Invariant Smoothing on Lie Groups
Paul Chauchat, Axel Barrau, Silvere Bonnabel

To cite this version:

HAL Id: hal-01725847
https://hal.archives-ouvertes.fr/hal-01725847
Submitted on 7 Mar 2018

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 on Lie Groups

Paul CHAUCHAT, Axel BARRAU and Silvère BONNABEL

Abstract—In this paper we propose a (non-linear) smoothing algorithm for group-affine observation systems, a recently introduced class of estimation problems on Lie groups that bear a particular structure. As most non-linear smoothing methods, the proposed algorithm is based on a maximum a posteriori estimator, determined by optimization. But owing to the specific properties of the considered class of problems, the involved linearizations are proved to have a form of independence with respect to the current estimates, leveraged to avoid (partially or sometimes totally) the need to relinearize. The method is validated on a robot localization example, both in simulations and on real experimental data.

I. INTRODUCTION

Statistical state estimation based on sensor fusion has played a major role in localization for the last decades. Most of the proposed solutions revolved around filtering, especially under the framework of the standard Extended Kalman Filter (EKF) [20]. Many alternative filters have been introduced to overcome the inherent limitations of the EKF, such as particle filters or the Unscented Kalman Filter. In the robotics field, and especially in the Simultaneous Localization and Mapping (SLAM) community, the smoothing approach, based on the Gauss-Newton (GN) algorithm, was proposed in order to take advantage of the sparsity of the involved matrix, and reduce the consequences of wrong linearization points [18].

It had a tremendous impact and led to many of the state-of-the-art algorithms for SLAM and visual odometry systems [22,28,32], to cite a few. It was more recently applied to GPS aided inertial navigation, showing promising results [27,36]. In the meantime, the advantages of the Lie group structure for probabilistic robotics, and especially that of the configuration spaces $SE(d)$, have been widely recognized [4,16,17,19,26,35]. This led to new non-linear filtering algorithms [7,10,13] which showed remarkable properties [5,7], and to geometrical methods at the core of all the recent solvers for smoothing problems involving relative poses [22,32].

The smoothing technique proposed in this paper builds upon the invariant filtering framework, whose main results can be found in [7,10]. An overview of the theory and its applications is made in [9], and a focus on the case of SLAM in [5]. Similar results were achieved with observers [30]. A more tutorial approach to the theory, along with some new results, can also be found in the recent paper [8] dedicated to the discrete-time case. In the present paper, we devise a smoothing framework adapted to robot localization, where the dependence of the whole information matrix of the system with respect to the linearization points is reduced, or even removed under mild conditions. This brings immediate benefits: 1) The number of iterations to convergence is reduced

2) The consistency of the smoother is improved. As usual with estimation methods derived from the theory of invariant filtering this comes at a price: the framework is adapted to a specific class of systems called group-affine observation systems defined in [7] (the equations of inertial navigation [7], as well as SLAM [5], fall within this framework).

A. Links and differences with the previous literature

Smoothing on manifolds is well established, and a thorough study of a Gauss-Newton algorithm on Riemannian manifolds can be found in [1]. In the SLAM community, a lot of work has been dedicated to studying the non-Euclidean structure of the state-space [3,16,17]. Kuemmerle et al. introduced pose-graph algorithms for general manifolds [24,29]. Most of the recent work on SLAM focused on the 2D and 3D Special Euclidean groups $SE(2)$ and $SE(3)$ which are used to represent position-orientation states, especially in the pose-SLAM framework. The majority uses the group structure to define the cost function, and we then recover a group synchronization problem [11]. To solve it, most methods revolve around using the manifold structure of the state-space and custom retractions [21] or using the matrix structure to relax the problem [31]–[33]. The recent work of Bourmaud et al [12,14] is, to the authors’ knowledge, the only one which fully relies on the Lie group structure for the estimate. In a way, the proposed method can be seen as an instance of this algorithm, but where the choice of the uncertainty representation is commanded by the system instead of being free, and an additional trick regarding the observation factors provides new properties.

These approaches are focused on leveraging the intrinsic tools of a given structure of the state space (Riemannian manifold or Lie group). For instance in [14], the proposed algorithm only required the state to belong to a Lie group, but no assumption is made regarding the dynamics and observation functions. In the present paper, the equations of the system are assumed to have specific properties with respect to the chosen group structure. This reduces the application field, but ensures in turn striking properties.

B. Organization of the paper

The paper is divided as follows. Section II contains mathematical preliminaries. Section III recalls the standard framework of smoothing, both in the vector and manifold...
settings. Section IV presents our smoothing algorithm for group-affine systems on Lie groups, which we call invariant smoothing, and which is the main contribution of the paper. Section V illustrates the theory with both a simulated and an experimental 2D robot localization problems, showing the advantage of the proposed method over classical smoothing algorithms, and also over the recently developed (invariant) Kalman filter based UKF-LG and IEKF (Invariant Extended Kalman Filter) [7,15].

II. MATHEMATICAL PRELIMINARIES

A. Lie Groups

In this section we recall the definitions and basic properties of matrix Lie groups, Lie algebra and random variables on Lie groups. A matrix Lie group $G \subset \mathbb{R}^{N \times N}$ is a set of square invertible matrices that is a group, i.e., the following properties hold:

$$I_N \in G; \forall \chi \in G, \chi^{-1} \in G; \forall \chi_1, \chi_2 \in G, \chi_1 \chi_2 \in G \quad (1)$$

Locally about the identity matrix $I_N$, the group $G$ can be identified with a vector space $\mathbb{R}^g$ using the matrix exponential map $\exp_m(\cdot)$, where $g = \dim G$. Indeed, to any $\xi \in \mathbb{R}^g$ one can associate a matrix $\xi^\wedge$ of the tangent space of $G$ at $I_N$, called the Lie algebra $\mathfrak{g}$. We then define the exponential map $\exp : \mathbb{R}^g \to G$ for Lie groups as

$$\exp(\xi) = \exp_m(\xi^\wedge), \quad (2)$$

Locally, it is a bijection, and one can define the Lie logarirthm map $\log : G \to \mathbb{R}^g$ as its inverse: $\log(\exp(\xi)) = \xi$.

As Lie groups are not necessarily commutative, in general we have $\exp(x) \exp(y) \neq \exp(x+y)$ for $x, y \in \mathfrak{g}$. However, the vector $z$ such that $\exp(z) \exp(y) = \exp(z+y)$ satisfies the Baker-Campbell-Hausdorff (BCH) formula [23]

$$z = \text{BCH}(x, y) = x + y + r(x, y),$$

where $r(x, y)$ contains higher order terms.

A last classical tool is the (inner) automorphism $\Psi_a \in \text{Aut}(G)$ defined for each $a \in G$ as $\Psi_a : g \mapsto aga^{-1}$. Its differential at the identity element $I_d$ of $G$ is called adjoint operator and denoted by $\text{Ad}_a : \mathfrak{g} \to \mathfrak{g}$. It satisfies:

$$\forall a \in G, u \in \mathfrak{g}, \ a \exp(u)a^{-1} = \exp(\text{Ad}_a u).$$

B. Uncertainties on Lie Groups

To define random variables on Lie groups, we cannot apply the usual approach of additive noise for $\chi_1, \chi_2 \in G$ as $G$ is not a vector space, i.e., generally $\chi_1 + \chi_2 \notin G$ does not hold. In contrast, we adopt the framework of [4], see also [6], which is slightly different from the pioneering approach of [16,17]. Indeed, we define the probability distribution $\chi \sim \mathcal{N}_L(\bar{\chi}, \mathbf{P})$ for the random variable $\chi \in G$ as

$$\chi = \bar{\chi}\exp(\xi), \quad \xi \sim \mathcal{N}(0, \mathbf{P}), \quad (3)$$

where $\mathcal{N}(\cdot, \cdot)$ is the classical Gaussian distribution in Euclidean space and $\mathbf{P} \in \mathbb{R}^{g \times g}$ is a covariance matrix. In the sequel, we will refer to (3) as the left-invariant Gaussian distribution on $G$, owing to the fact that the discrepancy $\bar{\chi}^{-1}\chi$ which is invariant to left multiplications $(\bar{x}, x) \mapsto (\Gamma \bar{x}, \Gamma x)$, is the exponential of a Gaussian $\bar{\xi}$. In (3), the noise-free quantity $\bar{x}$ is viewed as the mean, and the dispersion arises through left multiplication with the exponential of a Gaussian random variable. Similarly, the distribution $\chi \sim \mathcal{N}_L(\bar{\chi}, \mathbf{P})$ can be defined through right multiplication as

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

We stress that we have defined these probability density functions directly in the vector space $\mathbb{R}^g$ such that both $\mathcal{N}_L(\cdot, \cdot)$ and $\mathcal{N}_R(\cdot, \cdot)$ are not Gaussian distributions.

III. CLASSICAL SMOOTHING

A. Maximum a posteriori (MAP) estimate

Contrary to filtering approaches, which only consider the latest state of the system, smoothing methods aim at recovering the maximum a posteriori estimate (MAP) of the full trajectory given a set of measurements $Z = (y_1, \ldots, y_n)$:

$$\chi_0^*, \ldots, \chi_n^* = \arg\max_{(\chi_i)_{i \leq n}} p((\chi_i)_{i \leq n} | Z) \quad (5)$$

Under a markovian assumption on the trajectory and conditional independence of the measurements, and given a prior on the initial state $\chi_0$, this becomes

$$\chi_0^*, \ldots, \chi_n^* = \arg\max_{(\chi_i)_{i \leq n}} p(\chi_0) \prod_{i} p(\chi_{i+1} | \chi_i) \prod_{k} p(y_k | (\chi_i)_{i \leq n}) \quad (6)$$

In this paper, we consider partial measurements of a single state, i.e. we have for some instant $t_k$:

$$\forall k, \ p(y_k | (\chi_i)_{i \leq n}) = p(y_k | \chi_{t_k}) \quad (7)$$

B. Gaussian Smoothing on vector spaces

In the general case, (6) represents a difficult high dimensional optimization problem. However, if the noises are supposed to be additive and Gaussian, this boils down to a non-linear least-squares problem [3].

Consider the following prior, dynamics, and measurement model:

$$\chi_0 = \bar{\chi}_0 + z_0 \quad \chi_{i+1} = f_i(\chi_i) + w_i \quad y_k = h(\chi_k) + v_k \quad (8)$$

where $z_0$, $w_i$ and $v_k$ are gaussian white noises of respective covariances $P_0$, $Q_i$ and $R_k$. Then the MAP estimates satisfies:

$$\chi_0^*, \ldots, \chi_n^* = \arg\min_{(\chi_i)_{i \leq n}} \frac{1}{2} \|\chi_0 - \bar{\chi}_0\|_P^2 + \sum_j \|f_j(\chi) - \chi_{t+1}\|_Q^2 + \sum_k \|h(\chi_k) - y_k\|_R^2 \quad (9)$$

where we used the Mahalanobis norm $\|e\|^2 = e^T\Sigma^{-1}e$.

Solving (9) is done through iterative algorithms involving sequential linearizations, such as Gauss-Newton (GN) [18, 36].

For $i \leq n$, denoting by $\delta \chi_i$ the current estimates and letting $\delta \chi_i$ be the search parameter for the current iteration, each iteration consists in solving the following least-squares problem, based on a first-order Taylor expansion of (9)

$$\begin{equation}
(\delta \chi_i)_{i \leq n} = \arg\min_{(\delta \chi_i)_{i \leq n}} \frac{1}{2} \|p_0 + \delta \chi_0\|_P^2 + \sum_j \|a_j + F_j \delta \chi_i - \delta \chi_{t+1}\|_Q^2 + \sum_k \|c_k + H_k \delta \chi_i\|_R_k^2
\end{equation} \quad (10)$$
where \( p_0 = \hat{x}_0 - \tilde{x}_0, \ a_i = f_i(\hat{x}_k) - \tilde{x}_{i+1}, \ c_k = h(\tilde{x}_k) - y_k. \) \( F_i \) and \( H_k \) are the Jacobians of \( f_i \) and \( h_k \) evaluated at \( \hat{x}_i \) and \( \tilde{x}_k \) respectively. Each of the terms of the sum are referred to as factors. This name comes from the link between the MAP estimation problem and graphical models, in particular factor graphs [18], which lead to powerful solvers [28,29,34] still used in many recent applications [22].

In this paper, we focus on the non-linear least-squares problem which is to be solved, and argue that a better parametrization taking into account the geometry of the system will lead to better convergence properties, regardless of the solver used.

C. Smoothing on manifolds

A large body of literature was dedicated to smoothing of systems which do not live in vector spaces but on a manifold \( \mathcal{M} [21,24]. \) The main idea is to replace the \( - \) and \(+\) operators of the vector space by adapted ones \( \boxplus, \boxminus, \) which highlight the fact that \( \mathcal{M} \) is not a vector space and that linearization takes place on its tangent space \( T.\mathcal{M} \), where the \( \delta \chi_i \) should live. Thus, these operators are defined such that \( \boxplus : \mathcal{M} \times \mathcal{M} \to T.\mathcal{M} \) and \( \boxminus : \mathcal{M} \times T.\mathcal{M} \to \mathcal{M}. \)

Therefore, a smoothing algorithm can be divided into two main parts, the non-linear and the linear ones. The former dictates how the jacobians and errors are computed, and how the updates are handled, while the latter is the solver, only concerned with linear algebra, as illustrated on Figure 1. In this work, we focus on the modeling part, and propose a new framework which can be used with any existing solver.

Fig. 1: Schematic representation of the two main components of a smoothing algorithm, the non linear model of the system which handles the transitions from the manifold to the tangent space and back, and the solver which inverts the obtained linear system.

When the manifold is a Lie group, the group multiplication and its intrinsic operators \( \log \) and \( \exp \) naturally yield a family of such operators, through the following definitions:

\[
\boxplus_{L}(\chi, \xi) = \exp(\xi) \chi \quad \boxplus_{R}(\chi, \xi) = \chi \exp(\xi) \\
\boxminus_{L}(\chi_1, \chi_2) = \log(\chi_1 \chi_2^{-1}) \quad \boxminus_{R}(\chi_1, \chi_2) = \log(\chi_2^{-1} \chi_1)
\]

Note that, these operators were used in the definition of Section II. Smoothing methods on (Lie) groups were used extensively for the particular cases of pose-SLAM, visual odometry, and group synchronization problems in general [11,22,32], which only consider relative measurements of the full states. They were recently extended to a more general case by Bourmaud et al [14], but focusing on the Lie group structure essentially, without discussing the best choice of group structure with respect to the system’s equations. In the next section, we focus our study on a particularly rich class of models on Lie groups.

IV. INVARIANT SMOOTHING

In this section, we consider smoothing on Lie groups for a special class of systems. The remarkable result is that the dependency of the information matrix of smoothing algorithms (partially) disappears, owing to the interplay between the group structure and the system’s properties.

A. Group-affine observation systems

In this work we argue that, just as vector spaces are particularly well-suited to linear systems, the Lie group formalism is beneficial to a particular class of dynamics called “group-affine observation systems”. They were initially introduced for continuous-time dynamics [7,8] but are easily adapted to the discrete case [8,9]. Here we briefly introduce them and recall some of their properties.

**Definition 1:** A discrete time system (17) with state \( \chi_i \) at time \( i \) being an element of a Lie group \( G \) is said to be “group-affine” if its dynamics/observation couple writes:

\[
\forall i, k, \quad \chi_{i+1} = f_i(\chi_i) \quad y_k = \chi_{i,h+1}(\chi_i d)
\]

where \( \forall a, b \in G, f_i(ab) = f_i(a)f_i((Id)^{-1}f_i(b) \)

The dynamics part of this definition is called group-affine dynamics and has a very powerful property, that will lead to interesting results in the sequel. Note that a continuous-time counterpart of these dynamics also exists in the theory of control on Lie groups [2].

**Theorem 1** (from [7]): \( f \) defines a group-affine dynamics if and only if \( g^L(\chi) = f((Id)^{-1}f(\chi) \) is a group automorphism, i.e., satisfies

\[
\forall a, b, g^L(ab) = g^L(a)g^L(b),\quad \text{and thus} \quad g^L(a^{-1}) = g^L(a)^{-1}
\]

In that case, the Lie-group/Lie algebra morphism correspondence ensures the existence of a \( q \times q \) matrix \( G^L \) such that

\[
\forall \xi, g^L(\exp(\xi)) = \exp(G^L \xi). \quad (16)
\]

B. Noisy system

For the remainder of this paper we will focus on the following dynamics, observation and noise models:

\[
\chi_{i+1} = f_i(\chi_i) \exp(w_i) \quad y_k = \chi_{i,h+1} + v_k \quad (17)
\]

where \( w_i \) and \( v_k \) are white noise similar to those of (8), \( d \) is a given vector of \( \mathbb{R}^q \), and \( f_i \) possesses the group-affine property (13) for all \( i \).

In this model, the measurements are left-equivariant, a fact which according to [7] prompts the use of the left multiplication based uncertainty model (3). Therefore, given a current guess \( \hat{\chi} \), the cost function will be linearized, noting \( \xi \) the searched parameter, using

\[
\chi = \hat{\chi} \exp(\xi). \quad (18)
\]
C. Factors definition

Under the Gaussian and conditional independence assumptions, defining the factors boils down to isolating the noise in (17). We have \( \exp(w_i) = f_i(\chi)\chi^{i-1} \). Let \( \tilde{\chi}_i, \tilde{\chi}_{i+1} \) be some guesses, and let us write the true \( \chi_i, \chi_{i+1} \) using the Lie-group based error model (18). This yields

\[
\exp(w_i) = f_i(\chi_i)^{-1} \chi_{i+1} = f_i(\tilde{\chi}_i, \exp(\xi_i))^{-1} \tilde{\chi}_{i+1} \exp(\xi_{i+1})
\]

Using (13), and then (16) this is equal to

\[
= (f_i(\tilde{\chi}_i)f_i(Id)^{-1}f_i(\exp(\xi_i)))^{-1} \tilde{\chi}_{i+1} \exp(\xi_{i+1})
\]

\[
= (f_i(\tilde{\chi}_i)g_i^{-1}(\exp(\xi_i)))^{-1} \tilde{\chi}_{i+1} \exp(\xi_{i+1})
\]

\[
= \exp(-g_i^{-1}(\xi_i))f_i(\tilde{\chi}_i)^{-1} \tilde{\chi}_{i+1} \exp(\xi_{i+1})
\]

Let \( a_i = \log(f_i(\tilde{\chi}_i)^{-1} \tilde{\chi}_{i+1}) \) be the estimation error for the guess. Then, taking the logarithm of both hands sides and using the BCH formula, only keeping the first order terms in \( \xi \) and \( a_i \) leads to

\[
w_i \approx a_i - g_i^{-1}(\xi_i) \approx a_i - \xi_i
\]

The major advantage (and very remarkable feature) of this parametrization is the fact that, if the dynamics is accurate enough the jacobians do not depend on the current estimate since \( G_i^k \) is independent of \( \tilde{\chi}_i \) from Theorem 1.

To get the measurement factors we propose to consider the following quantities

\[
\tilde{\chi}_i^{-1}v_k = \tilde{\chi}_i^{-1}v_k - \exp(\xi_i) \frac{d}{d}.
\]

By letting \( c_k = \tilde{\chi}_k^{-1}v_k \), linearization then leads to

\[
\tilde{\chi}_k^{-1}v_k \approx c_k - H\tilde{\chi}_k
\]

where \( H \) is the matrix defined by \( H\xi = \xi^4 \frac{d}{d} \), where we used a Taylor expansion of the matrix exponential in (2).

Finally, the factor associated to the prior must also be assessed. Indeed, the noise on the prior now writes

\[
\exp(z_0) = \tilde{\chi}_0^{-1}z_0 \exp(\tilde{\chi}_0)
\]

Contrarily to the propagation factor, here the first-order terms of the BCH formula cannot be reduced to the sum of its arguments, as the term \( \log(\tilde{\chi}_0^{-1} \tilde{\chi}_k) \) will tend to grow over the iterations, sometimes to a point where the jacobian of the Lie group [3] must be taken into account. We thus have in general

\[
z_0 \approx p_0 + J_0 \tilde{\chi}_0
\]

where \( p_0 = \log(\tilde{\chi}_0^{-1} \tilde{\chi}_0) \), and \( J_0 \) is defined as \( \text{BCH}(p_0, \tilde{\chi}) = p_0 + J_0 \tilde{\chi} + \mathcal{O}(\|\tilde{\chi}\|^2) \).

Therefore, the linear least-squares problem to be solved at each iteration writes

\[
(\xi_i)_{i \leq n} = \arg\min_{(\xi_i)_{i \leq n}} \|p_0 + J_0 \tilde{\chi}_0\|^2 + \sum_{i} \|a_i - g_i^{-1}(\xi_i) + \xi_{i+1}\|^2 + \sum_{k} \|c_k + H\tilde{\chi}_k\|^2
\]

where \( \tilde{N}_k = \tilde{\chi}_k^{-1}N_k \tilde{\chi}_k^T \) is the covariance of \( \tilde{\chi}_k^{-1}v_k \), see (23).

Algorithm 1: Smoothing for a group-affine system with a left multiplication based parametrization (18)

Input: \( (\tilde{\chi}_i)_{1 \leq i \leq n}, P_0, (f_i)_i, (Q_i)_i, (y_k)_i, (N_i)_i \);

Initialization

1. Set \( \tilde{\chi}_i^0 = \tilde{\chi}_i, i = 1, \cdots, n \); Until convergence do

2. Linearize around \( (\tilde{\chi}_i^k)_i \) according to (26);

3. Solve for \( (\xi_i)_i) \);

4. Update: \( \tilde{\chi}_{i+1} = \tilde{\chi}_i \exp(\xi_i), i = 1, \cdots, n \);

Output: \( (\tilde{\chi}_i)_{i \leq n} \);

D. Final algorithm and benefits of the proposed approach

Injecting (26) into the smoothing framework, we get Algorithm 1 for the left multiplication based parametrization (18) (as opposed to \( \chi = \exp(\xi)\tilde{\chi} \)). The main advantage of this framework is the fact that the current estimates do not appear in the definition of the propagation and observation jacobians, in (21) and (23) respectively (matrices \( G^k \) and \( H \)), although the latter comes at the expense of a modified covariance for the measurement factors. However, this dependency can often disappear, leaving only the prior jacobian of (24) to be a function of the estimate. If the initial guess is good, this can also be harmless relaxed by approximating \( J_0 \approx Id \). This is summarized in the following proposition.

Proposition 1: We have the following properties: (i) Consider the system (17). If \( f_i \) is group-affine for all \( i \), then all the propagation jacobians are independent from the current estimate. (ii) Moreover, if the measurement covariance \( N_i \) is such that \( N_i = \chi N_i \chi^T \) for all \( \chi \), then the observation jacobians are also independent from the estimates. (iii) Finally, if the initial guess is good, i.e. \( J_0 \approx Id \), then the information matrix associated to (26) is independent from the current estimate.

Although the condition on the measurement covariance can seem restrictive at first, it often boils down to an isotropy assumption, as will be the case for the model of Section V.

V. APPLICATION TO MOBILE ROBOT LOCALIZATION

A. Considered problem: robot localization

To evaluate the performances of the developed approach, tests were conducted for a wheeled robot localization problem, using the standard non-linear equations of the 2D differential drive car modeling the position \( x_i \in \mathbb{R}^2 \) and heading \( \theta_i \in \mathbb{R} \) of the robot. The odometric velocity is integrated between two time steps to give a position shift \( u_i \in \mathbb{R}^2 \) and the angular shift \( \omega_i \in \mathbb{R} \) is measured through (differential) odometry and/or gyroscopes. The discrete noisy model writes:

\[
\begin{align*}
\theta_{i+1} &= \theta_i + \omega_i + w_i^\theta \\
x_{i+1} &= x_i + R(\theta_i)(u_i + w_i^x)
\end{align*}
\]

where \( w_i^\theta \) is the angular measurement error, \( w_i^x \) contains both the odometry and transversal shift errors, and \( R(\theta) \in SO(2) \).
denotes the planar rotation of angle θ. The vehicle also gets noisy position measurements (through e.g., GNSS) with standard deviation σ of the form

\[ y_k = x_k + v_k, \quad v_k \sim \mathcal{N}(0, \sigma^2 I_2) \]  \hspace{1cm} (28)

The state (and the increments) can be embedded in the matrix Lie group \( SE(2) \), which is detailed in Appendix A, using the homogeneous matrix representation

\[ \chi_i = \begin{pmatrix} R(\theta)^{1/2} & x_i \\ 0 & 1 \end{pmatrix}, \quad U_i = \begin{pmatrix} R(\alpha) & u_i \\ 0 & 1 \end{pmatrix} \]

and letting \( w_i = \begin{pmatrix} w_i^x \\ R(\alpha)^T w_i^\gamma \end{pmatrix} \) the stacked noise vector on the increments, we have using a first order expansion of the exponential map of \( SE(2) \) (the increment noises are small if the time step is small)

\[ \exp(w_i) \approx \begin{pmatrix} R(w_i^\theta) & R(\alpha)^T w_i^\gamma \\ 0 & 1 \end{pmatrix}. \]

Thus, the dynamics and the observations respectively rewrite in the desired form (8):

\[ \chi_{i+1} = \chi_i U_i \exp(w_i) = f_i(\chi_i) \exp(w_i) \]  \hspace{1cm} (29)
\[ y_k = \chi_i \begin{pmatrix} 0_{2 \times 1} \\ 1 \end{pmatrix} + v_k \]  \hspace{1cm} (30)

This system is group-affine and its measurement covariance is rotation-invariant (i.e., isotropic), therefore Proposition 1 (i) and (ii) hold. Moreover, we have, with the notations of Section IV and according to Section II,

\[ \psi_i^T(\chi) = \Psi_i U_i^{-1}(\chi) \]  \hspace{1cm} (31)

B. Compared smoothing frameworks

In Sections V-C and V-D, the proposed smoothing method is compared with two non-invariant parametrizations, but which account for the non linear structure of the state space, from respectively [25] and [21], and a standard linear parametrization. The two former were developed for applications such as pose-SLAM or VIO, without directly considering navigation with absolute measurements. They were reimplemented in a batch setting, so that the focus was put exclusively on the parametrization. Writing the propagation linearized factor in the general form \( a_i + F_i^j \xi_j + F_i^{j+1} \), the following jacobians were used, see [21,25] for the respective residuals’ definitions:

<table>
<thead>
<tr>
<th>IS (ours)</th>
<th>(-Ad_{ij}^{-1})</th>
<th>(F_{ij}^\bullet)</th>
<th>(F_{ij}^{j+1})</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lin [25]</td>
<td>(-t_2 R_{ij} - 1)</td>
<td>Id</td>
<td>Id</td>
</tr>
<tr>
<td>[21]</td>
<td>([-\Omega_f^T \hat{R}_i \Omega_f^T \hat{R}<em>i (\hat{x}</em>{i+1} - \hat{x}_i)) ]</td>
<td>(\Omega_f^T \hat{R}_i^{-1})</td>
<td>(\hat{R}_{i+1}^{-1})</td>
</tr>
</tbody>
</table>

The observation factors are constructed using the following jacobians \( H_k \), errors \( c_k \) and covariances:

<table>
<thead>
<tr>
<th>IS (ours)</th>
<th>(t_2 0)</th>
<th>(R_{ij}^p (y_k - \hat{x}_k))</th>
<th>(R_{ij}^p N_i R_{ij}^p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lin [25]</td>
<td>(t_2 0)</td>
<td>(y_k - \hat{x}_k)</td>
<td>(N_k)</td>
</tr>
<tr>
<td>[21]</td>
<td>(\hat{x}_k 0)</td>
<td>(y_k - \hat{x}_k)</td>
<td>(N_k)</td>
</tr>
</tbody>
</table>

C. Simulation results

The four smoothing methods of the previous section were compared. The vehicle is initialized at the true position but with a wrong heading of \(-3\pi/4\), and an initial covariance of \( diag(0.0025, 0.0025, (-3\pi/4)^2) \). The robot moves along a line at 7m/s, where the odometry is polluted by a white noise of covariance \( (0.1(m/s))^2, 0.01(rad/s)^2 \), and acquired at a rate of 10Hz. Position measurements of covariance \( 0.01m^2 \) are received every five steps. In this batch setting, Proposition 1 (iii) does not hold for the invariant framework.

The iterations of the four optimization schemes are displayed on Figure 2. As can be seen, the invariant parametrization converges faster than all the other ones, and much more smoothly. This is important for a sliding window smoothing setting, as restricting the number of solver iterations at each step will have a weaker impact on the estimation.
case, leading to an information matrix fully independent from the estimate. The effect of the window’s size is also studied here. The raw odometer inputs were provided to each method. Artificial position measurements are delivered by adding a Gaussian noise to the ground truth, at a rate of 1.35Hz. For each setting, 100 Monte-Carlo simulations are run, each algorithm being initialized identically at each run. The trajectory followed by the robot, along with results of the various algorithms for one of the runs, are displayed in Figure 4. Three series of experiments were conducted, to study the evolution of the error with respect to the measurement covariance, the window size, and the initial error respectively. In the first case, the window size was fixed to 5, and the measurement covariance ranged from $10^{-1}$ to $10^{-5}m^2$, with noise turned on and off on the initial state. When on, $(\bar{x}_0, \bar{\theta}_0) \sim \mathcal{N}((0,0,0), \text{diag}(1/8, 1/8, (\pi/4)^2))$. When off, it was put at $\bar{x}_0 = (1/4, 1/4)$ and $\bar{\theta}_0 = \pi/4$. In the second one, the measurement covariance was fixed to $10^{-1}m^2$, while the window size ranged from 5 to 13. For the last one, the window size was set to 5, the measurement covariance to $10^{-5}m^2$. The initial estimate was put at $\bar{x}_0 = (1/4, 1/4)$ and $\bar{\theta}_0$ spanned $[-\pi, \pi]$ with only one Gauss-Newton per iteration, then it was fixed at $\bar{\theta}_0 = 9\pi/10$ with one and seven GN iterations. Figures 5 and 6 show the averaged Root Mean Square Error (RMSE) of the position and heading for the two first experiments. Figure 7 shows the RMSE of the heading for the last one.

**Remark 1:** Average computation times used to compute the full estimate, on a laptop with Intel i5-5300 2.3 GHz CPU, are given in the following table.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.82</td>
<td>0.81</td>
<td>0.83</td>
<td>0.62</td>
</tr>
</tbody>
</table>

It appears that the invariant smoothing, in a batch setting, does not induce extra computational load. Actually, it could prove more efficient in an incremental setting by taking advantage of the independence of the information matrix from the linearization points to prefactor the information matrix [28].

![Fig. 3: Wifibot robot in its testing arena, surrounded by Optitrack cameras](image1)

**E. Experimental results**

Figure 5 displays the evolution of the error with respect to the measurement noise covariance, when the initial state is fixed and when it is not. The first observation to be made is that, for a noise up to moderate ($10^{-2}m$), smoothing methods in general outperform the filtering ones, especially in terms of heading. This was expected, as the former explicitly contains the constraints between successive poses. If the initial state is fixed at $(1/4, 1/4, \pi/4)$, all smoothing approaches achieve almost identical results validating, in this case, the assumption made when deriving the propagation factor. However, invariant smoothing appeared to be the most robust to noise on the initial state, especially for low measurement covariance. This is further investigated in the third experiment.

The impact of the window size, illustrated in Figure 6, seems clear: increasing the size of the window is beneficial to all smoothing methods for strong noise. Still, none of them beats the UKF-LG in terms of position, highlighting its robustness. However, for lighter noise, the impact was negligible, as the odometry’s uncertainty was quickly reached.

Finally, the invariant smoothing proved to be the more robust to initial heading errors such that $|\bar{\theta}_0| > \pi/2$, as shown by Figure 7, top, while both non invariant methods behaved identically. The two other graphs explain this difference: it appears that non-invariant smoothing methods fall into a local minimum where the estimate is completely turned around. Indeed, when increasing the number of GN iterations, the non-invariants experiments clearly form two clusters, one
with the expected heading RMSE, and one with a RMSE close to $\pi$. Note that it is not a problem of the non-invariant methods, as they were devised for problems relative to the original position, i.e. without such an influence of the initial error. Moreover, this confirms the results of Section V-C, as the invariant smoothing reaches the optimal RMSE quicker than the non-invariant ones, in the right cases.

Fig. 5: Top and middle : average position and heading RMSE for the five estimation methods w.r.t $\sigma$, with fixed initial state. Down : heading RMSE w.r.t $\sigma$, with random initial state. The window size was set to 5 for the smoothing methods.

Fig. 6: Average position and heading RMSE for the smoothing methods w.r.t. the window size, for $\sigma^2 = 10^{-1}m^2$. The RMSE of the filtering methods are shown as a comparison.

Fig. 7: Heading RMSE for the smoothing methods w.r.t the initial angle $\theta_0$ for $\sigma^2 = 10^{-5}m^2$ and a window size of 5. Top : $\theta_0$ spans $[-\pi, \pi]$ with one GN iteration per step. Middle : RMSE for each run with $\theta_0 = 9\pi/10$ with one GN iteration. Down : RMSE for each run with $\theta_0 = 9\pi/10$ with seven GN iterations.

F. Discussion

For this experiment, the proposed method managed to combine the advantages of the smoothing approach and of the invariant framework. Indeed, on the first hand it provided better estimates than the filtering methods thanks to the smoothing paradigm. On the other hand, owing to the mathematical invariances leveraged by the algorithm, their computation are based on the measured odometry inputs, and not the estimated ones. This reduces the need for relinearization, while ensuring more robustness to initial heading error, especially when non-invariant methods exhibited local minima which have no impact on the position estimate, but completely shift the heading one. However, this also means that the method will be dependent of the odometry’s uncertainties.
VI. CONCLUSION

This paper presented the extension of the invariant Kalman filtering framework (see e.g., [9] for an overview) to smoothing. This resulted in Jacobians being independent from the current estimate, thus ensuring sound behavior of the estimate. To validate the results, this was applied to a robot localization problem, in simulations, and using experimental data. On the one hand, faster convergence to the MAP estimate than the other smoothing approaches was achieved in simulation. On the other hand, the invariant smoothing provided estimates similar to that of other smoothing methods, while being as fast to compute, and better than invariant filtering methods. It also appeared more robust to errors on the initial state, avoiding local minima of the non-invariant methods. Future work will include extending the framework to relative measurements of the state.

VII. ACKNOWLEDGEMENTS

The authors would like to thank Martin BARCZYK and Tony NOËL for their precious help with the experiments. This work is supported by the company Safran through the CIFRE convention 2016/1444.

APPENDIX A

OPERATORS OF THE SPECIAL EUCLIDEAN 2D GROUP

The Special Euclidean 2D $SE(2)$ group represents rigid transformations in 2D space defined by their heading $\theta$ and position $x$. The exponential, logarithm and adjoint operators on $SE(2)$ are defined for $\chi \in SE(2)$ and $\xi \in se(2)$ as

$$
\exp(\xi) = \begin{bmatrix} R(\xi^3) & V(\xi^2) & \xi^1 \\ 0 & 1 & 0 \end{bmatrix},
$$

$$
\log(\chi) = \begin{bmatrix} V_\theta^1 x \\ \theta \end{bmatrix},
$$

where $V_\alpha = \frac{1}{\alpha} \begin{bmatrix} \sin(\alpha) & -1 + \cos(\alpha) & \sin(\alpha) \\ 1 - \cos(\alpha) & \sin(\alpha) & 0 \end{bmatrix}$.

$$
Ad_\chi = \begin{bmatrix} R(\theta) & -J x \\ 0 & 1 \end{bmatrix},
$$

where $J = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

REFERENCES


