for it explicitly, (11) then becomes

$$\begin{cases}\mathbf{R}_{i+1} = \mathbf{R}_i \exp_m\left[\Delta t(\omega_i - \mathbf{b}_i^\omega)_\times\right]\\ \mathbf{v}_{i+1} = \mathbf{v}_i + \Delta t\ \left(\mathbf{g} + \mathbf{R}_i\left(\mathbf{a}_i - \mathbf{b}_i^a\right)\right)\\ \mathbf{p}_{i+1} = \mathbf{p}_i + \Delta t\ \mathbf{v}_i\\ \mathbf{b}_{i+1}^a = \mathbf{b}_i^a, \quad \mathbf{b}_{i+1}^\omega = \mathbf{b}_i^\omega\end{cases}, \tag{14}$$

with $\mathbf{b}_i^\omega$ the gyroscope bias. The TFG law includes the latter as follows, where $\chi^{\text{acc}} = \left(\mathbf{R},\mathbf{v},\mathbf{p},\mathbf{b}^a\right)$:

$$\begin{pmatrix}\chi_1^{\text{acc}}\\ \mathbf{b}_1^\omega\end{pmatrix} \bullet \begin{pmatrix}\chi_2^{\text{acc}}\\ \mathbf{b}_2^\omega\end{pmatrix} = \begin{pmatrix}\chi_1^{\text{acc}} \bullet \chi_2^{\text{acc}}\\ \mathbf{b}_2^\omega + \mathbf{R}_2^\top\mathbf{b}_1^\omega\end{pmatrix} \quad \text{(TFG-gyro law)} \tag{15}$$

(14) is not group-affine in general since $\exp_m\left[\Delta t(\omega_i - \mathbf{b}^\omega)_\times\right]$ contains state element $\mathbf{b}_i^\omega$. This structure can still bring sensible improvements over existing methods [4].

A. Computing the Jacobians

The Jacobian of (11) can be directly retrieved from [4] when $\omega_i = \mathbf{0}$ and there is no gyro bias. It can be extended to the more difficult present setting as follows. Let $\Omega_i = \exp_m\left[\Delta t(\omega_i - \mathbf{b}_i^\omega)_\times\right]$ and $\mathbf{D}_i = D_{\exp_m}(\Delta t(\omega_i - \mathbf{b}_i^\omega))$ with $D_{\exp_m}$ the differential of $\exp_m$. Then we have:

$$\mathbf{F}_i = \begin{bmatrix} \Omega_i^\top - \Delta t\ \mathbf{D}_i(\mathbf{b}_i^\omega)_\times & & & -\Delta t\ \mathbf{D}_i\\ -\Omega_i^\top(\Delta t\ \mathbf{a}_i)_\times & \Omega_i^\top & -\Delta t\ \Omega_i^\top & \\ & \Delta t\ \Omega_i^\top & \Omega_i^\top & \\ (\mathbf{b}_i^a)_\times\left(\mathbf{I} - \Omega_i^\top + \Delta t\mathbf{D}_i(\mathbf{b}_i^\omega)_\times\right) & & \mathbf{I} & \\ (\mathbf{b}_i^\omega)_\times\left(\mathbf{I} - \Omega_i^\top + \Delta t\mathbf{D}_i(\mathbf{b}_i^\omega)_\times\right) & & & \mathbf{I}\end{bmatrix} \tag{16}$$

It coincides with the Jacobian of "imperfect IS" which relies on $SE_2(3) \times \mathbb{R}^3 \times \mathbb{R}^3$ (see [8,11]), except for the first block column, whose computation is detailed in Appendix A.

B. Differences with other parametrisations

Let us summarise the differences between IS based on the TFG structure, imperfect IS which relies on $SE_2(3) \times \mathbb{R}^3$, and the smoothing method implemented for NavState in GTSAM [17], which is a refinement of [20]. Although the considered residuals and their covariances are essentially the same, the main difference lies in the parametrisation of the state (i.e. the retraction) used to update the state variables at each optimization descent step. Obviously, the treatment of the bias $\mathbf{b} = (\mathbf{b}^a, \mathbf{b}^\omega)$ is a key difference, since all other parametrisations consider it linearly. As concerns the navigational part (attitude, velocity, position), only imperfect IS coincides with IS. More precisely, the retractions used in GTSAM [17] and imperfect IS are respectively

$$(\hat{\mathbf{R}},\hat{\mathbf{v}},\hat{\mathbf{x}},\mathbf{b}) \leftarrow (\hat{\mathbf{R}}\delta_R, \hat{\mathbf{v}} + \hat{\mathbf{R}}\delta_v, \hat{\mathbf{x}} + \hat{\mathbf{R}}\delta_x, \mathbf{b} + \delta_{\mathbf{b}}). \tag{17}$$

$$(\hat{\mathbf{R}},\hat{\mathbf{v}},\hat{\mathbf{x}},\mathbf{b}) \leftarrow ((\hat{\mathbf{R}},\hat{\mathbf{v}},\hat{\mathbf{x}})\exp_{SE_2(3)}(\delta_R,\delta_v,\delta_x), \mathbf{b} + \delta_{\mathbf{b}}). \tag{18}$$

(17) is linear by nature whereas imperfect IS (18) uses the exponential map of $SE_2(3)$, which offers a more appropriate nonlinear map. Note that (17) is a first-order approximation of (18). By constrast, the method proposed herein proposed the full exponential of the TFG, see (22) below.

The Jacobian for GTSAM (17) is similar to the one of imperfect IS, replacing $\Omega_i^\top$ and $\mathbf{a}_i$ with their estimated counterparts: $\hat{\Omega}_i^\top = \hat{\mathbf{R}}_{i+1}^\top\hat{\mathbf{R}}_i$ and $\Delta t\ \hat{\mathbf{a}}_i = \hat{\mathbf{R}}_i^\top(\hat{\mathbf{v}}_{i+1} - \hat{\mathbf{v}}_i - \Delta t\ \mathbf{g}) - \hat{\mathbf{b}}_i^a$. Though, it differs more starkly with that proposed herein.

The populated lower-left block of (16) has a slight computational impact on the computations, but only marginal with regard to the whole process.

V. Experimental Results

We evaluate the proposed smoothing method, based on the TFG, on the KITTI dataset [21], which contains raw

and synchronised IMU data, and a ground-truth. Smoothing estimates are implemented using sliding windows, where the oldest state is marginalised once a given size is reached. TFG is compared with the two other parametrisations (17), (18), based on a common smoothing implementation to ensure fair comparison.

When a robot is started, typically a terrestrial wheeled vehicle, it may have no information about its position and its orientation. However, in outdoors applications, GNSS provides a quick coarse estimate of the position, and the accelerometers that of the vertical direction. The yaw, on the other hand, is much harder to estimate, as no coarse estimate is directly available (magnetometers may be very unreliable due to metallic masses in vehicles and robots). To highlight the differences between the various smoothing methods, we consider the difficult and relevant localization problem where initial orientation is unknown and thus orientation error may be large (100 $^\circ$ standard deviation), and we focus on the transitory phase during which orientation is recovered (this is called the alignment problem in aerospace engineering, see e.g., [16,29,39]).

A. Implementation details

The IMU data is sampled at 100Hz. We simulate noisy position measurements at 1 Hz by adding white noise to the ground truth with $\sigma_y = 1m$, which is typically the accuracy of a GNSS sensor.

The estimators are compared using three different window sizes of 5, 10 and 15 instants for the state. The initial position is based on the ground truth, roll, pitch, velocity and biases are initialised at zero, and the yaw is randomly sampled. The initial uncertainty is given by $\sigma_p^0 = 1$m, $\sigma_v^0 = 10$m/s, $\sigma_R^0 = 100\ ^\circ, \sigma_{b^a}^0 = 0.06$m/s^2, $\sigma_{b^\omega}^0 = 0.07$rad/s. The process uncertainty uses $\sigma_a = 0.05$m/s^2, $\sigma_\omega = 0.01$rad/s, $\sigma_{b^a} = 0.002$m/s^2, $\sigma_{b^\omega} = 3.10^{-5}$rad/s. For each experiment, 50 Monte Carlo runs are carried out.

B. Results

When facing large initial errors, optimisation-based estimators easily fall into local minima. In this context, a relevant criterion to compare estimators is to check whether the difficult-to-estimate yaw error stays inside its believed 3σ envelope output by the estimator. Consistency is thus defined as the ability to convey a consistent (or conservative) estimate of the extent of uncertainty associated with the predictions.

Figure 1 shows the yaw errors over time for the three smoothing methods, highlighting trajectories which yield inconsistent estimates for each of the 50 random initializations, over a typical KITTI trajectory example. We observe a much larger ratio of orientation errors fail to stay inside the 3σ envelope (red curves vs blue curves) when using alternative methods to the TFG-based one proposed in the present paper. Besides, we note the 3σ envelopes are less dispersed over Monte-Carlo runs when using the TFG, which is consistent with them having less dependency on the estimates, see [4].

To assess the method over a larger number of experiments, Table I reports, for the various KITTI trajectories, window sizes and methods, the ratio of Monte Carlo runs which are consistent. As expected, larger windows ensure better convergence (but require more onboard computation capabilities) except when the ratio is already close to one. Moreover, smoothing based on the TFG proves much more robust, in that it systematically outperforms other methods when using the smallest window size (and hence being computationally efficient). It exhibits a special ability to avoid local minima.

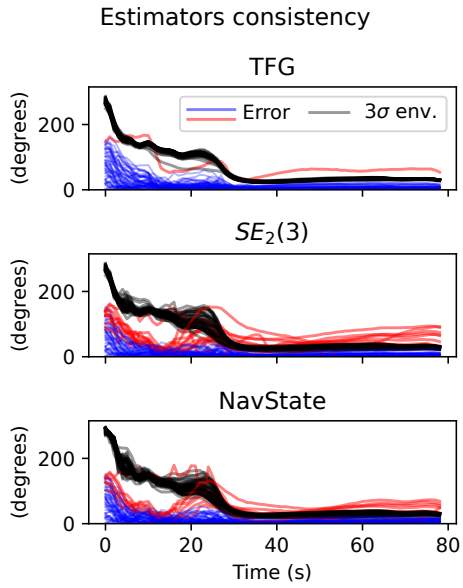

Fig. 1: Convergence and consistency comparison of smoothing based on three parametrisations: using the TFG (ours), $SE_2(3)$ [13], and NavState (GTSAM [17]), with a window size of 5. For 50 Monte Carlo runs over the KITTI trajectory 01, during the transitory phase after an initialization with random yaw, the yaw error and the 3σ envelope obtained from the estimated covariance are plotted over time. Consistent trajectories are in blue, inconsistent ones in red.

VI. Conclusion

In this paper we proposed to leverage the recently introduced Two-Frames Group structure in an Invariant Smoothing framework for localization, in a difficult setting where the window size is relatively low, and the inital error may be large. The necessary derivations, notably to retrieve the Jacobians, helped understanding the properties of the TFG. The proposed method showed increased robustness over the state-of-the-art in terms of convergence on real data. This supports the relevance of the TFG for more complex navigation problems, which opens up for a wide range of future work directions. It may be interesting in the future to investigate notably whether one could use the framework of GTSAM combined with this new kind of Lie-group based parameterization instead of the current on-manifold parametrization, as to date only improvements have been observed. Further investigations are thus desirable.

seq.	window	TFG	$SE_2(3)$	(17) (GTSAM)
01	5	**0.98**	0.78	0.86
	10	**0.98**	**0.98**	0.96
	15	**0.98**	**0.98**	0.84
05	5	**0.94**	0.7	0.7
	10	**0.96**	0.92	0.92
	15	0.96	**0.98**	0.96
06	5	**0.96**	0.68	0.66
	10	0.94	0.96	**1**
	15	0.94	0.98	**1**
07	5	**0.96**	0.94	**0.96**
	10	**0.98**	**0.98**	**0.98**
	15	0.98	**1**	**1**
08	5	**0.98**	0.84	0.86
	10	**0.98**	**0.98**	**0.98**
	15	**1**	0.98	0.94
09	5	**0.84**	0.68	0.7
	10	**0.94**	0.86	0.88
	15	0.96	**1**	0.96
10	5	**0.96**	0.88	0.88
	10	**1**	0.96	0.98
	15	**1**	0.96	0.94

TABLE I: Ratio of consistent trajectories, out of 50 random Monte Carlo initialization, for various smoothing methods on the KITTI dataset. Consistency is defined as the yaw error staying inside the 3-σ envelope.

Appendix A
Computation of (16)

Let $\chi = (R, \mathbf{v}, \mathbf{p}, \mathbf{b}^a, \mathbf{b}^\omega)$ be the state. For readability, we denote $\exp(\theta) = \exp_m[(\theta)_\times]$ on $SO(3)$, i.e. for $\theta \in \mathbb{R}^3$. Consider an update $\delta_\chi = \begin{pmatrix}\delta_\mathbf{R} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{pmatrix}$, with $\delta_\mathbf{R} = \exp(\xi^R)$. Then we have $\chi \bullet \delta_\chi = (R\delta_\mathbf{R}, \mathbf{v}, \mathbf{p}, \delta_\mathbf{R}^\top \mathbf{b}^a, \delta_\mathbf{R}^\top \mathbf{b}^\omega)$, and

$$f_i(\chi \bullet \delta_\chi) = \begin{pmatrix} \mathbf{R}\delta_\mathbf{R}\exp(\Delta t(\omega_i - \delta_\mathbf{R}^\top\ \mathbf{b}^\omega)) \\ \mathbf{v} + \Delta t\ (\mathbf{g} + \mathbf{R}\delta_\mathbf{R}(\mathbf{a}_i - \mathbf{b}^a)) \\ \mathbf{p} + \Delta t\ \mathbf{v} \\ \delta_\mathbf{R}^\top\ \mathbf{b}^a \\ \delta_\mathbf{R}^\top\ \mathbf{b}^\omega \end{pmatrix} \tag{19}$$

Let $\Omega_i = \exp(\Delta t(\omega_i - \mathbf{b}^\omega))$. Since $\delta_\mathbf{R}^\top \approx \mathbf{I} - (\xi^R)_\times$, we can approximate $\exp(\Delta t(\omega_i - \delta_\mathbf{R}^\top\ \mathbf{b}^\omega)) \approx \Omega_i \exp(-\Delta t \mathbf{D}_i(\xi^R)_\times \mathbf{b}^\omega)$ with $\mathbf{D}_i = D_{\exp}(\Delta t(\omega_i - \mathbf{b}^\omega))$ and $D_{\exp}$ the differential of exp on $SO(3)$. We know that $\mathbf{R}\delta_\mathbf{R}\Omega_i = \mathbf{R}\Omega_i \exp(\Omega_i^\top \xi^R)$, so $\mathbf{R}\delta_\mathbf{R}\exp(\Delta t(\omega_i - \delta_\mathbf{R}^\top\ \mathbf{b}^\omega)) \approx \mathbf{R}\Omega_i \exp(\Omega_i^\top \xi^R - \Delta t \mathbf{D}_i(\mathbf{b}^\omega)_\times \xi^R)$. Therefore, given (15), we can write

$$f_i(\chi \bullet \delta_\chi) \approx f_i(\chi) \bullet \begin{pmatrix} \exp(\Omega_i^\top \xi^R - \Delta t \mathbf{D}_i(\mathbf{b}^\omega)_\times \xi^R) \\ -\Delta t \Omega_i^\top (\xi^R)_\times \mathbf{a}_i \\ 0 \\ \delta_\mathbf{R}^\top \mathbf{b}^a - \exp(-\Omega_i^\top \xi^R + \Delta t \mathbf{D}_i(\mathbf{b}^\omega)_\times \xi^R)\mathbf{b}^a \\ \delta_\mathbf{R}^\top \mathbf{b}^\omega - \exp(-\Omega_i^\top \xi^R + \Delta t \mathbf{D}_i(\mathbf{b}^\omega)_\times \xi^R)\mathbf{b}^\omega \end{pmatrix} \tag{20}$$

The last terms usually do not appear when considering other parametrisations. Linearising the exponential, we get

$$\begin{aligned} &\left(\delta_\mathbf{R}^\top - \exp(-\Omega_i^\top \xi^R + \Delta t \mathbf{D}_i(\mathbf{b}^\omega)_\times \xi^R)\right)\mathbf{b}^a \\ &\approx \left(-(\xi^R)_\times + (\Omega^\top \xi^R)_\times - \Delta t(\mathbf{D}_i(\mathbf{b}^\omega)_\times \xi^R)_\times\right)\mathbf{b}^a \\ &= (\mathbf{b}^a)_\times \left(\mathbf{I} - \Omega^\top + \Delta t \mathbf{D}_i(\mathbf{b}^\omega)_\times\right)\xi^R \end{aligned} \tag{21}$$

and similarly for $\mathbf{b}^\omega$, thus recovering the first block column of (16). In particular, this highlights why $\Omega_i \neq \mathbf{I}$ breaks the group affine property of (11) in the 3D case (and without the gyro bias).

Appendix B
Useful Formulas for the TFG

We recall the results of [4] being useful herein.

A. Exponential and Logarithm

The exponential on the considered TFG is given by the following formula:

$$\exp_{TFG}\begin{pmatrix}\xi^R \\ \xi^v \\ \xi^p \\ \xi^{b^a} \\ \xi^{b^\omega}\end{pmatrix} = \begin{pmatrix}\exp(\xi^R) \\ \nu(\xi^R)\xi^v \\ \nu(\xi^R)\xi^p \\ \nu(-\xi^R)\xi^{b^a} \\ \nu(-\xi^R)\xi^{b^\omega}\end{pmatrix} \tag{22}$$

Where ν is given by:

$$\nu(\xi) = \mathbf{I} + \frac{1-\cos(||\xi||)}{||\xi||^2}(\xi)_\times + \frac{||\xi|| - \sin(||\xi||)}{||\xi||^3}(\xi)_\times^2$$

Its inverse, the logarithm, writes:

$$\log_{TFG}\begin{pmatrix}\mathbf{R} \\ \mathbf{v} \\ \mathbf{p} \\ \mathbf{b}^a \\ \mathbf{b}^\omega\end{pmatrix} = \begin{pmatrix}\xi^R \\ \nu(\xi^R)^{-1}\mathbf{v} \\ \nu(\xi^R)^{-1}\mathbf{p} \\ \nu(-\xi^R)^{-1}\mathbf{b}^a \\ \nu(-\xi^R)^{-1}\mathbf{b}^\omega\end{pmatrix} \tag{23}$$

B. Adjoint Matrices and left Jacobian

In order to apply (8) to the TFG, one needs to compute $\widetilde{\mathbf{P}}_0 = \mathbf{J}_0^{-1}\mathbf{P}_0\mathbf{J}_0^{-T}$, where $\mathbf{J}_0$ is the left Jacobian of the group computed at $\mathbf{p}_0$, which is given by the following sum $\mathbf{J}_0 = \sum_{j\geq 0}\frac{1}{(n+1)!}(\mathbf{ad}_{\mathbf{p_0}})^j$ [31]. Moreover $\mathbf{ad}_{\mathbf{p_0}}$ can be determined through identification, thanks to the fact that $\exp_m \mathbf{ad}_{\mathbf{p_0}} = \mathbf{Ad}_{\exp_{TFG}\mathbf{p}_0}$. Thus, what is left to do is compute $\mathbf{Ad}_\chi$ for an element of the TFG. In general it is given by the differential of $g \mapsto \chi \bullet g \bullet \chi^{-1}$ around the identity. Let $\chi = (\mathbf{R}, \mathbf{v}, \mathbf{p}, \mathbf{b}^a, \mathbf{b}^\omega)$ be a state. Then, following [4], we get

$$\mathbf{Ad}_\chi = \begin{bmatrix} \mathbf{R} & & & & \\ (\mathbf{v})_\times\mathbf{R} & \mathbf{R} & & & \\ (\mathbf{p})_\times\mathbf{R} & & \mathbf{R} & & \\ \mathbf{R}(\mathbf{b}^a)_\times & & & \mathbf{R} & \\ \mathbf{R}(\mathbf{b}^\omega)_\times & & & & \mathbf{R} \end{bmatrix} \tag{24}$$

Now, let $\chi = \exp_{TFG}(\xi)$, with $\xi = (\xi^R, \xi^p, \xi^p, \xi^{b^a}, \xi^{b^\omega})$. Using (22), and the facts that $\mathbf{R}\nu(-\xi^R) = \nu(\xi^R)$, and $\mathbf{R}(\mathbf{x})_\times = (\mathbf{R}\mathbf{x})_\times\mathbf{R}$, we get

$$\mathbf{Ad}_{\exp_{TFG}(\xi)} = \begin{bmatrix} \mathbf{R} & & & & \\ (\nu(\xi^R)\xi^v)_\times\mathbf{R} & \mathbf{R} & & & \\ (\nu(\xi^R)\xi^p)_\times\mathbf{R} & & \mathbf{R} & & \\ (\nu(\xi^R)\xi^{b^a})_\times\mathbf{R} & & & \mathbf{R} & \\ (\nu(\xi^R)\xi^{b^\omega})_\times\mathbf{R} & & & & \mathbf{R} \end{bmatrix} \tag{25}$$

Therefore, we can directly identify

$$\mathbf{ad}_{\xi} = \begin{bmatrix} (\xi^R)_\times & & & & \\ (\xi^v)_\times & (\xi^R)_\times & & & \\ (\xi^p)_\times & & (\xi^R)_\times & & \\ (\xi^{b^a})_\times & & & (\xi^R)_\times & \\ (\xi^{b^\omega})_\times & & & & (\xi^R)_\times \end{bmatrix} \tag{26}$$

REFERENCES

[1] Axel Barrau. *Non-linear state error based extended Kalman filters with applications to navigation*. PhD thesis, Mines Paristech, 2015.

[2] Axel Barrau and Silvère Bonnabel. The invariant extended kalman filter as a stable observer. *IEEE Transactions on Automatic Control*, 62(4):1797–1812, April 2017.

[3] Axel Barrau and Silvere Bonnabel. Invariant kalman filtering. *Annual Review of Control, Robotics, and Autonomous Systems*, 1:237–257, 2018.

[4] Axel Barrau and Silvere Bonnabel. The geometry of navigation problems. *IEEE Transactions on Automatic Control*, 68(2):689–704, 2022.

[5] Axel Barrau and Silvère Bonnabel. Linear observed systems on groups. *Systems and Control Letters*, 129:36 – 42, 2019.

[6] Silvère Bonnabel, Philippe Martin, and Pierre Rouchon. Symmetry-Preserving Observers. *IEEE Transactions on Automatic Control*, 53(11):2514–2526, December 2008.

[7] Guillaume Bourmaud, Rémi Mégret, Audrey Giremus, and Yannick Berthoumieu. Discrete extended Kalman filter on Lie groups. In *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European*, pages 1–5. IEEE, 2013.

[8] Martin Brossard, Axel Barrau, Paul Chauchat, and Silvère Bonnabel. Associating uncertainty to extended poses for on Lie group IMU preintegration with rotating Earth. *IEEE Transactions on Robotics*, 38(2):998–1015, 2021.

[9] D. Caruso, A. Eudes, M. Sanfourche, D. Vissière, and G. L. Besnerais. Magneto-visual-inertial dead-reckoning: Improving estimation consistency by invariance. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 7923–7930, 2019.

[10] Lubin Chang, Fangjun Qin, and Jiangning Xu. Strapdown inertial navigation system initial alignment based on group of double direct spatial isometries. *IEEE Sensors Journal*, 22(1):803–818, 2021.

[11] Paul Chauchat. *Smoothing algorithms for navigation, localisation and mapping based on high-grade inertial sensors*. Theses, Université Paris sciences et lettres, February 2020.

[12] Paul Chauchat, Axel Barrau, and Silvere Bonnabel. Invariant Smoothing on Lie Groups. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018*, Madrid, Spain, October 2018.

[13] Paul Chauchat, Silvére Bonnabel, and Axel Barrau. Invariant smoothing with low process noise. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 4758–4763, 2022.

[14] Gregory S Chirikjian. *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications*. Springer Science & Business Media, 2011.

[15] Mitchell R Cohen and James Richard Forbes. Navigation and control of unconventional vtol uavs in forward-flight with explicit wind velocity estimation. *IEEE Robotics and Automation Letters*, 5(2):1151–1158, 2020.

[16] Xiao Cui, Chunbo Mei, Yongyuan Qin, Gongmin Yan, and Qiangwen Fu. In-motion alignment for low-cost sins/gps under random misalignment angles. *Journal of Navigation*, 70(6):1224–1240, 2017.

[17] Frank Dellaert. The new imu factor. URL https://raw.githubusercontent.com/borglab/gtsam/develop/doc/ImuFactor.pdf. [Online; accessed 30-July-2024].

[18] Frank Dellaert, David M Rosen, Jing Wu, Robert Mahony, and Luca Carlone. Shonan rotation averaging: Global optimality by surfing so (p)ˆ n so (p) n. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 292–308. Springer, 2020.

[19] Alessandro Fornasier, Yonhon Ng, Robert Mahony, and Stephan Weiss. Equivariant filter design for inertial navigation systems with input measurement biases. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4333–4339. IEEE, 2022.

[20] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. On-manifold preintegration for real-time visual–inertial odometry. *IEEE Transactions on Robotics*, 33(1):1–21, Feb 2017.

[21] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013.

[22] Ross Hartley, Maani Ghaffari, Ryan M Eustice, and Jessy W Grizzle. Contact-aided invariant extended kalman filtering for robot state estimation. *The International Journal of Robotics Research*, 39(4):402–430, 2020.

[23] Seyed Hamed Hashemi and Jouni Mattila. Global asymptotic convergent observer for slam. *IEEE Access*, 10:122414–122422, 2022.

[24] Hashim A Hashim. Gps-denied navigation: Attitude, position, linear velocity, and gravity estimation with nonlinear stochastic observer. In *2021 American Control Conference (ACC)*, pages 1149–1154. IEEE, 2021.

[25] S. Heo and C. G. Park. Consistent ekf-based visual-inertial odometry on matrix lie group. *IEEE Sensors Journal*, 18(9):3780–3788, May 2018.

[26] Minh-Duc Hua, Guillaume Ducard, Tarek Hamel, Robert Mahony, and Konrad Rudin. Implementation of a nonlinear attitude estimator for aerial robotic vehicles. *Control Systems Technology, IEEE Transactions on*, 22(1):201–213, 2014.

[27] V. Indelman, S. Williams, Michael Kaess, and F. Dellaert. Information fusion in navigation systems via factor graph based incremental smoothing. *Journal of Robotics and Autonomous Systems, RAS*, 61(8):721–738, August 2013.

[28] Robert E. Mahony and Tarek Hamel. A geometric nonlinear observer for simultaneous localisation and mapping. In *56th IEEE Annual Conference on Decision and Control, CDC 2017, Melbourne, Australia, December 12-15, 2017*, pages 2408–2415, 2017.

[29] Wei Ouyang and Yuanxin Wu. Optimization-based strapdown attitude alignment for high-accuracy systems: Covariance analysis with applications. *IEEE Transactions on Aerospace and Electronic Systems*, 2022.

[30] Natalia Pavlasek, Alex Walsh, and James Richard Forbes. Invariant extended kalman filtering using two position receivers for extended pose estimation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5582–5588. IEEE, 2021.

[31] Joan Sola, Jeremie Deray, and Dinesh Atchuthan. A micro lie theory for state estimation in robotics. *arXiv preprint arXiv:1812.01537*, 2018.

[32] Niels van Der Laan, Mitchell Cohen, Jonathan Arsenault, and James Richard Forbes. The invariant rauch-tung-striebel smoother. *IEEE Robotics and Automation Letters*, 5(4):5067–5074, 2020.

[33] Pieter van Goor and Robert Mahony. Autonomous error and constructive observer design for group affine systems. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 4730–4737. IEEE, 2021.

[34] Pieter van Goor and Robert Mahony. Eqvio: An equivariant filter for visual-inertial odometry. *IEEE Transactions on Robotics*, 39(5):3567–3585, 2023.

[35] A. Walsh, J. Arsenault, and J. R. Forbes. Invariant sliding window filtering for attitude and bias estimation. In *2019 American Control Conference (ACC)*, pages 3161–3166, July 2019.

[36] Miaomiao Wang and Abdelhamid Tayebi. Hybrid nonlinear observers for inertial navigation using landmark measurements. *IEEE Transactions on Automatic Control*, 65(12):5173–5188, 2020.

[37] Kevin C. Wolfe, Michael Mashner, and Gregory S. Chirikjian. Bayesian fusion on Lie groups. *Journal of Algebraic Statistics*, 2(1):75–97, 2011.

[38] K. Wu, T. Zhang, D. Su, S. Huang, and G. Dissanayake. An invariant-ekf vins algorithm for improving consistency. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1578–1585, Sep. 2017.

[39] Yuanxin Wu and Xianfei Pan. Velocity/position integration formula part i: Application to in-flight coarse alignment. *IEEE Transactions on Aerospace and Electronic Systems*, 49(2):1006–1023, APRIL 2013.

[40] Bowen Yi, Chi Jin, Lei Wang, Guodong Shi, and Ian R. Manchester. An almost globally convergent observer for visual slam without persistent excitation. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 5441–5446, 2021.