LIÈGE université
Sciences Appliquées

# Iterated invariant Kalman filtering for articulated rigid-body systems

## from theoretical foundations to inertial motion estimation for robotics

Sven Goffin

advised by Pierre Sacré and co-advised by Olivier Brüls

Thesis submitted in partial fulfillment of the requirements
for the degree of doctorate (PhD) in Engineering Science

Doctoral College in Electrical Engineering, Electronics, and Computer Science

University of Liège

February 2026

 Sven Goffin is a FRIA grantee of the Fonds de la Recherche Scientifique – FNRS.

All figures were created by the author himself using the Affinity suite. ChatGPT (OpenAI) was used solely for editorial assistance (spelling, grammar, and phrasing). The author reviewed all AI-assisted edits and remains fully responsible for the content, results, and interpretations.

# Jury

**Jean-Michel Redouté** (President)
Department of Electrical Engineering and Computer Science
University of Liège

**Pierre Sacré** (supervisor)
Department of Electrical Engineering and Computer Science
University of Liège

**Olivier Brüls** (co-supervisor)
Department of Aerospace and Mechanical Engineering
University of Liège

**Silvère Bonnabel**
Centre for Robotics
Mines Paris — PSL

**Manon Kok**
Delft Center for Systems and Control
Technische Universiteit Delft

**Frank Naets**
Department of Mechanical Engineering
Katholieke Universiteit Leuven

**Pieter Van Goor**
School of Aerospace, Mechanical, and Mechatronic Engineering
University of Sydney

Published online at the Institutional Repository of the University of Liège:

# Abstract

The invariant extended Kalman filter (IEKF) is known for its strong convergence and consistency properties. Using an invariant definition of the estimation error, it yields propagation and update Jacobians that are independent of the current state for systems with group-affine dynamics and measurements written in invariant form. Although it is a state-of-the-art method for single-body pose estimation, its use in articulated rigid-body systems remains limited. Kinematic constraints, coupled poses, and nontrivial state representations hinder a direct extension of the theory. This thesis addresses this gap by extending invariant Kalman filtering to inertial pose estimation in articulated rigid-body systems.

This thesis makes three main contributions. First, it revisits the treatment of state equality constraints in Kalman filtering. By modeling such constraints as noise-free pseudo-measurements, it formalizes the properties a stochastic filter should satisfy when enforcing exact information, and highlights the challenges of doing so. Second, it introduces the iterated invariant extended Kalman filter (IterIEKF), an iterative variant of the IEKF that systematically improves the accuracy of its update through Gauss–Newton relinearization. In the noise-free case, the IterIEKF comes with theoretical guarantees: it is able to incorporate perfect information consistently and globally on the matrix Lie group state space, which the classical IEKF cannot ensure. Third, this thesis introduces the relative L-extended pose, a matrix Lie group representation of multibody pose. This representation yields group-affine dynamics when each body is equipped with an inertial measurement unit (IMU), and it allows a broad class of joint constraints to be expressed in invariant form. Combined with the IterIEKF, it provides a principled way to handle kinematic constraints within the invariant framework and to transfer the properties of invariant filtering to multibody pose estimation problems.

The proposed framework is validated on two real-world experiments: inertial pose estimation of a UR5e robot during a pick-and-place task and of a human leg during forward lunges. In both cases, the IterIEKF combined with the relative L-extended pose exhibits faster convergence and substantially reduced estimation variance compared with EKF and iterated EKF (IterEKF) baselines, as well as invariant filtering based on a free-segment multibody representation. Overall, this thesis provides a principled route from invariant filtering theory to practical inertial motion estimation for articulated rigid-body systems.

# Acknowledgments

I would first like to thank my supervisor, Pierre Sacré, for sparking my curiosity through this PhD project and for believing in me from the very beginning of this journey. He always followed my work closely and offered his support whenever I needed it. He shared with me his critical mindset, teaching me to step back, gain perspective, and grow into an independent researcher. Pierre also quickly understood how I work, reassuring me about my abilities when it mattered and pushing me when necessary, which allowed me to make progress under the best possible conditions. On a personal level, he was consistently understanding and attentive, and few supervisors would have been as supportive in allowing me to work remotely so that I could spend extended time with my partner, who lives abroad. For all of this, I am deeply grateful to him.

This work would never have come to fruition without the support and guidance of Professor Silvère Bonnabel. I learned a great deal from his expertise in invariant filtering. I also warmly thank him for welcoming me into his team during my research stay at École des Mines de Paris, and for introducing me to Marseille. It is not every day that one meets someone who is both so brilliant and so approachable.

I also thank my co-supervisor, Olivier Brüls, for sharing his expertise in mechanical engineering and robotics. He was always available whenever I needed him. I learned a great deal from his experience, and, like Pierre Sacré and Silvère Bonnabel, he is undoubtedly a role model for the researcher I aspire to become.

I would also like to thank my friends Anaëlle and Florent, my office mates and partners in crime, for putting up with my sometimes sharp sense of humor over the past four years. Our mutual support helped me stay motivated in my research, and our short breaks allowed me to step back, regain perspective, and lift my head from the grind. I also thank my fellow doctoral friends Astrid, Laura, Glaudy, Vicol, Loris, and Antoine for their daily support and good spirits. I would also like to acknowledge all the other members of the neuroengineering lab for the stimulating and insightful discussions we shared.

My family also played an important role in the success of this PhD journey. They did not always understand every detail I tried to explain, but they consistently showed tremendous support throughout. I must also mention my roommates and friends, Gilles and Glaudy, who were there through both the good and the difficult times over the past four years. They kept me from going crazy by listening

whenever I needed it, and for that I am truly grateful. I would also like to mention my friend Cyril, who was incredibly supportive as well.

Finally, I am deeply grateful to Kathleen, who guided me throughout this journey. Without her, I would not be as proud of my work, and, more importantly, I would not be as proud of the person I have become over the past four years thanks to her presence in my life.

With a strong sense of completion, I am ready to close this chapter of my life, carry forward what I have learned along the way, and embark on new adventures with same curiosity and motivation.

# Contents

# 1 Introduction

*"Hot chocolate, as requested," the robot said as it rolled up beside the workbench, a steaming mug balanced carefully in its gripper. The scientist, absorbed in his work, extended a hand without looking up and murmured, "Thank you, SideKick. I do not know what I would do without you." SideKick paused for a second, attempting to estimate the scientist's pose before performing the handover. "Trajectory acquired," it announced while beginning a smooth approach. "Handing you the mug," SideKick added, as the cup was set down roughly 10 cm away from the outstretched hand, spilling its content across the scientist's notebooks in a chocolate-flavored disaster. The scientist finally looked up, half irritated and half resigned. "And your kind is supposed to take over the world one day?"*

Pose estimation and motion tracking are central to human-robot interaction: even a simple handover requires the robot to continuously infer where the human is and how they are moving. The same capability underpins more advanced tasks such as learning from demonstration, where the robot must first track human motion in order to reproduce it, and balance control in humanoid or legged robots, which depends on accurate self-pose estimation. These problems also extend well beyond robotics. In healthcare and biomechanics, motion tracking enables gait analysis and supports rehabilitation protocols [1], including robot-assisted therapy. Exoskeletons similarly rely on reliable pose estimation of the wearer to provide assistance that is both effective and safe [2, 3].

Most motion tracking systems rely on either vision-based technologies or inertial sensors [4]. Vision-based systems provide direct measurements of motion with respect to (w.r.t.) an inertial reference frame, and mostly rely on computer-vision, machine learning and deep-learning algorithms [5]. They generally offer very good performances [6]. Going from a single depth-camera [7, 8] to multiple cameras and advanced motion capture systems [9, 10], vision-based motion tracking setups operate only within a limited capture volume. In cluttered or dynamic environments, occlusions further restrict the field of view and reduce robustness. Inertial measurement units (IMUs), in contrast, are small, self-contained, and energy efficient, which makes them well suited for applications involving large scale motion in constrained or open spaces [11]. Their accuracy, however, depends critically on the algorithms used to reconstruct motion from noisy IMU data.

Pose estimation based on inertial measurements is typically performed using stochastic filtering methods, which combine predictions from imperfect motion models with noisy, and often partial, sensor measurements to obtain statistically optimal estimates. For more than sixty years, the Kalman filter and its extended version (EKF) have been the workhorse of the field [12]. The paradigm has naturally been widely adopted for inertial pose estimation of articulated rigid-body systems [13–15], including legged robots [16, 17] and exoskeletons [18, 19], where Kalman-type filters are commonly used to combine inertial data with biomechanical, contact and kinematic constraints, and human body [20–23], where full-body kinematics are reconstructed by combining inertial integration with anatomical constraints. Alternative approaches include complementary filtering [24–26], as well as particle filtering [27] and optimization-based methods [28, 29].

In many robotics and navigation problems, the state does not evolve in a Euclidean space but on a geometric manifold, often a matrix Lie group. This observation has motivated the development of geometric filtering methods, which explicitly account for the structure of the state space [30–36]. Among them, the invariant extended Kalman filter (IEKF) has become a state-of-the-art approach in inertial navigation [37–39]. Compared with the standard EKF, the IEKF leverages invariant error parametrizations on matrix Lie groups and, when the dynamics are group-affine and the measurements admit an invariant form (see Chapter 4), offers strong theoretical guarantees, including convergence results [38] and improved consistency in the presence of unobservability [39]. Advances in invariant filtering have enabled successful applications in attitude estimation [40, 41], inertial navigation [42–44], simultaneous localization and mapping [45, 46], single rigid-body pose estimation [47, 48], legged locomotion [49–51], and aerospace [39]. Despite this success, the invariant filtering framework remains much less prevalent in motion tracking problems involving articulated rigid-body systems, where kinematic constraints, multiple coupled poses, and nontrivial state representations complicate a direct extension of the theory.

This thesis addresses this gap by extending the invariant Kalman filtering framework to pose estimation for articulated rigid-body systems. Its contributions are threefold. First, it revisits the treatment of state equality constraints in Kalman filtering by modeling them as noise-free pseudo-measurements. It formalizes the properties that a filter should satisfy when enforcing such constraints and identifies the main theoretical and practical difficulties that arise. Second, it introduces the iterated invariant extended Kalman filter (IterIEKF), which combines invariant error parametrization with Gauss–Newton relin-

earization of the measurement model. This yields an update step capable of incorporating noise-free information globally on the Lie group state space, thereby addressing the remaining issues associated with equality-constraint enforcement in the invariant filtering framework and providing a practical mechanism for handling joint-induced kinematic constraints in articulated rigid-body systems. Finally, it proposes a relative L-extended pose representation tailored to multibody extended-pose estimation. This representation preserves the Lie group structure, admits group-affine dynamics when inertial measurements are available, and expresses a broad class of joint constraints in invariant form. Together, these contributions provide a consistent extension of invariant filtering to articulated rigid-body systems, validated on both robotic and biomechanical experimental setups.

## 1.1 Thesis outline

This thesis is divided into two main parts. Part I provides a technical yet intuitive overview of the background required to understand the contributions. It is divided into three chapters. Chapter 2 reviews the main results from probability theory that underpin filtering, from the definition of a probability space to recursive Bayesian inference for hidden Markov models. It also introduces the notions of random variables and stochastic processes, with particular emphasis on Gaussian distributions and processes. Chapter 3 presents classical Kalman filtering methods. It begins with the linear case and the standard Kalman filter, then moves to nonlinear systems through the extended Kalman filter (EKF), and finally introduces the iterated EKF as an improvement of the EKF update. Chapter 4 provides a succinct introduction to matrix Lie groups before diving into the theory of invariant Kalman filtering.

Part II contains the core of the thesis and presents its contributions in three chapters. Chapter 5 investigates state equality constraints in the Kalman filtering framework and discusses how invariant filtering can address some of the associated difficulties. Chapter 6 introduces the IterIEKF, an iterated variant of the IEKF designed to accommodate noise-free and low-noise measurements. Chapter 7 presents the proposed extension of invariant Kalman filtering to pose estimation for articulated rigid-body systems, leveraging the properties of the IterIEKF to handle joint-induced kinematic constraints.

Finally, Chapter 8 summarizes the main results, discusses the limitations of the proposed approach, and outlines directions for future work.

# Part I

# Fundamentals of Kalman filtering

As growing often rhymes with thriving in today's economic model, successful companies tend to expand. In these evolving environments, increasing attention is directed toward task automation, and human workers are more and more compelled to collaborate closely with autonomous machines. In such shared spaces, safety is crucial, and these machines must remain aware of their surroundings at all time.

Consider a simple scenario in which an autonomous robot collaborates with a human co-worker. The robot must ensure that it avoids collisions with its human partner. Achieving this requires knowledge of the human's state, such as their position and velocity, which informs the robot's decision-making process as it carries out its primary task. However, this information is usually not directly accessible and must instead be inferred through estimation. To this end, the robot typically relies on two models.

The first one is a model of the dynamics: given the current state, the robot can predict what the next state is likely to be in the near future. This dynamical model is inherently imperfect, since sudden changes in direction or speed cannot be anticipated with certainty. The second model is a measurement model: the robot uses several sensors that provide partial and noisy information about the state. These measurements may come from a camera that offers a coarse or occasionally occluded view, a motion sensor mounted on the worker, or proximity sensors that detect the worker within a limited range. Intuitively, fusing information coming from these two noisy, uncertain sources should yield a more reliable state estimate. This raises the question of how such a fusion should be performed.

Filtering provides a systematic way to address this situation. It consists in estimating the hidden state of a system over time by combining a dynamic model with incoming measurements. When a new observation arrives, the filter updates its estimate by balancing what the model predicts with what the sensors report. In other words, it fuses the available information to obtain the most reliable estimate at each moment.

This part of the manuscript offers a gradual and intuitive introduction to filtering. We begin with a brief recap of probability theory, which forms the foundation of state estimation. We then present the Kalman filter, a linear filtering algorithm that has played a central role in the field for more than sixty years, following the seminal work of Rudolf E. Kalman. After that, we introduce extensions suited for nonlinear systems, starting with the extended Kalman filter and then the invariant Kalman filtering framework, which is the main focus of this thesis. To build intuition, many examples are provided throughout. Readers already familiar with these concepts may skip directly to part II.

# 2 A primer on probabilities

Probabilities provide a mathematical framework for representing and manipulating uncertainty. They can be interpreted from different perspectives, often depending on the context. Consider a die that you roll one hundred times. Suppose that out of the 100 rolls, 43 land on the value 6. If you are asked whether the die is loaded, you will probably say yes, since you would expect roughly half that many sixes for a fair die. In this case, your answer is based on the observed frequency of occurrence, interpreted as the proportion of successes in a large number of similar trials. This is the frequentist approach to probability, which is often the most natural.

There are situations, however, where thinking in terms of long-run frequencies is not appropriate. Imagine a historian claiming that the hypothesis that the Titanic sank because it collided with an iceberg is probable at 97 percent. A frequency-based interpretation does not make sense here, as the event occurred only once. Rather, the number expresses the historian's degree of belief given the available evidence. This illustrates the Bayesian approach to probability, named after the British mathematician Thomas Bayes.

The frequentist and Bayesian interpretations are the two standard approaches adopted in the literature. As will become clear later, filtering is closely tied to the notion of belief in the Bayesian sense. For this reason, this chapter provides a brief introduction to the probabilistic concepts used throughout this manuscript, presented from a Bayesian perspective. Readers seeking more comprehensive treatments of probability theory are referred to [52], [53], and [54]. The structure and presentation of this chapter are partially inspired by the corresponding chapter in [55].

## 2.1 Probabilistic modeling

Behind every uncertain situation lies an underlying process, called an experiment, that produces exactly one outcome among several possible ones. To reason about the likelihood of these outcomes and to support decisions that depend on them, we describe the problem using a probabilistic model. Concretely, this consists in choosing an appropriate probability space that represents, with sufficient accuracy, the experiment we aim to model.

▶ **Definition 2.1 (Probability space).** A probability space is a mathematical triplet $(\Omega, \mathcal{F}, \Pr)$ composed of:

- a set $\Omega$, called the sample space, which contains all possible outcomes of a random experiment;
- a $\sigma$-algebra $\mathcal{F}$ on $\Omega$, called the event space, which is a collection of subsets of $\Omega$ that contains the empty set and is closed under complementation and countable unions;
- a probability measure $\Pr : \mathcal{F} \to [0, 1]$, which assigns a probability to each event in $\mathcal{F}$. ◀

The pair $(\Omega, \mathcal{F})$ defines a measurable space, on which $\Pr$ acts as a measure. For any event $A \in \mathcal{F}$, the probability $\Pr(A)$ quantifies the likelihood that the outcome of the experiment belongs to $A$. The axioms below formalize the properties required of a probability measure.

▶ **Definition 2.2 (Kolmogorov axioms).** Let $(\Omega, \mathcal{F}, \Pr)$ be a probability space. The probability measure $\Pr$ satisfies the following axioms:

1. (Normalization) $\Pr(\Omega) = 1$.
2. (Non-negativity) $\Pr(A) \geq 0$ for all $A \in \mathcal{F}$.
3. (Additivity) If $A, B \in \mathcal{F}$ are disjoint events, that is, $A \cap B = \varnothing$, then

$$\Pr(A \cup B) = \Pr(A) + \Pr(B).$$

More generally, if $\{A_i\}_{i \geq 1}$ is a sequence of pairwise disjoint events, then

$$\Pr\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \Pr(A_i).$$

◀

Probabilistic models can be either discrete or continuous, depending on the structure of their sample space $\Omega$. In the discrete case, the sample space contains a finite or countably infinite number of outcomes $\omega_i$, and the probability measure $\Pr$ is completely determined by the probabilities $\Pr(\{\omega_i\})$ of the singletons $\{\omega_i\}$. This contrasts with continuous models, whose sample spaces contain an uncountable infinite number of outcomes, and where the probabilities of singletons are not

sufficient to characterize the probability measure, and are in fact equal to zero. The probability measure must therefore be defined by specifying how it assigns probabilities to the measurable subsets contained in the event space $\mathcal{F}$. The following example illustrates this difference.

▶ **Example 2.1.** A die roll has six possible outcomes, and can therefore be described mathematically using a discrete model. Knowing the probability of each face is enough to determine the probability of any event, such as "the outcome is 3 or 6."

In contrast, a Light Detection and Ranging (LiDAR) sensor that measures distances up to $5\,\mathrm{m}$ possesses an uncountably infinite number of possible outcomes. A single value reported by the sensor cannot have positive probability; otherwise, by selecting sufficiently many distinct values, the additivity axiom would imply a total probability greater than 1. In this case, a possible way to define a probability measure is to assign to each interval $[a, b] \subseteq [0, 5]$ the probability $\frac{b-a}{5}$. ◀

### 2.1.1 Joint probability and conditioning

For a given experiment, it is often useful to evaluate the probability that several events occur simultaneously. Given two events $A$ and $B$, their joint probability is defined as

$$\Pr(A, B) \coloneqq \Pr(A \cap B). \tag{2.1}$$

The probability of the union of two events satisfies

$$\Pr(A \cup B) = \Pr(A) + \Pr(B) - \Pr(A, B). \tag{2.2}$$

This relation is a generalization of Kolmogorov's additivity axiom and accounts for the fact that the probability of the intersection $A \cap B$ would otherwise be counted twice if one simply summed $\Pr(A)$ and $\Pr(B)$. This can be visualized using the Venn diagram shown in Figure 2.1.

▶ **Example 2.2.** Consider a regular die and the events $A =$ "the landed face carries an odd number" and $B =$ "the landed face is 3 or 4". The joint probability of these events is the probability of their intersection $A \cap B =$ "the landed face is 3", so that $\Pr(A, B) = \frac{1}{6}$. Similarly, the probability of their union $A \cup B =$ "the landed face is 1, 3, 4 or 5" is given by $\Pr(A \cup B) = \Pr(A) + \Pr(B) - \Pr(A, B) = \frac{3}{6} + \frac{2}{6} - \frac{1}{6} = \frac{4}{6}$. ◀

Beyond joint probabilities, we are often interested in how the occurrence of one event influences the likelihood of another. In many situations involving uncertainty, partial information is available and should be taken into account when assigning probabilities. Conditional probability provides a systematic framework

for reasoning under such partial information. Given a probability space $(\Omega, \mathcal{F}, \Pr)$ and an event $B$ with $\Pr(B) > 0$, the conditional probability of an event $A$ given $B$, denoted $\Pr(A \mid B)$, represents the probability that the outcome lies in $A$, knowing that it lies in $B$. It is defined as

$$\Pr(A \mid B) = \frac{\Pr(A, B)}{\Pr(B)}. \tag{2.3}$$

▶ **Example 2.3.** Consider a robot that estimates the position of a human co-worker using a LiDAR sensor. When the LiDAR does not detect the co-worker, the probability mass associated with the co-worker's presence is concentrated outside the sensor's ranging area. When the LiDAR detects the co-worker, this new information conditions the probability distribution, increasing the probability assigned to locations close to the detected position while decreasing the probability assigned to more distant locations. ◀

For a fixed event $B$ with nonzero probability, the conditional probability $\Pr(\cdot \mid B)$ satisfies the Kolmogorov axioms. Equivalently, one may regard $B$ as a new sample space and interpret $\Pr(\cdot \mid B)$ as a probability measure defined on the measurable subsets of $B$.

Conditional probability leads to an important identity known as the multiplication rule, which expresses the probability of the intersection of multiple events as a product of conditional probabilities:

$$P\left(\cap_{i=1}^{n} A_i\right) = P(A_1)P(A_2 \mid A_1)P(A_3 \mid A_1 \cap A_2) \cdots P\left(A_n \mid \cap_{i=1}^{n-1} A_i\right). \tag{2.4}$$

▶ **Example 2.4.** Consider a LiDAR sensor that, whenever an obstacle is present within its sensing range, correctly detects it with probability $0.97$. A missed detection corresponds to the situation in which an obstacle is present but not detected. Let $A$ denote the event "an obstacle is present in the sensing range" and $B$ the event "the LiDAR detects an obstacle." Denoting by $B^c$ the complement of $B$, a missed detection corresponds to the event $A \cap B^c$. Assuming that the probability of an obstacle being present in the sensing zone is $0.1$, the probability of a missed detection is

$$\Pr(A \cap B^c) = \Pr(A)\Pr(B^c \mid A) = 0.1 \times 0.03 = 0.003.$$

◀

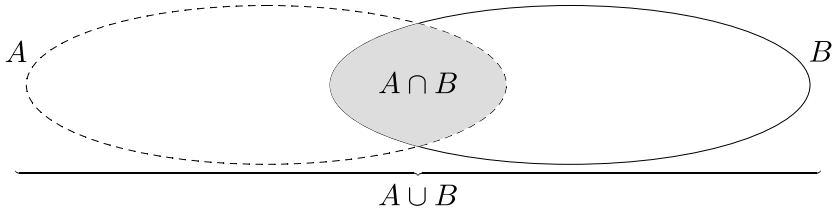



**Figure 2.1:** Venn diagram illustrating the union and intersection of two events $A$ and $B$.

## 2.2 Random variables

The outcome of most experiments is of a numerical nature. In the few cases where it is not, it is often useful to associate each outcome with a numerical quantity of interest. This idea is formalized through the notion of a random variable.

▶ **Definition 2.3.** Given a random experiment with sample space $\Omega$, a random variable $\mathbf{X} : \Omega \to \mathcal{S}$ is a function that assigns to each outcome $\omega \in \Omega$ an element $\mathbf{X}(\omega)$ of a measurable set $\mathcal{S}$. ◀

This definition allows random variables to take values in any measurable set $\mathcal{S}$. In practice, $\mathcal{S}$ is most often a subset of $\mathbb{R}^n$, so that the realized value of the random variable is a scalar or a vector of numerical values. For simplicity of exposition, we implicitly restrict attention to random variables taking values in $\mathcal{S} \subseteq \mathbb{R}^n$. In the sequel, we slightly abuse notation and write $\Pr(\mathbf{X} \in \mathcal{M})$ to denote the probability that the value realized by the random variable lies in a subset $\mathcal{M} \subseteq \mathcal{S}$.

▶ **Example 2.5.** In many board games, a pawn is moved according to the sum of the results of two dice rolls. The corresponding random experiment consists of rolling two dice, and the sample space contains all ordered pairs of integers between $1$ and $6$. The sum of the two outcomes defines a random variable. As another example, consider an experiment consisting of randomly selecting $10$ people in the street. The vector formed by stacking their heights is a random variable. ◀

The notion of probability naturally extends to random variables. The probability that a random variable $\mathbf{X}$ takes a value in a set $\mathcal{M}$ is defined as

$$\Pr(\mathbf{X} \in \mathcal{M}) = \Pr(\{\omega \in \Omega \mid \mathbf{X}(\omega) \in \mathcal{M}\}).$$

A random variable is said to be discrete if it takes values in a finite or countably infinite set, and continuous if it takes values in an uncountably infinite set. This distinction leads to different ways of characterizing their probability laws, in much the same way as for probabilistic models. In the remainder of this work, we focus exclusively on continuous random variables, as they are the only type considered.

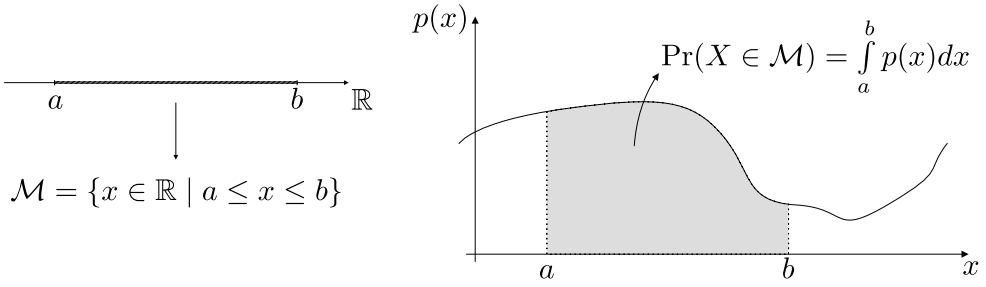



**Figure 2.2:** Illustration of the probability density function of a scalar random variable $X$.

### 2.2.1 Probability density of a continuous random variable

For continuous random variables, the probability of taking any single value $\mathbf{x} \in \mathbb{R}^n$ is zero. Probabilities are instead defined over sets through integration of a probability density function.

▶ **Definition 2.4 (Probability density function).** The probability density function (PDF) of a continuous random variable $\mathbf{X} : \Omega \to \mathcal{S}$ is a non-negative function $p : \mathcal{S} \to \mathbb{R}^+$ such that

$$\Pr(\mathbf{X} \in \mathcal{M}) = \int_{\mathcal{M}} p(\mathbf{x})\mathbf{dx},$$

for all measurable subsets $\mathcal{M} \subseteq \mathcal{S}$. ◀

The integral is interpreted in the classical Riemann sense: the probability associated with a set $\mathcal{M}$ can be viewed as the area under the curve defined by $p(\mathbf{x})$ for $\mathbf{x} \in \mathcal{M}$. This concept is illustrated in Figure 2.2 for a scalar random variable. From the normalization axiom of Kolmogorov, it follows that

$$\int_{\mathcal{S}} p(\mathbf{x})\mathbf{dx} = 1. \tag{2.5}$$

For highly irregular functions or sets, the integral in Definition 2.4 may fail to exist. Such cases are not considered here. It is nevertheless reassuring to note that these mathematical issues do not arise when $p$ is piecewise continuous with a finite number of discontinuities and when $\mathcal{M}$ is a union of a finite or countably infinite number of intervals. Under these mild assumptions, all integrals required in this work are well defined.

▶ **Example 2.6.** Consider the LiDAR sensor from Example 2.1. Its output can be modeled as a random variable $X$ taking values in $\mathbb{R}$, whose probability law is

specified by the uniform probability density function

$$p(x) = \begin{cases} \frac{1}{5} & \text{if } x \in [0, 5], \\ 0 & \text{otherwise.} \end{cases}$$

It is straightforward to verify that

$$\int_{-\infty}^{\infty} p(x)dx = \int_{0}^{5} \frac{1}{5}dx = 1.$$

◀

Similarly to the joint probability of two events, one can define the joint probability density function of two continuous random variables. Let $\mathbf{X} : \Omega \to \mathcal{S}_{\mathbf{X}}$ and $\mathbf{Y} : \Omega \to \mathcal{S}_{\mathbf{Y}}$ be two continuous random variables. Their joint probability density function is a non-negative function $p : \mathcal{S}_{\mathbf{X}} \times \mathcal{S}_{\mathbf{Y}} \to \mathbb{R}^+$ such that, for any measurable sets $\mathcal{M}_{\mathbf{X}} \subseteq \mathcal{S}_{\mathbf{X}}$ and $\mathcal{M}_{\mathbf{Y}} \subseteq \mathcal{S}_{\mathbf{Y}}$,

$$\Pr(\mathbf{X} \in \mathcal{M}_{\mathbf{X}}, \mathbf{Y} \in \mathcal{M}_{\mathbf{Y}}) = \int_{\mathcal{M}_{\mathbf{X}}} \int_{\mathcal{M}_{\mathbf{Y}}} p(\mathbf{x}, \mathbf{y}) \mathbf{dydx}. \tag{2.6}$$

The marginal probability density functions of $\mathbf{X}$ and $\mathbf{Y}$ are obtained by marginalization of the joint PDF:

$$p(\mathbf{x}) = \textstyle\int_{\mathcal{S}_{\mathbf{Y}}} p(\mathbf{x}, \mathbf{y}) \mathbf{dy}, \quad p(\mathbf{y}) = \textstyle\int_{\mathcal{S}_{\mathbf{X}}} p(\mathbf{x}, \mathbf{y}) \mathbf{dx}. \tag{2.7}$$

It follows that the joint PDF satisfies the normalization axiom of Kolmogorov:

$$\int_{\mathcal{S}_{\mathbf{X}}} \int_{\mathcal{S}_{\mathbf{Y}}} p(\mathbf{x}, \mathbf{y}) \mathbf{dydx} = 1. \tag{2.8}$$

Conditioning can also be defined for probability density functions, in analogy with conditional probabilities. Assuming that the marginal density $p(\mathbf{y})$ is strictly positive at $\mathbf{y}$ and using the multiplication rule in Equation (2.4), the conditional probability density function of $\mathbf{X}$ given $\mathbf{Y} = \mathbf{y}$ is defined as

$$p(\mathbf{x} \mid \mathbf{y}) := \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{y})}. \tag{2.9}$$

### 2.2.2 Bayes' rule and Bayesian inference

Consider two continuous random variables $\mathbf{X} : \Omega \to \mathcal{S}_\mathbf{X}$ and $\mathbf{Y} : \Omega \to \mathcal{S}_\mathbf{Y}$ that model, respectively, a state $\mathbf{x}$ to be estimated and a measurement $\mathbf{y}$ of this state. Equation (2.9) allows the joint probability density function to be factorized as

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x} \mid \mathbf{y})p(\mathbf{y}) = p(\mathbf{y} \mid \mathbf{x})p(\mathbf{x}). \tag{2.10}$$

Injecting the last term back in Equation (2.9) yields Bayes' rule.

▶ **Definition 2.5 (Bayes' rule).** The conditional probability density function of a random variable $\mathbf{X}$ given another random variable $\mathbf{Y}$ is

$$p(\mathbf{x} \mid \mathbf{y}) = \frac{p(\mathbf{y} \mid \mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}. \tag{2.11}$$

◀

The marginal density $p(\mathbf{y})$ can be obtained by marginalizing the joint density,

$$p(\mathbf{y}) = \int_{\mathcal{S}_\mathbf{X}} p(\mathbf{x}, \mathbf{y})\mathbf{dx} = \int_{\mathcal{S}_\mathbf{X}} p(\mathbf{y} \mid \mathbf{x})p(\mathbf{x})\mathbf{dx}. \tag{2.12}$$

Bayes' rule can therefore be written explicitly as

$$p(\mathbf{x} \mid \mathbf{y}) = \frac{p(\mathbf{y} \mid \mathbf{x})p(\mathbf{x})}{\int_{\mathcal{S}_\mathbf{X}} p(\mathbf{y} \mid \mathbf{x})p(\mathbf{x})\mathbf{dx}}. \tag{2.13}$$

In this formulation, $p(\mathbf{x})$ is called the *prior* density, as it encodes all *a priori* information available about the state $\mathbf{x}$ before observing the measurement. The conditional density $p(\mathbf{y} \mid \mathbf{x})$ is called the *likelihood* and is described by the measurement model. Finally, the conditional density $p(\mathbf{x} \mid \mathbf{y})$ is called the *posterior* density, as it incorporates both the prior information and the information brought by the measurement. Bayesian inference consists in computing the posterior density from the prior density and the measurement model.

▶ **Example 2.7.** Consider an autonomous robot that seeks to estimate the distance $x$ to an obstacle located somewhere within the interval $[5, 15]$ m in front of it. The robot has a sensor that measures $y = x + e$, where $e$ is measurement noise uniformly distributed over $[-a, a]$, with $a < 5$. The likelihood of observing $y$ given $x$ is thus

$$p(y \mid x) = \begin{cases} \frac{1}{2a} & \text{if } |y - x| \leq a, \\ 0 & \text{otherwise.} \end{cases}$$

The prior density is assumed uniform on $[5, 15]$,

$$p(x) = \begin{cases} \frac{1}{10} & \text{if } x \in [5, 15], \\ 0 & \text{otherwise.} \end{cases}$$

Upon receiving a measurement $y$, the robot updates its belief by computing the posterior density $p(x \mid y)$. Solving the integral in Equation (2.12) yields

$$p(y) = \begin{cases} \frac{y-5+a}{20a} & \text{if } 5 - a < y < 5 + a, \\ \frac{1}{10} & \text{if } 5 + a \leq y \leq 15 - a, \\ \frac{15-y+a}{20a} & \text{if } 15 - a < y < 15 + a, \\ 0 & \text{otherwise.} \end{cases}$$

Applying Bayes' rule then gives the posterior density in explicit piecewise form:

- If $5 - a < y < 5 + a$:

$$p(x \mid y) = \begin{cases} \frac{1}{y-5+a} & \text{if } x \in [5,\ y + a], \\ 0 & \text{otherwise.} \end{cases}$$

- If $5 + a \leq y \leq 15 - a$:

$$p(x \mid y) = \begin{cases} \frac{1}{2a} & \text{if } x \in [y - a,\ y + a], \\ 0 & \text{otherwise.} \end{cases}$$

- If $15 - a < y < 15 + a$:

$$p(x \mid y) = \begin{cases} \frac{1}{15-y+a} & \text{if } x \in [y - a,\ 15], \\ 0 & \text{otherwise.} \end{cases}$$

In all cases, the posterior density is uniform over the intersection $[5, 15] \cap [y-a, y+a]$ and zero elsewhere. ◀

Although simple in appearance, the previous example highlights that performing full Bayesian inference to compute the posterior can quickly become cumbersome. In practice, the denominator in Equation (2.13) is often intractable, making exact inference impossible. As a result, updating our prior knowledge of the state $\mathbf{x}$ using the information provided by the measurement $\mathbf{y}$ must often rely on approximations.

### 2.2.3 Moments of a continuous random variable

Probability density functions are often described by parametrized families of functions. The parameters of these families are frequently related to quantities known as the moments of the random variable. We begin by defining raw moments in the case of a scalar random variable.

▶ **Definition 2.6 ($n^{\text{th}}$ raw moment).** The $n^{\text{th}}$ raw moment of a continuous random variable $X : \Omega \to \mathbb{R}$ with probability density $p : \mathbb{R} \to \mathbb{R}^+$ is defined as

$$\mathrm{E}\{X^n\} = \int_{-\infty}^{+\infty} x^n p(x) dx. \tag{2.14}$$

◀

To gain intuition, it is useful to draw an analogy with mass distributions. If probability is interpreted as mass, then the zeroth moment corresponds to the total mass, while the first moment corresponds to the center of mass. The later plays a central role in probability theory and is called the mean of the random variable. It indicates where the probability distribution is centered. In the case of a multivariate random variable $\mathbf{X} : \Omega \to \mathbb{R}^n$, the mean takes the vector form

$$\mathrm{E}\{\mathbf{X}\} = \int_{\mathbb{R}^n} \mathbf{x} p(\mathbf{x}) \mathbf{dx}. \tag{2.15}$$

More generally, for any matrix-valued function $\mathbf{F}(\mathbf{x})$, the expectation is given by

$$\mathrm{E}\{\mathbf{F}(\mathbf{X})\} = \int_{\mathbb{R}^n} \mathbf{F}(\mathbf{x}) p(\mathbf{x}) \mathbf{dx}. \tag{2.16}$$

Having introduced the mean, we can now define central moments.

▶ **Definition 2.7 ($n^{\text{th}}$ central moment).** The $n^{\text{th}}$ central moment of a continuous random variable $X : \Omega \to \mathbb{R}$ with probability density $p : \mathbb{R} \to \mathbb{R}^+$ is defined as

$$\mathrm{E}\{(X - \mathrm{E}\{X\})^n\} = \int_{-\infty}^{+\infty} (x - \mathrm{E}\{X\})^n p(x) dx. \tag{2.17}$$

◀

Whereas raw moments describe the location of a distribution, central moments characterize its spread and shape. In particular, the second central moment quan-

tifies how the probability mass is distributed around the mean. In the univariate case, it is called the variance and is denoted by $\mathrm{Var}\{X\}$. For a multivariate random variable $\mathbf{X} : \Omega \to \mathbb{R}^n$, the second central moment takes the form of a matrix, called the covariance matrix, and is defined as

$$\mathrm{Cov}\{\mathbf{X}\} = \int_{\mathbb{R}^n} (\mathbf{x} - \mathrm{E}\{\mathbf{X}\})(\mathbf{x} - \mathrm{E}\{\mathbf{X}\})^T p(\mathbf{x}) \mathbf{dx}. \tag{2.18}$$

The covariance matrix admits the decomposition

$$\mathrm{Cov}\{\mathbf{X}\} = \mathrm{E}\{\mathbf{X}\mathbf{X}^T\} - \mathrm{E}\{\mathbf{X}\}\mathrm{E}\{\mathbf{X}\}^T. \tag{2.19}$$

One can also define the covariance between two random variables $\mathbf{X} : \Omega \to \mathbb{R}^n$ and $\mathbf{Y} : \Omega \to \mathbb{R}^m$ as

$$\mathrm{Cov}\{\mathbf{X}, \mathbf{Y}\} = \int_{\mathbb{R}^n} \int_{\mathbb{R}^m} (\mathbf{x} - \mathrm{E}\{\mathbf{X}\})(\mathbf{y} - \mathrm{E}\{\mathbf{Y}\})^T p(\mathbf{x}, \mathbf{y}) \mathbf{dydx}. \tag{2.20}$$

This expression can similarly be decomposed as

$$\mathrm{Cov}\{\mathbf{X}, \mathbf{Y}\} = \mathrm{E}\{\mathbf{X}\mathbf{Y}^T\} - \mathrm{E}\{\mathbf{X}\}\mathrm{E}\{\mathbf{Y}\}^T. \tag{2.21}$$

### 2.2.4 Statistical independence and uncorrelatedness

Two continuous random variables $\mathbf{X} : \Omega \to \mathcal{S}_\mathbf{X}$ and $\mathbf{Y} : \Omega \to \mathcal{S}_\mathbf{Y}$ are said to be statistically independent if their joint probability density function factors into the product of their marginal densities, that is,

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y}). \tag{2.22}$$

They are said to be uncorrelated if their covariance vanishes,

$$\mathrm{Cov}\{\mathbf{X}, \mathbf{Y}\} = \mathbf{0}, \tag{2.23}$$

or, equivalently, if

$$\mathrm{E}\{\mathbf{X}\mathbf{Y}^T\} = \mathrm{E}\{\mathbf{X}\}\mathrm{E}\{\mathbf{Y}\}^T. \tag{2.24}$$

Statistical independence is a stronger property than uncorrelatedness: independence always implies zero cross-correlation. This can be shown as follows:

$$\mathrm{E}\{\mathbf{X}\mathbf{Y}^T\} = \int_{\mathcal{S}_\mathbf{X}} \int_{\mathcal{S}_\mathbf{Y}} \mathbf{x}\mathbf{y}^T p(\mathbf{x}, \mathbf{y}) \mathbf{dy}\mathbf{dx}, \tag{2.25a}$$

$$= \int_{\mathcal{S}_\mathbf{X}} \int_{\mathcal{S}_\mathbf{Y}} \mathbf{x}\mathbf{y}^T p(\mathbf{x}) p(\mathbf{y}) \mathbf{dy}\mathbf{dx}, \tag{2.25b}$$

$$= \left( \int_{\mathcal{S}_\mathbf{X}} \mathbf{x} p(\mathbf{x}) \mathbf{dx} \right) \left( \int_{\mathcal{S}_\mathbf{Y}} \mathbf{y}^T p(\mathbf{y}) \mathbf{dy} \right), \tag{2.25c}$$

$$= \mathrm{E}\{\mathbf{X}\}\mathrm{E}\{\mathbf{Y}\}^T, \tag{2.25d}$$

where the second equality follows from the assumption of statistical independence. The converse does not hold in general: uncorrelated random variables are not necessarily statistically independent.

▶ **Example 2.8.** Let $X$ be a random variable uniformly distributed on $[-1, 1]$, and define another random variable $Y = X^2$. The random variables $X$ and $Y$ are statistically dependent, since $Y$ is entirely determined by $X$. However,

$$\mathrm{E}\{X\} = 0, \quad \mathrm{E}\{Y\} = \mathrm{E}\{X^2\} = \frac{1}{3}, \quad \mathrm{E}\{XY\} = \mathrm{E}\{X^3\} = 0.$$

It follows that $\mathrm{E}\{XY\} = \mathrm{E}\{X\}\mathrm{E}\{Y\}$, showing that $X$ and $Y$ are uncorrelated despite being statistically dependent. ◀

## 2.3 Gaussian random variables

Among random variables, one family is ubiquitous across many fields of application: the family of Gaussian random variables. Gaussian distributions naturally arise when modeling uncertainty and noise, as they provide a compact and mathematically tractable description of random fluctuations around a nominal value.

▶ **Definition 2.8 (Gaussian probability density function).** A continuous random variable $\mathbf{X} : \Omega \to \mathbb{R}^n$ is said to follow a Gaussian probability density function if its probability density is given by

$$p(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n \det(\boldsymbol{\Sigma})}} \exp\left( -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right), \tag{2.26}$$

where $\boldsymbol{\mu} = \mathrm{E}\{\mathbf{X}\}$ is the mean vector and $\boldsymbol{\Sigma} = \mathrm{Cov}\{\mathbf{X}\}$ is the covariance matrix, which is assumed to be symmetric and positive definite. ◀

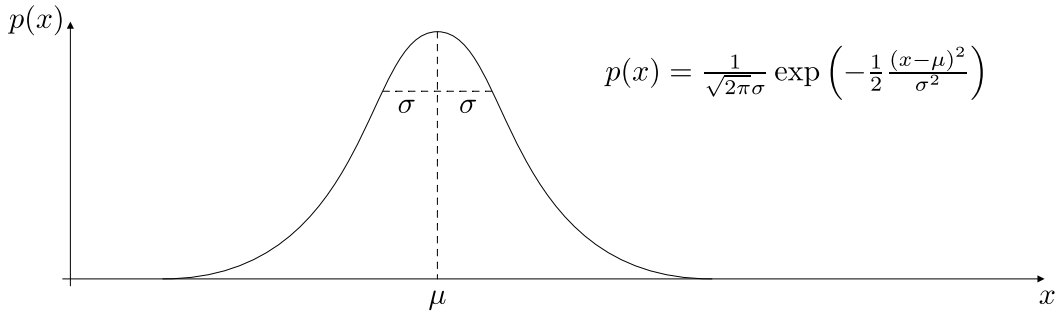



**Figure 2.3:** Probability density function of a scalar Gaussian random variable.

To indicate that a random variable $\mathbf{X}$ follows a Gaussian (or normal) distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, we use the notations

$$\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}). \tag{2.27}$$

Strictly speaking, the density of such a variable should be written as $p(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$. Throughout this thesis, this conditioning is left implicit to simplify the notation.

An illustration of a Gaussian probability density function is provided in Figure 2.3 for a scalar random variable. In this case, the Gaussian density simplifies to

$$p(x) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}\right), \tag{2.28}$$

where $\mu = \mathrm{E}\{\mathbf{X}\}$, $\sigma^2 = \mathrm{Var}\{\mathbf{X}\}$, and $\sigma$ is called the standard deviation. The Gaussian density is symmetric about its mean, and the extent of its spread around the mean is encoded by the variance, or by the covariance matrix in the multivariate case. Larger values of $\sigma^2$, or larger eigenvalues of $\boldsymbol{\Sigma}$, correspond to distributions that are more spread out from the mean.

Gaussian probability density functions are commonly used to model stochastic behavior, such as noise. For modeling purposes, it is useful to recall that if $X \sim \mathcal{N}(\mu, \sigma^2)$, then

$$\Pr(\mu - \sigma \leq X \leq \mu + \sigma) \approx 68.27\%, \tag{2.29a}$$
$$\Pr(\mu - 2\sigma \leq X \leq \mu + 2\sigma) \approx 95.45\%, \tag{2.29b}$$
$$\Pr(\mu - 3\sigma \leq X \leq \mu + 3\sigma) \approx 99.73\%. \tag{2.29c}$$

### 2.3.1 Joint density and Bayesian inference

Gaussian distributions play a central role in probabilistic modeling and Bayesian estimation, as they admit simple analytical expressions and are closed under marginalization and conditioning. Consider two random variables $\mathbf{X} : \Omega \to^n$ and $\mathbf{Y} : \Omega \to^m$ that are jointly Gaussian:

$$(\mathbf{X}, \mathbf{Y}) \sim \mathcal{N}\left( \begin{bmatrix} \boldsymbol{\mu}_\mathbf{x} \\ \boldsymbol{\mu}_\mathbf{y} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}} \\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}} \end{bmatrix} \right), \tag{2.30}$$

where $\boldsymbol{\Sigma}_{\mathbf{xy}} = \boldsymbol{\Sigma}_{\mathbf{yx}}^T = \mathrm{E}\{(\mathbf{X} - \mathrm{E}\{\mathbf{X}\})(\mathbf{Y} - \mathrm{E}\{\mathbf{Y}\})^T\}$ is the cross-covariance matrix. We now show that the marginal densities $p(\mathbf{x})$ and $p(\mathbf{y})$, as well as the conditional density $p(\mathbf{x} \mid \mathbf{y})$, are also Gaussian.

Using the Schur complement of $\boldsymbol{\Sigma}_{\mathbf{yy}}$, the covariance matrix can be factored as

$$\begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}} \\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\mathbf{yy}} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}} & \mathbf{I} \end{bmatrix}. \tag{2.31}$$

This allows the quadratic form in the exponent of the joint density to be written as

$$\begin{aligned} \begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_\mathbf{x} \\ \mathbf{y} - \boldsymbol{\mu}_\mathbf{y} \end{bmatrix}^T \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}} \\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{x} - \boldsymbol{\mu}_\mathbf{x} \\ \mathbf{y} - \boldsymbol{\mu}_\mathbf{y} \end{bmatrix} = \\ \big(\mathbf{x} - \boldsymbol{\mu}_\mathbf{x} - \boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}(\mathbf{y} - \boldsymbol{\mu}_\mathbf{y})\big)^T \big(\boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}\big)^{-1} \times \\ \big(\mathbf{x} - \boldsymbol{\mu}_\mathbf{x} - \boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}(\mathbf{y} - \boldsymbol{\mu}_\mathbf{y})\big) + (\mathbf{y} - \boldsymbol{\mu}_\mathbf{y})^T \boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}(\mathbf{y} - \boldsymbol{\mu}_\mathbf{y}), \end{aligned} \tag{2.32}$$

which is the sum of two quadratic forms: $C_1(\mathbf{x}, \mathbf{y})$, which depends on both $\mathbf{x}$ and $\mathbf{y}$, and $C_2(\mathbf{y})$, which depends only on $\mathbf{y}$. Using the determinant identity associated with the Schur complement of $\boldsymbol{\Sigma}_{\mathbf{yy}}$,

$$\det\left( \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}} \\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}} \end{bmatrix} \right) = \det(\boldsymbol{\Sigma}_{\mathbf{yy}}) \det\big(\boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}\big), \tag{2.33}$$

marginalizing the joint density over $\mathbf{x}$ yields

$$p(\mathbf{y}) = \int_{\mathbb{R}^n} p(\mathbf{x}, \mathbf{y}) \mathbf{dx}, \tag{2.34a}$$

$$= \frac{\exp\big(-\frac{1}{2}C_2(\mathbf{y})\big) \int_{\mathbb{R}^n} \exp\big(-\frac{1}{2}C_1(\mathbf{x}, \mathbf{y})\big) \mathbf{dx}}{\sqrt{(2\pi)^{n+m} \det(\boldsymbol{\Sigma}_{\mathbf{yy}}) \det\big(\boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}\big)}}, \tag{2.34b}$$

$$= \frac{\exp\left(-\frac{1}{2}C_2(\mathbf{y})\right)\sqrt{(2\pi)^n \det\left(\boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}\right)}}{\sqrt{(2\pi)^{n+m}\det(\boldsymbol{\Sigma}_{\mathbf{yy}})\det\left(\boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}\right)}}, \tag{2.34c}$$

$$= \frac{\exp\left(-\frac{1}{2}C_2(\mathbf{y})\right)}{\sqrt{(2\pi)^m \det(\boldsymbol{\Sigma}_{\mathbf{yy}})}}, \tag{2.34d}$$

where the integral is evaluated using Equation (2.5). This yields

$$\mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{y}}, \boldsymbol{\Sigma}_{\mathbf{yy}}). \tag{2.35}$$

A similar derivation, using the Schur complement of $\boldsymbol{\Sigma}_{\mathbf{xx}}$ and marginalizing over $\mathbf{y}$, gives the marginal density $p(\mathbf{x})$ and the associated marginal distribution.

Finally, since $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x} \mid \mathbf{y})p(\mathbf{y})$, combining Equation (2.34) and Equation (2.32) allows us to identify the conditional distribution

$$\mathbf{X} \mid \mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}} + \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}(\mathbf{y} - \boldsymbol{\mu}_{\mathbf{y}}), \boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}), \tag{2.36}$$

which provides the explicit expression for Bayesian inference with Gaussian probability density functions.

### 2.3.2 Independence and uncorrelatedness

In the case of Gaussian random variables, statistical independence and uncorrelatedness are equivalent notions: each implies the other. We have already shown in Section 2.2.4 that statistical independence implies uncorrelatedness. We now show that, for Gaussian random variables, the converse is also true.

Let us consider two Gaussian random variables $\mathbf{X} : \Omega \to \mathbb{R}^n$ and $\mathbf{Y} : \Omega \to \mathbb{R}^m$ that are uncorrelated. Their joint probability density can always be written as $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x} \mid \mathbf{y})p(\mathbf{y})$. Since $\mathbf{X}$ and $\mathbf{Y}$ are uncorrelated, their cross-covariance matrix satisfies

$$\boldsymbol{\Sigma}_{\mathbf{xy}} = \mathrm{E}\{(\mathbf{X} - \mathrm{E}\{\mathbf{X}\})(\mathbf{Y} - \mathrm{E}\{\mathbf{Y}\})^T\} = \mathrm{E}\{\mathbf{X}\mathbf{Y}^T\} - \mathrm{E}\{\mathbf{X}\}\mathrm{E}\{\mathbf{Y}\}^T = \mathbf{0}. \tag{2.37}$$

Substituting $\boldsymbol{\Sigma}_{\mathbf{xy}} = \mathbf{0}$ into Equation (2.36) yields

$$\mathbf{X} \mid \mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{xx}}), \tag{2.38}$$

in such a way that $p(\mathbf{x} \mid \mathbf{y}) = p(\mathbf{x})$. The joint density thus factorizes as $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x})p(\mathbf{y})$, proving that $\mathbf{X}$ and $\mathbf{Y}$ are statistically independent.

### 2.3.3 Linear transformations

Consider a random variable $\mathbf{X} : \Omega \to \mathbb{R}^n$ distributed according to the Gaussian

$$\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{x}, \boldsymbol{\Sigma}_{\mathbf{xx}}). \tag{2.39}$$

A fundamental property of Gaussian random variables is that they are closed under affine transformations. In particular, for any matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ and vector $\mathbf{b} \in \mathbb{R}^m$, both assumed known and constant, the random variable

$$\mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{b} \tag{2.40}$$

is also Gaussian. The parameters of the resulting distribution follow directly from the definitions of the mean and covariance. The mean of $\mathbf{Y}$ is

$$\mathrm{E}\{\mathbf{Y}\} = \mathrm{E}\{\mathbf{A}\mathbf{X} + \mathbf{b}\} = \mathbf{A}\mathrm{E}\{\mathbf{X}\} + \mathbf{b} = \mathbf{A}\boldsymbol{\mu}_\mathbf{x} + \mathbf{b}, \tag{2.41}$$

and its covariance matrix is

$$\begin{aligned} \mathrm{Cov}\{\mathbf{Y}\} &= \mathrm{E}\big\{(\mathbf{Y} - \mathrm{E}\{\mathbf{Y}\})(\mathbf{Y} - \mathrm{E}\{\mathbf{Y}\})^T\big\} \\ &= \mathrm{E}\big\{(\mathbf{A}\mathbf{X} - \mathbf{A}\boldsymbol{\mu}_\mathbf{x})(\mathbf{A}\mathbf{X} - \mathbf{A}\boldsymbol{\mu}_\mathbf{x})^T\big\} \\ &= \mathbf{A}\mathrm{E}\big\{(\mathbf{X} - \boldsymbol{\mu}_\mathbf{x})(\mathbf{X} - \boldsymbol{\mu}_\mathbf{x})^T\big\}\mathbf{A}^T \\ &= \mathbf{A}\boldsymbol{\Sigma}_{\mathbf{xx}}\mathbf{A}^T. \end{aligned} \tag{2.42}$$

Hence, $\mathbf{Y} \sim \mathcal{N}(\mathbf{A}\boldsymbol{\mu}_\mathbf{x} + \mathbf{b}, \mathbf{A}\boldsymbol{\Sigma}_{\mathbf{xx}}\mathbf{A}^T)$.

More generally, let $\mathbf{X} : \Omega \to \mathbb{R}^n$ and $\mathbf{Y} : \Omega \to \mathbb{R}^m$ be two Gaussian random variables such that

$$\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{x}, \boldsymbol{\Sigma}_{\mathbf{xx}}), \quad \mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{y}, \boldsymbol{\Sigma}_{\mathbf{yy}}). \tag{2.43}$$

Then, for any known matrices $\mathbf{A} \in \mathbb{R}^{l \times n}$ and $\mathbf{B} \in \mathbb{R}^{l \times m}$, the random variable

$$\mathbf{Z} = \mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{Y} \tag{2.44}$$

is also Gaussian. Its mean is given by

$$\mathrm{E}\{\mathbf{Z}\} = \mathrm{E}\{\mathbf{A}\mathbf{X} + \mathbf{B}\mathbf{Y}\} = \mathbf{A}\mathrm{E}\{\mathbf{X}\} + \mathbf{B}\mathrm{E}\{\mathbf{Y}\} = \mathbf{A}\boldsymbol{\mu}_\mathbf{x} + \mathbf{B}\boldsymbol{\mu}_\mathbf{y}, \tag{2.45}$$

and its covariance matrix is

$$\begin{aligned}\mathrm{Cov}\{\mathbf{Z}\} &= \mathrm{E}\big\{(\mathbf{Z}-\mathrm{E}\{\mathbf{Z}\})(\mathbf{Z}-\mathrm{E}\{\mathbf{Z}\})^T\big\},\\ &= \mathbf{A}\boldsymbol{\Sigma}_{\mathbf{xx}}\mathbf{A}^T + \mathbf{B}\boldsymbol{\Sigma}_{\mathbf{yy}}\mathbf{B}^T + \mathbf{A}\boldsymbol{\Sigma}_{\mathbf{xy}}\mathbf{B}^T + \mathbf{B}\boldsymbol{\Sigma}_{\mathbf{yx}}\mathbf{A}^T,\end{aligned} \tag{2.46}$$

where $\boldsymbol{\Sigma}_{\mathbf{xy}} = \mathrm{Cov}\{\mathbf{X},\mathbf{Y}\}$ and $\boldsymbol{\Sigma}_{\mathbf{yx}} = \boldsymbol{\Sigma}_{\mathbf{xy}}^T$. In the special case where $\mathbf{X}$ and $\mathbf{Y}$ are statistically independent, the cross-covariance terms vanish, yielding

$$\mathbf{Z} \sim \mathcal{N}(\mathbf{A}\boldsymbol{\mu}_{\mathbf{x}} + \mathbf{B}\boldsymbol{\mu}_{\mathbf{y}}, \mathbf{A}\boldsymbol{\Sigma}_{\mathbf{xx}}\mathbf{A}^T + \mathbf{B}\boldsymbol{\Sigma}_{\mathbf{yy}}\mathbf{B}^T). \tag{2.47}$$

## 2.4 Stochastic processes

So far, we have considered random variables that describe uncertainty at a single, fixed point in time and space. In many real-world situations, however, the underlying random experiment evolves over time, space, or both, causing the statistical properties of the associated random variables to change. Such evolving randomness is captured by the notion of a stochastic process.

▶ **Definition 2.9 (Stochastic process).** A stochastic process (or random process) with index set $\mathcal{T}$ and state space $(\mathcal{S},\mathcal{B})$, defined on a probability space $(\Omega,\mathcal{F},\Pr)$, is a sequence of random variables

$$(\mathbf{X}_k)_{k\in\mathcal{T}}, \tag{2.48}$$

where each

$$\mathbf{X}_k : \Omega \to \mathcal{S} \tag{2.49}$$

is a random variable taking values in $\mathcal{S}$. ◀

This definition leaves the interpretation of the index set $\mathcal{T}$ open. In most applications, including those considered in this thesis, the index represents time. One typically distinguishes between discrete-time and continuous-time stochastic processes, depending on whether $\mathcal{T}$ is countable or uncountable. In this work, we focus exclusively on discrete-time stochastic processes.

An alternative and often useful perspective is to view a stochastic process as a probability distribution over a space of trajectories, where each trajectory represents a possible evolution of the state of a system over time. In deterministic systems, this trajectory is uniquely determined. In contrast, a stochastic process admits many possible trajectories, each occurring with a certain probability.

In the context of filtering and state estimation, a useful mental picture is that of an unknown underlying trajectory that the system follows, while only partial and

noisy information about this trajectory is available through observations. The goal of filtering is then to infer, as accurately as possible, the current state and future evolution of this hidden trajectory based on the measurements collected so far.

▶ **Example 2.9.** Consider a person who has had a bit too much to drink and is attempting to walk home from a Christmas market. Her home lies somewhere along a long, straight street. At each step, she moves either one unit forward or one unit backward, each with probability $0.5$, independently of all previous steps. Let time be indexed by the step count. The sequence of steps can be modeled as a process $(X_k)_{k\geq 1}$, where each $X_k$ is an independent random variable satisfying

$$\Pr(X_k = 1) = \Pr(X_k = -1) = \frac{1}{2}.$$

Here, the value $1$ represents a step forward, while $-1$ represents a step backward. This process is a classical example known as a *random walk*. Modeling the motion as a stochastic process provides a probabilistic framework for reasoning about questions such as whether the person is more likely to eventually reach her home or instead wander far enough in the opposite direction to give up and call a cab. ◀

### 2.4.1 Gaussian processes

Just as Gaussian random variables play a central role in probability theory, their extension to stochastic processes leads to the important class of *Gaussian processes*.

▶ **Definition 2.10.** A discrete-time stochastic process $(\mathbf{X}_k)_{k\in\mathcal{T}}$ is said to be Gaussian if all its finite-dimensional marginal distributions are Gaussian. That is, for any finite set of indices $k_1, \ldots, k_N \in \mathcal{T}$, the stacked random vector

$$\begin{bmatrix} \mathbf{X}_{k_1} \\ \vdots \\ \mathbf{X}_{k_N} \end{bmatrix} \tag{2.50}$$

follows a multivariate Gaussian distribution. ◀

In particular, each individual random variable $\mathbf{X}_k$ is a Gaussian random vector,

$$\mathbf{X}_k \sim \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_{kk}), \tag{2.51}$$

where the mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_{kk}$ may depend on time. However, the converse is not true: the fact that each $\mathbf{X}_k$ is Gaussian does not, by itself, imply

that the process is Gaussian, since the joint distributions across time must also be Gaussian.

The temporal dependence structure of a Gaussian process is fully characterized by its cross-covariance matrices

$$\mathbf{\Sigma}_{kl} = \mathrm{Cov}\{\mathbf{X}_k, \mathbf{X}_l\}, \qquad k, l \in \mathcal{T}. \tag{2.52}$$

An important class of Gaussian processes is *Gaussian white noise*, which is widely used to model process and measurement noises in dynamical systems.

▶ **Definition 2.11 (Zero-mean Gaussian white noise).** A discrete-time stochastic process $(\mathbf{w}_k)_{k\in\mathcal{T}}$, with $\mathbf{w}_k \in \mathbb{R}^n$, is called a zero-mean Gaussian white noise process if it is Gaussian and satisfies

$$\mathrm{E}\{\mathbf{w}_k\} = \mathbf{0}, \qquad \mathbf{\Sigma}_{kl} = \begin{cases} \mathbf{\Sigma}_{kk} & \text{if } k = l, \\ \mathbf{0} & \text{if } k \neq l. \end{cases} \tag{2.53}$$

for some positive semi-definite matrix $\mathbf{\Sigma}_{kk}$. ◀

The term white refers to the absence of temporal correlation. Since the process is Gaussian, this implies statistical independence between samples at different time indices. Gaussian white noise therefore provides a convenient and tractable model for temporally uncorrelated stochastic disturbances acting on a system.

### 2.4.2 Markov processes

While stochastic processes describe how uncertainty evolves over time, an essential modeling question concerns how the current state depends on past states. In many dynamical systems, the influence of the past can be summarized by a limited history, rather than the entire trajectory. This motivates the notion of a *Markov chain*.

▶ **Definition 2.12 (Markov chain of order $m$).** A discrete-time stochastic process $(\mathbf{X}_k)_{k\in\mathcal{T}}$ is called a Markov chain of order $m \in \mathbb{N}$ if, for all $k \in \mathcal{T}$ and all realizations $\mathbf{x}_1, \ldots, \mathbf{x}_k$, it satisfies

$$p(\mathbf{x}_k \mid \mathbf{x}_{1:k-1}) = p(\mathbf{x}_k \mid \mathbf{x}_{k-m:k-1}), \tag{2.54}$$

with $\mathbf{x}_{i:j}$ the sequence of realized values of the process from time index $i$ to $j$. ◀

In other words, the conditional distribution of the current state $\mathbf{X}_k$ depends only on the $m$ most recent past states and is independent of all earlier states.

This property makes Markov chains a natural modeling choice for many causal dynamical systems arising in practice.

▶ **Example 2.10.** Consider a robot that aims to estimate the position of a human co-worker at discrete time intervals. For sufficiently small time steps, it is reasonable to assume that the position at the current time depends only on a limited number of previous time steps, rather than on the entire past trajectory. The motion can therefore be modeled as a Markov chain of order $m$, where the value of $m$ reflects the desired level of modeling fidelity. Setting $m = 1$ yields a first-order Markov model, in which the current position depends only on the most recent position. Larger values of $m$ allow the model to capture richer motion characteristics, such as velocity or acceleration, thereby accounting for inertial effects. Alternatively, these quantities can be explicitly included in the state vector, resulting in a first-order Markov model with an augmented state. ◀

### 2.4.3 Hidden Markov models

In many applications, the system state of interest cannot be observed directly. Instead, one has access only to indirect and noisy measurements that provide partial information about the underlying state. A complete probabilistic description must therefore account for both the state dynamics and the measurement process. Hidden Markov models (HMMs) provide a principled framework for this purpose by combining a Markovian dynamical model with a probabilistic observation model.

▶ **Definition 2.13 (Hidden Markov model).** Consider the discrete-time random processes $(\mathbf{X}_k)_{k\in\mathcal{T}}$ and $(\mathbf{Y}_k)_{k\in\mathcal{T}}$. The pair $(\mathbf{X}_k, \mathbf{Y}_k)$ forms a hidden Markov model if

1. the state process $(\mathbf{X}_k)_{k\in\mathcal{T}}$ is a first-order Markov chain whose realizations are not directly observable,
2. the observation process $(\mathbf{Y}_k)_{k\in\mathcal{T}}$ satisfies the conditional independence property $p(\mathbf{y}_k \mid \mathbf{x}_{1:k}, \mathbf{y}_{1:k-1}) = p(\mathbf{y}_k \mid \mathbf{x}_k)$, for all $k \in \mathcal{T}$.

◀

The process $(\mathbf{X}_k)$ is referred to as the latent state process and models the hidden dynamics of the system. Although the Markov property restricts dependence to the previous state, the state evolution may also involve known control inputs and stochastic disturbances that account for modeling inaccuracies. The observation

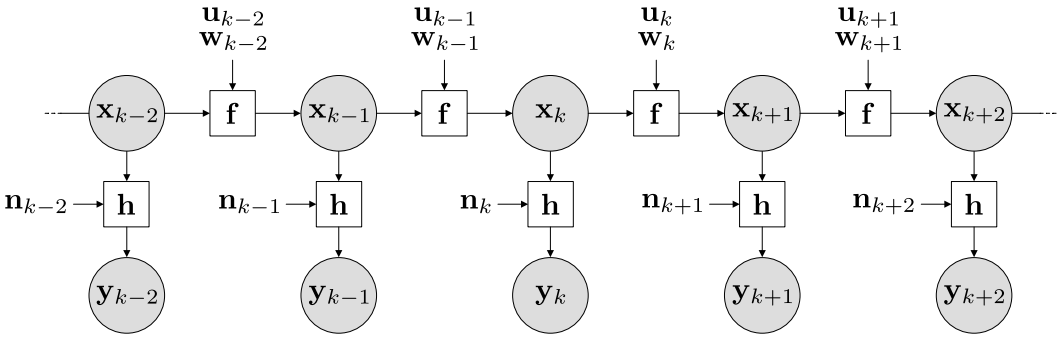



**Figure 2.4:** Block diagram of a hidden Markov model. The random vector $\mathbf{x}_k$ represents the latent state at time index $k$, $\mathbf{y}_k$ denotes the observation, $\mathbf{u}_k$ is the system input, and $\mathbf{w}_k$ and $\mathbf{n}_k$ respectively represent the process and measurement noises. The dynamics function $\mathbf{f}$ and the measurement function $\mathbf{h}$ may be nonlinear.

process $(\mathbf{Y}_k)$ provides the only source of information about the state and is typically corrupted by measurement noise. The structure of a hidden Markov model is illustrated in Figure 2.4.

▶ **Example 2.11.** Consider a robotic arm collaborating with a human worker. To define a danger zone that must be avoided in order to prevent collisions, the robot estimates the planar position $\mathbf{p}_k$ and velocity $\mathbf{v}_k$ of the worker, expressed in an inertial reference frame attached to its base. The latent state is thus defined as

$$\mathbf{x}_k = (\mathbf{v}_k, \mathbf{p}_k) \in \mathbb{R}^4.$$

The robot assumes constant-velocity motion and has access to a GPS beacon mounted on the worker's body, providing a coarse measurement of the planar position. This situation can be modeled as a hidden Markov chain governed by

$$\begin{aligned}\mathbf{x}_k &= \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{I}\,dt & \mathbf{I}\end{bmatrix}\mathbf{x}_{k-1} + \begin{bmatrix}\mathbf{w}_{k-1}\\ \mathbf{0}\end{bmatrix},\\ \mathbf{y}_k &= \begin{bmatrix}\mathbf{0} & \mathbf{I}\end{bmatrix}\mathbf{x}_k + \mathbf{n}_k,\end{aligned}$$

where $dt$ denotes the sampling period. The process noise $(\mathbf{w}_{k-1}, \mathbf{0}) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$ accounts for deviations from the constant-velocity model, while the measurement noise $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ captures the uncertainty inherent to the GPS sensor. ◀

### Recursive Bayesian estimation

Under the assumptions of a hidden Markov model, Bayesian inference can be carried out recursively. Specifically, the posterior distribution $p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$ can be computed from the previous posterior $p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1})$ using a two-stage procedure consisting of a prediction stage and an update stage. Starting from Bayes' rule,

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k \mid \mathbf{x}_k, \mathbf{y}_{1:k-1}) p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})}{\int_{\mathbb{R}^n} p(\mathbf{y}_k \mid \mathbf{x}_k, \mathbf{y}_{1:k-1}) p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) \mathbf{dx}_k}, \tag{2.55a}$$

$$= \frac{p(\mathbf{y}_k \mid \mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})}{\int_{\mathbb{R}^n} p(\mathbf{y}_k \mid \mathbf{x}_k) p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) \mathbf{dx}_k}, \tag{2.55b}$$

where the second equality follows from the conditional independence property of the observation process. The remaining term $p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1})$ corresponds to the predicted prior and can be expressed by marginalizing over the previous state:

$$p(\mathbf{x}_k \mid \mathbf{y}_{1:k-1}) = \int_{\mathbb{R}^n} p(\mathbf{x}_k, \mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) \mathbf{dx}_{k-1}, \tag{2.56a}$$

$$= \int_{\mathbb{R}^n} p(\mathbf{x}_k \mid \mathbf{x}_{k-1}, \mathbf{y}_{1:k-1}) p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) \mathbf{dx}_{k-1}, \tag{2.56b}$$

$$= \int_{\mathbb{R}^n} p(\mathbf{x}_k \mid \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1}) \mathbf{dx}_{k-1}, \tag{2.56c}$$

where the last line follows from the first-order Markov property of the state process.

Thus, recursive Bayesian estimation proceeds by first propagating the previous posterior through the system dynamics, using the transition model $p(\mathbf{x}_k \mid \mathbf{x}_{k-1})$, to obtain the prior at time $k$ (prediction step). This prior is then combined with the new measurement through the observation model $p(\mathbf{y}_k \mid \mathbf{x}_k)$ to yield the posterior at time $k$ (update step). This recursive structure forms the foundation of the Kalman filter and its extensions, which are introduced in the next chapter.

# 3 Kalman filtering in Euclidean spaces

The Kalman filter is one of the most widely used state estimation algorithms, with applications spanning many fields, including robotics, control, signal processing, aerospace, and finance. It is named after Rudolf E. Kalman, who in 1960 published a paper that reshaped modern estimation theory and helped establish the state-space viewpoint as a central tool in modern estimation [12]. Although the method was not immediately adopted across engineering practice, its impact became clear once it was implemented in demanding real-world settings, notably in aerospace navigation and tracking. Early work at NASA, including contributions by Stanley F. Schmidt, helped establish the Kalman filter as a practical tool for real-time trajectory estimation [56, 57], including within the Apollo program.

The Kalman filter did not emerge in isolation. Related ideas appeared earlier in different forms and communities. In the late 1950s, Peter Swerling, motivated by satellite orbit estimation, developed recursive least-squares procedures with stagewise error propagation, in which an error covariance is carried through a dynamical model [58]. Going further back, Thorvald N. Thiele introduced linear time-series models that combine deterministic components with stochastic dynamics and measurement noise, and derived recursive estimation and prediction procedures that anticipate key elements of Kalman filtering [59, 60]. In parallel, foundational developments in continuous-time stochastic estimation led to the Kalman–Bucy filter, named to reflect Richard S. Bucy's major contributions to the continuous-time theory [61].

Against this historical backdrop, we focus on the discrete-time formulation used in most modern systems. Specifically, this chapter presents the Kalman filter for discrete-time linear systems and its extension to nonlinear systems, the extended Kalman filter (EKF). For contemporary perspectives on Kalman filtering, see [62, 63]; for more recent pedagogical introductions, see [64–66]. This chapter also draws, in part, on the robotics-oriented viewpoint of Barfoot [55].

## 3.1 The linear Kalman filter

The Kalman filter seeks to estimate the hidden state $\mathbf{x}_k \in \mathbb{R}^n$ of a linear dynamical system represented as a hidden Markov model. The system dynamics and

measurement equations are given by

$$\mathbf{x}_k = \mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1} + \mathbf{w}_{k-1}, \tag{3.1a}$$
$$\mathbf{y}_k = \mathbf{H}_k\mathbf{x}_k + \mathbf{n}_k, \tag{3.1b}$$

where $\mathbf{u}_{k-1} \in \mathbb{R}^b$ is the system input, $\mathbf{y}_k \in \mathbb{R}^m$ is the measurement vector, and $\mathbf{F}_{k-1} \in \mathbb{R}^{n\times n}$, $\mathbf{B}_{k-1} \in \mathbb{R}^{n\times b}$, and $\mathbf{H}_k \in \mathbb{R}^{m\times n}$ are known matrices. The terms $\mathbf{w}_{k-1}$ and $\mathbf{n}_k$ represent the process and measurement noises, respectively. They are assumed to be zero-mean, mutually uncorrelated Gaussian white noise processes,

$$\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1}), \quad \mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k), \quad \mathrm{E}\{\mathbf{w}_l \mathbf{n}_m^T\} = \mathbf{0}, \tag{3.2}$$

for all $k, l, m \in \mathbb{N}$.

The Kalman filter models uncertainty through the estimation error

$$\mathbf{e}_{k|l} = \mathbf{x}_k - \hat{\mathbf{x}}_{k|l}, \tag{3.3}$$

which is assumed to be a zero-mean Gaussian random vector,

$$\mathbf{e}_{k|l} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{k|l}), \tag{3.4}$$

where $\hat{\mathbf{x}}_{k|l}$ denotes the estimate of the state at time index $k$ based on measurements available up to time index $l$. The estimation error is assumed to be independent of both the process and measurement noises. By definition of the estimation error, the true state $\mathbf{x}_k$ is assumed to follow the Gaussian distribution

$$\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_{k|l}, \mathbf{\Sigma}_{k|l}). \tag{3.5}$$

The Kalman filter is a recursive Bayesian estimator. At each time step, its belief about the system state is fully characterized by the Gaussian distribution $\mathcal{N}(\hat{\mathbf{x}}_{k|l}, \mathbf{\Sigma}_{k|l})$, which is propagated forward in time and updated as new measurements become available. The Kalman filter equations follow directly from the linear, Gaussian, and hidden Markov model assumptions: they are obtained by applying the rules of recursive Bayesian inference, as described in Section 2.4.3, to this special case. Stanley F. Schmidt is credited with recognizing that this estimation procedure fits the recursive Bayesian framework and naturally decomposes into two stages, prediction and update [57].

### 3.1.1 The prediction stage

The prediction stage of the Kalman filter, also called propagation, consists in propagating the current estimate through the system dynamics between two measurement updates. Assume that we are at time index $k-1$ and that all measurements up to this time have been incorporated. The posterior distribution is given by

$$\mathbf{x}_{k-1} \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-2}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k-1|k-1}, \boldsymbol{\Sigma}_{k-1|k-1}). \tag{3.6}$$

Because the dynamics are linear, the system input $\mathbf{u}_{k-1}$ is deterministic, and the process noise is a zero-mean white Gaussian noise process, the propagated distribution remains Gaussian, as shown in Section 2.3.3. It is fully characterized by the pair of its two first moments $(\hat{\mathbf{x}}_{k|k-1}, \boldsymbol{\Sigma}_{k|k-1})$. The state dynamics in Equation (3.1a), the predicted state estimate is given by

$$\begin{aligned}
\hat{\mathbf{x}}_{k|k-1} &= \mathrm{E}\big\{\mathbf{x}_k \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}\big\}, \\
&= \mathrm{E}\big\{\mathbf{F}_{k-1}\mathbf{x}_{k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1} + \mathbf{w}_{k-1} \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}\big\}, \\
&= \mathbf{F}_{k-1}\mathrm{E}\big\{\mathbf{x}_{k-1} \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}\big\} + \mathbf{B}_{k-1}\mathbf{u}_{k-1}, \\
&= \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1}.
\end{aligned} \tag{3.7}$$

Thus, the predicted state estimate is obtained by propagating the previous estimate through the noise-free dynamics. This yields the following evolution of the estimation error:

$$\mathbf{e}_{k|k-1} = \mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}, \tag{3.8a}$$
$$= \mathbf{F}_{k-1}(\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1|k-1}) + \mathbf{w}_{k-1}, \tag{3.8b}$$
$$= \mathbf{F}_{k-1}\mathbf{e}_{k-1|k-1} + \mathbf{w}_{k-1}. \tag{3.8c}$$

From Equation (2.47), it follows that

$$\boldsymbol{\Sigma}_{k|k-1} = \mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}. \tag{3.9}$$

The presence of the process noise covariance matrix in this equation highlights how uncertainty is introduced when the estimate is propagated through an imperfect dynamical model. As a result, the obtained distribution $\mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, \boldsymbol{\Sigma}_{k|k-1})$, which becomes the prior at time $k$, is wider and flatter than the past posterior $\mathcal{N}(\hat{\mathbf{x}}_{k-1|k-1}, \boldsymbol{\Sigma}_{k-1|k-1})$. The prediction stage is illustrated in Figure 3.1.

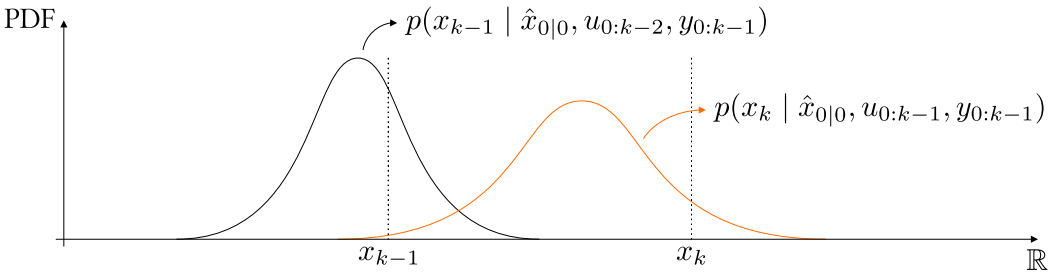



**Figure 3.1:** Kalman filter prediction step for a one-dimensional state: the Gaussian posterior at time $k-1$ is propagated to a Gaussian prior at time $k$.

### 3.1.2 The update stage

Whenever a new measurement becomes available, the Kalman filter updates its estimate using this information. Assume we are at time index $k$ and that the measurement $\mathbf{y}_k$ has just been received. At this stage, the prior distribution of the state and the likelihood of observing $\mathbf{y}_k$ given $\mathbf{x}_k$ are

$$\mathbf{x}_k \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, \boldsymbol{\Sigma}_{k|k-1}), \tag{3.10}$$

$$\mathbf{y}_k \mid (\mathbf{x}_k, \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\mathbf{H}_k \mathbf{x}_k, \mathbf{N}_k), \tag{3.11}$$

and the joint distribution of $\mathbf{x}_k$ and $\mathbf{y}_k$ is therefore given by

$$(\mathbf{x}_k, \mathbf{y}_k) \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}\left( \begin{bmatrix} \hat{\mathbf{x}}_{k|k-1} \\ \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{k|k-1} & \boldsymbol{\Sigma}_{k|k-1} \mathbf{H}_k^T \\ \mathbf{H}_k \boldsymbol{\Sigma}_{k|k-1} & \mathbf{H}_k \boldsymbol{\Sigma}_{k|k-1} \mathbf{H}_k^T + \mathbf{N}_k \end{bmatrix} \right). \tag{3.12}$$

Using Equation (2.36), the posterior distribution is computed according to

$$\mathbf{x}_k \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k}) \sim \mathcal{N}(\underbrace{\hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \mathbf{z}_k}_{\hat{\mathbf{x}}_{k|k}}, \underbrace{(\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \boldsymbol{\Sigma}_{k|k-1}}_{\boldsymbol{\Sigma}_{k|k}}), \tag{3.13}$$

where the Kalman gain $\mathbf{K}_k$ and the innovation $\mathbf{z}_k$ are defined as

$$\mathbf{K}_k = \boldsymbol{\Sigma}_{k|k-1} \mathbf{H}_k^T \left( \mathbf{H}_k \boldsymbol{\Sigma}_{k|k-1} \mathbf{H}_k^T + \mathbf{N}_k \right)^{-1}, \quad \mathbf{z}_k = \mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1}. \tag{3.14}$$

The covariance update reduces the eigenvalues of the estimation error covariance matrix along the directions informed by the measurement, reflecting the increased

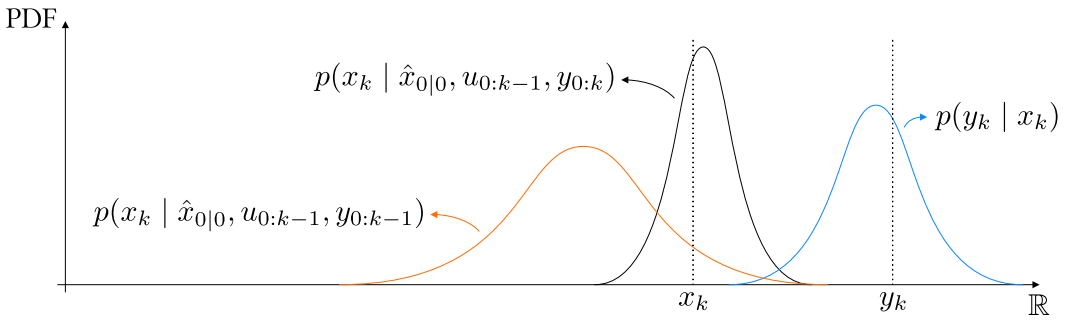



**Figure 3.2:** Kalman filter update step for a one-dimensional state: the Gaussian prior is updated to an exactly computed Gaussian posterior.

confidence in the corresponding components of the state estimate. In probabilistic terms, this corresponds to a posterior distribution that is narrower and taller than the prior. The update stage of the Kalman filter is illustrated in Figure 3.2.

The covariance update admits an equivalent form, known as the Joseph form,

$$\mathbf{\Sigma}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{\Sigma}_{k|k-1}(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T + \mathbf{K}_k\mathbf{N}_k\mathbf{K}_k^T, \tag{3.15}$$

which is obtained by analyzing how the update of the state estimate affects the estimation error. Although computationally more expensive, the Joseph form is numerically more stable, as it explicitly preserves the symmetry and positive semi-definiteness of the covariance matrix.

### 3.1.3 Summary and discussion

The Kalman filter is an optimal recursive Bayesian estimator in the sense that it computes the exact posterior distribution of the system state given the available measurements. This optimality holds under the assumptions of linear system dynamics and measurement models, and of additive white Gaussian process and measurement noises. Under these conditions, the Kalman estimate is also the Best Linear Unbiased Estimator (BLUE), meaning that it achieves the minimum estimation error covariance among all linear and unbiased estimators.

The update equations were derived through Bayesian inference. An equivalent result can be obtained using a Maximum A Posteriori (MAP) approach, in which the updated state estimate is chosen as the maximizer of the posterior probability density function rather than by explicitly computing the full posterior. Since the posterior is Gaussian, its mean and mode coincide, yielding the same estimate.

The process and measurement noise covariance matrices $\mathbf{Q}_k$ and $\mathbf{N}_k$ play a

critical role in filter behavior, as they determine the relative confidence assigned to the dynamical model and to the measurements, respectively. An inaccurate specification of these noise statistics can significantly degrade performance, making their careful tuning or identification essential in practical applications.

Finally, the Kalman filter must be initialized with an initial state estimate $\hat{\mathbf{x}}_{0|0}$ and an associated error covariance matrix $\boldsymbol{\Sigma}_{0|0}$. Although poor initialization may temporarily affect performance, the filter typically converges as informative measurements are incorporated over time.

The Kalman filter algorithm is summarized in Algorithm 1, and an illustrative example of its application is presented in Example 3.1.

**Algorithm 1** The Kalman filter

Choose the initial estimate $(\hat{\mathbf{x}}_{0|0}, \boldsymbol{\Sigma}_{0|0})$.
**loop**
  ▷ Prediction
  Define the matrices $\mathbf{F}_{k-1}, \mathbf{B}_{k-1}, \mathbf{Q}_{k-1}$ and the input vector $\mathbf{u}_{k-1}$.
  $\hat{\mathbf{x}}_{k|k-1} \leftarrow \mathbf{F}_{k-1}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_{k-1}\mathbf{u}_{k-1}$
  $\boldsymbol{\Sigma}_{k|k-1} \leftarrow \mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}$
  ▷ Update
  **if** measurement $\mathbf{y}_k$ available **then**
    Define the matrices $\mathbf{H}_k$ and $\mathbf{N}_k$.
    $\mathbf{z}_k \leftarrow \mathbf{y}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}$
    $\mathbf{K}_k \leftarrow \boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T(\mathbf{H}_k\boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T + \mathbf{N}_k)^{-1}$
    $\hat{\mathbf{x}}_{k|k} \leftarrow \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\mathbf{z}_k$
    $\boldsymbol{\Sigma}_{k|k} \leftarrow (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\boldsymbol{\Sigma}_{k|k-1}$
  **end if**
**end loop**

▶ **Example 3.1.** Consider the situation described in Example 2.11, where the latent state is composed of the velocity $\mathbf{v}_k$ and position $\mathbf{p}_k$ of a human co-worker, expressed in the inertial frame attached to the base of a robotic arm, that is,

$$\mathbf{x}_k = (\mathbf{v}_k, \mathbf{p}_k) \in \mathbb{R}^4.$$

We recall the state-space representation of the system,

$$\mathbf{x}_k = \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{I}dt & \mathbf{I} \end{bmatrix}}_{\mathbf{F}_{k-1}} \mathbf{x}_{k-1} + \underbrace{\begin{bmatrix} \bar{\mathbf{w}}_{k-1} \\ \mathbf{0} \end{bmatrix}}_{\mathbf{w}_{k-1}},$$

$$\mathbf{y}_k = \underbrace{\begin{bmatrix}\mathbf{0} & \mathbf{I}\end{bmatrix}}_{\mathbf{H}_k} \mathbf{x}_k + \mathbf{n}_k.$$

The dynamical model is based on a constant-velocity assumption, while the measurement model corresponds to a GPS beacon mounted on the human co-worker. The process noise $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$ accounts solely for deviations from the constant-velocity assumption and does not directly affect the position propagation equation. The measurement noise $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ models the uncertainty inherent to the GPS sensor. For illustration purposes, the sampling period is set to $dt = 0.2\,\mathrm{s}$, although much smaller values are typically used in practice.

We apply a Kalman filter to estimate $\mathbf{x}_k$ and consider the first full iteration of the algorithm. The ground truth is such that

$$\mathbf{x}_0 = (-1, -0.5, 1, 2), \qquad \mathbf{x}_1 = (-1.04, -0.55, 0.80, 1.90).$$

The following process and measurement noise covariance matrices are used:

$$\mathbf{Q}_{k-1} = \mathrm{block\_diag}(0.1^2\mathbf{I}, \mathbf{0}), \qquad \mathbf{N}_k = 0.3^2\mathbf{I},$$

where $\mathrm{block\_diag}(\cdot)$ denotes the operator that constructs a block-diagonal matrix from its arguments. The filter is initialized with the following values:

$$\hat{\mathbf{x}}_{0|0} = (-0.9, 0.1, 1.4, 2.7), \qquad \mathbf{\Sigma}_{0|0} = \mathbf{I}.$$

**Prediction:** The prediction step of the Kalman filter yields:

$$\hat{\mathbf{x}}_{1|0} = (-0.90, 0.10, 1.22, 2.72), \qquad \mathbf{\Sigma}_{1|0} = \begin{bmatrix} 1.01\mathbf{I} & 0.2\mathbf{I} \\ 0.2\mathbf{I} & 1.04\mathbf{I} \end{bmatrix}.$$

The prediction step increases the diagonal entries of the covariance matrix, reflecting the growth of uncertainty in the state estimate. It also introduces nonzero off-diagonal terms, which indicate coupling between the position and velocity components. This coupling is a direct consequence of the dynamics, since the position update depends on both the previous position and the velocity.

**Update:** The GPS beacon provides the following measurement:

$$\mathbf{y}_1 = (0.97, 1.80).$$

Incorporating this information yields

$$\hat{\mathbf{x}}_{1|1} = (-0.94, -0.06, 0.99, 1.87), \quad \boldsymbol{\Sigma}_{1|1} = \begin{bmatrix} 0.97\mathbf{I} & 0.02\mathbf{I} \\ 0.02\mathbf{I} & 0.08\mathbf{I} \end{bmatrix}.$$

The update step reduces both the estimation error and the uncertainty, as reflected by the decrease in the covariance. Although the measurement informs only the position, the cross-covariance terms generated during prediction propagate this information to the velocity component, allowing it to be updated. ◀

## 3.2 The extended Kalman filter

The Kalman filter provides a closed-form, optimal solution to the state estimation problem under linear dynamics, linear measurement models, and Gaussian noise assumptions. Many practical systems, however, deviate from this linear setting. In this case, propagating a Gaussian prior through nonlinear dynamics, or updating it with a nonlinear measurement model, generally produces non-Gaussian distributions, making exact Bayesian inference intractable.

The discrete-time EKF builds upon the same probabilistic framework as the linear Kalman filter, with the key distinction that it allows the system dynamics and measurement models to be nonlinear. Specifically, the EKF considers nonlinear, time-varying hidden Markov models driven by zero-mean white Gaussian process and measurement noises. The system is described by

$$\mathbf{x}_k = \mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \tag{3.16}$$

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k), \tag{3.17}$$

where the state transition function $\mathbf{f}_{k-1} : \mathbb{R}^n \times \mathbb{R}^b \times \mathbb{R}^q \to \mathbb{R}^n$ and the measurement function $\mathbf{h}_k : \mathbb{R}^n \times \mathbb{R}^r \to \mathbb{R}^m$ are generally nonlinear. As in the linear case, the process and measurement noises are assumed to be mutually independent, with distributions

$$\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1}), \quad \mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k). \tag{3.18}$$

The EKF adopts the same definition of the estimation error as the linear Kalman filter and assumes it to be Gaussian:

$$\mathbf{e}_{k|l} = \mathbf{x}_k - \hat{\mathbf{x}}_{k|l} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{k|l}), \tag{3.19}$$

which implies the approximate posterior distribution

$$\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_{k|l}, \boldsymbol{\Sigma}_{k|l}). \tag{3.20}$$

The estimation error is further assumed to be uncorrelated with both the process noise $\mathbf{w}_{k-1}$ and the measurement noise $\mathbf{n}_k$.

As in the linear case, the hidden Markov model structure enables the EKF to apply the recursive formulation of Bayesian inference to propagate and update its belief over time. However, due to the nonlinear nature of the state transition and measurement functions, exact Bayesian inference is no longer tractable. Even under Gaussian noise assumptions, nonlinear transformations of random variables generally result in non-Gaussian distributions. The EKF addresses this difficulty by locally linearizing the nonlinear models around the current estimate, thereby approximating the true posterior distribution with a Gaussian at each time step.

### 3.2.1 The prediction stage

Assume that at time index $k-1$ the posterior distribution of the state is Gaussian,

$$\mathbf{x}_{k-1} \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-2}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k-1|k-1}, \boldsymbol{\Sigma}_{k-1|k-1}). \tag{3.21}$$

The objective of the prediction stage is to propagate this distribution through the system dynamics in order to obtain a prior density for $\mathbf{x}_k$. Since nonlinear transformations of Gaussian random variables are generally non-Gaussian, the EKF relies on a first-order Taylor expansion of the dynamics function $\mathbf{f}_{k-1}$ about the operating point $(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})$ to maintain a Gaussian representation. This yields the local linear approximation

$$\begin{aligned}\mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) \approx \\ \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}) + \mathbf{F}_{k-1}\mathbf{e}_{k-1|k-1} + \mathbf{G}_{k-1}\mathbf{w}_{k-1},\end{aligned} \tag{3.22}$$

where the Jacobian matrices $\mathbf{F}_{k-1}$ and $\mathbf{G}_{k-1}$ are defined as

$$\mathbf{F}_{k-1} = \left.\frac{\partial \mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})}{\partial \mathbf{x}_{k-1}}\right|_{(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})}, \tag{3.23}$$

$$\mathbf{G}_{k-1} = \left.\frac{\partial \mathbf{f}_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1})}{\partial \mathbf{w}_{k-1}}\right|_{(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})}. \tag{3.24}$$

Using this linearized model, the predicted mean is obtained by taking the condi-

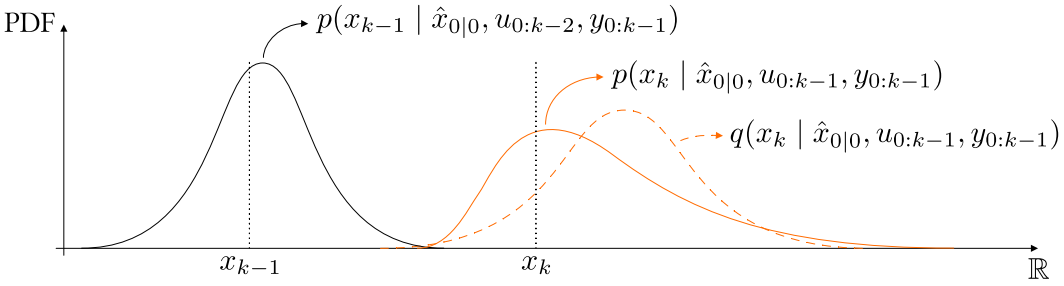



**Figure 3.3:** EKF prediction step for a one-dimensional state: nonlinear dynamics generally transform a Gaussian posterior at time $k-1$ into a non-Gaussian prior at time $k$. The EKF maintains a Gaussian representation by linearizing the dynamics about the current estimate. True densities are denoted by $p$, while Gaussian approximations are denoted by $q$.

tional expectation,

$$\begin{aligned}
&\mathrm{E}\{\mathbf{x}_k \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}\} \\
&\approx \mathrm{E}\{\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}) + \mathbf{F}_{k-1}\mathbf{e}_{k-1|k-1} + \mathbf{G}_{k-1}\mathbf{w}_{k-1} \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}\}, && \text{(3.25a)} \\
&\approx \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}), && \text{(3.25b)}
\end{aligned}$$

where the zero-mean assumptions on $\mathbf{e}_{k-1|k-1}$ and $\mathbf{w}_{k-1}$ have been used. Similarly, the covariance of the predicted prior is approximated as

$$\begin{aligned}
&\mathrm{Cov}\{\mathbf{x}_k \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}\} \\
&\quad\approx \mathrm{E}\{(\mathbf{F}_{k-1}\mathbf{e}_{k-1|k-1} + \mathbf{G}_{k-1}\mathbf{w}_{k-1})(\mathbf{F}_{k-1}\mathbf{e}_{k-1|k-1} + \mathbf{G}_{k-1}\mathbf{w}_{k-1})^T\}, && \text{(3.26a)} \\
&\quad\approx \mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}_{k-1}^T + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^T. && \text{(3.26b)}
\end{aligned}$$

where independence between $\mathbf{e}_{k-1|k-1}$ and $\mathbf{w}_{k-1}$ has been assumed.

Altogether, the prediction stage yields the following Gaussian approximation:

$$\begin{aligned}
&\mathbf{x}_k \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \\
&\qquad \mathcal{N}(\underbrace{\mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})}_{\hat{\mathbf{x}}_{k|k-1}}, \underbrace{\mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}_{k-1}^T + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^T}_{\boldsymbol{\Sigma}_{k|k-1}}). \qquad \text{(3.27)}
\end{aligned}$$

An illustration of the prediction stage of the EKF is shown in Figure 3.3.

### 3.2.2 The update stage

Assume we are at time index $k$ and that the measurement $\mathbf{y}_k$ has just been received. In the linear Kalman filter, the update stage could be carried out using exact Bayesian inference, since the measurement model was linear and the prior distribution was Gaussian. This is no longer the case for nonlinear measurement models. The EKF therefore linearizes the measurement model and perform full Bayesian inference using this linear model approximation.

A first-order Taylor expansion of $\mathbf{h}_k$ about the operating point $(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0})$ yields

$$\mathbf{y}_k \approx \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0}) + \mathbf{H}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}) + \mathbf{M}_k\mathbf{n}_k, \tag{3.28}$$

where the Jacobian matrices $\mathbf{H}_k$ and $\mathbf{M}_k$ are defined as

$$\mathbf{H}_k = \left.\frac{\partial \mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)}{\partial \mathbf{x}_k}\right|_{(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0})}, \quad \mathbf{M}_k = \left.\frac{\partial \mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)}{\partial \mathbf{n}_k}\right|_{(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0})}. \tag{3.29}$$

This yields the following Gaussian approximations of the prior and the likelihood:

$$\mathbf{x}_k \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, \mathbf{\Sigma}_{k|k-1}), \tag{3.30}$$

$$\mathbf{y}_k \mid (\mathbf{x}_k, \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0}) + \mathbf{H}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}), \mathbf{M}_k\mathbf{N}_k\mathbf{M}_k^T). \tag{3.31}$$

Therefore, the joint distribution conditioned on past information is approximately Gaussian:

$$(\mathbf{x}_k, \mathbf{y}_k) \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}\left(\begin{bmatrix} \hat{\mathbf{x}}_{k|k-1} \\ \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0}) \end{bmatrix}, \begin{bmatrix} \mathbf{\Sigma}_{k|k-1} & \mathbf{\Sigma}_{k|k-1}\mathbf{H}_k^T \\ \mathbf{H}_k\mathbf{\Sigma}_{k|k-1} & \mathbf{H}_k\mathbf{\Sigma}_{k|k-1}\mathbf{H}_k^T + \mathbf{M}_k\mathbf{N}_k\mathbf{M}_k^T \end{bmatrix}\right). \tag{3.32}$$

Applying Equation (2.36) gives the Gaussian approximation of the posterior:

$$\mathbf{x}_k \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k}) \sim \mathcal{N}(\underbrace{\hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\mathbf{z}_k}_{\hat{\mathbf{x}}_{k|k}}, \underbrace{(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{\Sigma}_{k|k-1}}_{\mathbf{\Sigma}_{k|k}}), \tag{3.33}$$

where the Kalman gain $\mathbf{K}_k$ and the innovation $\mathbf{z}_k$ are now given by

$$\mathbf{K}_k = \mathbf{\Sigma}_{k|k-1}\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{\Sigma}_{k|k-1}\mathbf{H}_k^T + \mathbf{M}_k\mathbf{N}_k\mathbf{M}_k^T\right)^{-1}, \tag{3.34}$$

$$\mathbf{z}_k = \mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0}). \tag{3.35}$$

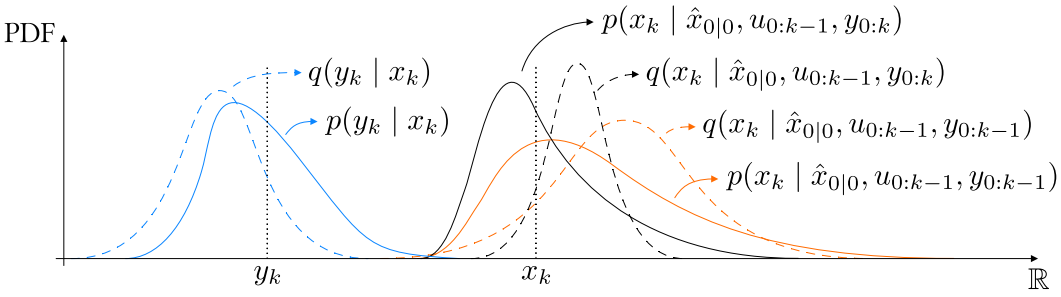



**Figure 3.4:** EKF update stage for a one-dimensional state: under a nonlinear measurement model, the posterior is generally non-Gaussian. The EKF maintains a Gaussian representation by linearizing the measurement model about the current estimate. True densities are denoted by $p$, while Gaussian approximations are denoted by $q$.

An illustration of the update stage of the EKF is provided in Figure 3.4.

### 3.2.3 Summary and discussion

Although it considers nonlinear dynamics and measurement models, the EKF relies heavily on first-order approximations of these models around the current state estimate and essentially follows the same methodology as the linear Kalman filter. This simplicity has contributed to the widespread adoption of the EKF in practical applications. From an implementation standpoint, the EKF requires only minor modifications to the linear filter, namely the propagation of the state estimate through the nonlinear dynamics and measurement models, and the computation of the Jacobian matrices. When these matrices admit closed-form expressions, the computational cost of the EKF remains relatively low, making it one of the most efficient state estimation techniques for nonlinear systems.

Nevertheless, the EKF suffers from several well-known limitations. First, its performance strongly depends on the degree of nonlinearity present in the system. As the dynamics and measurement models become more nonlinear, the accuracy of first-order approximations far from the linearization point degrades, which directly deteriorates the quality of both the prediction and update steps. This issue is further exacerbated by the fact that the linearization is performed around the current state estimate rather than the true state. Second, because the linearization Jacobians depend on the estimate itself, a critical sensitivity is introduced: poor estimates lead to inaccurate Jacobians, which in turn degrade subsequent estimates. This positive feedback mechanism can cause estimation errors to accumulate over time and may ultimately prevent the filter from converging. Note that this mechanism is

primarily due to the dependence of the Jacobians $\mathbf{F}_{k-1}$ and $\mathbf{H}_k$ on the state estimate, as the impact of the Jacobians $\mathbf{G}_{k-1}$ and $\mathbf{M}_k$ is often mitigated by the typically low amplitude of the process and measurement noises.

Finally, when the state is not fully observable, linearizing the measurement model around the estimated state rather than the true state introduces an additional issue. In this case, the kernel of the Jacobian matrix $\mathbf{H}_k$, which is generated by the unobservable directions of the state, is incorrectly represented. As a result, spurious information is artificially injected into the unobservable components of the state, causing the filter to become increasingly overconfident in quantities that are, in fact, unobservable. This well-known limitation of the EKF is commonly referred to as the false observability issue and has been the subject of many studies, particularly in the context of Simultaneous Localization And Mapping (SLAM) [67–69].

For these reasons, the EKF does not admit general convergence or optimality guarantees, and its successful application therefore depends heavily on proper initialization. In practice, the initial state estimate must be sufficiently close to the true state, and the initial covariance matrix should accurately reflect the uncertainty associated with this estimate.

The EKF algorithm is summarized in Algorithm 2, and an illustrative example of its application is presented in Example 3.2.

**Algorithm 2** The extended Kalman filter

Choose the initial estimate $(\hat{\mathbf{x}}_{0|0}, \boldsymbol{\Sigma}_{0|0})$.
**loop**
  ▷ Prediction
  Compute Jacobians $\mathbf{F}_{k-1}$ and $\mathbf{G}_{k-1}$, define $\mathbf{f}_{k-1}$, $\mathbf{Q}_{k-1}$ and $\mathbf{u}_{k-1}$.
  $\hat{\mathbf{x}}_{k|k-1} \leftarrow \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})$
  $\boldsymbol{\Sigma}_{k|k-1} \leftarrow \mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}_{k-1}^T + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^T$
  ▷ Update
  **if** measurement $\mathbf{y}_k$ available **then**
    Compute Jacobians $\mathbf{H}_k$ and $\mathbf{M}_k$, define $\mathbf{h}_k$ and $\mathbf{N}_k$.
    $\mathbf{z}_k \leftarrow \mathbf{y}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0})$
    $\mathbf{K}_k \leftarrow \boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T(\mathbf{H}_k\boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T + \mathbf{N}_k)^{-1}$
    $\hat{\mathbf{x}}_{k|k} \leftarrow \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\mathbf{z}_k$
    $\boldsymbol{\Sigma}_{k|k} \leftarrow (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\boldsymbol{\Sigma}_{k|k-1}$
  **end if**
**end loop**

▶ **Example 3.2.** We extend the setup of Example 3.1 by augmenting the state with the heading angle $\theta_k$ of the human co-worker. The resulting state vector is

$$\mathbf{x}_k = (\theta_k, \mathbf{v}_k, \mathbf{p}_k) \in \mathbb{R}^5.$$

We assume that the worker is equipped with an inertial measurement unit (IMU) providing noisy measurements of angular velocity $\omega_k \in \mathbb{R}$ and specific force $\mathbf{a}_k \in \mathbb{R}^2$, expressed in the worker's body frame. Rather than relying on a constant-velocity model, the IMU measurements are used as system inputs,

$$\mathbf{u}_k = (\omega_k, \mathbf{a}_k) \in \mathbb{R}^3,$$

and the system dynamics are described using the equations of inertial navigation:

$$\begin{aligned}
\theta_{k+1} &= \theta_k + (\omega_k + w_k^{\omega})\, dt, \\
\mathbf{v}_{k+1} &= \mathbf{v}_k + \mathbf{R}(\theta_k)(\mathbf{a}_k + \mathbf{w}_k^{\mathbf{a}})\, dt, \\
\mathbf{p}_{k+1} &= \mathbf{p}_k + \mathbf{v}_k\, dt + \mathbf{R}(\theta_k)(\mathbf{a}_k + \mathbf{w}_k^{\mathbf{a}})\, \tfrac{dt^2}{2},
\end{aligned}$$

where we neglected the IMU biases and where the process noise vector $\mathbf{w}_k = (w_k^{\omega}, \mathbf{w}_k^{\mathbf{a}}) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ stacks the gyroscope and accelerometer noise. The matrix $\mathbf{R}(\theta)$ is the rotation matrix associated with the heading angle $\theta$ and is given by

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}.$$

Linearizing the dynamics yields the Jacobians

$$\mathbf{F}_{k-1} = \begin{bmatrix} 1 & \mathbf{0} & \mathbf{0} \\ \frac{\partial \mathbf{R}(\theta)}{\partial \theta}\Big|_{\hat{\theta}_{k-1|k-1}} \mathbf{a}_{k-1}\, dt & \mathbf{I} & \mathbf{0} \\ \frac{\partial \mathbf{R}(\theta)}{\partial \theta}\Big|_{\hat{\theta}_{k-1|k-1}} \mathbf{a}_{k-1}\, \frac{dt^2}{2} & \mathbf{I}\, dt & \mathbf{I} \end{bmatrix}, \qquad \mathbf{G}_{k-1} = \begin{bmatrix} dt & \mathbf{0} \\ 0 & \mathbf{R}(\hat{\theta}_{k-1|k-1})\, dt \\ 0 & \mathbf{R}(\hat{\theta}_{k-1|k-1})\, \frac{dt^2}{2} \end{bmatrix},$$

where the Jacobian of the rotation matrix w.r.t. the heading angle is given by

$$\frac{\partial \mathbf{R}(\theta)}{\partial \theta} = \begin{bmatrix} -\sin(\theta) & -\cos(\theta) \\ \cos(\theta) & -\sin(\theta) \end{bmatrix}.$$

Instead of a GPS receiver, the robot has access to measurements transmitted by a range-and-bearing sensor mounted on the worker. We model the sensor output as

the relative positions of three beacons expressed in the worker's body frame:

$$\mathbf{y}_k = \begin{bmatrix} \mathbf{R}(\theta_k)^T(\mathbf{b}_1 - \mathbf{p}_k) \\ \mathbf{R}(\theta_k)^T(\mathbf{b}_2 - \mathbf{p}_k) \\ \mathbf{R}(\theta_k)^T(\mathbf{b}_3 - \mathbf{p}_k) \end{bmatrix} + \mathbf{n}_k,$$

where $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ models the sensor noise and where $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3 \in \mathbb{R}^2$ denote the fixed beacon positions in the inertial frame, which are given by

$$\mathbf{b}_1 = (0, 0), \qquad \mathbf{b}_2 = (10, 0), \qquad \mathbf{b}_3 = (0, 10).$$

The Jacobians of the measurement model admit the closed-form expressions

$$\mathbf{H}_k = \begin{bmatrix} \left( \frac{\partial \mathbf{R}(\theta)}{\partial \theta} \Big|_{\hat{\theta}_{k|k-1}} \right)^T (\mathbf{b}_1 - \hat{\mathbf{p}}_{k|k-1}) & \mathbf{0} & -\mathbf{R}(\hat{\theta}_{k|k-1})^T \\ \left( \frac{\partial \mathbf{R}(\theta)}{\partial \theta} \Big|_{\hat{\theta}_{k|k-1}} \right)^T (\mathbf{b}_2 - \hat{\mathbf{p}}_{k|k-1}) & \mathbf{0} & -\mathbf{R}(\hat{\theta}_{k|k-1})^T \\ \left( \frac{\partial \mathbf{R}(\theta)}{\partial \theta} \Big|_{\hat{\theta}_{k|k-1}} \right)^T (\mathbf{b}_3 - \hat{\mathbf{p}}_{k|k-1}) & \mathbf{0} & -\mathbf{R}(\hat{\theta}_{k|k-1})^T \end{bmatrix}, \qquad \mathbf{M}_k = \mathbf{I}.$$

We apply an EKF to estimate $\mathbf{x}_k$ and consider the first iteration of the algorithm. The ground truth trajectory is such that

$$\begin{aligned} \mathbf{x}_0 &= (3.9270, -0.9899, -0.9899, 5.0000, 5.0000), \\ \mathbf{x}_1 &= (3.9270, -0.9915, -0.9915, 4.9505, 4.9505), \\ \mathbf{u}_0 &= (-0.0541, 0.0411, -0.1459). \end{aligned}$$

The simulation is performed with the following process and measurement noise covariance matrices:

$$\mathbf{Q}_k = 0.1^2\mathbf{I}, \qquad \mathbf{N}_k = 0.1^2\mathbf{I}.$$

The sampling period is set to $dt = 0.05\,\mathrm{s}$ and the filter is initialized with

$$\hat{\mathbf{x}}_{0|0} = (4.5406, -2.2593, 0.0718, 5.0449, 3.9764),$$

$$\boldsymbol{\Sigma}_{0|0} = \begin{bmatrix} 0.2742 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 1.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 1.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{bmatrix}.$$

**Prediction:** The EKF prediction step produces

$$\hat{\mathbf{x}}_{1|0} = (4.5379, -2.2669, 0.0711, 4.9317, 3.9800),$$
$$\boldsymbol{\Sigma}_{1|0} = \begin{bmatrix} 0.2742 & 0.0002 & -0.0021 & 0.0000 & -0.0001 \\ 0.0002 & 1.0000 & 0.0000 & 0.0500 & 0.0000 \\ -0.0021 & 0.0000 & 1.0000 & 0.0000 & 0.0500 \\ 0.0000 & 0.0500 & 0.0000 & 1.0025 & 0.0000 \\ -0.0001 & 0.0000 & 0.0500 & 0.0000 & 1.0025 \end{bmatrix}.$$

The prediction stage introduces correlations between the heading angle, velocity, and position. It also reduces the confidence of the filter in its estimate, as reflected by the slightly larger diagonal entries of the covariance matrix.

**Update:** The range-and-bearing sensor outputs the measurement

$$\mathbf{y}_1 = (7.1900, -0.0301, -0.0233, 6.9132, -0.1647, -7.2182).$$

Incorporating this new information yields the following updated estimate:

$$\hat{\mathbf{x}}_{1|1} = (3.9799, -2.3067, 0.1141, 4.1417, 4.7583),$$
$$\boldsymbol{\Sigma}_{1|1} = \begin{bmatrix} 0.0001 & 0.0000 & 0.0000 & 0.0000 & 0.0001 \\ 0.0000 & 0.9975 & 0.0000 & 0.0002 & 0.0000 \\ 0.0000 & 0.0000 & 0.9975 & 0.0000 & 0.0002 \\ 0.0000 & 0.0002 & 0.0000 & 0.0034 & -0.0001 \\ 0.0001 & 0.0000 & 0.0002 & -0.0001 & 0.0035 \end{bmatrix}$$

This measurement is highly informative about the heading angle and the position of the worker, which increases the confidence of the filter in its updated heading and position estimates. Although the measurement model does not depend directly on the velocity, the filter still corrects $\mathbf{v}_k$ through the correlations between velocity, position, and heading introduced by the prediction step.

The simulation is run over the first 20 time steps. Figure 3.5 shows the evolution of the norm of the estimation error for the worker's heading angle, velocity, and position. The estimation errors in the heading angle and position decrease rapidly and quickly reach low values. By contrast, the velocity estimation error initially increases before slowly converging to a low value. This slower convergence is mainly due to poor linearization of the measurement model and to the fact that the measurement does not directly depend on the velocity. ◀

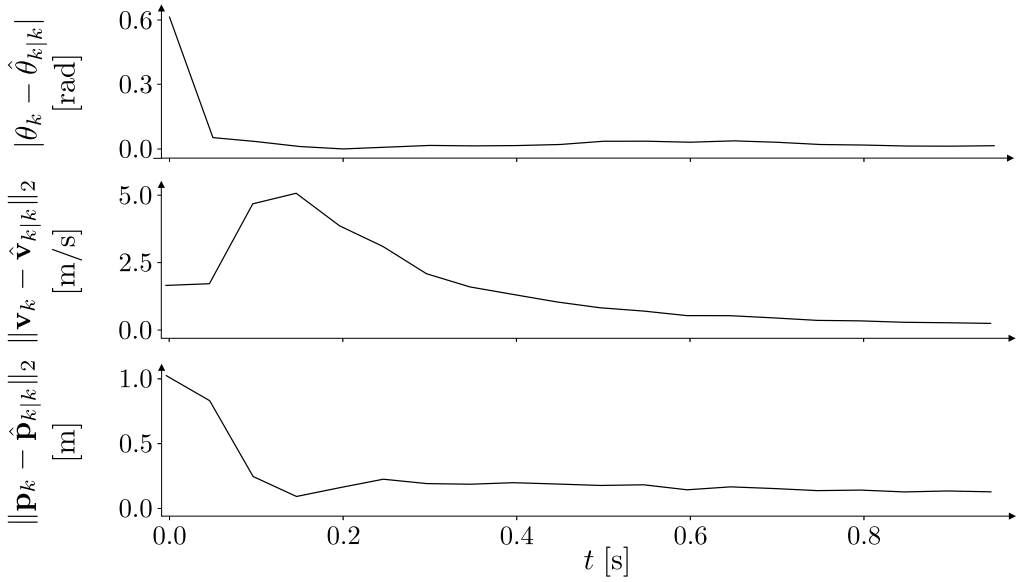



**Figure 3.5:** Evolution of the norm of the EKF estimation error for the worker's heading angle, velocity and position, over the first 20 time steps.

## 3.3 The iterated extended Kalman filter

The well-known limitations of the EKF have motivated a number of refinements. Among them, the iterated extended Kalman filter (IterEKF) improves the measurement update by refining the linearization point of the measurement model within each time step. The underlying idea is commonly attributed to Wishner et al. [70] and was later formalized by Jazwinski [71]. Since the IterEKF prediction stage is essentially identical to that of the EKF, we focus on the measurement update only.

### 3.3.1 the update stage

Assume we are at time index $k$ and that the measurement $\mathbf{y}_k$ given by Equation (3.17) has just been received. Rather than linearizing $\mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)$ around $(\hat{\mathbf{x}}_{k|k-1}, \mathbf{0})$, we linearize it around the arbitrary operating point $(\mathbf{x}_k^i, \mathbf{0})$, giving the approximation

$$\mathbf{y}_k \approx \underbrace{\mathbf{h}_k(\mathbf{x}_k^i, \mathbf{0}) + \mathbf{H}_k^i(\mathbf{x}_k - \mathbf{x}_k^i)}_{\mathbf{h}_k^i(\mathbf{x}_k)} + \mathbf{M}_k^i \mathbf{n}_k, \tag{3.36}$$

where the Jacobians $\mathbf{H}_k^i$ and $\mathbf{M}_k^i$ are defined as

$$\mathbf{H}_k^i = \left.\frac{\partial \mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)}{\partial \mathbf{x}_k}\right|_{(\mathbf{x}_k^i, \mathbf{0})}, \qquad \mathbf{M}_k^i = \left.\frac{\partial \mathbf{h}_k(\mathbf{x}_k, \mathbf{n}_k)}{\partial \mathbf{n}_k}\right|_{(\mathbf{x}_k^i, \mathbf{0})}. \tag{3.37}$$

Based on the prediction step and this first-order approximation of the measurement model, the prior and the measurement likelihood are approximated by

$$\mathbf{x}_k \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\hat{\mathbf{x}}_{k|k-1}, \boldsymbol{\Sigma}_{k|k-1}), \tag{3.38}$$

$$\mathbf{y}_k \mid (\mathbf{x}_k, \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \mathcal{N}(\mathbf{h}_k^i(\mathbf{x}_k), \mathbf{M}_k^i \mathbf{N}_k (\mathbf{M}_k^i)^T). \tag{3.39}$$

Consequently, the joint distribution of $\mathbf{x}_k$ and $\mathbf{y}_k$ conditioned on past information is approximately Gaussian:

$$\begin{aligned}(\mathbf{x}_k, \mathbf{y}_k) \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) \sim \\ \mathcal{N}\left(\begin{bmatrix}\hat{\mathbf{x}}_{k|k-1} \\ \mathbf{h}_k^i(\hat{\mathbf{x}}_{k|k-1})\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{k|k-1} & \boldsymbol{\Sigma}_{k|k-1}(\mathbf{H}_k^i)^T \\ \mathbf{H}_k^i \boldsymbol{\Sigma}_{k|k-1} & \mathbf{H}_k^i \boldsymbol{\Sigma}_{k|k-1}(\mathbf{H}_k^i)^T + \mathbf{M}_k^i \mathbf{N}_k (\mathbf{M}_k^i)^T\end{bmatrix}\right).\end{aligned} \tag{3.40}$$

Using Equation (2.36), we find the approximated posterior distribution

$$\begin{aligned}\mathbf{x}_k \mid (\hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k}) \sim \\ \mathcal{N}\big(\hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^i\big(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k^i, \mathbf{0}) - \mathbf{H}_k^i(\hat{\mathbf{x}}_{k|k-1} - \mathbf{x}_k^i)\big), (\mathbf{I} - \mathbf{K}_k^i \mathbf{H}_k^i)\boldsymbol{\Sigma}_{k|k-1}\big),\end{aligned} \tag{3.41}$$

where the Kalman gain is computed according to

$$\mathbf{K}_k^i = \boldsymbol{\Sigma}_{k|k-1}(\mathbf{H}_k^i)^T \big(\mathbf{H}_k^i \boldsymbol{\Sigma}_{k|k-1}(\mathbf{H}_k^i)^T + \mathbf{M}_k^i \mathbf{N}_k (\mathbf{M}_k^i)^T\big)^{-1}. \tag{3.42}$$

Starting from $\mathbf{x}_k^0 = \hat{\mathbf{x}}_{k|k-1}$, the IterEKF iteratively recomputes the mean in Equation (3.41) by choosing the operating point at iteration $i+1$ as the posterior mean obtained at iteration $i$. In this way, the linearization point is refined until the change in $\mathbf{x}_k^i$ becomes sufficiently small. Assuming convergence after $i^\star$ iterations, the IterEKF updates its estimate according to

$$\hat{\mathbf{x}}_{k|k} = \mathbf{x}_k^{i^\star}, \tag{3.43}$$

$$\boldsymbol{\Sigma}_{k|k} = (\mathbf{I} - \mathbf{K}_k^{i^\star} \mathbf{H}_k^{i^\star})\boldsymbol{\Sigma}_{k|k-1}. \tag{3.44}$$

Note that the posterior covariance matrix can be computed once after convergence, using the final linearization point $\mathbf{x}_k^{i^\star}$.

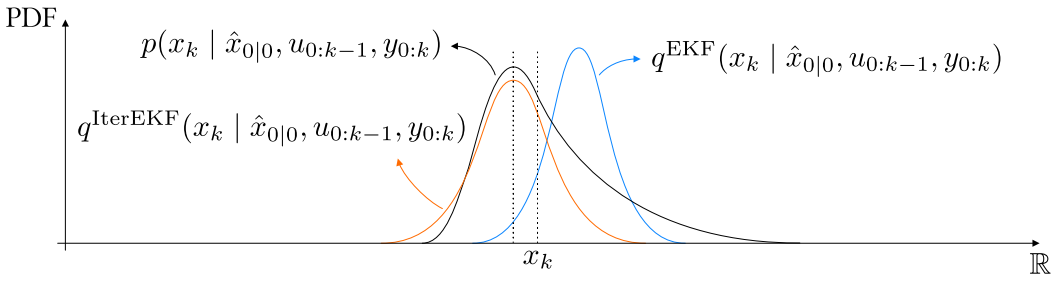



**Figure 3.6:** True posterior density $p(x_k \mid \hat{x}_{0|0}, u_{0:k-1}, y_{0:k})$ and Gaussian approximations $q^{\mathrm{EKF}}(x_k \mid \hat{x}_{0|0}, u_{0:k-1}, y_{0:k})$ and $q^{\mathrm{IterEKF}}(x_k \mid \hat{x}_{0|0}, u_{0:k-1}, y_{0:k})$ after the EKF and IterEKF updates, respectively. When the iterative scheme converges, the IterEKF yields a Gaussian whose mean coincides with the (local) MAP estimate of the true posterior.

Figure 3.6 illustrates, for a scalar state $x_k$, the difference between the true posterior density $p(x_k \mid \hat{x}_{0|0}, u_{0:k-1}, y_{0:k})$ and the Gaussian approximations $q^{\mathrm{EKF}}(x_k \mid \hat{x}_{0|0}, u_{0:k-1}, y_{0:k})$ and $q^{\mathrm{IterEKF}}(x_k \mid \hat{x}_{0|0}, u_{0:k-1}, y_{0:k})$ produced by the EKF and IterEKF updates, respectively.

### 3.3.2 Summary and discussion

The IterEKF improves the accuracy of the measurement update at the expense of additional computational resources. It is therefore natural to ask how much is gained compared to the classical EKF update. This question is easier to address from an optimization viewpoint than from the purely Bayesian perspective adopted so far. From this viewpoint, Bell et al. [72] showed that, under additive Gaussian measurement noise, the successive linearizations performed by the IterEKF generate a sequence of state corrections that can be interpreted as Gauss-Newton iterations[1] applied to the MAP estimation problem associated with the nonlinear measurement update:

$$\hat{\mathbf{x}}^{\star}_{k|k} = \operatorname*{argmax}_{\mathbf{x}_k} \quad p(\mathbf{x}_k \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k}), \tag{3.45a}$$

$$= \operatorname*{argmax}_{\mathbf{x}_k} \quad p(\mathbf{x}_k \mid \hat{\mathbf{x}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k-1}) p(\mathbf{y}_k \mid \mathbf{x}_k). \tag{3.45b}$$

1 The Gauss-Newton method emerged as a practical approach to solving overdetermined nonlinear least-squares problems in the context of Gauss's astronomical work [73]. It can be viewed as an adaptation of Newton's method [74] to the least-squares setting, obtained by linearizing the residuals and solving the resulting linear least-squares subproblem at each iteration.

In this sense, the IterEKF update seeks a local maximizer of the posterior density. In other words, its updated state estimate is a (local) MAP estimate. This interpretation, however, applies to the update step conditioned on the prior at time $k$. Since this prior is itself only an approximation, the posterior density considered in this optimization problem is also approximate. In addition, the practical performance of the IterEKF depends on the convergence of the underlying Gauss-Newton iterations, which is not guaranteed in general. For results on convergence rates, see [75].

The entire IterEKF procedure is summarized in Algorithm 3. We deliberately do not specify a stopping criterion for the iterative update. Since this choice directly determines the trade-off between estimation accuracy and computational cost, we leave it to the user. An illustrative example of the IterEKF is presented in Example 3.3.

**Algorithm 3** The iterated extended Kalman filter

Choose the initial estimate $(\hat{\mathbf{x}}_{0|0}, \boldsymbol{\Sigma}_{0|0})$.
**loop**
  ▷ Prediction
  Compute Jacobians $\mathbf{F}_{k-1}$ and $\mathbf{G}_{k-1}$, define $\mathbf{f}_{k-1}$, $\mathbf{Q}_{k-1}$ and $\mathbf{u}_{k-1}$.
  $\hat{\mathbf{x}}_{k|k-1} \leftarrow \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0})$
  $\boldsymbol{\Sigma}_{k|k-1} \leftarrow \mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}_{k-1}^T + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^T$
  ▷ Update
  **if** measurement $\mathbf{y}_k$ available **then**
    Define $\mathbf{h}_k$ and $\mathbf{N}_k$.
    $\mathbf{x}_k^i \leftarrow \hat{\mathbf{x}}_{k|k-1}$
    **while** $\mathbf{x}_k^i$ has not converged **do**
      Compute Jacobian $\mathbf{H}_k^i$ and $\mathbf{M}_k^i$
      $\mathbf{K}_k^i \leftarrow \boldsymbol{\Sigma}_{k|k-1}(\mathbf{H}_k^i)^T(\mathbf{H}_k^i\boldsymbol{\Sigma}_{k|k-1}(\mathbf{H}_k^i)^T + \mathbf{M}_k^i\mathbf{N}_k(\mathbf{M}_k^i)^T)^{-1}$
      $\mathbf{x}_k^{i+1} \leftarrow \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^i\big(\mathbf{y}_k - \mathbf{h}_k(\mathbf{x}_k^i, \mathbf{0}) - \mathbf{H}_k^i(\hat{\mathbf{x}}_{k|k-1} - \mathbf{x}_k^i)\big)$
      $\mathbf{x}_k^i \leftarrow \mathbf{x}_k^{i+1}$
    **end while**
    $\hat{\mathbf{x}}_{k|k} \leftarrow \mathbf{x}_k^i$
    $\boldsymbol{\Sigma}_{k|k} \leftarrow (\mathbf{I} - \mathbf{K}_k^i\mathbf{H}_k^i)\boldsymbol{\Sigma}_{k|k-1}$
  **end if**
**end loop**

▶ **Example 3.3.** We consider the same setting and ground truth as in Example 3.2, except that we use an IterEKF rather than an EKF to estimate the system state. Since the prediction stage is identical for both filters, it yields the same result as in

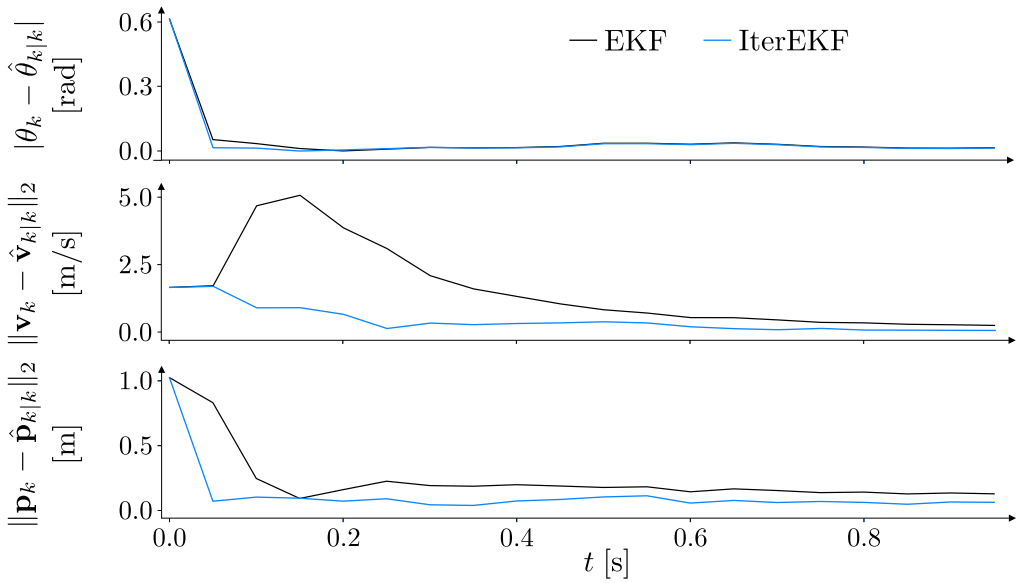



**Figure 3.7:** Evolution of the norm of the IterEKF estimation error for the worker's heading angle, velocity, and position over the first 20 time steps. The results obtained with the EKF are shown for comparison.

the EKF case. During the update, the inner iterations are stopped as soon as

$$\|\mathbf{K}_k^i \mathbf{z}_k^i - \mathbf{K}_k^{i-1} \mathbf{z}_k^{i-1}\|_2 < 10^{-4} \text{or} \quad i > i_{max},$$

where the maximum number of iterations is set to $i_{\max} = 20$.

**Update** The update stage yields the following estimate:

$$\hat{\mathbf{x}}_{1|1} = (3.9430, -2.2639, 0.1265, 5.0014, 5.0017),$$
$$\boldsymbol{\Sigma}_{1|1} = \begin{bmatrix} 0.0001 & 0.0000 & 0.0000 & -0.0001 & 0.0001 \\ 0.0000 & 0.9975 & 0.0000 & 0.0002 & 0.0000 \\ 0.0000 & 0.0000 & 0.9975 & 0.0000 & 0.0002 \\ -0.0001 & 0.0002 & 0.0000 & 0.0035 & -0.0002 \\ 0.0001 & 0.0000 & 0.0002 & -0.0002 & 0.0035 \end{bmatrix}.$$

The simulation is run over the first 20 time steps. Figure 3.7 shows the evolution of the norm of the estimation error for the worker's heading angle, velocity, and position. Compared to the EKF, the iterative update significantly improves the convergence behavior, particularly for the velocity estimate. ◀

# 4 The invariant extended Kalman filter

The Kalman filtering methods introduced in the previous chapters assume that the system state evolves in a Euclidean space. In many practical estimation problems, however, the state naturally lives on a nonlinear manifold. This is particularly common in navigation and robotics, where the state may include attitudes (quaternions or rotation matrices) or rigid-body poses. Accounting for this geometry has motivated the development of geometric filtering methods, which are designed to respect and exploit the structure of the underlying state space.

Several such methods have become state-of-the-art in navigation and pose estimation. Recent examples include equivariant filtering [76, 77] and its applications to inertial navigation [78, 79], visual-inertial odometry [80], and simultaneous localization and mapping [81, 82]. Geometric extensions of the Kalman filter family have also been proposed. The multiplicative EKF (MEKF), originally developed in aerospace, handles attitude estimation through a quaternion-based error representation [83] and remains widely used in practice [84–86]. Other approaches include the EKF on Lie groups [87, 88] and its iterated variant [89], as well as unscented Kalman filtering on Lie groups [90, 91]. Among these methods, the invariant extended Kalman filter (IEKF) [38, 39, 44] has attracted particular attention, as it provides strong theoretical guarantees while remaining close to the classical EKF in both spirit and implementation.

This chapter is devoted to the IEKF. Since the method relies heavily on matrix Lie group theory, we begin with a concise introduction to the necessary concepts before presenting the invariant filtering framework. For more comprehensive treatments of Lie groups, the reader is referred to [92] and [93].

## 4.1 A primer on matrix Lie groups

Before proceeding further, it is essential to introduce the mathematical notion of a group, which lies at the core of the theory developed in the sequel.

▶ **Definition 4.1 (Group).** A group is a set $G$ equipped with a binary operation, called the group composition law, that combines any two elements $a, b \in G$ into a third element $a \cdot b \in G$. This operation must satisfy the following axioms:

- (Associativity) $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in G$,

- (Identity element) There exists a unique element $e \in G$, called the identity, such that $e \cdot a = a \cdot e = a$ for all $a \in G$,
- (Inverse element) For each $a \in G$, there exists a unique element $a^{-1} \in G$ such that $a \cdot a^{-1} = a^{-1} \cdot a = e$.

◀

From this definition, it is clear that the notion of a group is more general than that of a vector space: every vector space satisfies the group axioms when equipped with vector addition as the binary operation. We now add an additional layer of structure by introducing the concept of a Lie group.

▶ **Definition 4.2 (Lie group).** A Lie group is a group $G$ that is also a smooth manifold, such that the group composition and inversion operations are smooth (i.e., infinitely differentiable) maps. ◀

In other words, a Lie group is a group whose elements vary continuously and smoothly, allowing tools from differential calculus to be applied to its algebraic structure. Lie groups are named after the Norwegian mathematician Marius Sophus Lie, who made foundational contributions to the theory of continuous symmetries.

So far, nothing has been specified about the nature of the elements of $G$. In this work, we are particularly interested in Lie groups whose elements are matrices. A fundamental example of such a Lie group is the set of all invertible $N \times N$ matrices with complex entries, called the general linear group:

$$GL(N, \mathbb{C}) = \boldsymbol{\chi} \in \mathbb{C}^{N \times N} \mid \det(\boldsymbol{\chi}) \neq 0, \tag{4.1}$$

with matrix multiplication as the group composition law, the identity matrix as the identity element, and classical matrix inversion as the inverse operation. This group plays a central role, as any matrix Lie group can be viewed as a subgroup of $GL(N, \mathbb{C})$, as formalized in the following definition.

▶ **Definition 4.3 (Matrix Lie group).** A matrix Lie group is a closed subgroup $G$ of $GL(N, \mathbb{C})$: for any sequence $(\boldsymbol{\chi}_m)_{m \in \mathbb{N}}$ with $\boldsymbol{\chi}_m \in G$ and $\lim_{m \to \infty} \boldsymbol{\chi}_m = \boldsymbol{\chi}$, the limit $\boldsymbol{\chi}$ either belongs to $G$ or is not invertible. ◀

The closedness condition on $G$ can be regarded as a technical requirement. In most practical cases, the matrix Lie groups of interest satisfy the stronger property that any convergent sequence in $G$ converges to an element of $G$. All matrix Lie groups considered in this work fall into this category.

▶ **Example 4.1.** A common and important example of a matrix Lie group is the special orthogonal group, denoted by $SO(d)$ and defined as

$$SO(d) = \{\mathbf{R} \in GL(d, \mathbb{R}) \mid \mathbf{R}^T\mathbf{R} = \mathbf{I},\ \det(\mathbf{R}) = 1\}. \tag{4.2}$$

For $d = 2$ or 3, the group $SO(d)$ is the set of rotation matrices. It is widely used in robotics and navigation to encode orientations and represent rotations in $\mathbb{R}^d$.

Another important matrix Lie group that will be used extensively in the sequel is the group of extended poses, denoted by $SE_k(d)$. This group generalizes the special Euclidean group of homogeneous transformations $SE(d)$ by augmenting the rotational component with $k$ translational components. It is defined as

$$SE_k(d) = \left\{ \left[\begin{array}{c|ccc} \mathbf{R} & \mathbf{x}_1 & \cdots & \mathbf{x}_k \\ \hline \mathbf{0} & \multicolumn{3}{c}{\mathbf{I}} \end{array}\right] \middle| \begin{array}{c} \mathbf{R} \in SO(d), \\ \mathbf{x}_1, \ldots, \mathbf{x}_k \in \mathbb{R}^d \end{array} \right\}. \tag{4.3}$$

◀

### 4.1.1 Tangent spaces and Lie algebra

Since matrix Lie groups are smooth manifolds, the tools of differential geometry naturally apply to them, in particular the notion of tangent spaces. To build intuition for this concept in the context of matrix Lie groups, consider smooth curves $\gamma(t) \in G$, with $t \in \mathbb{R}$, such that $\gamma(0) = \mathbf{I}$, the identity element of the group. Two such curves are said to coincide to first order at the identity if they have the same derivative at $t = 0$. The derivative

$$\left.\frac{d\gamma}{dt}\right|_{t=0}$$

is a matrix that represents the infinitesimal motion along the curve at the identity. The collection of all such derivatives, taken over all admissible curves $\gamma(t)$, forms a vector space called the tangent space at the identity element, denoted $T_{\mathbf{I}}G$.

While vector addition is commutative by definition, the composition law of a group is generally not, as is the case for matrix multiplication. As a consequence, composing two infinitesimal motions in different orders typically yields different results. This non-commutativity is not merely a higher-order effect, but already appears at first order, and must therefore be reflected in the structure of the tangent space itself. This observation motivates the abstract definition of a Lie algebra, which endows a vector space with additional structure capable of capturing this infinitesimal non-commutativity.

▶ **Definition 4.4 (Lie algebra).** A (finite-dimensional) real or complex Lie algebra is a finite-dimensional real or complex vector space $\mathfrak{g}$ equipped with a map

$$[\cdot, \cdot] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}, \tag{4.4}$$

called the Lie bracket, such that:

- (Bilinearity) $[\cdot, \cdot]$ is bilinear,
- (Skew-symmetry) $[\mathbf{X}, \mathbf{Y}] = -[\mathbf{Y}, \mathbf{X}]$ for all $\mathbf{X}, \mathbf{Y} \in \mathfrak{g}$,
- (Jacobi identity) for all $\mathbf{X}, \mathbf{Y}, \mathbf{Z} \in \mathfrak{g}$,

$$[\mathbf{X}, [\mathbf{Y}, \mathbf{Z}]] + [\mathbf{Y}, [\mathbf{Z}, \mathbf{X}]] + [\mathbf{Z}, [\mathbf{X}, \mathbf{Y}]] = \mathbf{0}. \tag{4.5}$$

◀

Every matrix Lie group $G$ has an associated Lie algebra $\mathfrak{g}$, defined as follows.

▶ **Definition 4.5 (Lie algebra of a matrix Lie group).** The Lie algebra $\mathfrak{g}$ of a matrix Lie group $G$ is the set of all matrices $\mathbf{X}$ such that the curve $\exp_m(t\mathbf{X})$ lies entirely in $G$ for all $t \in \mathbb{R}$, equipped with the Lie bracket defined by the matrix commutator

$$[\mathbf{X}, \mathbf{Y}] = \mathbf{XY} - \mathbf{YX}, \tag{4.6}$$

for all $\mathbf{X}, \mathbf{Y} \in \mathfrak{g}$. ◀

This definition relies on the matrix exponential $\exp_m : M_N(\mathbb{C}) \to GL(N, \mathbb{C})$ to parametrize a curve in $G$, where $M_N(\mathbb{C})$ denotes the set of all $N \times N$ matrices with complex entries. As will be seen in the next section, this function plays a fundamental role in the theory of matrix Lie groups. Since

$$\frac{d}{dt} \exp_m(t\mathbf{X}) \bigg|_{t=0} = \mathbf{X}, \tag{4.7}$$

it follows that the Lie algebra $\mathfrak{g}$ of $G$ is simply the tangent space $T_{\mathbf{I}}G$, with the matrix commutator serving as the Lie bracket.

So far, we have focused on the tangent space at the identity element. However, every point $\boldsymbol{\chi} \in G$ of a matrix Lie group also admits a tangent space, denoted $T_{\boldsymbol{\chi}}G$. This tangent space can be obtained from the Lie algebra $\mathfrak{g}$ through either left or right translation by $\boldsymbol{\chi}$: for any $\mathbf{X} \in \mathfrak{g}$, the matrices $\boldsymbol{\chi}\mathbf{X}$ and $\mathbf{X}\boldsymbol{\chi}$ both belong to $T_{\boldsymbol{\chi}}G$. They can be interpreted as the derivatives at $t = 0$ of the curves $\boldsymbol{\chi} \exp_m(t\mathbf{X})$ and $\exp_m(t\mathbf{X})\boldsymbol{\chi}$, respectively. An illustration summarizing the relationship between the Lie algebra and the tangent spaces is provided in Figure 4.1.

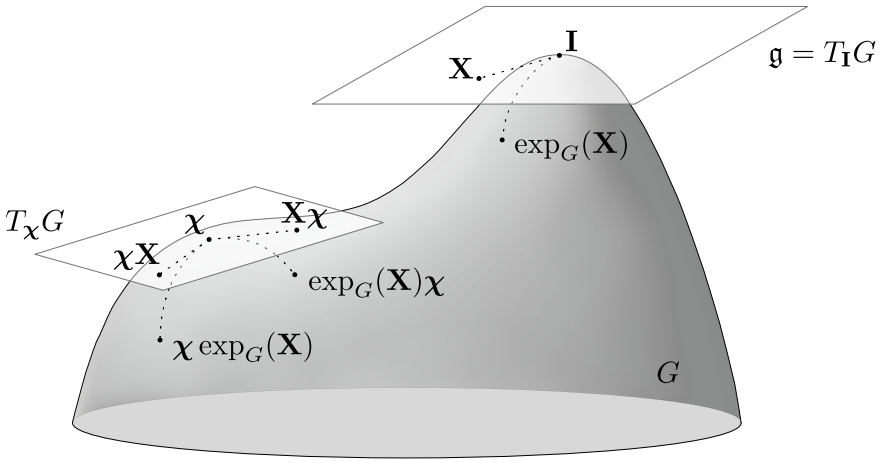



**Figure 4.1:** Illustration of the concept of Lie algebra and tangent spaces of a Lie group $G$.

### 4.1.2 The exponential and logarithmic maps

As suggested by Definition 4.5, the matrix exponential provides a fundamental link between the Lie algebra $\mathfrak{g}$, which is a linear vector space, and the nonlinear group structure. In particular, it allows elements of $\mathfrak{g}$ to be mapped onto the Lie group $G$, thereby enabling the transfer of local, linear information to the curved manifold on which the state evolves.

▶ **Definition 4.6 (Exponential map of a matrix Lie group).** Let $G \subseteq GL(N, \mathbb{C})$ be a matrix Lie group with Lie algebra $\mathfrak{g}$. The exponential map of $G$ is the mapping

$$\exp_G : \mathfrak{g} \to G$$

defined as the restriction to $\mathfrak{g}$ of the matrix exponential

$$\exp_m(\mathbf{X}) = \sum_{l=0}^{+\infty} \frac{\mathbf{X}^l}{l!}, \tag{4.8}$$

for all $\mathbf{X} \in \mathfrak{g}$. ◀

The above series converges absolutely for all matrices in $M_N(\mathbb{C})$, and therefore for all $\mathbf{X} \in \mathfrak{g}$. Since $\exp_G(\mathbf{X})$ depends smoothly on $\mathbf{X}$, the mapping $t \to \exp_m(t\mathbf{X})$ defines a smooth curve in $G$, whose derivative is given by

$$\frac{d}{dt} \exp_m(t\mathbf{X}) = \mathbf{X} \exp_m(t\mathbf{X}) = \exp_m(t\mathbf{X})\mathbf{X}. \tag{4.9}$$

Similarly, one can define a logarithmic map for a matrix Lie group.

▶ **Definition 4.7 (Logarithmic map of a matrix Lie group).** Let $G \subseteq GL(N, \mathbb{C})$ be a matrix Lie group with Lie algebra $\mathfrak{g}$. The logarithmic map of $G$ is the mapping

$$\log_G : G \to \mathfrak{g}$$

defined as the restriction to $G$ of the matrix logarithm

$$\log_m(\boldsymbol{\chi}) = \sum_{l=1}^{+\infty} (-1)^{l+1} \frac{(\boldsymbol{\chi} - \mathbf{I})^l}{l}, \tag{4.10}$$

for all $\boldsymbol{\chi} \in G$ for which the series converges. ◀

In contrast with the matrix exponential, the series in Equation (4.10) does not converge for all matrices in $M_N(\mathbb{C})$. However, it is well defined and continuous for all matrices $\boldsymbol{\chi} \in M_N(\mathbb{C})$ satisfying

$$\|\boldsymbol{\chi} - \mathbf{I}\| < 1, \tag{4.11}$$

where $\|\cdot\|$ denotes the Hilbert–Schmidt norm, defined by

$$\|\mathbf{A}\| = \sqrt{\sum_{i,j} |A_{ij}|^2}, \tag{4.12}$$

with $A_{ij}$ denoting the entry of $\mathbf{A}$ at the intersection of the $i^{\text{th}}$ row and $j^{\text{th}}$ column. This shows that the logarithmic map is not globally defined on $G$, nor is it a global inverse of the exponential map, as is sometimes mistakenly assumed. Nevertheless, for all $\boldsymbol{\chi} \in G$ satisfying Equation (4.11), one has

$$\exp_G\big(\log_G(\boldsymbol{\chi})\big) = \boldsymbol{\chi}. \tag{4.13}$$

Moreover, if $\mathbf{X} \in \mathfrak{g}$ is such that $\|\mathbf{X}\| < \log(2)$, then $\|\exp_G(\mathbf{X}) - \mathbf{I}\| < 1$ and

$$\log_G\big(\exp_G(\mathbf{X})\big) = \mathbf{X}. \tag{4.14}$$

The exponential map is therefore bijective only locally, on the neighborhood

$$\{\mathbf{X} \in \mathfrak{g} \mid \|\mathbf{X}\| < \log(2)\}.$$

The convergence of the power series in Equation (4.10) and the above results are established in the proof of Theorem 2.9 in [92].

Upon choosing a basis of $\mathfrak{g}$, there exists a linear isomorphism

$$\mathcal{L}_{\mathfrak{g}} : \mathbb{C}^n \to \mathfrak{g} \tag{4.15}$$

which provides a coordinate representation of elements of the Lie algebra. The dimension $n$ coincides with the dimension of the Lie group and satisfies $n \leq N^2$.

▶ **Example 4.2.** Any rotation in three dimensions can be represented by a rotation vector, whose direction specifies the axis of rotation and whose norm corresponds to the rotation angle. Given any rotation vector $\boldsymbol{\phi} \in \mathbb{R}^3$, the associated rotation matrix $\mathbf{R} \in SO(3)$ is obtained as

$$\mathbf{R} = \exp_{SO(3)}\big(\mathcal{L}_{\mathfrak{so}(3)}(\boldsymbol{\phi})\big), \tag{4.16}$$

where the bijective linear map $\mathcal{L}_{\mathfrak{so}(3)} : \mathbb{R}^3 \to \mathfrak{so}(3)$ associates each vector with the corresponding skew-symmetric matrix

$$\mathcal{L}_{\mathfrak{so}(3)}(\boldsymbol{\phi}) = (\boldsymbol{\phi})_\times = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix}, \tag{4.17}$$

with $\phi_i$ denoting the $i^{\text{th}}$ component of $\boldsymbol{\phi}$. For any given $\boldsymbol{\phi} \in \mathbb{R}^3$, the curve $\exp_{SO(3)}(t\,\mathcal{L}_{\mathfrak{so}(3)}(\boldsymbol{\phi}))$ belongs to $SO(3)$ for all $t \in \mathbb{R}$. The Lie algebra $\mathfrak{so}(3)$ can thus be defined as

$$\mathfrak{so}(3) = \{\mathcal{L}_{\mathfrak{so}(3)}(\boldsymbol{\phi}) \mid \boldsymbol{\phi} \in \mathbb{R}^3\}. \tag{4.18}$$

In the two-dimensional case, the skew-symmetric matrix reduces to

$$\mathcal{L}_{\mathfrak{so}(2)}(\phi) = (\phi)_\times = \begin{bmatrix} 0 & -\phi \\ \phi & 0 \end{bmatrix}, \tag{4.19}$$

where the same wedge notation $(\cdot)_\times$ is used as for $\mathfrak{so}(3)$. The distinction between the two cases is unambiguous from the dimension of the input. This leads to the following definition of the Lie algebra $\mathfrak{so}(2)$:

$$\mathfrak{so}(2) = \{\mathcal{L}_{\mathfrak{so}(2)}(\phi) \mid \phi \in \mathbb{R}\}. \tag{4.20}$$

The group $SO(2)$ is abelian, meaning that its elements commute. As a consequence, its Lie algebra is also commutative, and the Lie bracket vanishes identically, that is, $[\mathbf{X}, \mathbf{Y}] = \mathbf{0}$ for all $\mathbf{X}, \mathbf{Y} \in \mathfrak{so}(2)$.

Similarly to rotation matrices, any element $\boldsymbol{\chi} \in SE_k(3)$ can be expressed as

$$\boldsymbol{\chi} = \exp_{SE_k(3)}\big(\mathcal{L}_{\mathfrak{se}_k(3)}(\boldsymbol{\zeta})\big), \tag{4.21}$$

where the bijective linear map $\mathcal{L}_{\mathfrak{se}_k(3)} : \mathbb{R}^{3+3k} \to \mathfrak{se}_k(3)$ is defined by

$$\mathcal{L}_{\mathfrak{se}_k(3)}\left(\begin{bmatrix}\boldsymbol{\phi}\\ \boldsymbol{\zeta}_1\\ \vdots\\ \boldsymbol{\zeta}_k\end{bmatrix}\right) = \left[\begin{array}{c|ccc}(\boldsymbol{\phi})_\times & \boldsymbol{\zeta}_1 & \cdots & \boldsymbol{\zeta}_k\\ \hline \mathbf{0} & \multicolumn{3}{c}{\mathbf{0}}\end{array}\right], \tag{4.22}$$

with $\boldsymbol{\phi}, \boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_k \in \mathbb{R}^3$. This leads to the following definition of the Lie algebra:

$$\mathfrak{se}_k(3) = \{\mathcal{L}_{\mathfrak{se}_k(3)}(\boldsymbol{\xi}) \mid \boldsymbol{\xi} \in \mathbb{R}^{3+3k}\}. \tag{4.23}$$

In the two-dimensional case, the linear map $\mathcal{L}_{\mathfrak{se}_k(2)} : \mathbb{R}^{1+2k} \to \mathfrak{se}_k(2)$ reduces to

$$\mathcal{L}_{\mathfrak{se}_k(2)}\left(\begin{bmatrix}\phi\\ \boldsymbol{\zeta}_1\\ \vdots\\ \boldsymbol{\zeta}_k\end{bmatrix}\right) = \left[\begin{array}{c|ccc}(\phi)_\times & \boldsymbol{\zeta}_1 & \cdots & \boldsymbol{\zeta}_k\\ \hline \mathbf{0} & \multicolumn{3}{c}{\mathbf{0}}\end{array}\right], \tag{4.24}$$

where $\phi \in \mathbb{R}$ and $\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_k \in \mathbb{R}^2$, inducing the Lie algebra

$$\mathfrak{se}_k(2) = \{\mathcal{L}_{\mathfrak{se}_k(2)}(\boldsymbol{\xi}) \mid \boldsymbol{\xi} \in \mathbb{R}^{1+2k}\}. \tag{4.25}$$

◀

### 4.1.3 The adjoint map

Consider an element $\mathbf{X} \in \mathfrak{g}$, interpreted as an infinitesimal displacement in a neighborhood of the identity element $\mathbf{I}$ of $G$. By left or right multiplication, this infinitesimal displacement can be transported to a neighborhood of an arbitrary group element $\boldsymbol{\chi} \in G$. This naturally raises the following question: how does an infinitesimal displacement transform under such transport around the group? More specifically, does transporting $\mathbf{X}$ from $\mathbf{I}$ to $\boldsymbol{\chi}$ by left multiplication, and then transporting the result back to $\mathbf{I}$ by right multiplication by $\boldsymbol{\chi}^{-1}$, leave $\mathbf{X}$ unchanged? In general, the answer is no. The adjoint map provides a precise description of how infinitesimal displacements are modified under this change of reference.

▶ **Definition 4.8 (Adjoint map).** Let $G \subseteq GL(N, \mathbb{C})$ be a matrix Lie group with Lie algebra $\mathfrak{g}$. For each $\boldsymbol{\chi} \in G$, the adjoint map is the linear map $\mathrm{Ad}_{\boldsymbol{\chi}} : \mathfrak{g} \to \mathfrak{g}$ defined by

$$\mathrm{Ad}_{\boldsymbol{\chi}}(\mathbf{X}) = \boldsymbol{\chi}\mathbf{X}\boldsymbol{\chi}^{-1}, \tag{4.26}$$

for all $\mathbf{X} \in \mathfrak{g}$. ◀

An infinitesimal motion is a direction attached to the identity, and the adjoint map tells you how that direction must be rewritten when the same motion is applied at a different point of the group and then pulled back to the identity. Equivalently, the adjoint map $\mathrm{Ad}_{\boldsymbol{\chi}}$ is the differential at $\mathbf{I}$ of the conjugation operation by the group element $\boldsymbol{\chi}$, that is,

$$\mathrm{Ad}_{\boldsymbol{\chi}}(\mathbf{X}) = \frac{d}{dt}\boldsymbol{\chi}\exp_G(t\mathbf{X})\boldsymbol{\chi}^{-1}\Big|_{t=0}. \tag{4.27}$$

Upon choosing a basis of $\mathfrak{g}$ and the associated linear isomorphism $\mathcal{L}_{\mathfrak{g}} : \mathbb{C}^n \to \mathfrak{g}$, the adjoint map $\mathrm{Ad}_{\boldsymbol{\chi}}$ admits a matrix representation $\mathbf{Ad}_{\boldsymbol{\chi}} \in GL(n, \mathbb{C})$, called adjoint matrix. In particular, for any $\mathbf{X} \in \mathfrak{g}$ written as $\mathbf{X} = \mathcal{L}_{\mathfrak{g}}(\boldsymbol{\xi})$ with $\boldsymbol{\xi} \in \mathbb{C}^n$:

$$\mathrm{Ad}_{\boldsymbol{\chi}}(\mathbf{X}) = \mathcal{L}_{\mathfrak{g}}(\mathbf{Ad}_{\boldsymbol{\chi}}\boldsymbol{\xi}). \tag{4.28}$$

The differential of the adjoint map at the identity also has an important interpretation. It induces a linear map on the Lie algebra itself, called the adjoint map of the Lie algebra.

▶ **Definition 4.9 (Adjoint map of the Lie algebra).** Let $G \subseteq GL(N, \mathbb{C})$ be a matrix Lie group with Lie algebra $\mathfrak{g}$. For each $\mathbf{X} \in \mathfrak{g}$, the adjoint map of the Lie algebra is the linear mapping $\mathrm{ad}_{\mathbf{x}} : \mathfrak{g} \to \mathfrak{g}$ defined by

$$\mathrm{ad}_{\mathbf{X}} = \frac{d}{dt}\mathrm{Ad}_{\exp_G(t\mathbf{X})}\Big|_{t=0}. \tag{4.29}$$

◀

Computing the derivative in the previous definition yields

$$\mathrm{ad}_{\mathbf{X}}(\mathbf{Y}) = [\mathbf{X}, \mathbf{Y}], \tag{4.30}$$

so that the adjoint map of the Lie algebra coincides with the Lie bracket. This shows that $\mathrm{ad}_{\mathbf{X}}$ describes the infinitesimal effect of transporting an infinitesimal displacement $\mathbf{Y}$ along the infinitesimal motion generated by $\mathbf{X}$. The adjoint map

$\mathrm{Ad}$ and the Lie algebra adjoint map $\mathrm{ad}$ thus provide complementary descriptions of how infinitesimal motions transform under changes of reference: $\mathrm{Ad}$ describes this transformation under finite changes of reference on the group, while $\mathrm{ad}$ captures its infinitesimal counterpart within the Lie algebra itself.

▶ **Example 4.3.** Consider the group $SO(3)$. Identifying the Lie algebra $\mathfrak{so}(3)$ with $\mathbb{R}^3$ via the linear isomorphism $\mathcal{L}_{\mathfrak{so}(3)}(\cdot)$, it is straightforward to verify that

$$\mathrm{Ad}_{\mathbf{R}}\big(\mathcal{L}_{\mathfrak{so}(3)}(\boldsymbol{\phi})\big) = \mathcal{L}_{\mathfrak{so}(3)}(\mathbf{R}\boldsymbol{\phi}), \tag{4.31}$$

for all $\mathbf{R} \in SO(3)$ and all $\boldsymbol{\phi} \in \mathbb{R}^3$, so that the adjoint matrix is given by

$$\mathbf{Ad}_{\mathbf{R}} = \mathbf{R}. \tag{4.32}$$

In contrast, the abelian nature of the group $SO(2)$ implies the trivial adjoint action:

$$\mathrm{Ad}_{\mathbf{R}}\big(\mathcal{L}_{\mathfrak{so}(2)}(\boldsymbol{\phi})\big) = \mathcal{L}_{\mathfrak{so}(2)}(\boldsymbol{\phi}), \tag{4.33}$$

for all $\mathbf{R} \in SO(2)$ and all $\boldsymbol{\phi} \in \mathbb{R}$, so that the adjoint matrix reduces to the scalar

$$\mathbf{Ad}_{\mathbf{R}} = 1. \tag{4.34}$$

For a general element $\boldsymbol{\chi} \in SE_k(3)$, as defined in Equation (4.3), the adjoint matrix takes the block form

$$\mathbf{Ad}_{\boldsymbol{\chi}} = \left[\begin{array}{c|c} \mathbf{R} & \mathbf{0} \\ \hline (\mathbf{x}_1)_\times \mathbf{R} & \\ \vdots & \mathrm{block\_diag}(\underbrace{\mathbf{R}, \ldots, \mathbf{R}}_{k}) \\ (\mathbf{x}_k)_\times \mathbf{R} & \end{array}\right], \tag{4.35}$$

which explicitly shows how the rotational and translational components are coupled through the adjoint action.

In the two-dimensional case, for $\boldsymbol{\chi} \in SE_k(2)$, the adjoint matrix becomes

$$\mathbf{Ad}_{\boldsymbol{\chi}} = \left[\begin{array}{c|c} 1 & \mathbf{0} \\ \hline -(1)_\times \mathbf{x}_1 & \\ \vdots & \mathrm{block\_diag}(\underbrace{\mathbf{R}, \ldots, \mathbf{R}}_{k}) \\ -(1)_\times \mathbf{x}_k & \end{array}\right]. \tag{4.36}$$

◀

### 4.1.4 The Baker–Campbell–Hausdorff formula

Several familiar properties of the scalar exponential function do not carry over to the exponential map on a Lie group. In particular, for $\mathbf{X}, \mathbf{Y} \in \mathfrak{g}$, one generally has

$$\exp_G(\mathbf{X}+\mathbf{Y}) \neq \exp_G(\mathbf{X})\exp_G(\mathbf{Y}), \tag{4.37}$$

which is a direct consequence of the non-commutativity of matrix multiplication. To relate the product of two matrix exponentials to a single exponential, one uses the Baker–Campbell–Hausdorff (BCH) formula.

The BCH formula states that, for $\mathbf{X}$ and $\mathbf{Y}$ sufficiently close to the origin in $\mathfrak{g}$,

$$\begin{aligned}\log_G(\exp_G(\mathbf{X})\exp_G(\mathbf{Y})) = \\ \mathbf{X}+\mathbf{Y}+\tfrac{1}{2}[\mathbf{X},\mathbf{Y}]+\tfrac{1}{12}[\mathbf{X},[\mathbf{X},\mathbf{Y}]]-\tfrac{1}{12}[\mathbf{Y},[\mathbf{X},\mathbf{Y}]]+\cdots,\end{aligned} \tag{4.38}$$

where the omitted terms involve higher-order nested commutators.

More generally, the BCH formula admits the formal series expansion

$$\log_G(\exp_G(\mathbf{X})\exp_G(\mathbf{Y})) = \sum_{l=1}^{+\infty}\frac{(-1)^{l-1}}{l}\sum_{\substack{r_i+s_i>0\\1\leq i\leq l}}\frac{\left(\sum_{i=1}^{l}(r_i+s_i)\right)^{-1}}{\prod_{i=1}^{l}r_i!s_i!}[\mathbf{X}^{r_1}\mathbf{Y}^{s_1}\ldots\mathbf{X}^{r_l}\mathbf{Y}^{s_l}], \tag{4.39}$$

where the nested commutator is defined recursively by

$$[\mathbf{X}^{r_1}\mathbf{Y}^{s_1}\ldots\mathbf{X}^{r_l}\mathbf{Y}^{s_l}] = [\underbrace{\mathbf{X},\ldots[\mathbf{X}}_{r_1},\underbrace{\mathbf{Y},\ldots[\mathbf{Y}}_{s_1},\ldots\underbrace{[\mathbf{X},\ldots[\mathbf{X}}_{r_l},\underbrace{[\mathbf{Y},\ldots[\mathbf{Y},\mathbf{Y}]\ldots]}_{s_l}]. \tag{4.40}$$

In practice, the full Baker–Campbell–Hausdorff (BCH) series is rarely used, and truncated expressions are typically preferred. In particular, when one of the arguments is small, it is customary to retain only the terms that are linear in that argument. Keeping only the terms that are linear in $\mathbf{Y}$, the BCH formula reduces to

$$\log_G(\exp_G(\mathbf{X})\exp_G(\mathbf{Y})) \approx \mathbf{X}+\mathcal{J}_{-\mathbf{X}}^{-1}(\mathbf{Y}), \tag{4.41}$$

where $\mathcal{J}_{\mathbf{X}}^{-1}:\mathfrak{g}\to\mathfrak{g}$ is the operator defined by

$$\mathcal{J}_{\mathbf{X}}^{-1}(\mathbf{Y}) = \sum_{l=0}^{+\infty}\frac{B_l}{l!}\operatorname{ad}_{\mathbf{X}}^{l}(\mathbf{Y}) \tag{4.42}$$

with $B_l$ denoting the $l^{\text{th}}$ Bernoulli number and $\mathrm{ad}^l_{\mathbf{X}}(\mathbf{Y}) = [\overbrace{\mathbf{X}, [\mathbf{X}, \dots, [\mathbf{X}}^{l}, \mathbf{Y}] \dots]]$. Conversely, retaining only the terms that are linear in $\mathbf{X}$ yields

$$\log_G(\exp_G(\mathbf{X}) \exp_G(\mathbf{Y})) \approx \mathbf{Y} + \mathcal{J}_{\mathbf{Y}}^{-1}(\mathbf{X}). \tag{4.43}$$

The operator $\mathcal{J}_{\mathbf{X}}^{-1} : \mathfrak{g} \to \mathfrak{g}$ is called the inverse left-Jacobian of $G$. For $\mathbf{x} \in \mathfrak{g}$ belonging to a sufficiently small neighborhood of the origin, it is invertible. its inverse $\mathcal{J}_{\mathbf{X}}$ is naturally called the left-Jacobian of $G$ and is given by

$$\mathcal{J}_{\mathbf{X}}(\mathbf{Y}) = \sum_{l=0}^{+\infty} \frac{(-1)^l}{(l+1)!} \mathrm{ad}^l_{\mathbf{X}}(\mathbf{Y}). \tag{4.44}$$

Identifying the Lie algebra $\mathfrak{g}$ with $\mathbb{C}^n$ via the linear isomorphism $\mathcal{L}_{\mathfrak{g}}$, the operators $\mathcal{J}_{\mathbf{X}}$ and $\mathcal{J}_{\mathbf{X}}^{-1}$ admit matrix representations. Specifically, for $\mathbf{X} = \mathcal{L}_{\mathfrak{g}}(\boldsymbol{\xi})$ and $\mathbf{Y} = \mathcal{L}_{\mathfrak{g}}(\boldsymbol{\zeta})$, with $\boldsymbol{\xi}, \boldsymbol{\zeta} \in \mathbb{C}^n$, one has

$$\mathcal{J}_{\mathbf{X}}(\mathbf{Y}) = \mathcal{L}_{\mathfrak{g}}(\mathbf{J}_{\boldsymbol{\xi}} \boldsymbol{\zeta}), \tag{4.45}$$

where $\mathbf{J}_{\boldsymbol{\xi}} \in GL(n, \mathbb{C})$ is the matrix representation of the left-Jacobian operator $\mathcal{J}_{\mathbf{X}}$, and $\mathbf{J}_{\boldsymbol{\xi}}^{-1}$ the matrix representation of the inverse left-Jacobian operator $\mathcal{J}_{\mathbf{X}}^{-1}$.

▶ **Example 4.4.** Consider the group $SO(3)$ with the linear isomorphism $\mathcal{L}_{\mathfrak{so}(3)}$. The left-Jacobian operator admits the matrix representation given by

$$\mathbf{J}_{\boldsymbol{\phi}} = \frac{\sin(\phi)}{\phi}\mathbf{I} + \left(1 - \frac{\sin(\phi)}{\phi}\right)\mathbf{a}\mathbf{a}^T + \frac{1 - \cos(\phi)}{\phi}(\mathbf{a})_\times,$$

with $\boldsymbol{\phi} \in \mathbb{R}^3$, $\phi = \|\boldsymbol{\phi}\|_2$ and $\mathbf{a} = \boldsymbol{\phi}/\phi$. In the case $\phi = 0$, one recovers $\mathbf{J}_{\boldsymbol{\phi}} = \mathbf{I}$.

In the case of $SO(2)$, as the Lie bracket identically vanishes due to its abelian nature, the left-Jacobian reduces to the scalar $J_\phi = 1$ for all $\phi \in \mathbb{R}$.

For the group $SE_k(3)$, a closed-form expression of the left-Jacobian can be obtained as well. We refer the reader to the expression given for $SE_2(3)$ in [94].

The closed-form expression of the left-Jacobian of $SE_k(2)$ can be obtained from $SE_k(3)$ by restriction to the planar subgroup. ◀

## 4.2 Filter formulation

We now introduce the invariant Kalman filtering framework. For comprehensive treatments of the topic, the reader is referred to [37–39, 44].

The IEKF assumes that the state $\boldsymbol{\chi}_k$ evolves on a matrix Lie group $G \subseteq GL(N, \mathbb{R})$ of dimension $n$, with associated Lie algebra $\mathfrak{g}$. It further assumes that the system dynamics admit the hidden Markov model representation

$$\boldsymbol{\chi}_k = \mathbf{f}_{k-1}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \tag{4.46}$$
$$\mathbf{y}_k = \mathbf{h}_k(\boldsymbol{\chi}_k, \mathbf{n}_k), \tag{4.47}$$

where $\mathbf{u}_{k-1} \in \mathbb{R}^b$ denotes the system input, $\mathbf{y}_k \in \mathbb{R}^m$ the measurement output, and the process and measurement noises $\mathbf{w}_{k-1} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$ and $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ are mutually uncorrelated Gaussian white noise processes. The dynamics and measurement functions $\mathbf{f}_{k-1}$ and $\mathbf{h}_k$ are each assumed to satisfy a specific condition which, when met, confers the IEKF convergence properties reminiscent of those of the linear Kalman filter. The condition on $\mathbf{f}_{k-1}$ and the condition on $\mathbf{h}_k$ will be detailed in the subsections dedicated to the prediction and update stages, respectively.

The fact that the state space is no longer a vector space but a matrix Lie group entails several fundamental modifications to the Kalman filtering process. One of the most significant is that the group composition law is given by matrix multiplication and is, in general, non-commutative. As a consequence, the IEKF can be designed according to two distinct paradigms, referred to as the left- and right-invariant paradigms, depending on whether elements of $G$ are composed through left or right multiplication. This leads to two corresponding filters, the left- and right-IEKF, respectively denoted by L-IEKF and R-IEKF. In this context, the invariant nonlinear error is defined as

$$\boldsymbol{\eta}^L_{k|l} = \hat{\boldsymbol{\chi}}^{-1}_{k|l}\boldsymbol{\chi}_k, \qquad \text{(L-IEKF)} \tag{4.48}$$
$$\boldsymbol{\eta}^R_{k|l} = \boldsymbol{\chi}_k\hat{\boldsymbol{\chi}}^{-1}_{k|l}. \qquad \text{(R-IEKF)} \tag{4.49}$$

The term invariant stems from the fact that $\boldsymbol{\eta}_{k|l}$ is invariant under any left or right translation of both $\boldsymbol{\chi}_k$ and $\hat{\boldsymbol{\chi}}_{k|l}$ by any other element in $G$. Indeed, for all $\boldsymbol{\nu} \in G$,

$$\hat{\boldsymbol{\chi}}^{-1}_{k|l}\boldsymbol{\chi}_k = (\boldsymbol{\nu}\hat{\boldsymbol{\chi}}_{k|l})^{-1}(\boldsymbol{\nu}\boldsymbol{\chi}_k), \qquad \boldsymbol{\chi}_k\hat{\boldsymbol{\chi}}^{-1}_{k|l} = (\boldsymbol{\chi}_k\boldsymbol{\nu})(\hat{\boldsymbol{\chi}}_{k|l}\boldsymbol{\nu})^{-1}. \tag{4.50}$$

In the sequel, the superscripts $L$ and $R$ are dropped whenever both paradigms are treated in a similar manner.

Since the nonlinear error $\boldsymbol{\eta}_{k|l}$ takes values in $G$, the usual assumption that the estimation error is Gaussian distributed in a vector space does not apply. In order to define a probability density for the true state $\boldsymbol{\chi}_k$ that remains entirely on $G$, we instead define a distribution on $\mathbb{R}^n$ and use the coordinate representation $\mathcal{L}_{\mathfrak{g}} : \mathbb{R}^n \to \mathfrak{g}$ together with the exponential map to transfer this distribution from

$\mathbb{R}^n$ to $G$ around the identity element. The resulting dispersion is then transported around the estimate $\hat{\boldsymbol{\chi}}_{k|l}$ through group multiplication. This construction is known as a concentrated Gaussian on $G$ [87, 95–97] and is used in the invariant filtering framework to model uncertainty about the true state. Mathematically, this amounts to writing

$$\boldsymbol{\chi}_k = \hat{\boldsymbol{\chi}}_{k|l} \exp(\boldsymbol{\xi}^L_{k|l}) \quad \text{with} \quad \boldsymbol{\xi}^L_{k|l} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^L_{k|l}), \qquad \text{(L-IEKF)} \tag{4.51}$$

$$\boldsymbol{\chi}_k = \exp(\boldsymbol{\xi}^R_{k|l}) \hat{\boldsymbol{\chi}}_{k|l} \quad \text{with} \quad \boldsymbol{\xi}^R_{k|l} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}^R_{k|l}), \qquad \text{(R-IEKF)} \tag{4.52}$$

where $\exp = \exp_G \circ \mathcal{L}_{\mathfrak{g}}$. The Gaussian random vector $\boldsymbol{\xi}_{k|l} \in \mathbb{R}^n$ is referred to as the linearized error, and is assumed to be uncorrelated with both the process and measurement noises. For notational convenience, we write

$$\boldsymbol{\chi}_k \sim \mathcal{N}^L_G(\hat{\boldsymbol{\chi}}_{k|l}, \boldsymbol{\Sigma}^L_{k|l}), \qquad \text{(L-IEKF)} \tag{4.53}$$

$$\boldsymbol{\chi}_k \sim \mathcal{N}^R_G(\hat{\boldsymbol{\chi}}_{k|l}, \boldsymbol{\Sigma}^R_{k|l}), \qquad \text{(R-IEKF)} \tag{4.54}$$

to denote left- and right-concentrated Gaussian distributions on $G$.

At all times, the IEKF thus maintains a concentrated Gaussian distribution for the true state. Under the hidden Markov model assumption, this can be achieved using the recursive formulation of Bayesian inference, which naturally decomposes into separate prediction and update stages. However, propagating the filter belief through $\mathbf{f}_{k-1}$ and performing exact Bayesian inference with $\mathbf{h}_k$ do not, in general, preserve the concentrated Gaussian form. The IEKF therefore relies on linearization in order to maintain its concentrated Gaussian representation of the state.

### 4.2.1 The prediction stage

Assume that we are at time index $k-1$ and that the posterior distribution of the state is represented by a concentrated Gaussian with parameters $\hat{\boldsymbol{\chi}}_{k-1|k-1}$ and $\boldsymbol{\Sigma}_{k-1|k-1}$, either left- or right-invariant depending on the chosen paradigm. Similarly to the EKF, the IEKF propagates the state estimate through the noise-free dynamics,

$$\hat{\boldsymbol{\chi}}_{k|k-1} = \mathbf{f}_{k-1}(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}). \tag{4.55}$$

Using the current posterior to express $\boldsymbol{\chi}_{k-1}$ as a function of the state estimate and the linearized error, the dynamics of the invariant nonlinear error can be written as

$$\boldsymbol{\eta}^L_{k|k-1} = \mathbf{f}_{k-1}\big(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}\big)^{-1} \mathbf{f}_{k-1}\big(\hat{\boldsymbol{\chi}}_{k-1|k-1} \exp(\boldsymbol{\xi}^L_{k-1|k-1}), \mathbf{u}_{k-1}, \mathbf{w}_{k-1}\big), \tag{4.56}$$

$$\boldsymbol{\eta}^R_{k|k-1} = \mathbf{f}_{k-1}\big(\exp(\boldsymbol{\xi}^R_{k-1|k-1})\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}\big)\mathbf{f}_{k-1}\big(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{0}\big)^{-1}. \tag{4.57}$$

Denoting by $\log$ the composition $\mathcal{L}_{\mathfrak{g}}^{-1} \circ \log_G$, and assuming that the invariant nonlinear error remains sufficiently small, the dynamics of the linearized error are

$$\boldsymbol{\xi}_{k|k-1} = \tilde{\mathbf{f}}_{k-1}(\boldsymbol{\xi}_{k-1|k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) = \log(\boldsymbol{\eta}_{k|k-1}), \tag{4.58}$$

In order to preserve a concentrated Gaussian representation of the state, the IEKF propagates the error covariance using a first-order approximation of the linearized error dynamics with respect to $\boldsymbol{\xi}_{k-1|k-1}$ and $\mathbf{w}_k$. This yields the equation

$$\boldsymbol{\Sigma}_{k|k-1} = \mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}^T_{k-1} + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}^T_{k-1}, \tag{4.59}$$

where the Jacobian matrices are defined as

$$\mathbf{F}_{k-1} = \left.\frac{\partial \tilde{\mathbf{f}}_{k-1}}{\partial \boldsymbol{\xi}_{k-1|k-1}}\right|_{(\mathbf{0},\mathbf{u}_{k-1},\mathbf{0})}, \qquad \mathbf{G}_{k-1} = \left.\frac{\partial \tilde{\mathbf{f}}_{k-1}}{\partial \mathbf{w}_{k-1}}\right|_{(\mathbf{0},\mathbf{u}_{k-1},\mathbf{0})}. \tag{4.60}$$

Without any further assumption on the dynamics function $\mathbf{f}_{k-1}$, the prediction stage of the IEKF closely resembles that of the EKF, and similar convergence properties may therefore be expected. Part of the IEKF improved convergence properties stems from the additional assumption that the system dynamics are group-affine, as defined below.

▶ **Definition 4.10 (Group-affine dynamics).** The dynamics

$$\boldsymbol{\chi}_k = \mathbf{f}_{k-1}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) \tag{4.61}$$

are said to be group-affine if they can be written, up to first order in $\mathbf{w}_{k-1}$, as

$$\mathbf{f}_{k-1}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) = \bar{\mathbf{f}}_{k-1}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1})\mathbf{q}(\mathbf{w}_{k-1}, \mathbf{u}_{k-1}), \tag{4.62}$$

where $\mathbf{q}(\mathbf{0}, \mathbf{u}) = \mathbf{I}$ for all $\mathbf{u} \in \mathbb{R}^b$, and the mapping $\bar{\mathbf{f}}_{k-1} : G \times \mathbb{R}^b \to G$ satisfies at all time, for all $\boldsymbol{\nu}, \boldsymbol{\mu} \in G$ and all $\mathbf{u} \in \mathbb{R}^b$, the group-affine property

$$\bar{\mathbf{f}}_{k-1}(\boldsymbol{\nu}\boldsymbol{\mu}, \mathbf{u}) = \bar{\mathbf{f}}_{k-1}(\boldsymbol{\nu}, \mathbf{u})\bar{\mathbf{f}}_{k-1}(\mathbf{I}, \mathbf{u})^{-1}\bar{\mathbf{f}}_{k-1}(\boldsymbol{\mu}, \mathbf{u}). \tag{4.63}$$

◀

Group-affine dynamics ensure that the invariant nonlinear error evolves independently of the state estimate, a property that lies at the core of the IEKF.

▶ **Theorem 4.1 (Fundamental property of invariant filtering [39]).** In the absence of process noise, group-affine dynamics $\mathbf{f}_{k-1}$ yield a state-independent trajectory for the left- and right-invariant errors:

$$\boldsymbol{\eta}^L_{k|k-1} = \mathbf{g}_{k-1}(\boldsymbol{\eta}^L_{k-1|k-1}, \mathbf{u}_{k-1}), \quad \mathbf{g}_{k-1}(\boldsymbol{\eta}, \mathbf{u}) = \bar{\mathbf{f}}_{k-1}(\mathbf{I}, \mathbf{u})^{-1}\bar{\mathbf{f}}_{k-1}(\boldsymbol{\eta}, \mathbf{u}), \tag{4.64}$$

$$\boldsymbol{\eta}^R_{k|k-1} = \mathbf{g}_{k-1}(\boldsymbol{\eta}^R_{k-1|k-1}, \mathbf{u}_{k-1}), \quad \mathbf{g}_{k-1}(\boldsymbol{\eta}, \mathbf{u}) = \bar{\mathbf{f}}_{k-1}(\boldsymbol{\eta}, \mathbf{u})\bar{\mathbf{f}}_{k-1}(\mathbf{I}, \mathbf{u})^{-1}. \tag{4.65}$$

Moreover, in this case, for each $\mathbf{u}_{k-1} \in \mathbb{R}^b$, there exists $\mathbf{F}_{k-1} \in \mathbb{R}^{n\times n}$ such that

$$\mathbf{g}_{k-1}(\exp(\boldsymbol{\xi}), \mathbf{u}_{k-1}) = \exp(\mathbf{F}_{k-1}\boldsymbol{\xi}), \tag{4.66}$$

for all $\boldsymbol{\xi} \in \mathbb{R}^n$, meaning that $\mathbf{g}_{k-1}(\cdot, \mathbf{u}_{k-1})$ is wholly encoded in $\mathbf{F}_{k-1}$. ◀

This last result suggests that, in the absence of process noise, the linearized error evolves exactly linearly according to

$$\boldsymbol{\xi}_{k|k-1} = \mathbf{F}_{k-1}\boldsymbol{\xi}_{k-1|k-1}, \tag{4.67}$$

where the matrix $\mathbf{F}_{k-1}$ coincides with the Jacobian defined in Equation (4.60).

▶ **Theorem 4.2 (Log-Linear property [38]).** In the case of deterministic dynamics, consider the left- or right-invariant error $\boldsymbol{\eta}_{k|l}$ between two trajectories, which may be arbitrarily far apart. Suppose the initial error is given by $\boldsymbol{\eta}_{0|l} = \exp(\boldsymbol{\xi}_{0|l})$ for some $\boldsymbol{\xi}_{0|l} \in \mathbb{R}^n$. Then, for all $k \geq 0$,

$$\boldsymbol{\eta}_{k|l} = \exp(\boldsymbol{\xi}_{k|l}). \tag{4.68}$$

In other words, the nonlinear error $\boldsymbol{\eta}_{k|l}$ evolves entirely through its corresponding linear variable $\boldsymbol{\xi}_{k|l}$, which satisfies the linear time-varying Equation (4.67). ◀

The previous result is particularly significant, as it endows the IEKF with key properties of the linear Kalman filter in the absence of process noise, albeit in a nonlinear setting. Although deterministic dynamics are rare in practice, group-affine dynamics still substantially improve the convergence properties of the IEKF in the presence of process noise. In this case, Equation (4.67) becomes

$$\begin{aligned}\boldsymbol{\xi}_{k|k-1} = \mathbf{F}_{k-1}\boldsymbol{\xi}_{k-1|k-1} + \mathbf{G}_{k-1}\mathbf{w}_{k-1} \\ +\mathcal{O}(\|\boldsymbol{\xi}_{k-1|k-1}\|^2, \|\mathbf{w}_{k-1}\|^2, \|\boldsymbol{\xi}_{k-1|k-1}\|\|\mathbf{w}_{k-1}\|),\end{aligned} \tag{4.69}$$

where the Jacobian $\mathbf{G}_{k-1}$ typically depends on the state estimate $\hat{\boldsymbol{\chi}}_{k-1|k-1}$, whereas $\mathbf{F}_{k-1}$ does not. The independence of $\mathbf{F}_{k-1}$ from the state estimate significantly

mitigates the adverse effect that an inaccurate estimate can have on the linearized error dynamics. When $\mathbf{F}_{k-1}$ depends on the state estimate, a larger linearized error $\boldsymbol{\xi}_{k-1|k-1}$ leads to a more erroneous linearization point, and consequently to a less accurate Jacobian. The resulting term $\mathbf{F}_{k-1}\boldsymbol{\xi}_{k|k-1}$ then acts as a positive feedback mechanism that can destabilize the filter and hinder convergence. This situation occurs in the standard EKF, where $\mathbf{F}_{k-1}$ depends explicitly on the state estimate. By contrast, this feedback loop is eliminated when the Jacobian $\mathbf{F}_{k-1}$ is independent of the state estimate, as is the case in the IEKF for group-affine dynamics.

### 4.2.2 The update stage

Assume that we are at time index $k$ and that the measurement $\mathbf{y}_k$ has just been received. Rather than performing Bayesian inference directly on $G$ with a concentrated Gaussian prior, we map the problem to $\mathbb{R}^n$ and carry out inference on the linearized error. To this end, the measurement model is re-expressed as a function of the linearized error:

$$\mathbf{y}_k = \tilde{\mathbf{h}}_k(\boldsymbol{\xi}^L_{k|k-1}, \mathbf{n}_k) = \mathbf{h}_k\big(\hat{\boldsymbol{\chi}}_{k|k-1}\exp(\boldsymbol{\xi}^L_{k|k-1}), \mathbf{n}_k\big), \quad \text{(L-IEKF)} \tag{4.70}$$

$$\mathbf{y}_k = \tilde{\mathbf{h}}_k(\boldsymbol{\xi}^R_{k|k-1}, \mathbf{n}_k) = \mathbf{h}_k\big(\exp(\boldsymbol{\xi}^R_{k|k-1})\hat{\boldsymbol{\chi}}_{k|k-1}, \mathbf{n}_k\big). \quad \text{(R-IEKF)} \tag{4.71}$$

From this point onward, the update follows the same methodology as the EKF, with $\boldsymbol{\xi}_{k|k-1}$ playing the role of the state vector. The modified measurement model $\tilde{\mathbf{h}}_k$ is linearized around the operating point $(\mathbf{0}, \mathbf{0})$, yielding the following first-order approximation in $\boldsymbol{\xi}_{k|k-1}$ and $\mathbf{n}_k$:

$$\mathbf{y}_k \approx \mathbf{h}_k(\hat{\boldsymbol{\chi}}_{k|k-1}, \mathbf{0}) + \mathbf{H}_k\boldsymbol{\xi}_{k|k-1} + \mathbf{M}_k\mathbf{n}_k, \tag{4.72}$$

where the Jacobian matrices are defined as

$$\mathbf{H}_k = \left.\frac{\partial\tilde{\mathbf{h}}_k}{\partial\boldsymbol{\xi}_{k|k-1}}\right|_{(\mathbf{0},\mathbf{0})}, \qquad \mathbf{M}_k = \left.\frac{\partial\tilde{\mathbf{h}}_k}{\partial\mathbf{n}_k}\right|_{(\mathbf{0},\mathbf{0})}. \tag{4.73}$$

Using this approximation within the Bayesian inference process leads to the following Gaussian approximation of the posterior distribution of the linearized error:

$$\boldsymbol{\xi}_{k|k-1} \mid (\hat{\boldsymbol{\chi}}_{0|0}, \mathbf{u}_{0:k-1}, \mathbf{y}_{0:k}) \sim \mathcal{N}\big(\mathbf{K}_k\mathbf{z}_k, (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\boldsymbol{\Sigma}_{k|k-1}\big), \tag{4.74}$$

where the Kalman gain and the innovation are given by

$$\mathbf{K}_k = \boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T\big(\mathbf{H}_k\boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T + \mathbf{M}_k\mathbf{N}_k\mathbf{M}_k^T\big)^{-1}, \tag{4.75}$$

$$\mathbf{z}_k = \mathbf{y}_k - \mathbf{h}_k(\hat{\boldsymbol{\chi}}_{k|k-1}, \mathbf{0}). \tag{4.76}$$

To recover a proper concentrated Gaussian representation of the state, the linearized error is re-centered at $\mathbf{0}$ by defining the updated linearized error through

$$\boldsymbol{\xi}_{k|k-1} = \mathbf{K}_k\mathbf{z}_k + \boldsymbol{\xi}_{k|k}, \quad \boldsymbol{\xi}_{k|k} \sim \mathcal{N}\big(\mathbf{0}, (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\boldsymbol{\Sigma}_{k|k-1}\big). \tag{4.77}$$

The term $\mathbf{K}_k\mathbf{z}_k$ is then transferred to the state estimate by splitting the expression $\exp(\mathbf{K}_k\mathbf{z}_k + \boldsymbol{\xi}_{k|k})$ into two exponentials using a first-order truncation of the BCH formula, under the assumption that both $\mathbf{K}_k\mathbf{z}_k$ and $\boldsymbol{\xi}_{k|k}$ are small. This yields

$$\hat{\boldsymbol{\chi}}_{k|k-1}\exp(\boldsymbol{\xi}^L_{k|k-1}) \approx \hat{\boldsymbol{\chi}}_{k|k-1}\exp(\mathbf{K}_k\mathbf{z}_k)\exp(\boldsymbol{\xi}^L_{k|k}), \quad \text{(L-IEKF)} \tag{4.78}$$

$$\exp(\boldsymbol{\xi}^R_{k|k-1})\hat{\boldsymbol{\chi}}_{k|k-1} \approx \exp(\boldsymbol{\xi}^R_{k|k})\exp(\mathbf{K}_k\mathbf{z}_k)\hat{\boldsymbol{\chi}}_{k|k-1}. \quad \text{(R-IEKF)} \tag{4.79}$$

This leads to the IEKF update equations:

$$\hat{\boldsymbol{\chi}}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1}\exp(\mathbf{K}_k\mathbf{z}_k),\ \boldsymbol{\Sigma}^L_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\boldsymbol{\Sigma}^L_{k|k-1}, \quad \text{(L-IEKF)} \tag{4.80}$$

$$\hat{\boldsymbol{\chi}}_{k|k} = \exp(\mathbf{K}_k\mathbf{z}_k)\hat{\boldsymbol{\chi}}_{k|k-1},\ \boldsymbol{\Sigma}^R_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\boldsymbol{\Sigma}^R_{k|k-1}, \quad \text{(R-IEKF)} \tag{4.81}$$

Without any further assumption on the measurement model, the IEKF update closely resembles that of the EKF, and the advantage of using an IEKF does not appear clearly. However, similarly to the prediction stage, the update stage of the IEKF exhibits improved properties for a specific class of measurements, referred to as left- or right-invariant measurements.

▶ **Definition 4.11 (Invariant measurements).** Measurements are said to be left- or right-invariant if they take the form

$$\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k + \mathbf{n}_k, \quad \text{(L-IEKF)} \tag{4.82}$$

$$\mathbf{y}_k = \boldsymbol{\chi}_k^{-1}\mathbf{d}_k + \mathbf{n}_k, \quad \text{(R-IEKF)} \tag{4.83}$$

where $\mathbf{d}_k \in \mathbb{R}^N$. ◀

When left- or right-invariant measurements are used in combination with the corresponding invariant paradigm, one can define the alternative observation

$$\tilde{\mathbf{y}}_k = \hat{\boldsymbol{\chi}}^{-1}_{k|k-1}\mathbf{y}_k - \mathbf{d}_k, \quad \text{(L-IEKF)} \tag{4.84}$$

$$\tilde{\mathbf{y}}_k = \hat{\boldsymbol{\chi}}_{k|k-1}\mathbf{y}_k - \mathbf{d}_k, \quad \text{(R-IEKF)} \tag{4.85}$$

which expands into

$$\tilde{\mathbf{y}}_k = \tilde{\mathbf{h}}_k(\boldsymbol{\xi}^L_{k|k-1}, \mathbf{n}_k) = \exp(\boldsymbol{\xi}^L_{k|k-1})\mathbf{d}_k - \mathbf{d}_k + \hat{\boldsymbol{\chi}}^{-1}_{k|k-1}\mathbf{n}_k, \quad \text{(L-IEKF)} \tag{4.86}$$

$$\tilde{\mathbf{y}}_k = \tilde{\mathbf{h}}_k(\boldsymbol{\xi}^R_{k|k-1}, \mathbf{n}_k) = \exp(-\boldsymbol{\xi}^R_{k|k-1})\mathbf{d}_k - \mathbf{d}_k + \hat{\boldsymbol{\chi}}_{k|k-1}\mathbf{n}_k. \quad \text{(R-IEKF)} \tag{4.87}$$

Linearizing this alternative measurement model around $\boldsymbol{\xi}_{k|k-1} = \mathbf{0}$ yields a Jacobian $\mathbf{H}_k$ that depends solely on the known vector $\mathbf{d}_k$ and is completely independent of the state estimate $\hat{\boldsymbol{\chi}}_{k|k-1}$. This independence mitigates the adverse effect that an inaccurate state estimate can have on the linearization given in Equation (4.72), and consequently on the resulting update. Note that the the Jacobian $\mathbf{M}_k$, is still dependent on the state estimate. In practice, this dependence is typically of lesser importance, since the measurement noise often has low power, with $\mathbf{N}_k$ exhibiting small eigenvalues and thereby reducing the impact of the term $\mathbf{M}_k\mathbf{N}_k\mathbf{M}_k^T$ in the expression of the Kalman gain.

▶ **Remark 4.1.** The definition of the alternative observation $\tilde{\mathbf{y}}_k$ is primarily pedagogical. Performing the update using the original measurement $\mathbf{y}_k$, as described earlier, leads to a Jacobian $\mathbf{H}_k$ and an innovation $\mathbf{z}_k$ that depend on the state estimate in such a way that the products $\mathbf{K}_k\mathbf{z}_k$ and $\mathbf{K}_k\mathbf{H}_k$ are identical in both formulations. Consequently, both approaches yield exactly the same state estimate. ◀

## 4.3 Summary and discussion

The IEKF follows, in essence, the same methodology as the EKF, with the key difference that it intrinsically respects the matrix Lie group structure of the state space. This is made possible by a proper parametrization of the manifold using the exponential map. Since its introduction, the IEKF has become a state-of-the-art estimation method, primarily due to its improved convergence properties, which stem from two main structural assumptions: the group-affine nature of the system dynamics and the left- or right-invariant form of the measurements. The first condition affects the prediction stage by ensuring that the Jacobian $\mathbf{F}_k$ governing the linearized error dynamics is independent of the state estimate. The second condition impacts the update stage by eliminating the positive feedback mechanism that may otherwise destabilize the filter through state-dependent linearization, provided that the left- or right-invariant paradigm is chosen consistently with the form of the measurements. Note that this last property also resolves the false observability issue from which the EKF suffers, see Section 3.2.3.

Although the class of systems admitting group-affine dynamics and invariant measurements is broad, it is natural to ask how the IEKF behaves when one or both of these conditions are not satisfied. First, it is worth noting that, as made clear by the derivations presented in this chapter, the IEKF equations remain well defined even when one or both assumptions fail to hold. When only one of the two assumptions is satisfied, the IEKF still benefits from it, thereby improving either the prediction or the update stage compared to the EKF. When neither assumption holds, the IEKF no longer enjoys these specific convergence guarantees, but it retains the important advantage of intrinsically respecting the Lie group structure of the state space, a property that the EKF does not possess.

From an implementation standpoint, the IEKF algorithm remains very close to that of the EKF. Since the exponential map admits closed-form expressions in many practical cases, running an IEKF does not require significantly more computational resources than running an EKF.

For these reasons, the use of the invariant Kalman filtering framework is advocated whenever the system state can naturally be embedded in a matrix Lie group. The algorithms corresponding to the left- and right-invariant formulations of the IEKF are provided in Algorithm 4 and Algorithm 5, respectively. A numerical example illustrating one iteration of the left-invariant IEKF is presented in Example 4.5.

**Algorithm 4** The left-invariant extended Kalman filter

Choose the initial estimate $(\hat{\boldsymbol{\chi}}_{0|0}, \boldsymbol{\Sigma}_{0|0})$.
**loop**
  ▷ Prediction
  Compute Jacobians $\mathbf{F}_{k-1}$ and $\mathbf{G}_{k-1}$, define $\mathbf{f}_{k-1}$, $\mathbf{Q}_{k-1}$ and $\mathbf{u}_{k-1}$.
  $\hat{\boldsymbol{\chi}}_{k|k-1} \leftarrow \mathbf{f}_{k-1}(\hat{\boldsymbol{\chi}}_{k-1|k-1}, \mathbf{u}_k, \mathbf{0})$
  $\boldsymbol{\Sigma}_{k|k-1} \leftarrow \mathbf{F}_{k-1}\boldsymbol{\Sigma}_{k-1|k-1}\mathbf{F}_{k-1}^T + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^T$
  ▷ Update
  **if** measurement $\mathbf{y}_k$ available **then**
    Compute Jacobians $\mathbf{H}_k$ and $\mathbf{M}_k$, define $\mathbf{h}_k$ and $\mathbf{N}_k$.
    $\mathbf{z}_k \leftarrow \mathbf{y}_k - \mathbf{h}_k(\hat{\boldsymbol{\chi}}_{k|k-1}, \mathbf{0})$
    $\mathbf{K}_k \leftarrow \boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T(\mathbf{H}_k\boldsymbol{\Sigma}_{k|k-1}\mathbf{H}_k^T + \mathbf{N}_k)^{-1}$
    $\hat{\boldsymbol{\chi}}_{k|k} \leftarrow \hat{\boldsymbol{\chi}}_{k|k-1}\exp(\mathbf{K}_k\mathbf{z}_k)$
    $\boldsymbol{\Sigma}_{k|k} \leftarrow (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\boldsymbol{\Sigma}_{k|k-1}$
  **end if**
**end loop**

**Algorithm 5** The right-invariant extended Kalman filter

Same as Algorithm 4 with the following exception.

10: $\hat{\boldsymbol{\chi}}_{k|k} \leftarrow \exp(\mathbf{K}_k \mathbf{z}_k)\hat{\boldsymbol{\chi}}_{k|k-1}$

▶ **Example 4.5.** We return to Example 3.2 and embed the state in the matrix Lie group $SE_2(2)$, represented as

$$\boldsymbol{\chi}_k = \begin{bmatrix} \mathbf{R}_k & \mathbf{v}_k & \mathbf{p}_k \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}, \tag{4.88}$$

with $\mathbf{R}_k \in SO(2)$ encoding the worker's heading. The system input remains unchanged. Neglecting the IMU biases, the state evolves according to

$$\boldsymbol{\chi}_k = \bar{\mathbf{f}}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1})\mathbf{q}(\mathbf{w}_{k-1}, \mathbf{u}_{k-1}),$$

where $\mathbf{w}_{k-1} = (w^{\omega}_{k-1}, \mathbf{w}^{\mathbf{a}}_{k-1}) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_{k-1})$. The mappings $\bar{\mathbf{f}}$ and $\mathbf{q}$ are given by

$$\bar{\mathbf{f}}(\boldsymbol{\chi}_{k-1}, \mathbf{u}_{k-1}) = \begin{bmatrix} \mathbf{R}_{k-1}\exp(\omega_{k-1}dt) & \mathbf{v}_{k-1} + \mathbf{R}_{k-1}\mathbf{a}_{k-1}dt & \mathbf{p}_{k-1} + \mathbf{v}_{k-1}dt + \mathbf{R}_{k-1}\mathbf{a}_{k-1}\frac{dt^2}{2} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix},$$

$$\mathbf{q}(\mathbf{w}_{k-1}, \mathbf{u}_{k-1}) = \begin{bmatrix} \exp(w^{\omega}_{k-1}dt) & \exp(-\omega_{k-1}dt)\mathbf{w}^{\mathbf{a}}_{k-1}dt & \exp(-\omega_{k-1}dt)\mathbf{w}^{\mathbf{a}}_{k-1}\frac{dt^2}{2} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix},$$

with $\exp = \exp_{SO(2)} \circ \mathcal{L}_{\mathfrak{so}(2)}$. It is straightforward to verify that $\bar{\mathbf{f}}$ satisfies the group-affine property described in Equation (4.63).

Using this matrix Lie group representation, the measurement provided by the considered range-and-bearing sensor admits the following right-invariant form:

$$\mathbf{y}_k = \begin{bmatrix} \boldsymbol{\Pi}\boldsymbol{\chi}_k^{-1}\mathbf{d}_1 \\ \boldsymbol{\Pi}\boldsymbol{\chi}_k^{-1}\mathbf{d}_2 \\ \boldsymbol{\Pi}\boldsymbol{\chi}_k^{-1}\mathbf{d}_3 \end{bmatrix} + \mathbf{n}_k,$$

where $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ and where the vectors $\mathbf{d}_1, \mathbf{d}_2, \mathbf{d}_3 \in \mathbb{R}^4$ are given by

$$\mathbf{d}_1 = (\mathbf{b}_1, 0, 1), \quad \mathbf{d}_2 = (\mathbf{b}_2, 0, 1), \quad \mathbf{d}_3 = (\mathbf{b}_3, 0, 1),$$

with $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3 \in \mathbb{R}^2$ the fixed locations of the beacons in the inertial frame. The same values as in Example 3.2 are used. The last two elements of each vector $\boldsymbol{\chi}_k^{-1}\mathbf{d}_i$ do not convey any information about the state. To discard them, we left-multiply each vector by $\boldsymbol{\Pi} = \begin{bmatrix}\mathbf{I} & \mathbf{0}\end{bmatrix}$, which extracts the relevant two-dimensional components. The presence of $\boldsymbol{\Pi}$ in the measurement model does not affect the properties of the invariant filter.

We apply a R-IEKF to estimate $\boldsymbol{\chi}_k$. The propagation Jacobians in Equation (4.60) are

$$\mathbf{F}_{k-1} = \begin{bmatrix} 1 & \mathbf{0} & \mathbf{0} \\ 0 & \mathbf{I} & \mathbf{0} \\ 0 & \mathbf{I}dt & \mathbf{I} \end{bmatrix}, \qquad \mathbf{G}_{k-1} = \begin{bmatrix} dt & \mathbf{0} \\ -(1)_\times \hat{\mathbf{v}}_k dt & \hat{\mathbf{R}}_{k-1} dt \\ -(1)_\times \hat{\mathbf{p}}_k dt & \hat{\mathbf{R}}_{k-1} \frac{dt^2}{2} \end{bmatrix}.$$

Note that $\mathbf{G}_{k-1}$ depends on the velocity and position estimates at time $k$. This is not an issue, since $\mathbf{G}_{k-1}$ is not needed to compute these estimates. The measurement Jacobians in Equation (4.73) reduce to

$$\mathbf{H}_k = \begin{bmatrix} -\hat{\mathbf{R}}_k^T (1)_\times \mathbf{b}_1 & \mathbf{0} & -\hat{\mathbf{R}}_k^T \\ -\hat{\mathbf{R}}_k^T (1)_\times \mathbf{b}_2 & \mathbf{0} & -\hat{\mathbf{R}}_k^T \\ -\hat{\mathbf{R}}_k^T (1)_\times \mathbf{b}_3 & \mathbf{0} & -\hat{\mathbf{R}}_k^T \end{bmatrix}, \qquad \mathbf{M}_k = \mathbf{I}.$$

We use the same ground truth as in Example 3.2, as well as the same noise parameters and time step. The filter is initialized with the same initial state:

$$\hat{\boldsymbol{\chi}}_{0|0} = \begin{bmatrix} \cos(4.5406) & -\sin(4.5406) & -2.2593 & 5.0449 \\ \sin(4.5406) & \cos(4.5406) & 0.0718 & 3.9764 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{\Sigma}_{0|0} = \begin{bmatrix} 0.2742 & 0.0197 & 0.6194 & 1.0902 & -1.3831 \\ 0.0197 & 1.0014 & 0.0445 & 0.0783 & -0.0994 \\ 0.6194 & 0.0445 & 2.3995 & 2.4631 & -3.1249 \\ 1.0902 & 0.0783 & 2.4631 & 5.3350 & -5.4997 \\ -1.3831 & -0.0994 & -3.1249 & -5.4997 & 7.9775 \end{bmatrix}.$$

The covariance matrix is not initialized in the same way as in the EKF case. This is because the IEKF uses a concentrated Gaussian to model uncertainty, rather than

a standard Gaussian distribution. To ensure a fair comparison, we match these distributions to first order. In particular, up to first order in the EKF error $\mathbf{e}_{0|0}$, we have

$$\boldsymbol{\xi}_{0|0} \approx \underbrace{\begin{bmatrix} 1 & \mathbf{0} & \mathbf{0} \\ -(1)_\times \hat{\mathbf{v}}_{0|0} & \mathbf{I} & \mathbf{0} \\ -(1)_\times \hat{\mathbf{p}}_{0|0} & \mathbf{0} & \mathbf{I} \end{bmatrix}}_{\mathbf{A}} \mathbf{e}_{0|0}.$$

We therefore initialize the covariance matrix as $\boldsymbol{\Sigma}_{0|0}^{\text{IEKF}} = \mathbf{A}\boldsymbol{\Sigma}_{0|0}^{\text{EKF}}\mathbf{A}^T$.We now consider the first full iteration of the algorithm.

**Prediction:** The R-IEKF prediction step yields

$$\hat{\boldsymbol{\chi}}_{1|0} = \begin{bmatrix} \cos(4.5379) & -\sin(4.5379) & -2.2669 & 4.9317 \\ \sin(4.5379) & \cos(4.5379) & 0.0711 & 3.9800 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{\Sigma}_{1|0} = \begin{bmatrix} 0.2742 & 0.0197 & 0.6195 & 1.0912 & -1.3522 \\ 0.0197 & 1.0014 & 0.0445 & 0.1284 & -0.0971 \\ 0.6195 & 0.0445 & 2.3996 & 2.4655 & -3.0052 \\ 1.0912 & 0.1284 & 2.4655 & 5.3457 & -5.3819 \\ -1.3522 & -0.0971 & -3.0052 & -5.3819 & 7.6716 \end{bmatrix}.$$

**Update:** The range-and-bearing sensor provides the measurement

$$\mathbf{y}_1 = (7.1900, -0.0301, -0.0233, 6.9132, -0.1647, -7.2182)$$

Incorporating this measurement yields

$$\hat{\boldsymbol{\chi}}_{1|1} = \begin{bmatrix} \cos(3.9799) & -\sin(3.9799) & -2.2930 & 4.3936 \\ \sin(3.9799) & \cos(3.9799) & 0.1227 & 4.9333 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$\boldsymbol{\Sigma}_{1|1} = \begin{bmatrix} 0.0001 & 0.0000 & 0.0002 & 0.0003 & -0.0003 \\ 0.0000 & 0.9975 & 0.0000 & 0.0002 & 0.0000 \\ 0.0002 & 0.0000 & 0.9979 & 0.0006 & -0.0004 \\ 0.0003 & 0.0002 & 0.0006 & 0.0042 & -0.0008 \\ -0.0003 & 0.0000 & -0.0004 & -0.0008 & 0.0042 \end{bmatrix}.$$

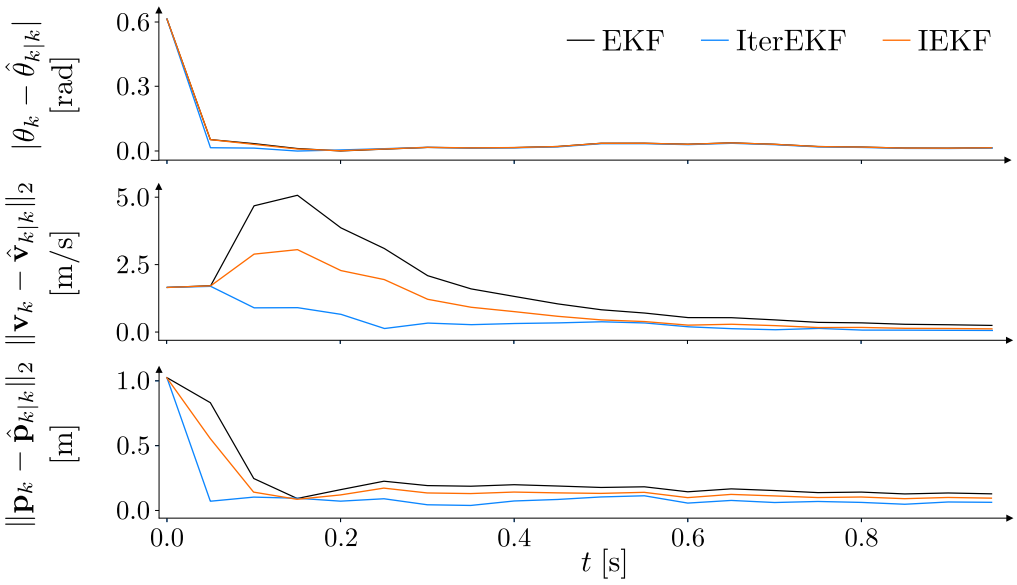



**Figure 4.2:** Evolution of the norm of the IEKF estimation error for the worker's heading angle, velocity, and position over the first 20 time steps. The results obtained with the EKF and IterEKF are shown for comparison.

The simulation is run over the first 20 time steps. To compare the R-IEKF with the EKF and the IterEKF, we compute the estimation error as in the EKF case, but using the R-IEKF estimates. Figure 4.2 shows the evolution of the norm of this error for the worker's heading angle, velocity, and position. The IEKF improves over the EKF in this setting, especially for the velocity estimate. This improvement is mainly due to the group-affine dynamics and to the invariant form of the measurement. The IterEKF performs better than the IEKF, as expected. The R-IEKF linearizes the measurement model around $\boldsymbol{\xi}_{k|k-1} = \mathbf{0}$, which can be far from the true value and lead to a coarse first-order approximation. By refining the linearization point, the IterEKF improves the linearization accuracy and yields a better update step. ◀

# Part II

# Building toward invariant Kalman filtering for pose estimation of rigid-body systems

The previous part equipped the reader with the theoretical foundations and intuition necessary to understand the principles of Kalman filtering and its invariant formulation. Through a series of illustrative examples, it presented several approaches to estimating the pose of a human co-worker in the context of collision avoidance, ensuring safety within a shared environment.

So far, the human co-worker has been modeled as a single rigid body, an assumption that considerably simplifies the estimation problem. While sufficient for basic collision avoidance, this abstraction quickly becomes limiting in more advanced applications. Consider, for instance, a rehabilitation robot that must perceive and react to a patient's body configuration. In such a setting, estimating the pose of the patient as a whole is not sufficient; the robot must instead infer the configuration of the patient's individual body segments in order to assess their mobility. A similar requirement arises in the context of exoskeletons, which must closely track the motion of their operator's body in order to provide effective assistance. Another illustrative example is programming by demonstration, where a robot learns a complex task by observing a human operator: for motions such as grasping, the robot must capture the configuration of the demonstrator's fingers in order to reproduce a similar behavior with its own morphology and actuation.

In all these situations, the estimation problem no longer concerns a single rigid body, but rather a rigid-body system, that is, a collection of rigid bodies interconnected through kinematic constraints. The presence of such interconnections introduces additional challenges in the filtering problem, as the estimated state must remain consistent with the constraints imposed by the system's structure.

Several approaches based on the EKF have been proposed to address this class of problems. However, to the best of our knowledge, the invariant Kalman filtering framework has not yet been systematically applied to the estimation of rigid-body systems. Given its favorable convergence properties compared to the EKF, extending the invariant framework to this setting is both a natural and promising direction.

This part of the thesis is devoted to this objective and presents the core contributions of the manuscript. It develops a progressive extension of the invariant Kalman filtering framework to the problem of pose estimation for rigid-body systems. The development is structured around three main building blocks, each addressed in a dedicated chapter. The first two chapters introduce essential components required to adapt the invariant filtering paradigm to rigid-body systems, thereby laying the theoretical and methodological groundwork for the final contribution. Building upon these elements, the third chapter presents the actual extension of the invariant Kalman filtering framework to rigid-body pose estimation, bringing the various concepts together into a unified formulation.

# 5 Enforcing state equality constraints in the invariant filtering framework

## Outline

This chapter develops a principled method for enforcing state equality constraints within the invariant Kalman filtering framework, which consists in considering them as noise-free pseudo-measurements. After examining the implications of such pseudo-measurements on the general Kalman filtering methodology and identifying the challenges associated with their proper handling, it is shown that invariant filtering provides a natural and effective way to address these issues. The proposed approach is illustrated through a pose estimation example involving a single rigid body subject to an equality constraint.

## 5.1 Prologue

Extending the invariant framework to rigid-body systems poses significant challenges and naturally lends itself to a staged treatment. In the divide-and-conquer strategy adopted in this thesis, the present chapter addresses the first of these stages and provides conceptual tools that are required for subsequent developments.

Rigid-body systems are characterized by kinematic constraints arising from mechanical interconnections, which couple the motion of multiple bodies. In an estimation context, such constraints can often be expressed as state equality constraints. This chapter isolates the problem of enforcing equality constraints within a class of problems for which the invariant filtering framework is well established: pose estimation of a single rigid body. The additional layer of complexity introduced by articulated systems is deferred to Chapter 7.

## 5.2 Invariant Kalman filtering with noise-free pseudo-measurements

This work was carried out in preparation for the $62^{\text{nd}}$ IEEE Conference on Decision and Control [98], held in Singapore in December 2023, and was presented as a 15-minute oral presentation.

### 5.2.1 Author contributions

As the first author, I led the research and was responsible for the entire simulation section. The manuscript was primarily written by Prof. Silvère Bonnabel and myself. Prof. Bonnabel also provided substantial support in the mathematical derivations. Prof. Pierre Sacré and Olivier Brüls offered close supervision on the research directions and contributed to the revision and refinement of the manuscript.

### 5.2.2 Reading tips

The reader is encouraged to pay close attention to Subsections C, D, and E of the second section of the paper, which present the challenges associated with handling noise-free pseudo-measurements in the Kalman filtering context. Understanding these challenges is essential for grasping the motivation of the following chapter. In particular, Properties 1 and 2 play a central role in this regard.

The paper adopts notation conventions that differ from those used throughout this thesis. First, matrix and vector quantities are not typeset in boldface. Second, the error covariance matrix is denoted by $P_{k|l}$ instead of $\mathbf{\Sigma}_{k|l}$. Moreover, we slightly abuse terminology by referring to the compositions $\exp_G \circ \mathcal{L}_{\mathfrak{g}}$ and $\mathcal{L}_{\mathfrak{g}}^{-1} \circ \log_G$ as the exponential and logarithmic maps of the matrix Lie group $G$, rather than to the maps $\exp_G$ and $\log_G$ themselves, see Section 4.1.2.

# Invariant Kalman Filtering with Noise-Free Pseudo-Measurements

Sven Goffin, Silvère Bonnabel, Olivier Brüls, and Pierre Sacré

*Abstract*— **In this paper, we focus on developing an Invariant Extended Kalman Filter (IEKF) for extended pose estimation for a noisy system with state equality constraints. We treat those constraints as noise-free pseudo-measurements. To this aim, we provide a formula for the Kalman gain in the limit of noise-free measurements and rank-deficient covariance matrix. We relate the constraints to group-theoretic properties and study the behavior of the IEKF in the presence of such noise-free measurements. We illustrate this perspective on the estimation of the motion of the load of an overhead crane, when a wireless inertial measurement unit is mounted on the hook.**

## I. Introduction

Since its introduction in the 1960s, the Extended Kalman Filter (EKF) is probably the most used filter in the industry [1]. Despite its use in many real world applications, it relies on strong assumptions that are rarely met in practice. Moreover, its development is solely based on a probabilistic description of a dynamical and a measurement model, disregarding the geometrical structure of the considered problem. The latter concern gave birth to a proper field that focuses on geometric filtering methods. This research direction provided improved solutions for many useful tasks, like Simultaneous Localization And Mapping (SLAM) [2]–[5], inertial navigation and localization [6]–[8], or attitude and pose estimation [9]–[11]. The invariance and equivariance properties of frameworks developed in [6], [8], [12], [13] are reminiscent of those of linear systems in many regards.

In various contexts, some additional (deterministic) information further constrain the state to belong to a subspace of the state space. Incorporating equality constraints in probabilistic filtering constitutes a significant challenge that has led to the realm of state constrained extended Kalman filtering, see [14]. In this paper, we consider nonlinear constraints, dictated by kinematic of mechanical relations, which arise for instance when estimating the configuration of a multi-body mechanical system whose parts are rigidly linked and equipped with one or several Inertial Measurement Units (IMUs). Especially, we consider the equations associated to an IMU that may move freely in space and assume it is mounted at the end of a rigid "link" whose other end's location is known, or fixed. This may be the case when mounting an IMU on a robotic arm, or on the hook of a crane (a pendulum), as in [15]. In those cases, the known location of the other end of the link provides an (nonlinear) equality constraint that may help estimating the state of the IMU.

One may then resort to state constrained Kalman filtering, see [14]. In the nonlinear case, such methods based on (re)projections are a means of enforcing the constraint. However, they come with no guarantee of consistency whatsoever with the estimation problem at hand, and a brutal (re)projection onto the constraint contains a degree of arbitrariness. Herein, we advocate that such nonlinear constraints may rather be incorporated in the extended Kalman filter and its variants through pseudo-measurements without noise, as they constitute information known with certainty. This technique is perfectly justified in the linear case, and allows for optimal fusion of noisy sensors and deterministic side information. This poses two challenges, though. First, the matrix that must be inverted in the Kalman filter may become rank-deficient, see *e.g.*, [16]. Second, in the nonlinear case, there is no guarantee the constraint is respected after the pseudo-measurement has been incorporated, contrary to the linear case.

We derive a formulation of the Kalman gain that accommodates the rank deficiency issues that may stem from the noise-free setting. Furthermore, we cast the problem into the framework of the Invariant Extended Kalman Filter (IEKF) by embedding the state space into a Lie group. Although [17] provides us with both an inspiration and a mathematical framework, we consider different problems: [17] considers the limit case of a noise-free (deterministic) dynamics with noisy measurements, along with an initial condition lying in a constrained set, so that deterministic information is propagated. Herein, we consider arbitrary noisy dynamics (and possibly noisy measurements) starting from an unrestricted initial configuration, and investigate how to incorporate noise-free measurements corresponding to deterministic side information. This comes with different challenges and allows for addressing different problems. Recently, [18] proposed to linearly constrain the Riccati equation for robust invariant filtering, which is a wholly different problem.

The paper is structured as follows. Section II specifies the considered class of estimation problems, provides two motivating examples of engineering interest, and highlights the properties one shall pursue and the challenges that are raised in both the linear and nonlinear cases. Section III derives a formulation for the Kalman gain in the limit of noise-free measurements in the linear case and shows its corresponding properties. Section IV extends the use of

Sven Goffin is a FRIA grantee of the Fonds de la Recherche Scientifique - FNRS.

S. Goffin and P. Sacré are with the Department of Electrical Engineering and Computer Science, University of Liège, Belgium (sven.goffin@uliege.be, p.sacre@uliege.be).

O. Brüls is with the Department of Aerospace and Mechanical Engineering, University of Liège, Belgium (o.bruls@uliege.be).

S. Bonnabel is with MINES ParisTech, PSL Research University, France (silvere.bonnabel@mines-paristech.fr).

this gain formulation in the framework of invariant Kalman filtering, and proposes an alternative update for the IEKF in the presence of noise-free measurements. Finally, Section V illustrates the performance of the proposed approach for the estimation of the extended pose of the hook of a crane.

## II. Motivating problem

Since the advent of smartphones, the use of cheap IMUs has increased significantly. In this paper, we consider the problem of fusing the IMU measurements with deterministic constraints, treated as noise-free pseudo-measurements.

### A. Mathematical formulation

Free from any prior knowledge about the motion, the general dynamics of an IMU write (*e.g.*, [6])

$$R_{k+1} = R_k \exp((\omega_k + w_k^\omega)_\times \, dt), \tag{1a}$$
$$v_{k+1} = v_k + (R_k(a_k + w_k^a) + g) \, dt, \tag{1b}$$
$$p_{k+1} = p_k + v_k \, dt, \tag{1c}$$

where we neglected sensors' biases. The matrix $R_k \in SO(3)$ is the rotation matrix between the IMU and world frames, and the vectors $v_k$, $p_k \in \mathbb{R}^3$ are respectively the IMU velocity and position vectors expressed in the world frame at time instant $k$. The angular velocity $\omega_k$ and linear acceleration $a_k$ output by the IMU are used as inputs to the dynamical system. The vector $g$ denotes the gravity vector expressed in the world frame. The map $\exp(\cdot)$ denotes the matrix exponential map and the notation $(b)_\times$ denotes the skew-symmetric matrix associated with cross product with vector $b \in \mathbb{R}^3$.

Even if no assumption is made on the dynamics, the structural and mechanical properties of the body to which the IMU is attached may constrain the IMU pose. When dealing with a rigid body, this comes as noise-free equality constraints of the form

$$R_k r_k + \alpha_k v_k + \beta_k p_k = y_k, \tag{2}$$

where $r_k$, $y_k \in \mathbb{R}^3$ and $\alpha_k$, $\beta_k \in \mathbb{R}$.

**Considered problem** We aim at devising a meaningful (Kalman) filter to estimate the state of the noisy system (1) from noise-free measurements (2) in real time.

### B. Application examples

Estimating the pose of an IMU under constraints like (2) applies to a wide range of applications. We provide two concrete applications for this problem, which is nonlinear, owing to the state variable $R_k$ being a rotation matrix.

*1) Crane state estimation:* As a first example, consider a crane on a construction site (see Figure 1). It is technically very feasible nowadays to mount an IMU on the hook that transmits its sensor readings [15]. Estimating the position and velocity of the load from the IMU measurements may open the door to new automation capabilities since it allows for feedback. However, the underlying dynamical model is unknown, as the hook does not necessarily follow a simple pendulum motion, because of external forces like friction and wind. The IMU dynamics (1), though, consist of kinematic relations which are satisfied at all times, whatever the motion. Moreover, when the cable is hanging, the distance between the hook and the cable hang-up point is the cable length $l_k$, which is very accurately measured in modern cranes for safety reasons, see, *e.g.*, [19]. This provides a constraint of the form of (2) with $r_k = (0, 0, -l_k)$, $\alpha_k = 0$, $\beta_k = 1$, and $y_k = 0_{3\times1}$.

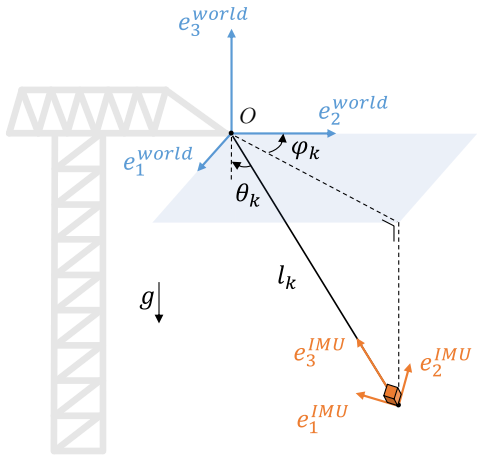



Fig. 1. Crane with a wireless IMU mounted on its hook.

*2) Robotic arm state estimation:* The field of robotics also uses IMUs extensively. As a second example, robotic arms are usually equipped with IMUs in order to precisely estimate the pose of their end effector. Although the dynamics of the arm cannot be predicted in advance, kinematic constraints enforce that the segments of the arm are attached to each other at the corresponding joints, providing constraints of the form (2). This is what is exploited in [7] in the form of noisy measurements.

### C. Kalman filtering with noise-free pseudo-measurements

Our problem of estimating noisy systems while treating noise-free equality constraints like (2) fits into the framework of probabilistic filtering. In this framework, the belief at time index $k$, taking into account measurements up to time $j$, consists of a multivariate Normal distribution parameterized by a mean $\hat{x}_{k|j} \in \mathbb{R}^n$ (the state estimate) and a covariance matrix $P_{k|j} \in \mathbb{R}^{n\times n}$ (the associated uncertainty). We advocate indeed that the constraints can be incorporated in a probabilistic filter by using $y_k$ as a noise-free pseudo-measurement $y_k = h(x_k)$, letting

$$h(x_k) = h(R_k, v_k, p_k) = R_k r_k + \alpha_k v_k + \beta_k p_k. \tag{3}$$

From the recent theory of two-frames [8], it can readily be checked (3) defines an output that fits into the framework of invariant filtering, and so do the IMU equations (1).

Noise-free pseudo-measurements convey information that is known for certain. Hence there are two desirable properties one would expect from a (Normal) probabilistic filter.

- *Property 1*: After a noise-free pseudo-measurement $y_k$, the state is known to belong to the subset

$$\{x \in \mathbb{R}^n \mid h(x) = y_k\}$$

owing to noise-free measurement. This should translate into $h(\hat{x}_{k|k}) = y_k$.

- *Property 2*: After a noise-free pseudo-measurement $y_k$, the error covariance matrix $P_{k|k}$ should encode some deterministic information: null variance is to be expected in the perfectly observed directions. Mathematically, it should translate into $H_k P_{k|k} H_k^T = 0_{m\times m}$ with $H_k \in \mathbb{R}^{m\times n}$ the Jacobian matrix of $h$ at the estimate $\hat{x}_{k|k}$.

### D. The linear case

In the linear case, the Kalman filter—whose equations are recalled in Appendix I—is optimal. Thus, we can expect both properties above to be satisfied. We have

$$H_k \hat{x}_{k|k} = H_k K_k y_k + (H_k - H_k K_k H_k)\hat{x}_{k|k-1} \tag{4}$$

and

$$H_k P_{k|k} H_k^T = (H_k - H_k K_k H_k) P_{k|k-1} H_k^T. \tag{5}$$

Both desirable properties above are satisfied indeed, as soon as $H_k K_k = I_m$, since it ensures that $H_k - H_k K_k H_k = 0_{m\times n}$. Recalling that $K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T)^{-1}$ whenever measurements are noise-free, the desired relation holds if $H_k P_{k|k-1} H_k^T$ is invertible. If it is not, the Kalman gain is not even defined. This may occur when a noise-free measurement is made for the second time: the rank of $H_k P_{k|k-1} H_k^T$ has dropped and the matrix cannot be inverted anymore. This may also occur when performing a subsequent noise-free measurement along directions that overlap with the span of $H_k$ so that the matrix to be inverted is not full-rank.

### E. The nonlinear case

In the nonlinear case, that is, when using an Extended Kalman Filter (EKF), problems accumulate. Indeed, even if $H_k P_{k|k-1} H_k^T$ is invertible leading to $H_k K_k = I_m$, there is no reason one should have $h(\hat{x}_{k|k}) = y_k$, as we see that (4) relies on $h$ being linear, which is not the case in our problem, see (3), as $R_k$ is a rotation matrix that cannot be treated linearly. Moreover, the Jacobian matrix $H_k$ depends on the linearization point in EKF design, so that the desired relation $H_k P_{k|k} H_k^T = 0_{m\times m}$ is not properly defined: the Jacobian matrices at different linearization points span different directions, making the condition unclear, in the sense that the dispersion encoded by $P_{k|k}$ has no clear relation to the subset defined by $h(x) = y_k$.

In the sequel, we address the issues that arise both in the linear and nonlinear case.

## III. Handling rank deficiency in the linear case

Let us consider the discrete-time linear system

$$x_{k+1} = F_k x_k + B_k u_k + w_k, \tag{6a}$$

$$y_k = H_k x_k. \tag{6b}$$

where $u_k \in \mathbb{R}^b$ is the system input, and where $F_k \in \mathbb{R}^{n\times n}$, $B_k \in \mathbb{R}^{n\times b}$, and $H_k \in \mathbb{R}^{m\times n}$. When the measurement $y_k$ is noise-free, as in (6b), the Kalman gain (35c) is not defined if $H_k P_{k|k-1} H_k^T$ is singular, since the measurement noise covariance $N_k$ is null. This is logical as then the same measurements does not bring additional information. However, in the noisy setting we may need to update the state regularly to combat the dispersion due to noise. Then $H_k P_{k|k-1} H_k^T$ may be nearly singular, and we may run into numerical issues. We now address this problem.

### A. Solving for the Kalman gain in the noise-free limit

The following result proves the limit of the Kalman gain when $N_k \to 0$ is well defined and provides its expression.

*Theorem 1:* Assume $P_{k|k-1}$ to be of rank $l \leq n$ and write it as $P_{k|k-1} = L_k L_k^T$ where $L_k \in \mathbb{R}^{n\times l}$ has linearly independent columns. The following Kalman gain

$$K_k := L_k (H_k L_k)^\dagger, \tag{7}$$

is the limit of $P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + N_k)^{-1}$ as the measurement noise covariance matrix $N_k$ shrinks to $0_{m\times m}$, where $A^\dagger$ denotes the Moore-Penrose pseudo-inverse of $A$.

Note that the given expression is valid whatever the invertibility of the innovation covariance $H_k P_{k|k-1} H_k^T$ as $(H_k L_k)^\dagger$ always exists.

*Proof:* Given any matrix $A$, its Moore-Penrose pseudo-inverse can be expressed as the limit

$$A^\dagger = \lim_{\delta\to 0} A^T (AA^T + \delta\, I)^{-1}. \tag{8}$$

The limit is finite and exists even if $AA^T$ is singular [20]. We pose $A_k = H_k L_k$ and assume for now that $N_k = \delta\, I_m$.

$$\lim_{\delta\to 0} K_k = \lim_{\delta\to 0} P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + \delta\, I_m)^{-1}, \tag{9}$$

$$= \lim_{\delta\to 0} L_k A_k^T (A_k A_k^T + \delta\, I_m)^{-1}, \tag{10}$$

$$= L_k \lim_{\delta\to 0} A_k^T \left(A_k A_k^T + \delta\, I_m\right)^{-1}, \tag{11}$$

$$\overset{(8)}{=} L_k A_k^\dagger, \tag{12}$$

$$= L_k (H_k L_k)^\dagger. \tag{13}$$

When $N_k$ is not of the form $\delta\, I_m$, we may upper and lower bound its eigenvalues by matrices of the desired form, and use the squeeze theorem (in the sense of positive semidefinite matrix inequalities), using the fact matrices $A_k A_k^T$ and $N_k$ are symmetric. ∎

### B. Properties of the Riccati update (Property 2)

When a noise-free measurement (6b) has been made, it is logical that the corresponding variance drops to zero, that is, $H_k P_{k|k} H_k^T = 0_{m\times m}$, along the lines of Property 2. We now show this is the case with the obtained limit gain.

*Theorem 2:* Assume $P_{k|k-1}$ to be of rank $l \leq n$ and write it as $P_{k|k-1} = L_k L_k^T$ where $L_k \in \mathbb{R}^{n\times l}$ has linearly independent columns. Consider the limit gain (7) and the corresponding Riccati update (35e). Then we have $H_k P_{k|k} H_k^T = 0$ or equivalently $H_k P_{k|k} = 0$.

*Proof:* We have

$$H_k(I - K_k H_k)P_{k|k-1} \tag{14}$$
$$= H_k(I - L_k(H_k L_k)^\dagger H_k)P_{k|k-1} \tag{15}$$
$$= (H_k L_k - H_k L_k (H_k L_k)^\dagger H_k L_k) L_k^T \tag{16}$$
$$= 0_{m\times n}, \tag{17}$$

where we used the property of the pseudo-inverse that that $AA^\dagger A = A$. This proves the result. ∎

## IV. Invariant Kalman filtering with noise-free pseudo-measurements

Consider the discrete-time nonlinear system (1) along with nonlinear output map (3). The EKF first linearizes the system before applying the machinery of linear Kalman filtering. Gain formulation (7) may thus be applied in the nonlinear case too. Nevertheless, the dependence of Jacobian matrix $H_k = \frac{\partial h(x)}{\partial x}$ on the linearization point makes the interpretation obscure: the update ensures Property 2 at the linearization point, which is not the true state. The invariant filtering framework brings clarification in this regard.

### A. The Invariant EKF for the considered problem

Let us modify the representation to cast the problem within the invariant framework of [3]. We embed the state into the matrix Lie group $SE_2(3)$, namely the group of extended poses [6], [21]. The state turns into

$$\chi_k = \begin{bmatrix} R_k & v_k & p_k \\ 0_{1\times 3} & 1 & 0 \\ 0_{1\times 3} & 0 & 1 \end{bmatrix}. \tag{18}$$

Its dynamics satisfies Equation (11) of [3] and is said to be group affine, insuring it fits the framework of the Invariant Extended Kalman Filter (IEKF). In this embedding, the measurement function may write as

$$h(\chi_k) = \chi_k d_k = \begin{bmatrix} R_k r_k + \alpha_k v_k + \beta_k p_k \\ \alpha_k \\ \beta_k \end{bmatrix}, \tag{19}$$

where $d_k = (r_k, \alpha_k, \beta_k)$, which fits into the theory of [6] and corresponds to left-invariant outputs (observations in the fixed frame). Along the lines of [6] we define the innovation as $z_k = \hat{\chi}_{k|k-1}^{-1} y_k - d_k$. The invariant framework introduces the nonlinear and linearized errors $\eta_{k|j} \in SE_2(3)$ and $\xi_{k|j} \in \mathbb{R}^9$ defined by

$$\eta_{k|j} = \hat{\chi}_{k|j}^{-1}\chi_k = \exp(\xi_{k|j}), \tag{20}$$

where $\exp(\cdot)$ now denotes the Lie exponential map of group $SE_2(3)$. To derive the Jacobian matrices, the fastest way is to readily retrieve them from the formulas of the recent two-frames theory [8]. In this context the left-invariant error is $\left(\hat{R}_{k|j}^{-1}R_k, \hat{R}_{k|j}^{-1}(v_k - \hat{v}_{k|j}), \hat{R}_{k|j}^{-1}(p_k - \hat{p})\right)$, the innovation is $z_k = (\hat{R}_{k|k-1}^{-1}R_k - I)r_k + \alpha_k \hat{R}_{k|k-1}^{-1}(v_k - \hat{v}_{k|k-1}) + \beta_k \hat{R}_{k|k-1}^{-1}(p_k - \hat{p}_{k|k-1})$ and the theory yields, see Proposition 13 of [8],

$$H_k = \begin{bmatrix} -(r_k)_\times & \alpha_k I_3 & \beta_k I_3 \end{bmatrix}, \tag{21}$$

and denoting $\Omega_k := \exp(dt(\omega_k)_\times)$, we have as in [8]

$$F_k = \begin{bmatrix} \Omega_k^{-1} & 0 & 0 \\ -dt\,\Omega_k^{-1}(a_k)_\times & \Omega_k^{-1} & 0 \\ 0 & dt\,\Omega_k^{-1} & \Omega_k^{-1} \end{bmatrix}. \tag{22}$$

We see we recover the important property of IEKF that the Jacobian matrices are state-independent. The IEKF updates its estimate according to

$$\hat{\chi}_{k|k} = \hat{\chi}_{k|k-1}\exp(K_k z_k). \tag{23}$$

The following sections presents how the invariant framework naturally ensures Property 2 and what can be done to satisfy at best Property 1. Since the problems tackled here arise from the noise-free nature of the considered pseudo-measurements, our work constitutes already a contribution in its own right compared to [3], as they only consider noisy measurements.

### B. What about desirable Property 2?

In the formalism of invariant filtering, the constraint (3) writes $\chi_k d_k = y_k$. This noise-free pseudo-measurement informs us that the state belongs to the subset

$$\mathcal{H} = \{\chi \in SE_2(3) \mid \chi d_k = y_k\}. \tag{24}$$

We first recall how the IEKF encodes uncertainty, see [3].

*Definition 1:* The IEKF belief at time instant $k$, accounting for measurements up to time $j$, for the true state $\chi_k$ is a concentrated Gaussian distribution on the Lie group [22]

$$\chi_k = \hat{\chi}_{k|j}\exp(\xi_{k|j}), \quad \xi_{k|j} \sim \mathcal{N}(0_{n\times 1}, P_{k|j}), \tag{25}$$

with $\hat{\chi}$ the "noise-free" mean estimate, and $P$ the covariance matrix encoding the statistical dispersion around the mean. As a result, for the belief to be consistent with the set $\mathcal{H}$, we would ideally like the entire updated distribution to lie within $\mathcal{H}$. It turns out to be the case indeed, as long as the updated mean is in the right subspace of course. This proves the structure of the invariant filtering framework is consistent with the physical problem indeed.

*Theorem 3:* Consider the IEKF's covariance matrix $P_{k|k}$ after updating its state via observation (3). Then if the updated estimate $\hat{\chi}_{k|k}$ satisfies $\hat{\chi}_{k|k} \in \mathcal{H}$, all the probabilistic dispersion encoded by the updated covariance matrix according to our belief model (25) lies within $\mathcal{H}$, that is, for all random $\xi \in \mathcal{N}(0, P_{k|k})$, we have $\hat{\chi}_{k|k}\exp(\xi) \in \mathcal{H}$.

*Proof:* The result is a direct application of the following property from [17], [23], which can be proved through a series expansion of the exponential, see the proof of (iv) of Theorem 1 in [23].

*Proposition 1:* Consider an element $\zeta$ of the Lie algebra of $SE_2(3)$ identified with $\mathbb{R}^9$. Let $H_k$ denote the Jacobian from invariant filtering associated to map $h(\chi) := \chi d_k$, i.e. given by (21) in our problem. Then we have $(\chi \in \mathcal{H} \text{ and } H_k\zeta = 0) \Rightarrow \chi\exp\zeta \in \mathcal{H}$.

Now, the Riccati update with the limit gain (7) ensures that $H_k P_{k|k} = 0_{m\times n}$, see Theorem 2. Thus any element $\xi$ in the span of $P_{k|k}$ satisfies $H_k\xi = 0$ ∎

We thus see that Property 2 is satisfied in a way that is meaningful and leads to a consistent belief: the covariance matrix $P_{k|k}$ being aligned with the actual dispersion, regardless of the linearization point. This is a clear improvement over the conventional EKF, that possesses none of the desired properties. In practice, this explains why the IEKF outperforms the EKF for the considered problem, as can be seen below. But before, let us turn to Property 1.

### C. What about Property 1?

In the previous theorem we had to assume Property 1 to be satisfied, that is, $\hat{\chi}_{k|k} \in \mathcal{H}$ for Property 2 to make complete sense. Unfortunately, due to the nonlinear nature of the problem, the IEKF does not satisfy Property 1. All we know is that the residual innovation (in other terms the prediction error) is null up to second order terms after the update, that is,

$$z_{k|k} = \hat{\chi}_{k|k}^{-1} y_k - d_k = \mathcal{O}\left(\|\xi_{k|k}\|^2\right), \tag{26}$$

and as a result $\hat{\chi}_{k|k}$ may not belong to the expected set $\mathcal{H}$.

To address this issue, we now propose an alternative IEKF update procedure to reduce the norm of the residual innovation. The rationale is the following. Pseudo-measurements come for free, in the sense that they are not actual sensor measurements, but side information that one has for certain. In principle, once a noise-free measurement is made, the drop in uncertainty is fully incorporated in the belief, so that making the same measurement immediately does not change the belief. In practice, we have seen this is not the case, owing to undesirable effects of nonlinearity. However, there is no reason one could not reuse this same information several times in a row.

Our idea is thus to update the estimate with the same noise-free pseudo-measurement $y_k$ and the same Kalman gain $K_k$ as long as it makes the prediction error decrease, in other words cycle on the noise-free measurement until $\|z_{k|k}\|$ stabilizes. This procedure makes sense if we keep the gain $K_k$ constant when performing the same noise-free measurement several times, since as soon as $K_k$ is updated it does not correct the state along the measurement direction, owing to Property 2 (as $K_k z_k$ lies in the span of $P_{k|k}$). The corresponding procedure is detailed in Algorithm 1.

*Remark 1:* Algorithm 1 reduces the residual innovation but is not always able to completely eliminate it. Indeed, the innovation can be decomposed as

$$z_k = z_{k,\parallel} + z_{k,\perp}, \tag{27}$$

where $z_{k,\parallel} \in \mathcal{R}(S_k)$ and $z_{k,\perp} \in \mathcal{R}(S_k)^\perp$, with $z_{k,\perp} = \mathcal{O}\left(\|\xi_{k|k-1}\|^2\right)$ being entirely due to linearization errors and $S_k = H_k P_{k|k-1} H_k^T$. The component $z_{k,\perp}$ lies in the kernel of $K_k$, *i.e.*, $K_k z_{k,\perp} = 0_{n\times 1}$. This means this part of the innovation accounts for a component $\xi_{k,\perp}$ of $\xi_{k|k-1}$ that lives within $\mathcal{R}(P_{k|k-1})^\perp$ and cannot be corrected using Kalman filtering techniques. As a consequence, the updated state $\hat{\chi}_{k|k}$ becomes slightly inconsistent with $P_{k|k}$. In most applications, this inconsistency issue is immediately fixed at the propagation stage as process noise reintroduces uncertainty in the directions for which the last noise-free update negated the variance. In the absence of process noise, this is problematic as it makes the filter become quickly overconfident. This problem is left unsolved here and will be the subject of future work.

**Algorithm 1** Update procedure for the IEKF with noise-free pseudo-measurements.

Compute $K_k$ using (7)
$z_k^0 \leftarrow \hat{\chi}_{k|k-1}^{-1} y_k - d_k$
$\chi \leftarrow \hat{\chi}_{k|k-1} \exp(K_k z_k^0)$
$z_k^1 \leftarrow \chi^{-1} y_k - d_k$
$i \leftarrow 1$
**while** $\|z_k^i - z_k^{i-1}\| > \mathrm{tol}$ **do**
    $\chi \leftarrow \chi \exp(K_k z_k^i)$
    $z_k^{i+1} \leftarrow \chi^{-1} y_k - d_k$
    $i \leftarrow i + 1$
**end while**
$\hat{\chi}_{k|k} \leftarrow \chi$
$P_{k|k} \leftarrow (I - K_k H_k) P_{k|k-1} (I - K_k H_k)^T$

## V. Estimation of the pendulum angle for a crane

We consider the problem of estimating the extended pose (orientation, position, velocity) of an IMU fixed on the hook of a crane as presented in section II and illustrated in Figure 1, and compare the three following filters:

- *EKF*: a conventional extended Kalman filter.
- *IEKF*: a conventional invariant extended Kalman filter.
- *Noise-free IEKF*: the invariant extended Kalman filter that implements gain (7) and uses Algorithm 1 to further refine its estimate at each pseudo-measurement.

For the conventional EKF and IEKF, the gain was computed by setting the noise covariance $N_k$ to the small value of $10^{-4} I_2$ and letting $K_k = P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + N_k)^{-1}$. Estimating the IMU pose in the 3D space with cable length as the unique constraint raises observability issues that might obscure the big picture of the present paper. For the sake of simplicity, the IMU is assumed to stay in the plane $\phi_k = 0$ so that we consider the associated 2D problem, letting $\chi_k \in SE_2(2)$ and $\xi_{k|j} \in \mathbb{R}^5$.

The ground-truth simulation is the trajectory followed by the hook when it starts with an initial angle $\theta_0 = 20°$ and no angular velocity. No other external force than gravity is simulated. The crane cable is assumed to stay straight during the entire simulation and its length varies according to the profile displayed in Figure 2. The simulation is stopped after 2 s. The IMU and the three filters operate at the same frequency of 100 Hz. The initial error covariance matrix is set to

$$P_{0|0} = \begin{bmatrix} 0.05^2 & 0_{1\times 4} \\ 0_{4\times 1} & 0.5^2 I_4 \end{bmatrix}. \tag{28}$$

The gyroscope and accelerometer of the IMU are affected by Normal noise of zero mean and covariance

$$E\left((w_k^\omega)^2\right) = (0.005)^2, \tag{29}$$

$$E\left(w_k^a (w_k^a)^T\right) = (0.005)^2 I_2, \tag{30}$$

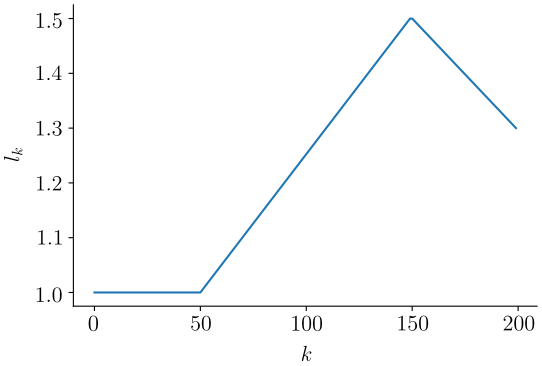



Fig. 2. Evolution of the length of the crane cable $l_k$ as a function of the time index.

where $E(\cdot)$ denotes the expectation operator. The error

$$\xi_k = \log\left(\hat{\chi}_{k|k}^{-1}\chi_k\right) \tag{31}$$

is used to compare the performances of the three filters, where $\log(\cdot) : SE_2(2) \to \mathbb{R}^5$ is the logarithmic map of $SE_2(2)$. The comparison is carried over 30 simulations in which the initial error $\xi_0$ is drawn randomly from the distribution $\mathcal{N}(0_{5\times1}, P_{0|0})$. The average and standard deviations of the norm of the error $\xi_k$ are computed for the three filters. Results are plotted in Figure 3. Two general comments can

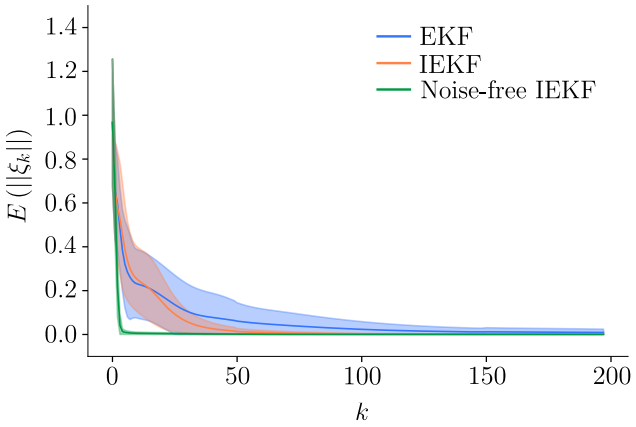



Fig. 3. Mean norm of the error function $\xi_k$ as a function of the time index, computed over 30 simulations. The standard deviation of the norm is displayed in light colors.

be made:

1) The conventional extended Kalman filter is outperformed by its invariant filtering counterparts: its convergence is way slower, with a large variability across simulations.
2) The noise-free IEKF exhibits the best convergence rate, with only 6 time steps needed on average for $\|\xi_k\|$ to go bellow 1% of the initial error $\|\xi_0\|$, against 59 and 191 for the classical IEKF and EKF. It is also the method with the lowest variability from one simulation to another. Note that with a tolerance $\text{tol} = 10^{-7}$, Algorithm 1 performs 2.03 cycles on average before $\|z_{k|k}\|$ stabilizes.

This simulation shows that treating equality constraints like (2) as proper noise-free pseudo-measurements clearly benefits the estimation and results in extremely fast convergence.

*Remark 2:* We indicate the failure to ensure Property 1 can sometimes make the noise-free IEKF diverge, when the residual innovation is too large. This leaves room for improvement, which is the subject of ongoing work.

## VI. Conclusion

This work presented a practical and efficient solution for incorporating equality constraints of the form (2) in the (extended) Kalman filtering framework. We advocated such constraints should be expressed as noise-free pseudo-measurements. We also derived a formulation of the Kalman gain that solves singular matrix inversion issues that may be encountered when dealing with noise-free pseudo-measurements. Using a proper matrix Lie group embedding and the theory of invariant filtering [3] leads to Normal filters that correctly encode the actual physical uncertainty in the problem: their belief is consistent with the constraint in that the covariance matrix after update corresponds to an uncertainty that is wholly contained within the constrained subset. Finally, an alternative update for the IEKF was proposed in order to mitigate the impact of linearization errors in the noise-free update process. The performance of this method was evaluated on the task of estimating the extended pose of the hook of a crane, and was proved to outperform both the EKF and IEKF in this context.

As a perspective, we would like to thoroughly treat the case of the 3D crane, which poses some potential observability issues. We also intend to derive a general theory for noise-free measurements in the context of invariant filtering, namely for two-frame systems [8]. In this context, we would like to explore how to fully enforce the constraint $y_k = h(\hat{\chi}_{k|k})$ at update in a natural way, that is, while retaining all the information that has been acquired before update. This might be achieved via the iterated Kalman filter [24], [25]. To do iterated EKF in the present context, we anticipate the results developed at Theorem 2 may prove useful.

## Appendix I
## Kalman Filter

Let $x_k \in \mathbb{R}^n$ be the state to estimate. The discrete-time Kalman Filter (KF) is an optimal probabilistic filter that fuses information coming from two noisy sources: the state dynamics model

$$x_{k+1} = F_k x_k + B_k u_k + w_k, \tag{32}$$

with $F_k \in \mathbb{R}^{n\times n}$, $B_k \in \mathbb{R}^{n\times b}$, input $u_k \in \mathbb{R}^b$, and process noise $w_k \in \mathbb{R}^n$ with covariance $Q_k$, and measurements

$$y_k = H_k x_k + n_k \tag{33}$$

with $H_k \in \mathbb{R}^{m\times n}$ and measurement noise $n_k \in \mathbb{R}^m$ with covariance $N_k$. The KF makes the assumption that $\xi_{k|j} = x_k - \hat{x}_{k|j} \sim \mathcal{N}(0_{n\times 1}, P_{k|j})$, where $\hat{x}$, $\xi$ and $P$ respectively denote the filter estimate, the estimation error and the associated covariance matrix. One iteration of the KF consists of two stages: the propagation and the update.

*a) Propagation:* The KF propagates its belief through the state dynamics model as follows:

$$\hat{x}_{k+1|k} = F_k\hat{x}_{k|k} + B_k u_k, \tag{34a}$$
$$P_{k+1|k} = F_k P_{k|k} F_k^T + Q_k. \tag{34b}$$

*b) Update:* Receiving measurement $y_k$, the KF updates its belief as follows:

$$z_k = y_k - H_k\hat{x}_{k|k-1}, \tag{35a}$$
$$S_k = H_k P_{k|k-1} H_k^T + N_k, \tag{35b}$$
$$K_k = P_{k|k-1} H_k^T S_k^{-1}, \tag{35c}$$
$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k z_k, \tag{35d}$$
$$P_{k|k} = (I_n - K_k H_k) P_{k|k-1} \tag{35e}$$

where $z_k$ is called the innovation, $S_k$ is the innovation covariance, and $K_k$ is the Kalman gain.

## REFERENCES


[1] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[2] R. Mahony and T. Hamel, "A geometric nonlinear observer for simultaneous localisation and mapping," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 2408–2415.

[3] A. Barrau and S. Bonnabel, "Invariant kalman filtering," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 237–257, 2018.

[4] P. van Goor, R. Mahony, T. Hamel, and J. Trumpf, "A geometric observer design for visual localisation and mapping," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 2543–2549.

[5] R. Mahony, T. Hamel, and J. Trumpf, "An homogeneous space geometry for simultaneous localisation and mapping," *Annual Reviews in Control*, vol. 51, pp. 254–267, 2021.

[6] A. Barrau and S. Bonnabel, "The invariant extended kalman filter as a stable observer," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2016.

[7] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended kalman filtering for robot state estimation," *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020.

[8] A. Barrau and S. Bonnabel, "The geometry of navigation problems," *IEEE Transactions on Automatic Control*, vol. 68, no. 2, pp. 689–704, 2023.

[9] S. Bonnabel and P. Rouchon, "On invariant observers," *Control and observer design for nonlinear finite and infinite dimensional systems*, pp. 53–65, 2005.

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

[11] M. R. Cohen and J. R. Forbes, "Navigation and control of unconventional vtol uavs in forward-flight with explicit wind velocity estimation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1151–1158, 2020.

[12] A. Barrau, "Non-linear state error based extended kalman filters with applications to navigation," Ph.D. dissertation, Mines Paristech, 2015.

[13] R. Mahony and J. Trumpf, "Equivariant filter design for kinematic systems on lie groups," *IFAC-PapersOnLine*, vol. 54, no. 9, pp. 253–260, 2021.

[14] D. Simon and T. L. Chia, "Kalman filtering with state equality constraints," *IEEE transactions on Aerospace and Electronic Systems*, vol. 38, no. 1, pp. 128–136, 2002.

[15] F. Rauscher, S. Nann, and O. Sawodny, "Motion control of an overhead crane using a wireless hook mounted IMU," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 5677–5682.

[16] K. S. Gurumoorthy, C. Grudzien, A. Apte, A. Carrassi, and C. K. Jones, "Rank deficiency of kalman error covariance matrices in linear time-varying system with deterministic evolution," *SIAM Journal on Control and Optimization*, vol. 55, no. 2, pp. 741–759, 2017.

[17] A. Barrau and S. Bonnabel, "Extended kalman filtering with nonlinear equality constraints: a geometric approach," *IEEE Transactions on Automatic Control*, vol. 65, no. 6, pp. 2325–2338, 2019.

[18] P. Chauchat, J. Vilà-Valls, and E. Chaumette, "Robust linearly constrained invariant filtering for a class of mismatched nonlinear systems," *IEEE Control Systems Letters*, vol. 6, pp. 223–228, 2021.

[19] S. Bonnabel and X. Claeys, "The industrial control of tower cranes: An operator-in-the-loop approach [applications in control]," *IEEE Control Systems Magazine*, vol. 40, no. 5, pp. 27–39, 2020.

[20] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[21] M. Brossard, A. Barrau, P. Chauchat, and S. Bonnabel, "Associating uncertainty to extended poses for on lie group imu preintegration with rotating earth," *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 998–1015, 2021.

[22] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, "Continuous-discrete extended kalman filter on matrix lie groups using concentrated gaussian distributions," *Journal of Mathematical Imaging and Vision*, vol. 51, pp. 209–228, 2015.

[23] P. Chauchat, A. Barrau, and S. Bonnabel, "Kalman filtering with a class of geometric state equality constraints," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 2581–2586.

[24] B. M. Bell and F. W. Cathey, "The iterated kalman filter update as a gauss-newton method," *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 294–297, 1993.

[25] G. Bourmaud, R. Mégret, A. Giremus, and Y. Berthoumieu, "From intrinsic optimization to iterated extended kalman filtering on lie groups," *Journal of Mathematical Imaging and Vision*, vol. 55, pp. 284–303, 2016.

## 5.3 Epilogue

Although this work addresses the central issue of incorporating kinematic constraints into the invariant filtering process and proposes solutions in this direction, it remains preliminary in several respects. In particular, two important issues remain unresolved at this point. This is especially critical since these issues can cause the IEKF to diverge.

The first unresolved issue is what we refer to as the residual innovation issue. As discussed in Remark 1 of the previous article, the algorithm describing the IEKF update for noise-free pseudo-measurements is not always able to fully compensate for the innovation $\mathbf{z}_k$. As a result, the updated state estimate may lie outside the observed set, that is, for a measurement $\mathbf{y}_k = \boldsymbol{\chi}_k \mathbf{d}_k$, the set $\{\boldsymbol{\chi} \mid \boldsymbol{\chi}\mathbf{d}_k = \mathbf{y}_k\}$. The next chapter is devoted to resolving this issue.

The second issue, which is not reported in the original paper because it was identified after publication, concerns Remark 2. That remark attributes the occasional divergence of the IEKF observed in simulations solely to the innovation residual problem described above. Subsequent analysis shows that the dominant cause was instead the noise-free formulation of the Kalman gain, which turns out to be numerically unstable. This formulation requires computing the Moore–Penrose pseudo-inverse $(\mathbf{H}_k\mathbf{L}_k)^\dagger$, where $\boldsymbol{\Sigma}_{k|k-1} = \mathbf{L}_k\mathbf{L}_k^T$. When the filter becomes extremely confident, the singular values of $\mathbf{H}_k\mathbf{L}_k$ become very small, making it difficult to distinguish them from truly zero singular values. Inappropriate thresholding may then lead to the inversion of extremely small values, resulting in an excessively large Kalman gain that can destabilize the filter.

Although this situation should not arise when the process noise re-injects sufficient uncertainty into the filter during the prediction step, it did occur in our experiments when the process noise standard deviation along all axes was set to a very small value. In the remainder of this thesis, this issue is addressed by using the classical Kalman gain formulation together with a regularization noise covariance matrix $\mathbf{N}_k^{\mathrm{r}} = \delta\mathbf{I}$, with $\delta > 0$. By continuity, this regularized gain converges to the theoretical noise-free gain as $\delta$ approaches zero. The value of $\delta$ is therefore chosen sufficiently small to remain close to this limit while avoiding numerical issues.

# 6 Handling noise-free measurements in the invariant filtering framework

## Outline

After identifying, in the previous chapter, how the IEKF can incorporate noise-free information and the challenges this entails, this chapter addresses the remaining innovation residual issue, namely the failure of the update step to yield a state estimate that lies in the observed subset. To overcome this limitation, we introduce the iterated invariant extended Kalman filter (IterIEKF), which improves upon the IEKF by enhancing its update stage through relinearization based on the Gauss–Newton iterative method. We further show that this iterative formulation provides theoretical guarantees ensuring that noise-free (pseudo-)measurements are properly incorporated not only locally, in accordance with Properties 1 and 2 of the previous chapter, but also globally.

## 6.1 Prologue

In many systems involving state equality constraints, it is often acceptable for the state estimate to satisfy the constraint only up to first order. However, some applications require extremely high estimation accuracy, in which case the residual innovation problem cannot be ignored. Consider, for instance, an articulated robotic arm teleoperated by a surgeon. In such a setting, feedback control relies on an estimate of the arm pose that must be accurate to fractions of a millimeter. Enforcing joint constraints only approximately is therefore inadequate, as even small local violations accumulate along the kinematic chain and lead to significant positional errors at the tool tip.

This issue is particularly problematic in view of the fact that the IEKF update for noise-free pseudo-measurements satisfies Property 2, namely that the updated error covariance matrix verifies $\mathbf{H}_k \boldsymbol{\Sigma}_{k|k} \mathbf{H}_k^T = \mathbf{0}$. To examine the implications of this property, consider a set of state equality constraints expressed as

$$\begin{cases} \boldsymbol{\chi}_k \mathbf{d}_k = \mathbf{y}_k, \\ \boldsymbol{\chi}_k \tilde{\mathbf{d}}_k = \tilde{\mathbf{y}}_k. \end{cases} \tag{6.1}$$

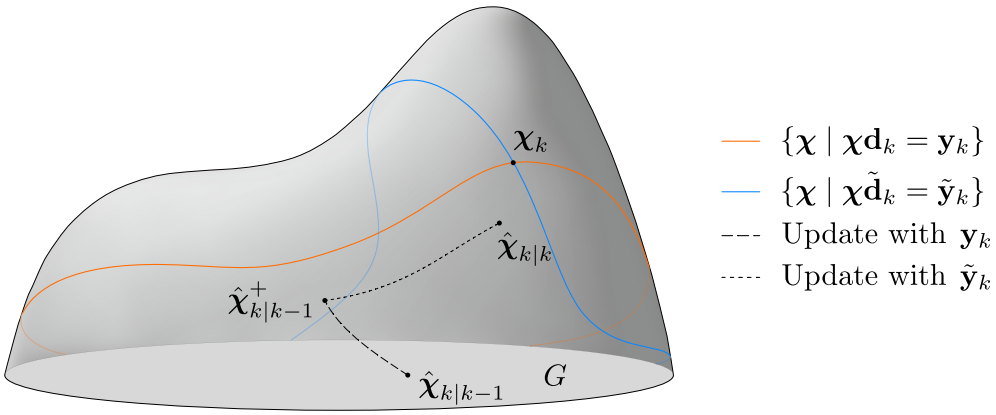



**Figure 6.1:** Illustration of two successive noise-free IEKF updates using the noise-free update scheme proposed in Algorithm 1 of Chapter 5.

These constraints can be processed sequentially, first updating the filter estimate using $\mathbf{y}_k$, and then using $\tilde{\mathbf{y}}_k$. We denote by $\hat{\chi}^+_{k|k-1}$ the intermediate estimate obtained after the first update. This situation is illustrated in Figure 6.1.

After the first update, the state estimate moves closer to the observed subset associated with $\mathbf{y}_k$, but does not lie exactly on it. More importantly, Property 2 ensures that $\mathbf{H}_k \boldsymbol{\Sigma}^+_{k|k-1} \mathbf{H}_k^T = \mathbf{0}$, or equivalently that $\mathbf{H}_k \boldsymbol{\Sigma}^+_{k|k-1} = \mathbf{0}$. This implies

$$\mathrm{Im}(\boldsymbol{\Sigma}^+_{k|k-1}) \subseteq \ker(\mathbf{H}_k), \tag{6.2}$$

where $\mathrm{Im}(\mathbf{A})$ and $\ker(\mathbf{A})$ denote the image and kernel of a matrix $\mathbf{A}$, respectively. By definition of the Kalman gain, any subsequent correction $\mathbf{K}_k \mathbf{z}_k$ therefore necessarily lies in $\ker(\mathbf{H}_k)$, and further updates are unable to move the state estimate closer to the observed set. The estimate is thus locked outside the constraint manifold.

This effect is visible during the second update with $\tilde{\mathbf{y}}_k$, for which the covariance $\boldsymbol{\Sigma}^+_{k|k-1}$ only permits motion along directions parallel to the first observed subset. Although process noise may later re-inject uncertainty in the constrained directions, this behavior renders the update step inefficient and can significantly impair filter convergence. This phenomenon motivates the need for an improved IEKF update mechanism capable of incorporating noise-free measurements in a globally consistent manner, which is the focus of the following paper.

## 6.2 Iterated invariant extended Kalman filter (IterIEKF)

This work has been published as a technical note in the journal *IEEE Transactions on Automatic Control* [99].

### 6.2.1 Author contributions

As the first author, I led the mathematical derivations of the work and was responsible for the numerical simulations. Prof. Silvère Bonnabel and Prof. Pierre Sacré provided close supervision throughout the project. Prof. Bonnabel and I were primarily responsible for writing the paper, with additional contributions and guidance from Prof. Sacré and Dr. Olivier Brüls. The original idea was proposed by Dr. Axel Barrau.

### 6.2.2 Reading tips

Local compatibility of the state estimate with a given noise-free measurement is defined in Definition 2 of the paper and is equivalent to Properties 1 and 2 of the previous article. This work adopts the notation $\mathbf{P}_{k|l}$ to denote the error covariance matrix, instead of $\mathbf{\Sigma}_{k|l}$.

# Iterated Invariant Extended Kalman Filter (IterIEKF)

Sven Goffin, Axel Barrau, Silvère Bonnabel, Olivier Brüls, and Pierre Sacré

***Abstract*—We study the mathematical properties of the Invariant Extended Kalman Filter (IEKF) when iterating on the measurement update step, following the principles of the well-known Iterated Extended Kalman Filter. This iterative variant of the IEKF (IterIEKF) systematically improves its accuracy through Gauss-Newton-based relinearization, and exhibits additional theoretical properties, particularly in the low-noise regime, that resemble those of the linear Kalman filter. We apply the proposed approach to the problem of estimating the extended pose of a crane payload using an inertial measurement unit. Our results suggest that the IterIEKF significantly outperforms the IEKF when measurements are highly accurate.**

***Index Terms*—Nonlinear state estimation, Invariant Kalman filtering, Lie groups, Nonlinear systems, Gauss-Newton method**

## I. Introduction

In the field of state estimation and observer design, the Extended Kalman Filter (EKF) is one of the most widespread methods used in practice. However, it is based on linearization, and the associated errors were early recognized as possibly degrading performance [1]. This motivated the development of iterative filtering algorithms such as the iterated EKF (IterEKF) [2]–[4], iterated EKF on Lie groups (LG-IterEKF) [5]–[7], and smoothers [8], where the discrepancy between the nonlinear output function and its first-order approximation is reduced, by refining the operating point.

On another note, observer design has benefited from geometric approaches over the past two decades. They may be traced back to attitude and pose estimation [9]–[11]. When turning to the more general problem of inertial navigation and localization, the Invariant Extended Kalman Filter (IEKF) [12], [13] has become a key alternative to the standard EKF. Its theoretical properties include convergence guarantees [12], consistency properties in the presence of unobservability [13], see also related works [14]–[16], and have led to applications in various fields, e.g., [8], [13], [14], [17]–[20] and in the industry [13]. The field of observers has benefited from the introduction of new groups brought by the IEKF theory, see e.g., [21]–[23], namely the groups $SE_2(3)$ and $SE_k(d)$ introduced in [12], [24] and their relation to the navigation equations, see [25] for a recent perspective. The equivariant observer framework, see [26], is closely related.

S. Goffin is a FRIA grantee of the Fonds de la Recherche Scientifique - FNRS.

S. Goffin and P. Sacré are with the Department of Electrical Engineering and Computer Science, University of Liège, Belgium (sven.goffin@uliege.be; p.sacre@uliege.be).

A. Barrau is with OFFROAD, France (axel@offroad.works).

S. Bonnabel is with MINES Paris, PSL Research University, Centre for Robotics, France (silvere.bonnabel@mines-paristech.fr).

O. Brüls is with the Department of Aerospace and Mechanical Engineering, University of Liège, Belgium (o.bruls@uliege.be).

In this technical note, we introduce the Iterated Invariant Extended Kalman Filter (IterIEKF), a refinement of the IEKF [13] that leverages the Gauss-Newton (GN) method to enhance its measurement update step, and which systematically improves its accuracy. Our main contribution is then a comprehensive analysis of the IterIEKF properties, revealing rare properties in nonlinear estimation that remind of the linear case, and offering new tools and insights into the theory of invariant filtering. Finally, to highlight the practical relevance of our approach, we apply the IterIEKF to a problem of engineering interest, and show it outperforms the IEKF when observations are very accurate (low measurement noise).

While the ultimate goal of invariant filtering is to recover properties of the linear Kalman filter in a nonlinear setting, and to identify systems of interest that lend themselves to this goal, a parallel line of work has been concerned with how to accommodate in general the geometric nature of the state space, without seeking further properties. In particular, the "Lie-Group EKF" (LG-EKF) from [27] proposes a generic intrinsic version of the EKF on Lie groups, that accounts for geometric features such as curvature. An iterated version, called LG-IterEKF was proposed in [5]. See, e.g., [28] and applications to inertial-lidar navigation [29] for recent references in this field. When tailored to the specific class of problems we consider, namely by using the groups from invariant filtering and the choice of errors that it advocates, the LG-IterEKF (that we call then *adapted*) is close to the proposed IterIEKF, although slightly different. As a result, we can prove the properties of the IterIEKF carry over to this adapted LG-IterEKF. This novel result constitutes a secondary contribution.

Section II introduces the proposed IterIEKF algorithm for both noisy and noise-free measurements. Section III derives theoretical properties of the IterIEKF in the context of noise-free measurements. We prove they carry over to the LG-IterEKF from [5] under some conditions. Section IV evaluates in simulations the performance of the proposed algorithm compared to other state-of-the-art filters for estimating the extended pose (orientation, velocity, and position) of a crane hook, equipped with an Inertial Measurement Unit (IMU).

In the following, the acronym IEKF is reserved for the invariant EKF, whereas IterEKF is used for the iterated EKF.

## II. The Iterated Invariant EKF (IterIEKF)

In this section, we first recall the equations of the IEKF and then introduce the IterIEKF, its iterated version.

### A. The IEKF equations

The invariant framework assumes the state $\boldsymbol{\chi}_k$ is an element of a matrix Lie group $G \subset GL_N(\mathbb{R})$ having dimension $n$,

where $GL_N(\mathbb{R})$ denotes the group of $N \times N$ invertible matrices. An example of such a state $\boldsymbol{\chi}$ is given below, in (15). The exponential map of $G$ is defined as $\exp_G(\cdot) := \exp_m(\mathcal{L}_{\mathfrak{g}}(\cdot))$, where $\exp_m(\cdot)$ is the matrix exponential and $\mathcal{L}_{\mathfrak{g}}(\cdot)$ is the bijective linear map identifying the Lie algebra $\mathfrak{g}$ with $\mathbb{R}^n$, see e.g., [12]. The IEKF comes in two versions, the left- or the right-invariant version, depending on the form of the observations, see [13]. This note focuses on the left-IEKF. Transposition to the right-IEKF is straightforward.

Consider the following nonlinear system in discrete time:

$$\boldsymbol{\chi}_{k+1} = \mathbf{f}(\boldsymbol{\chi}_k, \mathbf{u}_k, \mathbf{w}_k), \tag{1a}$$

$$\mathbf{y}_k = \boldsymbol{\chi}_k \mathbf{d}_k + \mathbf{n}_k, \tag{1b}$$

with $\mathbf{u}_k \in \mathbb{R}^m$ a control input, $\mathbf{f} : G \times \mathbb{R}^m \times \mathbb{R}^b \to G$ the function describing the system dynamics, $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$ an unknown process noise with $\mathbf{Q}_k \in \mathbb{R}^{b\times b}$, $\mathbf{y}_k \in \mathbb{R}^N$ the observation that consists of partial and noisy measurements of the state, $\mathbf{d}_k \in \mathbb{R}^N$ a known vector, and $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ an unknown measurement noise with $\mathbf{N}_k \in \mathbb{R}^{N\times N}$.

The (left) IEKF assumes that the state follows a concentrated Gaussian distribution on $G$ [30]–[33]. At time $k$, letting $l$ denote current time index $k$, or previous index $k-1$,

$$\boldsymbol{\chi}_k = \hat{\boldsymbol{\chi}}_{k|l} \exp_G(\boldsymbol{\xi}_{k|l}), \quad \text{where } \boldsymbol{\xi}_{k|l} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{k|l}), \tag{2}$$

where $\hat{\boldsymbol{\chi}}_{k|l} \in G$ is the (best) estimate of the state, and the linearized error $\boldsymbol{\xi}_{k|l} \in \mathbb{R}^n$ is a centered Gaussian with covariance matrix $\mathbf{P}_{k|l} \in \mathbb{R}^{n\times n}$. Using this model, the IEKF linearizes the equations of the system (1) at the current estimate, leading to the following update and propagation:

$$\text{Upd.} \quad \begin{cases} \mathbf{z}_k = \hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{y}_k - \mathbf{d}_k, \\ \hat{\mathbf{N}}_k = \hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{N}_k(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1})^T, \\ \mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \hat{\mathbf{N}}_k, \\ \mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k^T\mathbf{S}_k^{-1}, \\ \hat{\boldsymbol{\chi}}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1}\exp_G(\mathbf{K}_k\mathbf{z}_k), \\ \mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}, \end{cases}$$

$$\text{Prop.} \quad \begin{cases} \hat{\boldsymbol{\chi}}_{k+1|k} = \mathbf{f}(\hat{\boldsymbol{\chi}}_{k|k}, \mathbf{u}_k, \mathbf{0}), \\ \mathbf{P}_{k+1|k} = \mathbf{F}_k\mathbf{P}_{k|k}\mathbf{F}_k^T + \mathbf{G}_k\mathbf{Q}_k\mathbf{G}_k^T, \end{cases} \tag{3}$$

with $\mathbf{z}_k$ the innovation (i.e., prediction error), $\mathbf{S}_k$ the innovation covariance, and $\mathbf{K}_k$ the Kalman gain. As is customary in extended Kalman filtering, Jacobian matrices $\mathbf{F}_k, \mathbf{G}_k$ arise from first-order linearizations. However, in the case of invariant filtering, they are defined with respect to model (2), i.e.,

$$\begin{aligned} \mathbf{f}(\boldsymbol{\chi}\exp_G(\boldsymbol{\xi}), \mathbf{u}, \mathbf{w}) = \mathbf{f}(\boldsymbol{\chi}, \mathbf{u}, \mathbf{0}) \cdot \\ \exp_G\left(\mathbf{F}\boldsymbol{\xi} + \mathbf{G}\mathbf{w} + \mathcal{O}(\|\boldsymbol{\xi}\|^2, \|\mathbf{w}\|^2, \|\boldsymbol{\xi}\|\|\mathbf{w}\|)\right). \end{aligned} \tag{4}$$

When the function $\mathbf{f}$ possesses the group affine property (i.e., is of the form $\mathbf{f}(\boldsymbol{\chi}, \mathbf{u}, \mathbf{w}) = \bar{\mathbf{f}}(\boldsymbol{\chi}, \mathbf{u})\mathbf{g}(\mathbf{w})$ where $\bar{\mathbf{f}}$ satisfies $\bar{\mathbf{f}}(\mathbf{ab}, \mathbf{u}) = \bar{\mathbf{f}}(\mathbf{a}, \mathbf{u})\bar{\mathbf{f}}(\mathbf{I}, \mathbf{u})^{-1}\bar{\mathbf{f}}(\mathbf{b}, \mathbf{u})$, for all $\mathbf{a}, \mathbf{b} \in G$, $\mathbf{u} \in \mathbb{R}^m$, see [13]), the Jacobian $\mathbf{F}_k$ becomes independent of the current state estimate $\hat{\boldsymbol{\chi}}_{k|k}$ [12], [13]. Moreover, in the absence of process noise, this results in the exact error dynamics $\boldsymbol{\xi}_{k+1|k} = \mathbf{F}_k\boldsymbol{\xi}_{k|k}$, where the Jacobian $\mathbf{F}_k$, i.e., the first order, fully captures the nonlinearity of $\mathbf{f}$. This property called log-linearity is a key feature of invariant filtering. Note that an IEKF can be devised even if $\mathbf{f}$ is not group affine, based on expansion (4), and the theory to follow applies.

Similarly, we define $\mathbf{H}_k$ as the Jacobian of the innovation $\mathbf{z}_k$ w.r.t. $\boldsymbol{\xi}_{k|k-1}$. To compute it, we first express $\mathbf{y}_k$ as $\hat{\boldsymbol{\chi}}_{k|k-1}\exp_G(\boldsymbol{\xi}_{k|k-1})\mathbf{d}_k + \mathbf{n}_k$. We define the innovation $\mathbf{z}_k = \hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{y}_k - \mathbf{d}_k = \exp_G(\boldsymbol{\xi}_{k|k-1})\mathbf{d}_k - \mathbf{d}_k + \hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{n}_k$, and perform a first-order expansion w.r.t. $\boldsymbol{\xi}_{k|k-1}$. The corresponding Jacobian $\mathbf{H}_k$ is then defined through

$$\exp_G(\boldsymbol{\xi})\mathbf{d}_k = \mathbf{d}_k + \mathbf{H}_k\boldsymbol{\xi} + \mathcal{O}(\|\boldsymbol{\xi}\|^2), \tag{5}$$

whose detailed expression is to be found at Equation (10). Due to the specific form of the output and the definition of the innovation, the Jacobian $\mathbf{H}_k$ depends only on $\mathbf{d}_k$ and is thus independent of $\hat{\boldsymbol{\chi}}_{k|k-1}$, a feature of invariant filtering.

### *B. The IterIEKF equations*

When receiving a noisy measurement in the left-invariant form (1b), the IterIEKF aims to find the maximum *a posteriori* (MAP) estimate, described in the following result.

**Proposition 1.** *Starting from the prior* (2)*, with* $l = k-1$*, supposed to encode the state distribution at time* $k$ *conditional on past information* $\mathbf{y}_0, \dots, \mathbf{y}_{k-1}$*, the MAP estimate in the light of latest measurement* $\mathbf{y}_k$ *is given by* $\hat{\boldsymbol{\chi}}_{k|k}^{\star} = \hat{\boldsymbol{\chi}}_{k|k-1}\exp_G(\boldsymbol{\xi}^{\star})$*, where* $\boldsymbol{\xi}^{\star}$ *solves the optimization problem*

$$\boldsymbol{\xi}^{\star} = \underset{\boldsymbol{\xi}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{\xi}\|^2_{\mathbf{P}_{k|k-1}} + \frac{1}{2}\|\mathbf{z}_k - \exp_G(\boldsymbol{\xi})\mathbf{d}_k + \mathbf{d}_k\|^2_{\hat{\mathbf{N}}_k}, \tag{6}$$

*where* $\mathbf{z}_k$ *is the innovation from* (3)*,* $\|\boldsymbol{\eta}\|^2_{\Xi} := \boldsymbol{\eta}^T\Xi^{-1}\boldsymbol{\eta}$*, and where matrices* $\mathbf{P}_{k|k-1}$ *and* $\hat{\mathbf{N}}_k$ *are assumed to be invertible.*

*Proof.* From Bayes' rule $p(\boldsymbol{\xi}_{k|k-1} \mid \mathbf{y}_k) = p(\boldsymbol{\xi}_{k|k-1} \mid \mathbf{z}_k) \propto p(\boldsymbol{\xi}_{k|k-1})p(\mathbf{z}_k \mid \boldsymbol{\xi}_{k|k-1})$, with all densities $p$ implicitly conditional on past information $\mathbf{y}_0, \dots, \mathbf{y}_{k-1}$. Thus $p(\boldsymbol{\xi}_{k|k-1} \mid \mathbf{y}_k) \propto \exp(-\frac{1}{2}\|\boldsymbol{\xi}_{k|k-1}\|^2_{\mathbf{P}_{k|k-1}})\exp(-\frac{1}{2}\|\mathbf{z}_k - \exp_G(\boldsymbol{\xi}_{k|k-1})\mathbf{d}_k + \mathbf{d}_k\|^2_{\hat{\mathbf{N}}_k})$, maximized by $\boldsymbol{\xi}^{\star}$. □

The optimization problem (6) does not admit a closed-form solution. The IEKF addresses this by approximating $\exp_G(\boldsymbol{\xi})$ using its first-order Taylor expansion around $\mathbf{0}$. Inspired by the iterated EKF [2], a more accurate solution can be obtained by applying the GN algorithm to iteratively refine the estimate until convergence. This process results in a sequence of updates that closely resemble those of the IEKF. The resulting Iterated IEKF (IterIEKF) is outlined in Algorithm 1. Details are provided in Appendix A-A.

Iterating proves particularly useful when measurement noise is low, as we will see in the sequel. However, in the limit case where the measurement noise magnitude approaches zero, the innovation covariance $\mathbf{S}_k$ may become rank-deficient [34], in which case the Kalman gain is undefined. An adapted version of the IterIEKF for this specific scenario is detailed in Algorithm 2. Details are provided in Appendix A-B.

## III. Theoretical properties of the IterIEKF

Deriving properties in the presence of measurement noise seems out of reach, as is often the case in nonlinear filtering (e.g., the log-linear error property of [12] only holds in the

**Algorithm 1** The iterated invariant extended Kalman filter (IterIEKF)

Choose the initial state $\hat{\boldsymbol{\chi}}_{0|0} \in G$ and initial covariance $\mathbf{P}_{0|0} = \mathrm{Cov}(\boldsymbol{\xi}_{0|0}) \in \mathbb{R}^{n\times n}$.
**loop**
  Define Jacobians $\mathbf{F}_k$, $\mathbf{G}_k$, $\mathbf{H}_k$ of IEKF framework.
  Define noise covariances $\mathbf{Q}_k, \mathbf{N}_k$.

  {Update}
  $\mathbf{z}_k \leftarrow \hat{\boldsymbol{\chi}}^{-1}_{k|k-1}\mathbf{y}_k - \mathbf{d}_k$
  $\hat{\mathbf{N}}_k \leftarrow \hat{\boldsymbol{\chi}}^{-1}_{k|k-1}\mathbf{N}_k(\hat{\boldsymbol{\chi}}^{-1}_{k|k-1})^T$
  $\boldsymbol{\xi}^i_{k|k-1} \leftarrow \mathbf{0}$
  **while** $\boldsymbol{\xi}^i_{k|k-1}$ not converged **do**
    $\mathbf{H}^i_k \leftarrow \exp_G(\boldsymbol{\xi}^i_{k|k-1})\mathbf{H}_k\mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1})$
    $\mathbf{S}^i_k \leftarrow \mathbf{H}^i_k\mathbf{P}_{k|k-1}(\mathbf{H}^i_k)^T + \hat{\mathbf{N}}_k$
    $\mathbf{K}^i_k \leftarrow \mathbf{P}_{k|k-1}(\mathbf{H}^i_k)^T(\mathbf{S}^i_k)^{-1}$
    $\mathbf{z}^i_k \leftarrow \mathbf{z}_k - \exp_G(\boldsymbol{\xi}^i_{k|k-1})\mathbf{d}_k + \mathbf{d}_k + \mathbf{H}^i_k\boldsymbol{\xi}^i_{k|k-1}$
    $\boldsymbol{\xi}^i_{k|k-1} \leftarrow \mathbf{K}^i_k\mathbf{z}^i_k$
  **end while**
  $\hat{\boldsymbol{\chi}}_{k|k} \leftarrow \hat{\boldsymbol{\chi}}_{k|k-1}\exp_G(\boldsymbol{\xi}^i_{k|k-1})$
  $\mathbf{K}_k \leftarrow \mathbf{P}_{k|k-1}\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \hat{\mathbf{N}}_k)^{-1}$
  $\mathbf{P}_{k|k} \leftarrow (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}$

  {Propagation}
  $\hat{\boldsymbol{\chi}}_{k+1|k} \leftarrow \mathbf{f}(\hat{\boldsymbol{\chi}}_{k|k}, \mathbf{u}_k, \mathbf{0})$
  $\mathbf{P}_{k+1|k} \leftarrow \mathbf{F}_k\mathbf{P}_{k|k}\mathbf{F}_k^T + \mathbf{G}_k\mathbf{Q}_k\mathbf{G}_k^T$
**end loop**

$\mathcal{J}_r(\boldsymbol{\xi})$ denotes the right Jacobian [30] of $G$ at $\boldsymbol{\xi}$, see (30).

**Algorithm 2** The IterIEKF with noise-free measurements

Same as Algorithm 1 with the following exceptions.
5: Find $\mathbf{L}_{k|k-1}$ such that $\mathbf{P}_{k|k-1} = \mathbf{L}_{k|k-1}\mathbf{L}^T_{k|k-1}$.
13: $\mathbf{K}^i_k \leftarrow \mathbf{L}_{k|k-1}(\mathbf{H}^i_k\mathbf{L}_{k|k-1})^\dagger$
18: $\mathbf{K}_k \leftarrow \mathbf{L}_{k|k-1}(\mathbf{H}_k\mathbf{L}_{k|k-1})^\dagger$

Here, $(\cdot)^\dagger$ denotes the Moore-Penrose pseudo-inverse.

absence of process noise). However, we show that the IterIEKF possesses strong properties in the absence of measurement noise (whereas the state estimate and the propagation may be noisy). We therefore consider noise-free measurement

$$\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k. \tag{7}$$

In this case, a measurement defines a subset of the state space.

**Definition 1.** *We call* observed set *associated with the noise-free measurement* $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k$ *the subset*

$$\mathcal{S}_{\{\boldsymbol{\chi}\mathbf{d}_k=\mathbf{y}_k\}} := \{\boldsymbol{\chi} \in G \mid \boldsymbol{\chi}\mathbf{d}_k = \mathbf{y}_k\}. \tag{8}$$

Definition 2 establishes criteria for local compatibility of a Gaussian filter on Lie groups with measurement (7).

**Definition 2.** *The estimate* $(\hat{\boldsymbol{\chi}}_{k|l}, \mathbf{P}_{k|l})$ *is said to be* locally compatible with noise-free measurement $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k$ *if*

1) $\hat{\boldsymbol{\chi}}_{k|l} \in \mathcal{S}_{\{\boldsymbol{\chi}\mathbf{d}_k=\mathbf{y}_k\}}$,
2) $\mathbf{H}_k\mathbf{P}_{k|l}\mathbf{H}_k^T = \mathbf{0}$,

*where* $\mathbf{H}_k$ *denotes the Jacobian defined in* (5).

Since these criteria are not immediately obvious, we briefly motivate their formulation. Using model (2) and linearizing around $\boldsymbol{\xi}_{k|l} = \mathbf{0}$ via (5), we obtain $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k = \hat{\boldsymbol{\chi}}_{k|l}\exp_G(\boldsymbol{\xi}_{k|l})\mathbf{d}_k \approx \hat{\boldsymbol{\chi}}_{k|l}\mathbf{d}_k + \hat{\boldsymbol{\chi}}_{k|l}\mathbf{H}_k\boldsymbol{\xi}_{k|l}$. As $\boldsymbol{\xi}_{k|l} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{k|l})$, for this linear approximation of (7) to hold (almost surely) we must have both $\hat{\boldsymbol{\chi}}_{k|l}\mathbf{d}_k = \mathbf{y}_k$ (letting $\boldsymbol{\xi}_{k|l} = 0$), and $\mathbf{H}_k\boldsymbol{\xi}_{k|l} = \mathbf{0}$ whenever $\boldsymbol{\xi}_{k|l}$ in $\mathrm{Im}\,\mathbf{P}_{k|l}$, that is, 1) and 2).

Before turning to the IterIEKF and its properties in this context, let us further analyze the specific structure of the measurements under consideration.

### A. Properties of considered noise-free measurements

The theory of (left) invariant filtering focuses on measurements of the form (1b), see [12], which boil down to (7) when the noise is turned off. It turns out that the corresponding "observed set" (8) possesses an interesting structure.

**Proposition 2.** *Let* $\mathbf{H}_k$ *denote the Jacobian from invariant filtering defined in* (5)*. We have the following properties*

1) $\mathbf{H}_k\boldsymbol{\xi} = \mathbf{0},\ \boldsymbol{\xi} \in \mathbb{R}^n \Rightarrow \exp_G(\boldsymbol{\xi})\,\mathbf{d}_k = \mathbf{d}_k$,
2) $\mathfrak{s} := \{\mathcal{L}_\mathfrak{g}(\boldsymbol{\xi}) \in \mathfrak{g} \mid \mathbf{H}_k\boldsymbol{\xi} = \mathbf{0}\}$ *is a Lie subalgebra of* $\mathfrak{g}$.

*Proof.* As we are dealing with matrix Lie groups, the exponential coincides with the matrix exponential $\exp_m$ as follows

$$\exp_G(\boldsymbol{\xi}) = \exp_m(\mathcal{L}_\mathfrak{g}(\boldsymbol{\xi})) = \mathbf{I} + \sum_{l=1}^{+\infty}\frac{\mathcal{L}_\mathfrak{g}(\boldsymbol{\xi})^l}{l!}. \tag{9}$$

Keeping only first-order terms proves, in passing, that the Jacobian $\mathbf{H}_k$ from (5) writes

$$\mathbf{H}_k\boldsymbol{\xi} = \mathcal{L}_\mathfrak{g}(\boldsymbol{\xi})\mathbf{d}_k. \tag{10}$$

Thus $\mathbf{H}_k\boldsymbol{\xi} = \mathbf{0} \Rightarrow \mathcal{L}_\mathfrak{g}(\boldsymbol{\xi})^l\mathbf{d}_k = \mathbf{0}$ by induction, and then

$$\exp_G(\boldsymbol{\xi})\mathbf{d}_k = \mathbf{d}_k + \sum_{l=1}^{+\infty}\frac{\mathcal{L}_\mathfrak{g}(\boldsymbol{\xi})^l\mathbf{d}_k}{l!} = \mathbf{d}_k, \tag{11}$$

proving the first point. Regarding the second point, let $\boldsymbol{\delta}, \boldsymbol{\zeta} \in \mathbb{R}^n$ be such that $\mathcal{L}_\mathfrak{g}(\boldsymbol{\delta}), \mathcal{L}_\mathfrak{g}(\boldsymbol{\zeta}) \in \mathfrak{s}$. Considering the standard bilinear skew-symmetric Lie bracket defined as $[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$, we have $[\mathcal{L}_\mathfrak{g}(\boldsymbol{\delta}), \mathcal{L}_\mathfrak{g}(\boldsymbol{\zeta})]\mathbf{d}_k = \mathcal{L}_\mathfrak{g}(\boldsymbol{\delta})\mathcal{L}_\mathfrak{g}(\boldsymbol{\zeta})\mathbf{d}_k - \mathcal{L}_\mathfrak{g}(\boldsymbol{\zeta})\mathcal{L}_\mathfrak{g}(\boldsymbol{\delta})\mathbf{d}_k = \mathbf{0}$ by the definition of $\mathfrak{s}$ and (10). This proves that $[\mathcal{L}_\mathfrak{g}(\boldsymbol{\delta}), \mathcal{L}_\mathfrak{g}(\boldsymbol{\zeta})] \in \mathfrak{s}$, confirming that $\mathfrak{s}$ is closed under the Lie bracket and is therefore a subalgebra of $\mathfrak{g}$. □

The first point of Proposition 2 is directly useful for proving the compatibility properties of the IterIEKF, in the sense of Definition 2. This will be established in the next subsection, relying on the following key Lemma 1. We also note, in passing, that the second point will only become relevant when extending the present theory to the iterated EKF on Lie groups (LG-IterEKF) of [33], under certain conditions.

**Lemma 1.** *Let* $\mathbf{H}_k$ *denote the Jacobian from invariant filtering defined in* (5)*. We have*

$$\begin{gathered}\hat{\boldsymbol{\chi}}_{k|l} \in \mathcal{S}_{\{\boldsymbol{\chi}\mathbf{d}_k=\mathbf{y}_k\}} \quad \textit{and} \quad \mathbf{H}_k\boldsymbol{\xi} = \mathbf{0},\ \boldsymbol{\xi} \in \mathbb{R}^n \\ \Downarrow \\ \hat{\boldsymbol{\chi}}_{k|l}\exp_G(\boldsymbol{\xi}) \in \mathcal{S}_{\{\boldsymbol{\chi}\mathbf{d}_k=\mathbf{y}_k\}}.\end{gathered} \tag{12}$$

*Proof.* Using the first point of Proposition 2 yields

$$\mathbf{H}_k\boldsymbol{\xi} = \mathbf{0} \Rightarrow \hat{\boldsymbol{\chi}}_{k|l} \exp_G(\boldsymbol{\xi})\mathbf{d}_k = \hat{\boldsymbol{\chi}}_{k|l}\mathbf{d}_k = \mathbf{y}_k, \tag{13}$$

which proves $\hat{\boldsymbol{\chi}}_{k|l} \exp_G(\boldsymbol{\xi}) \in \mathcal{S}_{\{\boldsymbol{\chi}_{\mathbf{d}_k=\mathbf{y}_k}\}}$. □

A direct and important consequence is the following.

**Proposition 3.** *Assume the two points of Definition 2 are satisfied. Then, the entire probability distribution encoded by* (2) *is in fact (almost surely) contained within the observed set* $\mathcal{S}_{\{\boldsymbol{\chi}_{\mathbf{d}_k=\mathbf{y}_k}\}}$*, i.e., the estimate is* globally compatible with the noise-free measurement $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k$.

In other words, if the filter manages to encode the compatibility assumptions correctly locally, it will convey an estimated distribution being wholly consistent with available information, beyond the first order. This shall play a key role.

### *B. Compatibility of the IterIEKF with the measurements*

Let us first focus on the first point of Definition 2. As the magnitude of the measurement noise tends to zero, the optimization problem (6) boils down to finding the smallest $\boldsymbol{\xi}$ in the sense of the metric induced by $\mathbf{P}_k$, that satisfies the hard constraint $\|\mathbf{z}_k - \exp_G(\boldsymbol{\xi})\mathbf{d}_k + \mathbf{d}_k\|^2 = 0$. As $\mathbf{z}_k = \hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{y}_k - \mathbf{d}_k$, this constraint boils down to $\hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{y}_k = \exp_G(\boldsymbol{\xi})\,\mathbf{d}_k$ and the minimizer $\boldsymbol{\xi}^\star$ we are seeking necessarily satisfies $\hat{\boldsymbol{\chi}}_{k|k-1} \exp_G(\boldsymbol{\xi}^\star)\,\mathbf{d}_k = \mathbf{y}_k$. Provided that the IterIEKF iterate $\boldsymbol{\xi}_{k|k-1}^i$, which is a GN descent, indeed converges to such a minimizer, the updated state $\hat{\boldsymbol{\chi}}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1} \exp_G(\boldsymbol{\xi}^\star)$ will (by definition) belong to the observed set, i.e., $\hat{\boldsymbol{\chi}}_{k|k} \in \mathcal{S}_{\{\boldsymbol{\chi}_{\mathbf{d}_k=\mathbf{y}_k}\}}$, ensuring the first point of Definition 2. As the optimization problem is not convex, convergence of the GN descent is only guaranteed if the true error $\boldsymbol{\xi}_{k|k-1}$ is sufficiently close to $\mathbf{0}$. However, by contrast, non-iterative update schemes, such as the IEKF update, do not guarantee that $\hat{\boldsymbol{\chi}}_{k|k} \in \mathcal{S}_{\{\boldsymbol{\chi}_{\mathbf{d}_k=\mathbf{y}_k}\}}$ even for small initial errors. In the upcoming simulations of Section IV, and more generally in all simulations we have performed, the IterIEKF always converges to the observed set.

Let us now turn to the second point of Definition 2.

**Theorem 1.** *The IterIEKF updated covariance matrix* $\mathbf{P}_{k|k}$ *in the light of noise-free measurement* $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k$ *naturally ensures* $\mathbf{H}_k\mathbf{P}_{k|k}\mathbf{H}_k^T = \mathbf{0}$.

*Proof.* This is a consequence of linear Kalman theory, which may be proved as follows. When facing noise-free measurements, the Kalman gain is computed as $\mathbf{K}_k = \mathbf{L}_{k|k-1}(\mathbf{H}_k\mathbf{L}_{k|k-1})^\dagger$, where $\mathbf{P}_{k|k-1} = \mathbf{L}_{k|k-1}\mathbf{L}_{k|k-1}^T$. Letting $\mathbf{A}_k = \mathbf{H}_k\mathbf{L}_{k|k-1}$, we have: $\mathbf{H}_k\mathbf{P}_{k|k}\mathbf{H}_k^T = \mathbf{H}_k(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}\mathbf{H}_k^T, = (\mathbf{A}_k - \mathbf{A}_k(\mathbf{A}_k)^\dagger\mathbf{A}_k)\mathbf{A}_k^T = \mathbf{0}$, where we used $\mathbf{A}_k\mathbf{A}_k^\dagger\mathbf{A}_k = \mathbf{A}_k$. □

### *C. Consequences of the compatibility property*

When fed with a measurement (7) (resp. with a measurement (1b) with small noise), the information that the state lies in (resp. is close to) the observed subset $\mathcal{S}_{\{\boldsymbol{\chi}_{\mathbf{d}_k=\mathbf{y}_k}\}}$ should be well encoded in the filter, that is, no immediate subsequent measurement should be able to destroy that piece of information. The following result shows that the IterIEKF indeed can inherently "lock in" perfect information, which is akin to the behavior of the (linear) Kalman filter when confronted with a noise-free linear measurement.

**Theorem 2.** *Let* $\bar{\mathbf{y}}_k = \boldsymbol{\chi}_k\bar{\mathbf{d}}_k$ *represent a noise-free piece of information about the true state. Consider an IterIEKF whose current distribution is compatible with this information in the sense of Definition 2, meaning that* $\hat{\boldsymbol{\chi}}_{k|k-1} \in \mathcal{S}_{\{\boldsymbol{\chi}_{\bar{\mathbf{d}}_k=\bar{\mathbf{y}}_k}\}}$ *and that* $\bar{\mathbf{H}}_k\mathbf{P}_{k|k-1}\bar{\mathbf{H}}_k^T = \mathbf{0}$*, where* $\bar{\mathbf{H}}_k$ *is the Jacobian associated with* $\bar{\mathbf{d}}_k$*. If the estimate is subsequently updated using a (possibly noisy) measurement* $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k + \mathbf{n}_k$*, then the corresponding updated estimate* $(\hat{\boldsymbol{\chi}}_{k|k}, \mathbf{P}_{k|k})$*, which incorporates this new measurement* $\mathbf{y}_k$*, remains compatible with the deterministic information* $\bar{\mathbf{y}}_k = \boldsymbol{\chi}_k\bar{\mathbf{d}}_k$*. Mathematically,*

$$\begin{aligned} \hat{\boldsymbol{\chi}}_{k|k-1} \in \mathcal{S}_{\{\boldsymbol{\chi}_{\bar{\mathbf{d}}_k=\bar{\mathbf{y}}_k}\}} \quad &\text{and} \quad \bar{\mathbf{H}}_k\mathbf{P}_{k|k-1}\bar{\mathbf{H}}_k^T = \mathbf{0}, \\ &\Downarrow \\ \hat{\boldsymbol{\chi}}_{k|k} \in \mathcal{S}_{\{\boldsymbol{\chi}_{\bar{\mathbf{d}}_k=\bar{\mathbf{y}}_k}\}} \quad &\text{and} \quad \bar{\mathbf{H}}_k\mathbf{P}_{k|k}\bar{\mathbf{H}}_k^T = \mathbf{0}. \end{aligned} \tag{14}$$

*Proof.* Let us first consider the first point of Definition 2. During the update with $\mathbf{y}_k$, we see at line 17 of Algorithm 1 that $\hat{\boldsymbol{\chi}}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1} \exp_G(\boldsymbol{\xi}_{k|k-1}^i)$, where $\boldsymbol{\xi}_{k|k-1}^i = \mathbf{K}_k^i\mathbf{z}_k^i \in \operatorname{Im}\mathbf{K}_k^i \subseteq \operatorname{Im}\mathbf{P}_{k|k-1}$, with $\operatorname{Im}\mathbf{K}_k^i$ denoting the span of $\mathbf{K}_k^i$. Since $\bar{\mathbf{H}}_k\mathbf{P}_{k|k-1}\bar{\mathbf{H}}_k^T = \mathbf{0}$ by assumption, we necessarily have $\bar{\mathbf{H}}_k\boldsymbol{\xi}_{k|k-1}^i = \mathbf{0}$. Then, as $\hat{\boldsymbol{\chi}}_{k|k-1} \in \mathcal{S}_{\{\boldsymbol{\chi}_k\bar{\mathbf{d}}_k=\bar{\mathbf{y}}_k\}}$, a direct application of Lemma 1 ensures $\hat{\boldsymbol{\chi}}_{k|k} \in \mathcal{S}_{\{\boldsymbol{\chi}_{\bar{\mathbf{d}}_k=\bar{\mathbf{y}}_k}\}}$.

Now, regarding the second point of Definition 2, the condition $\bar{\mathbf{H}}_k\mathbf{P}_{k|k-1}\bar{\mathbf{H}}_k^T = \mathbf{0}$ implies $\mathbf{P}_{k|k-1}\bar{\mathbf{H}}_k^T = \mathbf{0}$ by symmetry. The update of the IterIEKF with measurement $\mathbf{y}_k$, see line 19 of Algorithm 1, gives $\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}$, where $\mathbf{H}_k$ is the Jacobian associated with $\mathbf{d}_k$. Since $\operatorname{Im}\mathbf{K}_k \subseteq \operatorname{Im}\mathbf{P}_{k|k-1}$, the updated covariance $\mathbf{P}_{k|k}$ also verifies $\mathbf{P}_{k|k}\bar{\mathbf{H}}_k^T = \mathbf{0}$, whatever $\mathbf{H}_k$. Consequently, $\bar{\mathbf{H}}_k\mathbf{P}_{k|k}\bar{\mathbf{H}}_k^T = \mathbf{0}$. □

Once the noise-free information has been properly accounted for, the IterIEKF effectively "focuses" on a problem of reduced dimensionality: the subsequent update correctly adjusts the state within the observed set $\mathcal{S}_{\{\boldsymbol{\chi}_{\bar{\mathbf{d}}_k=\bar{\mathbf{y}}_k}\}}$. This significantly improves the filter efficiency in practice, as will be demonstrated in the simulations. By contrast, a classical EKF or even an iterated EKF may step out of the appropriate subset, resulting in less efficient estimation updates.

*Illustration on $SO(3)$:* Albeit a simple example, it is pedagogical to illustrate the results on a problem involving rotation matrices. Let $\boldsymbol{\chi}_k \in SO(3)$ be an unknown rotation matrix representing the orientation of a drone in an inertial frame. Suppose an IterIEKF is used to estimate $\boldsymbol{\chi}_k$, and that prior observation has enabled perfect knowledge of the gravity vector in the drone body frame $\bar{\mathbf{d}}_k$. As a result, the current estimate $(\hat{\boldsymbol{\chi}}_{k|k-1}, \mathbf{P}_{k|k-1})$ is compatible, in the sense of Definition 2, with the noise-free information $\mathbf{g} = \boldsymbol{\chi}_k\bar{\mathbf{d}}_k$. Now, suppose a new measurement $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k + \mathbf{n}_k$ is received from another sensor. When updating the estimate using $\mathbf{y}_k$, all iterates of the IterIEKF, denoted by $\hat{\boldsymbol{\chi}}_{k|k-1} \exp_{SO(3)}(\boldsymbol{\xi}_{k|k-1}^i)$, remain within the observed set $\mathcal{S}_{\{\boldsymbol{\chi}_{\bar{\mathbf{d}}_k=\mathbf{g}}\}}$. Moreover, the update also ensures $\bar{\mathbf{H}}_k\mathbf{P}_{k|k}\bar{\mathbf{H}}_k^T = \mathbf{0}$. Consequently, once the noise-free information $\mathbf{g} = \boldsymbol{\chi}_k\bar{\mathbf{d}}_k$ is incorporated, the IterIEKF operates in a subset of reduced dimensionality—namely, it seeks planar rotations within a 2D subspace. This makes the filter significantly more efficient because 1) it performs updates

directly within a reduced subspace that contains the true state, and 2) it wholly preserves the information from the previous measurements. Note that, if noise is low instead, the latter remains approximately true—as there is a continuum between the noise-free and low-noise cases—which should make the filter very efficient in practice when measurement noise is low.

### *D. Extending the theory to the iterated EKF on Lie groups*

The LG-IterEKF, introduced in [5], transposes the standard iterated EKF to Lie group state spaces, by leveraging an intrinsic GN method on the group, that accounts for its manifold structure, notably curvature. LG-IterEKF of [5] is a general algorithm, meant to address general problems on Lie groups having no specific structure, beyond the intrinsic nature of the state space. By contrast, our IterIEKF addresses specific types of observations, namely (1b) (or its right-invariant counterpart $\mathbf{y}_k = \boldsymbol{\chi}_k^{-1}\mathbf{d}_k + \mathbf{n}_k$, in which case an iterated *right-invariant* EKF should be used, by mimicking the difference between the LIEKF and the RIEKF of [12]). It defines a modified innovation $\mathbf{z}_k = \hat{\boldsymbol{\chi}}_k^{-1}\mathbf{y}_k - \mathbf{d}_k$, allowing for the use of the invariant Kalman filter framework of [12], [13]. The standard invariant EKF has had various successes across control, navigation, and robotics, and is clearly a different algorithm than the LG-EKF of [27].

That said, the questions one could address are as follows. Assume we use the theory of invariant filtering to endow the LG-IterEKF with all the features of the IterIEKF—namely, measurements of the form (7), uncertainty model (2), and the Kalman gain from Algorithm 2 to address rank deficiency issues, as well as the appropriate choice regarding left or right versions (note that [5] chooses a systematic right-invariant error based setting whereas we advocate to choose the error depending on the form of the observations), resulting in an "invariant" version of the LG-IterEKF, that we call *adapted LG-IterEKF*, see Algorithm 3. Then, 1) do we actually recover the IterIEKF, and 2) if no, does it inherit any of the properties we have proved, given that no such properties have ever been proved for the LG-IterEKF? The answer to point 1 is no, as intrinsic GN on the group entails adding various Jacobians and differs from the GN algorithm addressing (6), that directly optimizes in a linear space. The answer to point 2 is yes, the properties above do carry over. This is detailed, proved, and discussed, in Appendix B.

**Algorithm 3** The adapted LG-IterEKF from [5] when measurements (7) are considered, uncertainty model (2) is used, potential rank deficiency issues in the Kalman gain are addressed via the noise-free gain from Algorithm 2, and a left-invariant error is used instead. See (30) for definition of $\mathcal{J}_r$.

Same as Algorithm 2 with the following exceptions.
11: $\mathbf{H}_k^i \leftarrow \hat{\boldsymbol{\chi}}_{k|k-1}\exp_G(\boldsymbol{\xi}_{k|k-1}^i)\mathbf{H}_k\mathcal{J}_r(\boldsymbol{\xi}_{k|k-1}^i)$
14: $\mathbf{z}_k^i \leftarrow \mathbf{y}_k - \hat{\boldsymbol{\chi}}_{k|k-1}\exp_G(\boldsymbol{\xi}_{k|k-1}^i)\mathbf{d}_k + \mathbf{H}_k^i\boldsymbol{\xi}_{k|k-1}^i$
19: $\mathbf{P}_{k|k} \leftarrow \mathcal{J}_r(\boldsymbol{\xi}_{k|k-1}^i)(\mathbf{I} - \mathbf{K}_k^i\mathbf{H}_k^i)\mathbf{P}_{k|k-1}\mathcal{J}_r(\boldsymbol{\xi}_{k|k-1}^i)^T$

## IV. Application of engineering interest

We propose using the IterIEKF to estimate the position, velocity, and orientation (extended pose) of the hook of a crane transporting a load, equipped with an IMU. Mounting an IMU on a crane hook for real-time sensor data transmission is technically very feasible nowadays [35], and may open the door to new automation capabilities since it allows for feedback. Leveraging the Lie group $SE_2(3)$ introduced by invariant filtering theory in [12], we represent the state as

$$\boldsymbol{\chi}_k = \begin{bmatrix} \mathbf{R}_k & \mathbf{v}_k & \mathbf{p}_k \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix} \in SE_2(3), \tag{15}$$

where $\mathbf{R}_k \in SO(3)$ is the rotation matrix between the IMU and inertial frames, and $\mathbf{v}_k$, $\mathbf{p}_k \in \mathbb{R}^3$ are the IMU velocity and position in the inertial frame. The inertial frame is fixed at the crane cable attachment point with the $z$-axis oriented upward, and remains stationary. The hook and IMU frames are assumed perfectly aligned. See our preliminary conference paper [34] for more details on this application.

Neglecting IMU biases, the system dynamics write

$$\mathbf{R}_{k+1} = \mathbf{R}_k \exp_{SO(3)}((\boldsymbol{\omega}_k + \mathbf{w}_k^{\omega})dt), \tag{16a}$$
$$\mathbf{v}_{k+1} = \mathbf{v}_k + (\mathbf{R}_k(\mathbf{a}_k + \mathbf{w}_k^{a}) + \mathbf{g})\, dt, \tag{16b}$$
$$\mathbf{p}_{k+1} = \mathbf{p}_k + \mathbf{v}_k\, dt, \tag{16c}$$

where $\boldsymbol{\omega}_k$ and $\mathbf{a}_k$ denote the IMU angular velocity and linear acceleration, $\mathbf{w}_k^{\omega}$ and $\mathbf{w}_k^{a}$ represent Gaussian noise in the gyroscope and accelerometer, and $\mathbf{g}$ is the gravity vector. The IMU is attached to a cable whose length $L_k$ is very accurately measured in modern cranes by motor encoders. We hence use it as noise-free measurement. Mathematically, this writes $\boldsymbol{\chi}_k\mathbf{d}_k = \mathbf{0}$, with $\mathbf{d}_k = \begin{bmatrix}0 & 0 & L_k & 0 & 1\end{bmatrix}^T$. The last two rows of $\boldsymbol{\chi}_k\mathbf{d}_k$ are discarded as they do not contain information.

A short simulation is carried out. The hook starts with an initial orientation of $-45^\circ$ around the $y$-axis, zero velocity, and position $\mathbf{p}_0 = \begin{bmatrix}\sqrt{2}/2 & 0 & -\sqrt{2}/2\end{bmatrix}^T$. The filter and IMU operate at $100\,\mathrm{Hz}$, with a time step $dt$ of $0.01\,\mathrm{s}$. The cable length follows $L_k = L_c(k\cdot dt)$, where $L_c(t)$ evolves as $\ddot{L}_c(t) + 12\dot{L}_c(t) + 16L_c(t) = 64$, with initial conditions $L_c(0) = 1\,\mathrm{m}$ and $\dot{L}_c(0) = 0\,\mathrm{m/s}$.

We compare five filters: EKF, IterEKF, (adapted) LG-IterEKF (implemented as in Algorithm 3), IEKF, and IterIEKF. Results are averaged over 500 runs, each with the same ground truth trajectory and a random initial error drawn from (2). The estimation is constrained to the $xz$-plane to ensure observability. Gyroscope noise has a standard deviation of $0.974\,^\circ/\mathrm{s}$ about the $y$-axis and $0\,^\circ/\mathrm{s}$ about the $x$- and $z$-axes. Accelerometer noise has a standard deviation of $0.1\,\mathrm{m/s^2}$ along the $x$- and $z$-axes and $0\,\mathrm{m/s^2}$ along the $y$-axis. The initial error covariance matrix encodes a standard deviation of $45^\circ$ for rotation about the $y$-axis and $5\,\mathrm{m/s}$ and $5\,\mathrm{m}$ for velocity and position along the $x$- and $z$-axes. No measurement noise is assumed. However, although the noise-free gain formulation used in Algorithm 2 is always well-defined and serves as a basis for the theoretical analysis, it is not well-suited to practical implementations, as it can be difficult to distinguish small singular values from actual zeros in the calculation of the pseudo-inverse. As a result, the noise-free gain formulation from Algorithm 2 is replaced with Algorithm 1, setting the measurement noise covariance

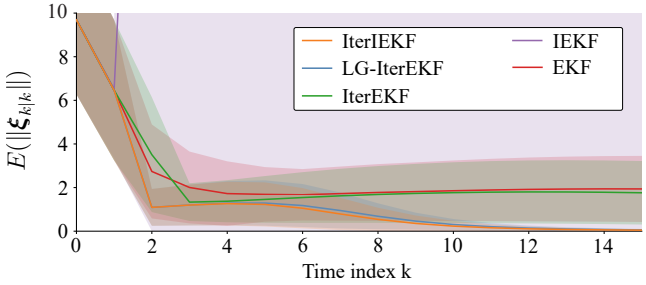



Fig. 1. Evolution of the average and standard deviation of the estimation error norm.

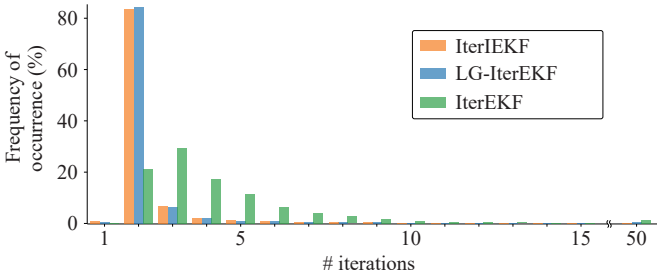



Fig. 2. Histogram of the number of iterations required at the update stage of the iterated filters, computed over 500 runs.

| RMSE | EKF | IterEKF | LG-IterEKF | IEKF | IterIEKF |
|---|---|---|---|---|---|
| Orientation | 0.805 | 0.792 | 0.592 | 1.112 | **0.574** |
| Velocity | 1.489 | 1.458 | 1.136 | 6.039 | **1.116** |
| Position | 1.054 | 1.041 | 0.871 | 1.852 | **0.858** |

TABLE I
RMSE COMPUTED OVER THE FIRST 15 TIMES STEPS OF THE 500 RUNS. THE SMALLEST VALUES ARE HIGHLIGHTED IN BOLD.

to a tiny value, namely $\mathbf{N}_k = 10^{-5}\mathbf{I}$. For iterative filters, the GN method stops when $\|\boldsymbol{\xi}^i_{k|k-1}\|_2$ changes by less than $10^{-5}$ between iterations or reaches a maximum of 50 iterations.

Figure 1 shows the mean estimation error norm over time, averaged across 500 runs, with standard deviation. Table I summarizes the global Root Mean Square Error (RMSE) computed over the first 15 time steps of the 500 runs. The IterIEKF and LG-IterEKF exhibit similar behavior, consistently outperforming all other filters. The first two updates significantly reduce the error on average, after which subsequent updates continue to decrease the error, albeit at a slower rate. Notably, both filters successfully converge across all 500 runs, regardless of the initial error. The IterIEKF performs slightly better on average proving that considering (6) instead of actual GN on the group does not pose problems.

The EKF and IterEKF exhibit similar performance, while the IterEKF is slightly better. They are outperformed by the introduced algorithm because their updates make them step out of the observed set. The IEKF surprisingly fails to converge in most cases. This behavior is expected with low measurement noise: in a noise-free setting, the IEKF update fails to satisfy the first condition in Definition 2, while still enforcing the second. Consequently, the estimate is confined to a subset that excludes the true state and cannot escape. This phenomenon underscores the necessity of an iterative update in the invariant filtering framework when handling accurate measurements.

As filtering is typically performed in real time, it is crucial to assess the number of iterations required during the update stage for the IterEKF, LG-IterEKF, and IterIEKF. Figure 2 presents a histogram of the iteration count for these iterative filters. The results indicate that in over $80\%$ of cases, both the IterIEKF and LG-IterEKF complete the update stage in just two iterations, making their computational cost relatively low. In contrast, the IterEKF generally requires a higher number of iterations on average, which could lead to increased computational expense.

## V. CONCLUSION

In this technical note, we introduced the IterIEKF, an iterated version of the IEKF inspired by the IterEKF. We analyzed its properties in the limit case of noise-free measurements, expressed in left-invariant form (7) (in case of right-invariant measurements, deriving the corresponding right-IterIEKF is straightforward, and the results immediately apply). Specifically, we derived criteria for assessing the local compatibility of Gaussian filters on Lie groups with this class of measurements, and leveraged their specific structure to demonstrate that these criteria also ensure global compatibility. Then, we proved the IterIEKF update produces estimates that are compatible, and moreover tend to inherently "hard code" this compatibility. We also proved that the same properties hold for the LG-IterEKF when tailored to the invariant framework.

We applied the IterIEKF to the problem of estimating the extended pose of a crane hook equipped with an IMU, where mechanical information is assimilated in the form of a noise-free pseudo-measurement. The IterIEKF demonstrated the best performance, closely followed by the adapted LG-IterEKF [5], a fact explained by our present theory. The superiority of both algorithms is consistent with the theoretical guarantees we have proved, and the IEKF is outperformed in this context.

However, we recommend using the IterIEKF over the LG-IterEKF, because the IEKF possesses convergence properties (notably log-linearity of the error [12]) and consistency properties that have never been proved for the LG-EKF. In particular, the unobservability consistency properties of the IEKF [13], [19], [20] may not be inherited by the LG-IterEKF, although proving this fact is left for future research.

Finally, simulations showed only a few iterations were needed for the GN method used in the IterIEKF update to converge, making the IterIEKF viable for real time.

As a perspective, we aim to apply this iterative estimation technique to other problems where the IEKF has proven effective, particularly in high-precision navigation. Moreover, we will explore its application in state estimation where constraints could be enforced as noise-free pseudo-measurements.

## APPENDIX A
## THE ITERIEKF DERIVATION

### A. Standard case: noisy measurements

Akin to the conventional iterated EKF [2], we intend to solve (6) iteratively using the Gauss-Newton method.

**Lemma 2** (from [2])**.** *Consider the optimization problem*

$$\mathbf{x}^\star = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{x}-\bar{\mathbf{x}}\|^2_{\mathbf{A}} + \frac{1}{2}\|\mathbf{b}-\mathbf{h}(\mathbf{x})\|^2_{\mathbf{B}}, \tag{17}$$

*with* $\mathbf{x}, \bar{\mathbf{x}} \in \mathbb{R}^n$*,* $\mathbf{b} \in \mathbb{R}^m$*,* $\mathbf{A} \in GL_n(\mathbb{R})$*,* $\mathbf{B} \in GL_m(\mathbb{R})$ *and* $\mathbf{h} \in C^1(\mathbb{R}^n, \mathbb{R}^m)$*. The Gauss-Newton method applied to* (17) *yields the sequence of estimates*

$$\mathbf{x}^{i+1} = \bar{\mathbf{x}} + \mathbf{K}^i(\mathbf{b} - \mathbf{h}(\mathbf{x}^i) - \mathbf{H}^i(\bar{\mathbf{x}} - \mathbf{x}^i)), \tag{18}$$

*with*

$$\mathbf{H}^i = \mathbf{h}'(\mathbf{x}^i), \tag{19a}$$
$$\mathbf{K}^i = \mathbf{A}(\mathbf{H}^i)^T(\mathbf{H}^i\mathbf{A}(\mathbf{H}^i)^T + \mathbf{B})^{-1}, \tag{19b}$$

*where* $\mathbf{h}'(\mathbf{x}^i)$ *denotes the Jacobian of* $\mathbf{h}$ *evaluated at* $\mathbf{x}^i$*.*

**Lemma 3.** *We have the following first-order expansion*

$$\exp_G(\bar{\boldsymbol{\xi}}+\boldsymbol{\delta})\mathbf{d}_k = \exp_G(\bar{\boldsymbol{\xi}})\mathbf{d}_k + \exp_G(\bar{\boldsymbol{\xi}})\mathbf{H}_k\mathcal{J}_r(\bar{\boldsymbol{\xi}})\boldsymbol{\delta} + \|\boldsymbol{\delta}\|^2,$$

*with* $\mathbf{H}_k$ *as in* (10)*, and where the right Lie Jacobian [30] is defined by* $\exp_G(\boldsymbol{\xi}+\boldsymbol{\delta}) \approx \exp_G(\boldsymbol{\xi})\exp_G(\mathcal{J}_r(\boldsymbol{\xi})\boldsymbol{\delta})$ *neglecting terms of order* $\|\boldsymbol{\delta}\|^2$*, see* (30) *for an expression.*

*Proof.* We have the first-order expansion in $\boldsymbol{\delta}$

$$\exp_G(\bar{\boldsymbol{\xi}}+\boldsymbol{\delta})\mathbf{d}_k \approx \exp_G(\bar{\boldsymbol{\xi}})\exp_G(\mathcal{J}_r(\bar{\boldsymbol{\xi}})\boldsymbol{\delta})\mathbf{d}_k, \tag{20a}$$
$$\approx \exp_G(\bar{\boldsymbol{\xi}})\big(\mathbf{d}_k + \mathcal{L}_{\mathfrak{g}}(\mathcal{J}_r(\bar{\boldsymbol{\xi}})\boldsymbol{\delta})\mathbf{d}_k\big), \tag{20b}$$
$$\approx \exp_G(\bar{\boldsymbol{\xi}})\big(\mathbf{d}_k + \mathbf{H}_k\mathcal{J}_r(\bar{\boldsymbol{\xi}})\boldsymbol{\delta}\big), \tag{20c}$$

where we applied (10) with $\boldsymbol{\xi} = \mathcal{J}_r(\bar{\boldsymbol{\xi}})\boldsymbol{\delta}$. □

Letting $\mathbf{b} := \mathbf{z}_k + \mathbf{d}_k$, $\mathbf{h}(\boldsymbol{\xi}) := \exp_G(\boldsymbol{\xi})\mathbf{d}_k$, and $\bar{\mathbf{x}} = \mathbf{0}$, we see objectives (6) and (17) coincide, and a direct application of Lemma 2 and Lemma 3 yields the following proposition.

**Proposition 4.** *The sequence of GN updates for the optimization problem* (6) *writes*

$$\boldsymbol{\xi}^{i+1}_{k|k-1} = \mathbf{K}^i_k\big(\mathbf{z}_k - \exp_G(\boldsymbol{\xi}^i_{k|k-1})\mathbf{d}_k + \mathbf{d}_k + \mathbf{H}^i_k\boldsymbol{\xi}^i_{k|k-1}\big), \tag{21}$$

*letting* $\mathbf{H}_k$ *be the IEKF standard Jacobian from* (5) *and*

$$\mathbf{H}^i_k = \exp_G(\boldsymbol{\xi}^i_{k|k-1})\mathbf{H}_k\mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1}), \tag{22}$$
$$\mathbf{K}^i_k = \mathbf{P}_{k|k-1}(\mathbf{H}^i_k)^T\left(\mathbf{H}^i_k\mathbf{P}_{k|k-1}(\mathbf{H}^i_k)^T + \hat{\mathbf{N}}_k\right)^{-1}. \tag{23}$$

Assuming the GN method converges after $i^\star$ iterations, the state estimate is updated according to

$$\hat{\boldsymbol{\chi}}_{k|k} = \hat{\boldsymbol{\chi}}_{k|k-1}\exp_G\left(\boldsymbol{\xi}^{i^\star}_{k|k-1}\right). \tag{24}$$

Since Jacobian $\mathbf{H}_k$ is independent from the current estimate, the Riccati update does not require any iteration, and the error covariance is updated once and for all as follows

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}, \tag{25}$$

with $\mathbf{H}_k = \mathbf{H}^0_k$ and $\mathbf{K}_k = \mathbf{K}^0_k$. The full algorithm is summarized in Algorithm 1. Note that we exactly recover the IEKF of [12] if we perform one iteration only.

### *B. Special case: noise-free measurements*

In the limit case of noise-free measurements, i.e., $\mathbf{N}_k = \mathbf{0}$, the innovation covariance becomes $\mathbf{S}_k = \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T$ and can cease to be invertible, in which case the Kalman gain $\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k^T\mathbf{S}_k^{-1}$ is undefined. This occurs when the perfectly observed directions overlap between subsequent updates for example. This issue is solved using the "noise-free limit gain" developed in our preliminary conference paper [34]:

$$\mathbf{K}^{\mathrm{nf}}_k = \lim_{\delta\to 0}\mathbf{P}_{k|k-1}\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \delta\mathbf{I})^{-1}, \tag{26a}$$
$$= \mathbf{L}_{k|k-1}(\mathbf{H}_k\mathbf{L}_{k|k-1})^\dagger, \tag{26b}$$

where $(\cdot)^\dagger$ is the Moore-Penrose pseudo-inverse, $\mathbf{P}_{k|k-1} = \mathbf{L}_{k|k-1}\mathbf{L}^T_{k|k-1}$, and nf stands for "noise-free". This gain is always defined.

Let us analyze the behavior of the proposed IterIEKF as the measurement noise magnitude approaches zero. Let $\hat{\mathbf{N}}_k = \delta\mathbf{I}$, with $\delta \ll 1$. In this context, Problem (6) becomes that of minimizing $f_\delta(\boldsymbol{\xi}) = \frac{1}{2}\|\boldsymbol{\xi}\|^2_{\mathbf{P}_{k|k-1}} + \frac{1}{2}\|\mathbf{z}_k - \exp_G(\boldsymbol{\xi})\mathbf{d}_k + \mathbf{d}_k\|^2_{\delta\mathbf{I}}$. The minimizer of $f_\delta$ may be sought using the GN sequence of estimates of Proposition 4

$$\boldsymbol{\xi}^{i+1}_{k|k-1,\delta} = \mathbf{K}^i_{k,\delta}\cdot \underbrace{\left(\mathbf{z}_k - \exp_G(\boldsymbol{\xi}^i_{k|k-1,\delta})\mathbf{d}_k + \mathbf{d}_k + \mathbf{H}^i_{k,\delta}\boldsymbol{\xi}^i_{k|k-1,\delta}\right)}_{\mathbf{z}^i_{k,\delta}}, \tag{27}$$

where $\mathbf{H}^i_{k,\delta} = \exp_G(\boldsymbol{\xi}^i_{k|k-1,\delta})\mathbf{H}_k\mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1,\delta})$ and where we let $\mathbf{K}^i_{k,\delta} = \mathbf{P}_{k|k-1}(\mathbf{H}^i_{k,\delta})^T\left(\mathbf{H}^i_{k,\delta}\mathbf{P}_{k|k-1}(\mathbf{H}^i_{k,\delta})^T + \delta\mathbf{I}\right)^{-1}$. Starting from $\boldsymbol{\xi}^0_{k|k-1,\delta} = \mathbf{0}$ and letting $\delta \to 0$, we get

$$\lim_{\delta\to 0}\boldsymbol{\xi}^{i+1}_{k|k-1,\delta} = \lim_{\delta\to 0}\mathbf{K}^i_{k,\delta}\cdot\lim_{\delta\to 0}\mathbf{z}^i_{k,\delta}, \tag{28}$$

so that the GN sequence approximating the solution to the noise-free optimization problem becomes, recalling (26),

$$\boldsymbol{\xi}^{i+1}_{k|k-1} = \mathbf{L}_{k|k-1}(\mathbf{H}^i_k\mathbf{L}_{k|k-1})^\dagger\mathbf{z}^i_k. \tag{29}$$

This provides the limit IterIEKF for noise-free measurements described in Algorithm 2.

## APPENDIX B
## ADAPTATION AND EXTENSION TO THE LG-ITEREKF

When providing the LG-IterEKF with features from invariant filtering, the two key differences between the LG-IterEKF and the IterIEKF lie in their Riccati update: 1) the LG-IterEKF computes $\mathbf{P}_{k|k}$ using $\mathbf{K}^i_k$ and $\mathbf{H}^i_k$, the gain and output Jacobian obtained in the final iteration of the GN method, and 2) it involves $\mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1})$, the right Jacobian of $G$, to adjust the linearization of the exponential map when expanding around $\boldsymbol{\xi}^i_{k|k-1}$ instead of $\mathbf{0}$. This Jacobian writes:

$$\mathcal{L}_{\mathfrak{g}}(\mathcal{J}_r(\boldsymbol{\delta})\boldsymbol{\zeta}) := \sum_{k=0}^{+\infty}\frac{[\ldots[\mathcal{L}_{\mathfrak{g}}(\boldsymbol{\zeta}), \overbrace{\mathcal{L}_{\mathfrak{g}}(\boldsymbol{\delta})], \ldots, \mathcal{L}_{\mathfrak{g}}(\boldsymbol{\delta})}^{k}]}{(k+1)!}. \tag{30}$$

Despite these differences, the following result holds.

**Corollary 1.** *If noise-free measurements* $\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d}_k$ *are considered, uncertainty model* (2) *is used (i.e., a left-invariant*

*error), and potential rank deficiency issues in the Kalman gain are addressed via the noise-free gain from Algorithm 2, then Theorems 1 and 2 hold, replacing IterIEKF with LG-IterEKF.*

*Proof.* As $\mathbf{H}_k \mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1}) = \exp_G(-\boldsymbol{\xi}^i_{k|k-1})\hat{\boldsymbol{\chi}}^{-1}_{k|k-1}\mathbf{H}^i_k$, see line 11 in Algorithm 3, we have the following equivalence:

$$\mathbf{H}_k \mathbf{P}_{k|k} \mathbf{H}_k^T = \mathbf{0} \Leftrightarrow \mathbf{H}^i_k(\mathbf{I} - \mathbf{K}^i_k \mathbf{H}^i_k)\mathbf{P}_{k|k-1}(\mathbf{H}^i_k)^T = \mathbf{0}.$$

The right-hand side of this equivalence follows directly from the same reasoning used in the proof of Theorem 1. Regarding implication (14), the same reasoning as in the proof of Theorem 2 is used to prove $\hat{\boldsymbol{\chi}}_{k|k} \in \mathcal{S}_{\{\boldsymbol{\chi}\bar{\mathbf{d}}_k = \bar{\mathbf{y}}_k\}}$. Let us show now that the equality $\bar{\mathbf{H}}_k \mathbf{P}_{k|k} \bar{\mathbf{H}}_k^T = \mathbf{0}$ also holds. The assumption $\bar{\mathbf{H}}_k \mathbf{P}_{k|k-1} \bar{\mathbf{H}}_k^T = \mathbf{0}$ implies that $\mathcal{L}_{\mathfrak{g}}(\boldsymbol{\xi}) \in \mathfrak{s}$, for all $\boldsymbol{\xi} \in \operatorname{Im} \mathbf{P}_{k|k-1}$, where $\mathfrak{s}$ is the subalgebra of $\mathfrak{g}$ defined in Proposition 2. As the Lie bracket is closed in $\mathfrak{s}$, the expression in (30) is such that $\bar{\mathbf{H}}_k \mathcal{J}_r(\boldsymbol{\delta})\boldsymbol{\zeta} = \mathbf{0}$, for all $\boldsymbol{\delta}, \boldsymbol{\zeta} \in \operatorname{Im} \mathbf{P}_{k|k-1}$. Recalling that $\mathbf{P}_{k|k} = \mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1})(\mathbf{I} - \mathbf{K}^i_k \mathbf{H}^i_k)\mathbf{P}_{k|k-1}\mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1})^T$, that $\boldsymbol{\xi}^i_{k|k-1} \in \operatorname{Im} \mathbf{P}_{k|k-1}$, and that $\operatorname{Im}(\mathbf{I} - \mathbf{K}^i_k \mathbf{H}^i_k)\mathbf{P}_{k|k-1}\mathcal{J}_r(\boldsymbol{\xi}^i_{k|k-1})^T \subseteq \operatorname{Im} \mathbf{P}_{k|k-1}$, we necessarily have $\bar{\mathbf{H}}_k \mathbf{P}_{k|k} = \mathbf{0}$, and thus $\bar{\mathbf{H}}_k \mathbf{P}_{k|k} \bar{\mathbf{H}}_k^T = \mathbf{0}$. □

We stress that this result–a consequence of the Lie subalgebra structure related to the observation (7)–does not, at any rate, extend to the general LG-IterEKF presented in [5].

## REFERENCES


[1] S. F. Schmidt, "The Kalman filter: Its recognition and development for aerospace applications," *Journal of Guidance and Control*, vol. 4, no. 1, pp. 4–7, 1981.

[2] B. M. Bell and F. W. Cathey, "The iterated Kalman filter update as a Gauss-Newton method," *IEEE Transactions on Automatic Control*, vol. 38, no. 2, pp. 294–297, 1993.

[3] J. Zhao, M. Netto, and L. Mili, "A robust iterated extended Kalman filter for power system dynamic state estimation," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 3205–3216, 2016.

[4] M. Bloesch, M. Burri, S. Omari, M. Hutter, and R. Siegwart, "Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1053–1072, 2017.

[5] G. Bourmaud, R. Mégret, A. Giremus, and Y. Berthoumieu, "From intrinsic optimization to iterated extended Kalman filtering on Lie groups," *Journal of Mathematical Imaging and Vision*, vol. 55, pp. 284–303, 2016.

[6] B. Liu, H. Chen, and W. Zhang, "A general iterative extended Kalman filter framework for state estimation on matrix Lie groups," in *2023 62nd IEEE Conference on Decision and Control (CDC)*. IEEE, 2023, pp. 1177–1182.

[7] C. Chahbazian, N. Merlinge, K. Dahia, B. Winter-Bonnet, J. Marini, and C. Musso, "Laplace particle filter on Lie groups applied to angles-only navigation," in *2021 IEEE 24th International Conference on Information Fusion (FUSION)*. IEEE, 2021, pp. 1–8.

[8] N. van Der Laan, M. Cohen, J. Arsenault, and J. R. Forbes, "The invariant Rauch-Tung-Striebel smoother," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5067–5074, 2020.

[9] S. Bonnabel and P. Rouchon, "On invariant observers," *Control and observer design for nonlinear finite and infinite dimensional systems*, pp. 53–65, 2005.

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

[11] M. R. Cohen and J. R. Forbes, "Navigation and control of unconventional VTOL UAVs in forward-flight with explicit wind velocity estimation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1151–1158, 2020.

[12] A. Barrau and S. Bonnabel, "The invariant extended Kalman filter as a stable observer," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2016.

[13] ——, "Invariant Kalman filtering," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 237–257, 2018.

[14] R. Mahony and T. Hamel, "A geometric nonlinear observer for simultaneous localisation and mapping," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 2408–2415.

[15] P. van Goor, R. Mahony, T. Hamel, and J. Trumpf, "A geometric observer design for visual localisation and mapping," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 2543–2549.

[16] R. Mahony, T. Hamel, and J. Trumpf, "An homogeneous space geometry for simultaneous localisation and mapping," *Annual Reviews in Control*, vol. 51, pp. 254–267, 2021.

[17] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended Kalman filtering for robot state estimation," *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020.

[18] N. Pavlasek, A. Walsh, and J. R. Forbes, "Invariant extended Kalman filtering using two position receivers for extended pose estimation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5582–5588.

[19] 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)*, Sep. 2017, pp. 1578–1585.

[20] S. Heo and C. G. Park, "Consistent EKF-based visual-inertial odometry on matrix Lie group," *IEEE Sensors Journal*, vol. 18, no. 9, pp. 3780–3788, May 2018.

[21] M. Wang and A. Tayebi, "Hybrid nonlinear observers for inertial navigation using landmark measurements," *IEEE Transactions on Automatic Control*, vol. 65, no. 12, pp. 5173–5188, 2020.

[22] H. A. Hashim, "GPS-denied navigation: Attitude, position, linear velocity, and gravity estimation with nonlinear stochastic observer," in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 1149–1154.

[23] P. van Goor and R. Mahony, "Autonomous error and constructive observer design for group affine systems," in *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE, 2021, pp. 4730–4737.

[24] A. Barrau, "Non-linear state error based extended Kalman filters with applications to navigation," Ph.D. dissertation, Mines ParisTech, Paris, France, 2015. [Online]. Available: https://theses.hal.science/tel-01344622/

[25] A. Barrau and S. Bonnabel, "The geometry of navigation problems," *IEEE Transactions on Automatic Control*, vol. 68, no. 2, pp. 689–704, 2023.

[26] R. Mahony and J. Trumpf, "Equivariant filter design for kinematic systems on Lie groups," *IFAC-PapersOnLine*, vol. 54, no. 9, pp. 253–260, 2021.

[27] G. Bourmaud, R. Mégret, A. Giremus, and Y. Berthoumieu, "Discrete extended Kalman filter on Lie groups," in *21st European Signal Processing Conference (EUSIPCO 2013)*. IEEE, 2013, pp. 1–5.

[28] D. He, W. Xu, and F. Zhang, "Kalman filters on differentiable manifolds," *arXiv preprint arXiv:2102.03804*, 2021.

[29] W. Xu, Y. Cai, D. He, J. Lin, and F. Zhang, "Fast-lio2: Fast direct lidar-inertial odometry," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2053–2073, 2022.

[30] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 1: Classical Results and Geometric Methods*. Boston, MA: Birkhäuser, 2009.

[31] Y. Wang and G. S. Chirikjian, "Error propagation on the Euclidean group with applications to manipulator kinematics," *IEEE Transactions on Robotics*, vol. 22, no. 4, pp. 591–602, 2006.

[32] K. C. Wolfe, M. Mashner, and G. S. Chirikjian, "Bayesian fusion on Lie groups," *Journal of Algebraic Statistics*, vol. 2, no. 1, 2011.

[33] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, "Continuous-discrete extended Kalman filter on matrix Lie groups using concentrated Gaussian distributions," *Journal of Mathematical Imaging and Vision*, vol. 51, pp. 209–228, 2015.

[34] S. Goffin, S. Bonnabel, O. Brüls, and P. Sacré, "Invariant Kalman filtering with noise-free pseudo-measurements," in *2023 62nd IEEE Conference on Decision and Control (CDC)*, 2023, pp. 8665–8671.

[35] F. Rauscher, S. Nann, and O. Sawodny, "Motion control of an overhead crane using a wireless hook mounted IMU," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 5677–5682.

## 6.3 Epilogue

This second paper provides a principled way to handle the kinematic constraints that arise in rigid-body systems within the invariant Kalman filtering framework. More specifically, assuming that the Gauss–Newton iterations converge, the IterIEKF update incorporates the noise-free information conveyed by such constraints in a globally consistent manner. In particular, for a left-invariant measurement of the form $\mathbf{y}_k = \boldsymbol{\chi}_k \mathbf{d}_k$, the updated estimate $\hat{\boldsymbol{\chi}}_{k|k}$ belongs to the observed subset $\{\boldsymbol{\chi} \mid \boldsymbol{\chi}\mathbf{d}_k = \mathbf{y}_k\}$, and the updated covariance correctly encodes the directions along which uncertainty vanishes, namely $\mathbf{H}_k \boldsymbol{\Sigma}_{k|k} \mathbf{H}_k^T = \mathbf{0}$. The same conclusions hold for right-invariant measurements.

This behavior is illustrated in Figure 6.2, which considers the same system of constraints as in the prologue of this chapter. After the first update yields

$$\hat{\boldsymbol{\chi}}^+_{k|k-1} \in \{\boldsymbol{\chi} \mid \boldsymbol{\chi}\mathbf{d}_k = \mathbf{y}_k\} \quad \text{and} \quad \mathbf{H}_k \boldsymbol{\Sigma}^+_{k|k-1} \mathbf{H}_k^T = \mathbf{0}, \tag{6.3}$$

the second update generates Gauss–Newton iterates of the form $\hat{\boldsymbol{\chi}}^+_{k|k-1} \exp(\boldsymbol{\xi}^i_{k|k-1})$ that all remain on the first observed set. Upon convergence, the updated estimate $\hat{\boldsymbol{\chi}}_{k|k}$ reaches the second observed set and lies in the intersection of both sets.

To build intuition, consider the special case $G = SO(3)$. Suppose the intermediate estimate enforces the noise-free constraint $\boldsymbol{\chi}_k \mathbf{d}_k = \mathbf{y}_k$, so that it satisfies Equation (6.3). In this case, the invariant Jacobian is $\mathbf{H}_k = -(\mathbf{d}_k)_\times$, whose kernel is $\mathrm{span}(\mathbf{d}_k)$. Since $\mathrm{Im}(\boldsymbol{\Sigma}^+_{k|k-1}) \subseteq \ker(\mathbf{H}_k)$, the Gauss–Newton correction can only lie in directions collinear with $\mathbf{d}_k$. As a result, each iterate $\hat{\boldsymbol{\chi}}^+_{k|k-1} \exp(\boldsymbol{\xi}^i_{k|k-1})$ composes $\hat{\boldsymbol{\chi}}^+_{k|k-1}$ with a rotation about the fixed axis $\mathbf{d}_k$. Such rotations leave $\mathbf{d}_k$ invariant, i.e., $\exp(\boldsymbol{\xi}^i_{k|k-1})\mathbf{d}_k = \mathbf{d}_k$, so all iterates remain on $\{\boldsymbol{\chi} \mid \boldsymbol{\chi}\mathbf{d}_k = \mathbf{y}_k\}$:

$$\hat{\boldsymbol{\chi}}^+_{k|k-1} \exp(\boldsymbol{\xi}^i_{k|k-1})\mathbf{d}_k = \hat{\boldsymbol{\chi}}^+_{k|k-1}\mathbf{d}_k = \mathbf{y}_k. \tag{6.4}$$

As noted in the example at the end of Subsection C of Section III of the previous paper, this improves efficiency by performing updates directly within a reduced subspace while preserving the information brought by previous measurements.

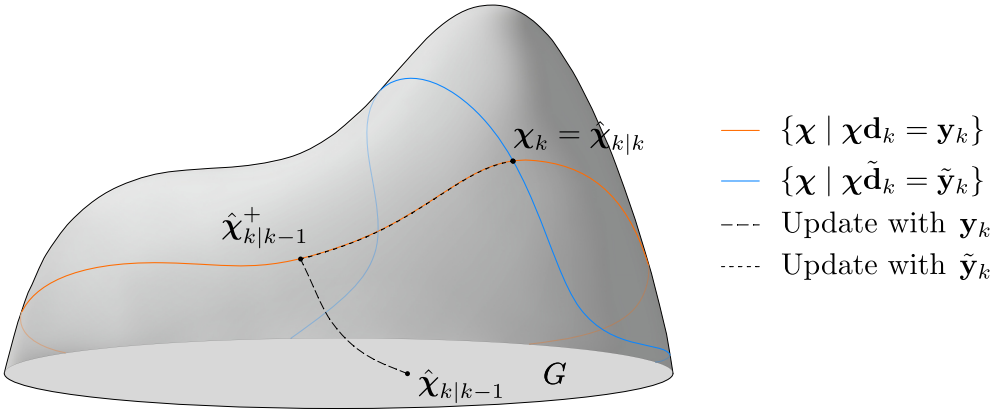



**Figure 6.2:** Illustration of two successive IterIEKF updates with two left-invariant noise-free measurements.

# 7 Extending the invariant framework to rigid-body systems

## Outline

This chapter introduces the *relative L-extended pose*, a representation of the extended pose of rigid-body systems with a kinematic tree structure. Assuming access to angular velocity and specific force measurements on each body, this representation yields group-affine dynamics. Moreover, it allows kinematic constraints associated with many common joints to be expressed as noise-free pseudo-measurements in invariant form. Combined with the IterIEKF, this framework is evaluated on two real-world experiments involving the estimation of the extended pose of a UR5e robotic arm and a human leg.

## 7.1 Prologue

The developments of the previous chapters provide the tools required to incorporate state equality constraints into the invariant filtering framework in a principled and globally consistent manner. The remaining challenge is to exploit these results in the context of rigid-body systems, where kinematic constraints naturally arise from mechanical interconnections between multiple bodies.

While the invariant filtering framework has proved effective for single rigid-body navigation, extending it to articulated systems raises additional modeling difficulties. Beyond the mere presence of joint constraints, the choice of a suitable state representation plays a central role. Such a representation must preserve the geometric structure of rigid-body motion, admit group-affine dynamics, and allow kinematic constraints to be expressed in invariant form. Satisfying all of these requirements simultaneously is nontrivial, and most existing formulations address them only partially.

The following paper makes a decisive step in this direction. By introducing a representation that fulfills these requirements, it enables the application of invariant filtering techniques to articulated rigid-body systems and thus directly addresses the central objective pursued throughout this thesis.

## 7.2 Invariant Kalman filtering for pose estimation in multi-IMU articulated rigid-body systems

This work was carried out in preparation for the 2026 IEEE International Conference on Intelligent Robots and Systems (IROS) and has not yet been published.

### 7.2.1 Author contributions

As the first author, I led the mathematical developments, the experimental work, and the writing of the paper, under the close supervision of Prof. Pierre Sacré and Dr. Olivier Brüls. Prof. Silvère Bonnabel provided guidance on specific theoretical questions as well as on the writing. Dr. Cédric Schwartz trained me in motion capture data acquisition and post-processing, and we jointly carried out the data acquisition for the experiments involving the human leg.

### 7.2.2 Reading tips

In the context of pose estimation for rigid-body systems, notation can quickly become cumbersome. For this reason, the reader is encouraged to first pay particular attention to Subsection B of the introduction, which presents the notation used throughout the paper, before delving into the core material. As in the previous papers, the error covariance matrix is denoted by $\mathbf{P}_{k|l}$ instead of $\mathbf{\Sigma}_{k|l}$.

# Invariant Kalman Filtering for Pose Estimation in Multi-IMU Articulated Rigid-Body Systems

Sven Goffin, Cédric Schwartz, Silvère Bonnabel, Olivier Brüls, and Pierre Sacré

***Abstract*—The invariant extended Kalman filter (IEKF) has become a state-of-the-art method for estimating the extended pose (orientation, velocity, position) of single rigid bodies. However, its application to systems of rigid bodies interconnected by joints has received little attention. This work introduces a Lie group representation, the relative $L$-extended pose, that enables the design of a proper IEKF for rigid-body systems equipped with inertial measurement units (IMUs). The proposed representation yields group-affine dynamics and allows a broad class of kinematic constraints to be written in invariant form. These constraints can then be incorporated as (potentially noise-free) pseudo-measurements using the iterated IEKF (IterIEKF), which preserves the strong convergence and consistency properties of invariant filtering. We evaluate the approach on two real-world systems: a UR5e robot performing a pick-and-place task and a human leg executing forward lunges. In both cases, the IterIEKF based on the proposed representation converges faster and achieves markedly lower estimation variance than the IterIEKF built on absolute poses, as well as all EKF and iterated EKF (IterEKF) baselines using either formulation. These results demonstrate that the relative $L$-extended pose provides a suitable Lie group representation for applying the invariant filtering framework to rigid-body systems.**

## I. Introduction

Pose estimation and motion tracking are central to many robotics problems such as human-robot interaction, imitation learning and balance control for humanoid and legged robots. Beyond robotics, they play a key role in healthcare and biomechanics, driven by advances in gait analysis technologies, assisted rehabilitation methods and wearable exoskeletons. Most motion tracking systems rely on either vision-based technologies or inertial sensors [1]. Vision-based systems provide direct measurements of motion with respect to (w.r.t.) an inertial reference frame, but require bulky external setups and operate only within a limited capture volume. In cluttered or dynamic environments, occlusions further restrict the field of view and reduce robustness. Inertial measurement units (IMUs), in contrast, are small, self-contained, and energy efficient, which makes them well suited for applications involving large scale motion in constrained or open spaces. Their accuracy, however, depends critically on the algorithms used to reconstruct motion from noisy accelerometer and gyroscope data.

S. Goffin is a FRIA grantee of the Fonds de la Recherche Scientifique - FNRS.

S. Goffin and P. Sacré are with the Department of Electrical Engineering and Computer Science, University of Liège, Belgium (sven.goffin@uliege.be; p.sacre@uliege.be).

C. Schwartz is with the Department of Physical Activity and Rehabilitation Sciences, University of Liège, Belgium (Cedric.Schwartz@uliege.be).

S. Bonnabel is with the Department of Mathematics and Systems, Mines Paris – PSL, France (silvere.bonnabel@mines-paristech.fr).

O. Brüls is with the Department of Aerospace and Mechanical Engineering, University of Liège, Belgium (o.bruls@uliege.be).

State estimation in this context is dominated by stochastic filtering methods, which combine predictions from imperfect motion models with noisy, often partial sensor measurements to produce statistically optimal estimates. This paradigm is widely used for rigid-body systems [2]. For more than sixty years, the extended Kalman filter (EKF) has been the standard approach [3], and its many variants remain prevalent in pose estimation for rigid-body systems [4], including exoskeletons [5], legged robots [6], [7], and human motion estimation [8]–[11]. Other approaches include complementary filtering [12], while more advanced alternatives rely on particle filtering [13] or optimization-based methods [14]. However, these algorithms generally neither exploit nor fully respect the geometric structure of rigid-body motion.

Geometric methods have recently gained prominence as principled tools that account for the Lie group structure inherent to pose estimation. They were first developed for single body attitude estimation and inertial navigation [15]–[17]. Within this context, the Invariant Extended Kalman Filter (IEKF) [18], [19] has emerged as a powerful alternative to the standard EKF. It offers strong theoretical guarantees, including convergence results [18] and consistency properties in the presence of unobservability [19]. Other geometric filtering frameworks have also been proposed [20]–[22]. These advances have supported applications in autonomous navigation, legged locomotion, robotics, and aerospace [19], [20], [23]–[27], and motivated the introduction of the Lie group of extended poses (orientation, velocity, position) $SE_2(3)$ [18], [28], which provides a natural representation for single-body inertial navigation [29]. Despite its success in single-body pose estimation, there is no obvious way to apply the IEKF to the problem of estimating the extended pose of rigid-body systems while preserving all its strong properties.

### A. Outline

This work extends the invariant filtering framework to the problem of extended pose estimation for rigid-body systems equipped with IMUs. Section II formulates the problem. Section III introduces a Lie group representation that yields group-affine dynamics and invariant expressions of common joint constraints. Section IV evaluates the approach on two real-world scenarios: a UR5e robot performing a pick-and-place motion and a human leg performing forward lunges. Section V summarizes the findings and discusses limitations and directions for future work.

### B. Notations

Frames are denoted by $\mathcal{F}_i$, with $\mathcal{F}_0$ reserved for the inertial frame. Bodies and joints of articulated systems are indexed

following Featherstone's convention (see Appendix A). Specifically, for each joint $J_j$, we denote by $p(j)$ and $s(j)$ the indices of the predecessor and successor bodies, and by $\boldsymbol{\rho}^j$ and $\boldsymbol{\varsigma}^j$ the calibrated offsets from $\mathcal{F}_{p(j)}$ and $\mathcal{F}_{s(j)}$ to the joint center.

Vectors and matrices follow a structured notation: the bottom-right index indicates the time step, the top-right superscript identifies the frames involved, and the top-left superscript specifies the frame in which the quantity is expressed. In particular, $\mathbf{R}_k^{ij} \in SO(3)$ denotes the rotation from $\mathcal{F}_j$ to $\mathcal{F}_i$, while ${}^l\mathbf{v}_k^{ij}, {}^l\mathbf{p}_k^{ij} \in \mathbb{R}^3$ denote the velocity and position of $\mathcal{F}_j$ with respect to $\mathcal{F}_i$, expressed in $\mathcal{F}_l$. The matrices $\mathbf{I}_N \in \mathbb{R}^{N\times N}$ and $\mathbf{0}_{M\times N} \in \mathbb{R}^{M\times N}$ denote the identity and zero matrix.

We also introduce a compact notation for lifting vector- and matrix-valued functions to block-diagonal operators. Consider the function $\mathbf{f}_{\mathrm{vec}} : \mathbb{R}^n \to \mathbb{R}^{N\times N}$. For any stacked vector $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_L) \in \mathbb{R}^{Ln}$ with $\mathbf{x}_i \in \mathbb{R}^n$, we define

$$\mathbf{f}_{\mathrm{vec}}^{\boxdot}(\mathbf{x}) \coloneqq \mathrm{blkdiag}\left(\mathbf{f}_{\mathrm{vec}}(\mathbf{x}_1), \ldots, \mathbf{f}_{\mathrm{vec}}(\mathbf{x}_L)\right).$$

Likewise, for $\mathbf{f}_{\mathrm{mat}} : \mathbb{R}^{N\times N} \to \mathbb{R}^{N\times N}$ and any block-diagonal matrix $\mathbf{X} = \mathrm{diag}(\mathbf{X}_1, \ldots, \mathbf{X}_L) \in \mathbb{R}^{LN\times LN}$ with $\mathbf{X}_i \in \mathbb{R}^{N\times N}$, we define

$$\mathbf{f}_{\mathrm{mat}}^{\boxdot}(\mathbf{X}) \coloneqq \mathrm{diag}\left(\mathbf{f}_{\mathrm{mat}}(\mathbf{X}_1), \ldots, \mathbf{f}_{\mathrm{mat}}(\mathbf{X}_L)\right).$$

## II. Problem Statement

Consider an articulated rigid-body system composed of $L$ bodies arranged in a kinematic tree, where each body carries a rigidly mounted IMU. An example of such system is depicted in Fig. 1. We assume that each body frame $\mathcal{F}_i$ coincides with its corresponding IMU frame. Without prior knowledge about the motion, the system dynamics are described by the inertial navigation equations. Neglecting IMU biases, this gives

$$\mathbf{R}_{k+1}^{0i} = \mathbf{R}_k^{0i} \exp_{SO(3)}((^i\boldsymbol{\omega}_k^i + \mathbf{w}_k^{i,\boldsymbol{\omega}})dt), \tag{1a}$$

$${}^0\mathbf{v}_{k+1}^{0i} = {}^0\mathbf{v}_k^{0i} + \left(\mathbf{R}_k^{0i}(^i\mathbf{a}_k^i + \mathbf{w}_k^{i,\mathbf{a}}) + {}^0\mathbf{g}\right)dt, \tag{1b}$$

$${}^0\mathbf{p}_{k+1}^{0i} = {}^0\mathbf{p}_k^{0i} + {}^0\mathbf{v}_k^{0i}dt + \left(\mathbf{R}_k^{0i}(^i\mathbf{a}_k^i + \mathbf{w}_k^{i,\mathbf{a}}) + {}^0\mathbf{g}\right)\frac{dt^2}{2}, \tag{1c}$$

where ${}^i\boldsymbol{\omega}_k^i, {}^i\mathbf{a}_k^i \in \mathbb{R}^3$ denote the angular velocity and specific force output by the IMU placed in $\mathcal{F}_i$, and where the process noise $\mathbf{w}_k^i = (\mathbf{w}_k^{i,\boldsymbol{\omega}}, \mathbf{w}_k^{i,\mathbf{a}}) \sim \mathcal{N}(\mathbf{0}_{6\times 1}, \mathbf{Q}_k^i)$ stacks the gyroscope and accelerometer noises. The vector ${}^0\mathbf{g}$ denotes gravity in $\mathcal{F}_0$ and $dt$ is the time step. The system joints impose kinematic constraints that couples the body poses they connect. For the system in Fig. 1, spherical joints give the constraints

$$J_1: \quad {}^0\mathbf{p}_k^{01} + \mathbf{R}_k^{01}\,{}^1\boldsymbol{\varsigma}^1 = {}^0\boldsymbol{\rho}^1, \tag{2a}$$

$$J_2: \quad {}^0\mathbf{p}_k^{02} + \mathbf{R}_k^{02}\,{}^2\boldsymbol{\varsigma}^2 = {}^0\mathbf{p}_k^{01} + \mathbf{R}_k^{01}\,{}^1\boldsymbol{\rho}^2. \tag{2b}$$

We seek to use the invariant filtering framework to estimate the extended pose $\mathbf{X}_k$ of the entire system, namely the orientation, velocity, and position of each body. In our recent work [30], we showed that state constraints such as those in (2) can be considered as noise-free pseudo-measurements within the filtering process. The remaining challenge is to define a matrix Lie group representation of $\mathbf{X}_k$ that satisfies the two assumptions required for the IEKF convergence and consistency properties to hold:

(i) Group-affine dynamics, that is, $\mathbf{X}_{k+1} = \mathbf{f}(\mathbf{X}_k, \mathbf{u}_k, \mathbf{w}_k)$ where $\mathbf{f}(\boldsymbol{\chi}, \mathbf{u}, \mathbf{w}) = \bar{\mathbf{f}}(\boldsymbol{\chi}, \mathbf{u})\mathbf{g}(\mathbf{w}, \mathbf{u})$ up to first order in the process noise $\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$, with $\mathbf{g}(\mathbf{0}, \mathbf{u}) = \mathbf{I}$ and

$$\bar{\mathbf{f}}(\boldsymbol{\mu}\boldsymbol{\nu}, \mathbf{u}) = \bar{\mathbf{f}}(\boldsymbol{\mu}, \mathbf{u})\bar{\mathbf{f}}(\mathbf{I}_5, \mathbf{u})^{-1}\bar{\mathbf{f}}(\boldsymbol{\nu}, \mathbf{u}), \tag{3}$$

for all inputs $\mathbf{u}$ and all states $\boldsymbol{\mu}, \boldsymbol{\nu}$.

(ii) Measurements expressed in an invariant form, that is,

$$\mathbf{y}_k = \mathbf{X}_k\mathbf{d}_k + \mathbf{n}_k \quad \text{or} \quad \mathbf{y}_k = \mathbf{X}_k^{-1}\mathbf{d}_k + \mathbf{n}_k, \tag{4}$$

with $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ and $\mathbf{d}_k$ a known column vector.

See Appendix B for more details.

The literature commonly distinguishes two representations of the extended pose of a rigid-body system. The first is the free-segment model, which assigns each body an independent pose in the inertial frame and enforces kinematic constraints as stochastic constraints [9], [13], [14]. Using the matrix Lie group of extended pose $SE_2(3)$, one defines, for the system in Fig. 1, the extended pose as $\mathbf{X}_k = ({}^0\boldsymbol{\chi}_k^{01}, {}^0\boldsymbol{\chi}_k^{02})$, with

$${}^0\boldsymbol{\chi}_k^{0i} \coloneqq \left[\begin{array}{c|cc} \mathbf{R}_k^{0i} & {}^0\mathbf{v}_k^{0i} & {}^0\mathbf{p}_k^{0i} \\ \hline \mathbf{0}_{2\times 3} & \multicolumn{2}{c}{\mathbf{I}_2} \end{array}\right] \in SE_2(3). \tag{5}$$

Following [31], the dynamics (1) can be written, up to first order in the process noise, as

$${}^0\boldsymbol{\chi}_{k+1}^{0i} = \boldsymbol{\Gamma}\boldsymbol{\Phi}({}^0\boldsymbol{\chi}_k^{0i})\boldsymbol{\Upsilon}(\mathbf{u}_k^i)\exp_{SE_2(3)}(\mathbf{G}_k^i\mathbf{w}_k^i), \tag{6}$$

where $\mathbf{u}_k^i = ({}^i\boldsymbol{\omega}_k^i, {}^i\mathbf{a}_k^i)$ and $\exp_{SE_2(3)}(\cdot)$ denotes the exponential map on $SE_2(3)$. The matrices $\boldsymbol{\Gamma}$, $\boldsymbol{\Phi}({}^0\boldsymbol{\chi}_k^{0i})$, $\boldsymbol{\Upsilon}(\mathbf{u}_k^i) \in \mathbb{R}^{5\times 5}$ and $\mathbf{G}_k^i \in \mathbb{R}^{9\times 6}$ are given in Appendix C. The mapping $\boldsymbol{\Phi}(\cdot)$ is an automorphism, i.e., $\boldsymbol{\Phi}(\boldsymbol{\mu}\boldsymbol{\nu}) = \boldsymbol{\Phi}(\boldsymbol{\mu})\boldsymbol{\Phi}(\boldsymbol{\nu})$ for all $\boldsymbol{\mu}, \boldsymbol{\nu} \in SE_2(3)$, which guarantees group-affine dynamics. However, with this representation the constraint (2b), which depends on the poses of two distinct moving bodies, cannot be expressed in invariant form.

A second common choice is the kinematic tree model, which represents the global pose of a root segment together with the relative poses between adjacent segments [32]–[34], typically in minimal coordinates. This formulation enforces joint constraints by construction, but minimal coordinates introduce singularities and often yield highly nonlinear dynamics that are not group-affine.

**In this paper, we combine the philosophy of the free-segment and kinematic tree models to propose a matrix Lie group representation of the extended pose of rigid-body systems with a kinematic tree structure that does not rely on minimal coordinates and enables the design of a proper IEKF—that is, a representation with group-affine dynamics that allows kinematic constraints to be expressed in invariant form and incorporated as (potentially noise-free) pseudo-measurements.**

## III. Modeling the Extended Pose of Rigid-Body Systems within the Invariant Framework

Although using the free-segment model does not preserve the desired IEKF properties, retaining the Lie group structure

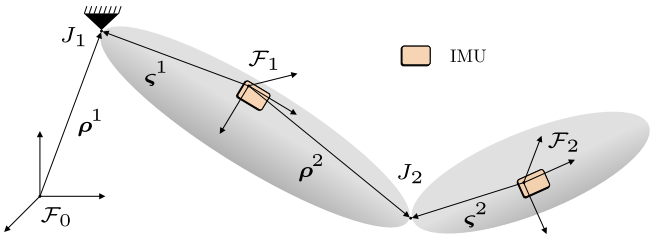



Fig. 1. Example of a rigid-body system with a kinematic tree structure, composed of two movable bodies and a base (the Earth), which is fixed in the inertial frame. An IMU is mounted on each movable body.

by using $SE_2(3)$ as a building block for the system state remains a valuable approach. With this in mind, we define the group of $L$-extended poses as follows.

**Definition 1.** *The Lie group of $L$-extended poses is defined as the set of block-diagonal matrices in $\mathbb{R}^{5L\times 5L}$, composed of $L$ individual elements of $SE_2(3)$:*

$$SE_2^L(3) \coloneqq \left\{ \operatorname{diag}(\boldsymbol{\chi}_1,\dots,\boldsymbol{\chi}_L) \;\middle|\; \begin{array}{c} \boldsymbol{\chi}_i \in SE_2(3), \\ i=1,\dots,L \end{array} \right\}.$$

*The linear map $\mathcal{L}_{SE_2^L(3)} : \mathbb{R}^{9L} \to \mathfrak{se}_2^L(3)$ is defined as*

$$\mathcal{L}_{SE_2^L(3)}(\cdot) \coloneqq \mathcal{L}^{\boxdot}_{SE_2(3)}(\cdot).$$

*Similarly,*

$$\exp_{SE_2^L(3)}(\cdot) \coloneqq \exp^{\boxdot}_{SE_2(3)}(\cdot).$$

While (2a) provides information about the extended pose of $\mathcal{F}_1$ w.r.t. the inertial frame $\mathcal{F}_0$, (2b) restricts the relative pose between two bodies whose extended poses are both being estimated. The representation introduced below is motivated by the relative nature of this second constraint.

**Definition 2** (Relative $L$-extended pose)**.** *The relative $L$-extended pose of a rigid-body system arranged as a kinematic tree is defined as the element $\mathbf{X}_k \in SE_2^L(3)$ given by*

$$\mathbf{X}_k = \operatorname{blkdiag}({}^{p(1)}\boldsymbol{\chi}_k^{p(1)s(1)},\dots,{}^{p(L)}\boldsymbol{\chi}_k^{p(L)s(L)})$$

*where the $j^{\text{th}}$ diagonal block $(\mathbf{X}_k)_j = {}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)}$ is the extended pose of $\mathcal{F}_{s(j)}$ w.r.t. $\mathcal{F}_{p(j)}$, expressed in $\mathcal{F}_{p(j)}$, with $p(j)$ and $s(j)$ the indices of the predecessor and successor bodies of joint $J_j$.*

This definition combines the structural simplicity of relative coordinates with the geometric consistency offered by the matrix Lie group $SE_2^L(3)$. In the following, we show that this formulation leads to group-affine dynamics and enables constraints (2) to both be expressed in invariant form.

### A. Dynamics of the relative L-extended pose

The key step in deriving the dynamics of the relative $L$-extended pose of a rigid-body system is to observe that the relative pose between any two bodies can be written as

$${}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)} = \left({}^{0}\boldsymbol{\chi}_k^{0p(j)}\right)^{-1}\left({}^{0}\boldsymbol{\chi}_k^{0s(j)}\right), \tag{7}$$

for $j=1,\dots,L$. With this observation, the following holds.

**Theorem 1.** *The relative $L$-extended pose of a rigid-body system arranged as a kinematic tree admits the following first-order dynamics approximation w.r.t. the process noise $\mathbf{w}_k$:*

$$\mathbf{X}_{k+1} = \boldsymbol{\Lambda}(\mathbf{X}_k,\mathbf{u}_k^p,\mathbf{u}_k^s)\exp_{SE_2^L(3)}(\mathbf{G}_k\mathbf{w}_k),$$

*where*

$$\begin{aligned}\boldsymbol{\Lambda}(\mathbf{X}_k,\mathbf{u}_k^p,\mathbf{u}_k^s) &= \boldsymbol{\Upsilon}^{\boxdot}(\mathbf{u}_k^p)^{-1}\boldsymbol{\Phi}^{\boxdot}(\mathbf{X}_k)\boldsymbol{\Upsilon}^{\boxdot}(\mathbf{u}_k^s),\\ \mathbf{u}_k^p &= \left(\mathbf{u}_k^{p(1)},\dots,\mathbf{u}_k^{p(L)}\right),\\ \mathbf{u}_k^s &= \left(\mathbf{u}_k^{s(1)},\dots,\mathbf{u}_k^{s(L)}\right),\\ \mathbf{w}_k &= \left(\mathbf{w}_k^1,\dots,\mathbf{w}_k^L\right),\end{aligned}$$

*and where the $j^{\text{th}}$ sub-vector of size 9 in $\mathbf{G}_k\mathbf{w}_k$ is given by*

$$(\mathbf{G}_k\mathbf{w}_k)_j = \mathbf{G}_k^{s(j)}\mathbf{w}_k^{s(j)} - \mathbf{Ad}_{\Delta_j^{-1}}\mathbf{G}_k^{p(j)}\mathbf{w}_k^{p(j)},$$

*with $\Delta_j = \boldsymbol{\Upsilon}(\mathbf{u}_k^{p(j)})^{-1}\boldsymbol{\Phi}({}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)})\boldsymbol{\Upsilon}(\mathbf{u}_k^{s(j)})$. Moreover, the function $\boldsymbol{\Lambda}(\cdot,\mathbf{u}_k^p,\mathbf{u}_k^s)$ satisfies group-affine property* (3)*.*

*Proof.* Injecting (6) into (7), we obtain:

$$\begin{aligned}\left(\mathbf{X}_{k+1}\right)_j =& \exp_{SE_2(3)}(-\mathbf{G}_k^{p(j)}\mathbf{w}_k^{p(j)})\\ &\cdot \boldsymbol{\Upsilon}(\mathbf{u}_k^{p(j)})^{-1}\boldsymbol{\Phi}({}^{0}\boldsymbol{\chi}_k^{0p(j)})^{-1}\boldsymbol{\Gamma}^{-1}\\ &\cdot \boldsymbol{\Gamma}\boldsymbol{\Phi}({}^{0}\boldsymbol{\chi}_k^{0s(j)})\boldsymbol{\Upsilon}(\mathbf{u}_k^{s(j)})\\ &\cdot \exp_{SE_2(3)}(\mathbf{G}_k^{s(j)}\mathbf{w}_k^{s(j)}),\\ =& \exp_{SE_2(3)}(-\mathbf{G}_k^{p(j)}\mathbf{w}_k^{p(j)})\\ &\cdot \boldsymbol{\Upsilon}(\mathbf{u}_k^{p(j)})^{-1}\boldsymbol{\Phi}({}^{0}\boldsymbol{\chi}_k^{0p(j)})^{-1}\\ &\cdot \boldsymbol{\Phi}({}^{0}\boldsymbol{\chi}_k^{0s(j)})\boldsymbol{\Upsilon}(\mathbf{u}_k^{s(j)})\\ &\cdot \exp_{SE_2(3)}(\mathbf{G}_k^{s(j)}\mathbf{w}_k^{s(j)}),\\ =& \exp_{SE_2(3)}(-\mathbf{G}_k^{p(j)}\mathbf{w}_k^{p(j)})\\ &\cdot \boldsymbol{\Upsilon}(\mathbf{u}_k^{p(j)})^{-1}\boldsymbol{\Phi}((\mathbf{X}_k)_j)\boldsymbol{\Upsilon}(\mathbf{u}_k^{s(j)})\\ &\cdot \exp_{SE_2(3)}(\mathbf{G}_k^{s(j)}\mathbf{w}_k^{s(j)}),\end{aligned}$$

where we used $\boldsymbol{\Phi}(\boldsymbol{\mu})^{-1}\boldsymbol{\Phi}(\boldsymbol{\nu}) = \boldsymbol{\Phi}(\boldsymbol{\mu}^{-1}\boldsymbol{\nu})$. Defining $\Delta_j \coloneqq \boldsymbol{\Upsilon}(\mathbf{u}_k^{p(j)})^{-1}\boldsymbol{\Phi}((\mathbf{X}_k)_j)\boldsymbol{\Upsilon}(\mathbf{u}_k^{s(j)})$ and using the adjoint $\mathbf{Ad}_{\Delta_j^{-1}}$ from $SE_2(3)$, we obtain:

$$\begin{aligned}\left(\mathbf{X}_{k+1}\right)_j =& \boldsymbol{\Upsilon}(\mathbf{u}_k^{p(j)})^{-1}\boldsymbol{\Phi}((\mathbf{X}_k)_j)\boldsymbol{\Upsilon}(\mathbf{u}_k^{s(j)})\\ &\cdot \exp_{SE_2(3)}(-\mathbf{Ad}_{\Delta_j^{-1}}\mathbf{G}_k^{p(j)}\mathbf{w}_k^{p(j)})\\ &\cdot \exp_{SE_2(3)}(\mathbf{G}_k^{s(j)}\mathbf{w}_k^{s(j)}),\\ =& \boldsymbol{\Upsilon}(\mathbf{u}_k^{p(j)})^{-1}\boldsymbol{\Phi}((\mathbf{X}_k)_j)\boldsymbol{\Upsilon}(\mathbf{u}_k^{s(j)})\\ &\cdot \exp_{SE_2(3)}(\mathbf{G}_k^{s(j)}\mathbf{w}_k^{s(j)} - \mathbf{Ad}_{\Delta_j^{-1}}\mathbf{G}_k^{p(j)}\mathbf{w}_k^{p(j)}),\end{aligned}$$

where the last line follows from a first-order approximation of the Baker–Campbell–Hausdorff formula in $\mathbf{w}_k^{p(j)}$ and $\mathbf{w}_k^{s(j)}$.

Building the matrix $\mathbf{G}_k$ in such a way that $(\mathbf{G}_k\mathbf{w}_k)_j = \mathbf{G}_k^{s(j)}\mathbf{w}_k^{s(j)} - \mathbf{Ad}_{\Delta_j^{-1}}\mathbf{G}_k^{p(j)}\mathbf{w}_k^{p(j)}$ allows us to write:

$$\begin{aligned}\mathbf{X}_{k+1} =& \boldsymbol{\Upsilon}^{\boxdot}(\mathbf{u}_k^p)^{-1}\boldsymbol{\Phi}^{\boxdot}(\mathbf{X}_k)\boldsymbol{\Upsilon}^{\boxdot}(\mathbf{u}_k^s)\\ &\cdot \exp^{\boxdot}_{SE_2(3)}(\mathbf{G}_k\mathbf{w}_k),\\ =& \boldsymbol{\Lambda}(\mathbf{X}_k,\mathbf{u}_k^p,\mathbf{u}_k^s)\exp_{SE_2^L(3)}(\mathbf{G}_k\mathbf{w}_k).\end{aligned}$$

The operator $(\cdot)^{\boxdot}$ preserves automorphisms, and $\mathbf{\Phi}^{\boxdot}$ satisfies $\mathbf{\Phi}^{\boxdot}(\boldsymbol{\mu}\boldsymbol{\nu}) = \mathbf{\Phi}^{\boxdot}(\boldsymbol{\mu})\mathbf{\Phi}^{\boxdot}(\boldsymbol{\nu})$ for all $\boldsymbol{\mu}, \boldsymbol{\nu} \in SE_2^L(3)$, yielding

$$\begin{aligned}
\mathbf{\Lambda}(\boldsymbol{\mu}\boldsymbol{\nu}, \mathbf{u}_k^p, \mathbf{u}_k^s) &= \mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^p)^{-1}\mathbf{\Phi}^{\boxdot}(\boldsymbol{\mu}\boldsymbol{\nu})\mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^s), \\
&= \mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^p)^{-1}\mathbf{\Phi}^{\boxdot}(\boldsymbol{\mu})\mathbf{\Phi}^{\boxdot}(\boldsymbol{\nu})\mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^s), \\
&= \mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^p)^{-1}\mathbf{\Phi}^{\boxdot}(\boldsymbol{\mu})\mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^s) \\
&\qquad \cdot \mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^s)^{-1}\mathbf{\Phi}^{\boxdot}(\mathbf{I})^{-1}\mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^p) \\
&\qquad \cdot \mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^p)^{-1}\mathbf{\Phi}^{\boxdot}(\boldsymbol{\nu})\mathbf{\Upsilon}^{\boxdot}(\mathbf{u}_k^s), \\
&= \mathbf{\Lambda}(\boldsymbol{\mu}, \mathbf{u}_k^p, \mathbf{u}_k^s)\mathbf{\Lambda}(\mathbf{I}, \mathbf{u}_k^p, \mathbf{u}_k^s)^{-1}\mathbf{\Lambda}(\boldsymbol{\nu}, \mathbf{u}_k^p, \mathbf{u}_k^s),
\end{aligned}$$

where $\mathbf{\Phi}^{\boxdot}(\mathbf{I}) = \mathbf{I}$. This proves that $\mathbf{\Lambda}(\cdot, \mathbf{u}_k^p, \mathbf{u}_k^s)$ satisfies (3) and is thus group-affine. □

It is noteworthy that gravity has no effect in the dynamics of the $L$-extended pose of a rigid-body system, as the matrix $\mathbf{\Gamma}$ cancels out, as shown in the proof. Consequently, the dynamics of the relative extended pose between two bodies depends solely on their angular velocities and linear accelerations.

### *B. Constraints formulation using the relative L-extended pose*

Consider the physical joint $J_j$ between two bodies of a rigid-body system. The constraints it imposes are holonomic, that is, equalities relating the positions and orientations of the two bodies and possibly time. The next result specifies the structure that these constraints, when expressed in the inertial frame, must satisfy in order to admit an invariant form under the relative $L$-extended pose representation.

**Theorem 2.** *Let a holonomic constraint enforced by joint $J_j$ between bodies $p(j)$ and $s(j)$ be given by*

$$\boldsymbol{\phi}_k\left({}^0\boldsymbol{\chi}_k^{0p(j)}, {}^0\boldsymbol{\chi}_k^{0s(j)}\right) = \mathbf{0}. \tag{9}$$

*If the function $\boldsymbol{\phi}_k$ can be written as*

$$\begin{aligned}
\boldsymbol{\phi}_k\left({}^0\boldsymbol{\chi}_k^{0p(j)}, {}^0\boldsymbol{\chi}_k^{0s(j)}\right) &= \mathbf{R}_k^{0p(j)}\mathbf{a}_k + \mathbf{R}_k^{0s(j)}\mathbf{b}_k \\
&\quad + \alpha_k\left({}^0\mathbf{p}_k^{0p(j)} - {}^0\mathbf{p}_k^{0s(j)}\right),
\end{aligned} \tag{10}$$

*where $\mathbf{a}_k, \mathbf{b}_k \in \mathbb{R}^3$ and $\alpha_k \in \mathbb{R}$ are known parameters independent of ${}^0\boldsymbol{\chi}_k^{0p(j)}$ and ${}^0\boldsymbol{\chi}_k^{0s(j)}$, then the constraint admits an invariant formulation based on the $j^{\text{th}}$ diagonal block $(\mathbf{X}_k)_j \in SE_2(3)$ of the relative L-extended pose $\mathbf{X}_k$, namely*

$$(\mathbf{X}_k)_j\mathbf{d}_k = \mathbf{y}_k, \tag{11}$$

*where $\mathbf{d}_k = (\mathbf{b}_k, 0, \alpha_k)$ and $\mathbf{y}_k = (\mathbf{a}_k, 0, \alpha_k)$.*

*Proof.* Left multiplying both sides of (9) by $(\mathbf{R}_k^{0p(j)})^T$, recalling (7), and rearranging terms give

$$\mathbf{R}_k^{p(j)s(j)}\mathbf{b}_k + \alpha_k\,{}^{p(j)}\mathbf{p}_k^{p(j)s(j)} = \mathbf{a}_k \quad \Leftrightarrow \quad (\mathbf{X}_k)_j\mathbf{d}_k = \mathbf{y}_k,$$

where the last two rows of the final equality do not carry information about the system state. □

In some cases, it is more convenient to express the joint constraint directly in terms of the relative pose between the two connected bodies. The next corollary provides the counterpart of Theorem 2 in this setting.

**Corollary 1.** *Let a holonomic constraint enforced by joint $J_j$ between bodies $p(j)$ and $s(j)$ be given by*

$$\boldsymbol{\phi}_k\left({}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)}\right) = \mathbf{0}. \tag{12}$$

*This constraint admits an invariant form if the function $\boldsymbol{\phi}_k$ can be expressed as*

$$\boldsymbol{\phi}_k\left({}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)}\right) = \mathbf{R}_k^{p(j)s(j)}\mathbf{b}_k + \alpha_k\,{}^{p(j)}\mathbf{p}_k^{p(j)s(j)} - \mathbf{a}_k, \tag{13}$$

*where parameters $\mathbf{a}_k, \mathbf{b}_k \in \mathbb{R}^3$ and $\alpha_k \in \mathbb{R}$ are known and independent of ${}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)}$.*

*Proof.* This follows directly from the proof of Theorem 2. □

It is important to note that constraints of the form (11) remove either two rotational or three translational relative degrees of freedom, depending on whether $\alpha_k$ is zero. To gain finer control over the number of constrained dimensions, such a constraint can be relaxed in the direction of the unit vector ${}^{p(j)}\mathbf{r}$ by introducing the following noisy pseudo-measurement:

$$(\mathbf{X}_k)_j\mathbf{d}_k = \mathbf{y}_k + \mathbf{n}_k^j, \tag{14}$$

where $\mathbf{n}_k^j \sim \mathcal{N}(\mathbf{0}_{3\times 1}, \mathbf{N}^j)$ with $\mathbf{N}^j = \sigma({}^{p(j)}\mathbf{r})({}^{p(j)}\mathbf{r})^T$. The parameter $\sigma \geq 0$ controls the degree of relaxation, with larger values producing a weaker constraint.

**Remark 1.** *Enforcing the noisy pseudo-measurement* (14) *with the IterIEKF requires a minor modification to Algorithm 1 in [35], where the measurement noise covariance $\mathbf{N}_k$ is implicitly assumed to be isotropic. Since $\mathbf{N}_k$ is rank-deficient, this assumption no longer holds, and the Riccati update at line 19 does not correctly encode the directions along which the constraint is relaxed. The issue originates at line 18, where the transformed covariance $\hat{\mathbf{N}}_k = \hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{N}_k(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1})^T$ is used to compute the Kalman gain. Because the update moves the estimate from $\hat{\boldsymbol{\chi}}_{k|k-1}$ to $\hat{\boldsymbol{\chi}}_{k|k}$, the relevant directions are those induced by $\hat{\boldsymbol{\chi}}_{k|k}^{-1}\mathbf{N}_k(\hat{\boldsymbol{\chi}}_{k|k}^{-1})^T$, not by $\hat{\boldsymbol{\chi}}_{k|k-1}^{-1}\mathbf{N}_k(\hat{\boldsymbol{\chi}}_{k|k-1}^{-1})^T$. Therefore, we replace line 18 in Algorithm 1 of [35] with*

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \hat{\boldsymbol{\chi}}_{k|k}^{-1}\mathbf{N}_k(\hat{\boldsymbol{\chi}}_{k|k}^{-1})^T\right)^{-1}.$$

Both constraints in (2) imposed by spherical joints follow the form (10). Using the relative $L$-extended pose therefore yields kinematic constraints expressed in invariant form. Many other joint types exist, each imposing its own set of constraints. Table I gathers most common ones together with the corresponding constraints expressed in invariant form. Note that in the prismatic joint case the last two constraints both restrict the relative orientation. Although each one independently removes two rotational degrees of freedom, together they provide only three independent equations and thus remove three relative degrees of freedom in total.

**Remark 2.** *This subsection focused on holonomic constraints since these are the constraints imposed by physical joints. Nevertheless, Theorem 2 and Corollary 1 can be adapted to*

*nonholonomic constraints. In that case, Equations* (10) *and* (13) *respectively become*

$$\begin{aligned}\boldsymbol{\phi}_k\left({}^{0}\boldsymbol{\chi}_k^{0p(j)},\,{}^{0}\boldsymbol{\chi}_k^{0s(j)}\right) &= \mathbf{R}_k^{0p(j)}\mathbf{a}_k + \mathbf{R}_k^{0s(j)}\mathbf{b}_k \\ &\quad + \alpha_k\left({}^{0}\mathbf{p}_k^{0p(j)} - {}^{0}\mathbf{p}_k^{0s(j)}\right) \\ &\quad + \beta_k\left({}^{0}\mathbf{v}_k^{0p(j)} - {}^{0}\mathbf{v}_k^{0s(j)}\right),\end{aligned} \tag{15}$$

$$\begin{aligned}\boldsymbol{\phi}_k\left({}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)}\right) &= \mathbf{R}_k^{p(j)s(j)}\mathbf{b}_k \\ &\quad + \alpha_k\,{}^{p(j)}\mathbf{p}_k^{p(j)s(j)} \\ &\quad + \beta_k\,{}^{p(j)}\mathbf{v}_k^{p(j)s(j)} - \mathbf{a}_k,\end{aligned} \tag{16}$$

*where* $\mathbf{a}_k, \mathbf{b}_k \in \mathbb{R}^3$ *and* $\alpha_k, \beta_k \in \mathbb{R}$ *are known parameters that do not depend on* ${}^{0}\boldsymbol{\chi}_k^{0p(j)}$, ${}^{0}\boldsymbol{\chi}_k^{0s(j)}$ *or* ${}^{p(j)}\boldsymbol{\chi}_k^{p(j)s(j)}$*, but may depend on the angular velocities* ${}^{p(j)}\boldsymbol{\omega}_k^{p(j)}$ *and* ${}^{s(j)}\boldsymbol{\omega}_k^{s(j)}$*. If such dependence exists, the IMU noise must be accounted for by adding a noise term to the pseudo-measurement.*

## IV. Real Experiments

We consider two rigid-body extended pose estimation tasks to evaluate the performance of an IterIEKF, an EKF, and an iterated EKF (IterEKF). For each filter, the system pose is represented either as the relative $L$-extended pose or as the set of individual body poses w.r.t. the base frame $\mathcal{F}_0$ (free-segment model), referred to as the relative and absolute representations, respectively. For clarity, quantities expressed in the absolute representation are denoted with a bar.

The IterIEKF is formulated using the left-invariant error:

$$(\boldsymbol{\xi}_{k|l})_j = \log_{SE_2(3)}\left((\hat{\mathbf{X}}_{k|l})_j^{-1}(\mathbf{X}_k)_j\right), \tag{17}$$

$$(\bar{\boldsymbol{\xi}}_{k|l})_j = \log_{SE_2(3)}\left({}^{0}\hat{\boldsymbol{\chi}}_{k|l}^{0s(j)})^{-1}\;{}^{0}\boldsymbol{\chi}_k^{0s(j)}\right). \tag{18}$$

In contrast, the EKF and IterEKF use the error:

$$(\mathbf{e}_{k|l})_j = \begin{bmatrix}\log_{SO(3)}\left((\hat{\mathbf{R}}_{k|l}^{p(j)s(j)})^T\mathbf{R}_k^{p(j)s(j)}\right) \\ {}^{p(j)}\mathbf{v}_k^{p(j)s(j)} - {}^{p(j)}\hat{\mathbf{v}}_{k|l}^{p(j)s(j)} \\ {}^{p(j)}\mathbf{p}_k^{p(j)s(j)} - {}^{p(j)}\hat{\mathbf{p}}_{k|l}^{p(j)s(j)}\end{bmatrix}, \tag{19}$$

$$(\bar{\mathbf{e}}_{k|l})_j = \begin{bmatrix}\log_{SO(3)}\left((\hat{\mathbf{R}}_{k|l}^{0s(j)})^T\mathbf{R}_k^{0s(j)}\right) \\ {}^{0}\mathbf{v}_k^{0s(j)} - {}^{0}\hat{\mathbf{v}}_{k|l}^{0s(j)} \\ {}^{0}\mathbf{p}_k^{0s(j)} - {}^{0}\hat{\mathbf{p}}_{k|l}^{0s(j)}\end{bmatrix}. \tag{20}$$

We model all error terms, $\boldsymbol{\xi}_{k|l}$, $\bar{\boldsymbol{\xi}}_{k|l}$, $\mathbf{e}_{k|l}$, and $\bar{\mathbf{e}}_{k|l}$, as zero-mean Gaussian, which induces distinct state distributions. To ensure a fair comparison, all filters are initialized from the same state—expressed in either absolute or relative form—and with covariance matrices chosen to make the resulting state distributions equal up to the first order. Details on how these covariances are computed are provided in Appendix D.

In the following, the joint kinematic constraints are divided into positional and rotational constraints and treated as noise-free pseudo-measurements. In practice, a small regularization measurement noise is introduced to avoid numerical issues in the computation of the Kalman gain. The corresponding standard deviations are reported in Tables II and III under the entries "Pos. constraint noise along each axis (std)" and "Rot. constraint noise along each axis (std)".

### A. UR5e robot extended pose estimation

The first task involves estimating the extended pose of a UR5e robot. The UR5e consists of a fixed base and six movable rigid bodies arranged in a kinematic chain, numbered as in Subsection A. All joints are hinges. Four Awinda (Xsens) IMUs are mounted on the first four movable bodies, while the last two bodies are held fixed relative to their parents due to the limited number of IMUs available in the laboratory. The setup is shown in Figure 2.

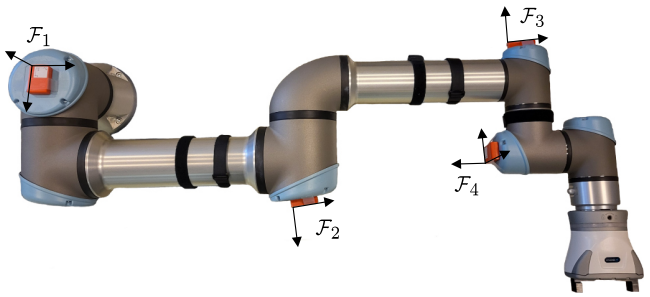



Fig. 2. UR5e robot equipped with four IMUs.

The ground-truth trajectory corresponds to the motion measured by the robot joint encoders during an arbitrary pick-and-place sequence. IMU biases are estimated offline from static measurements performed immediately prior to the experiment and subsequently subtracted from the inertial data recorded during the trials. The vectors ${}^{p(j)}\boldsymbol{\rho}^j$, ${}^{s(j)}\boldsymbol{\varsigma}^j$, ${}^{p(j)}\boldsymbol{\theta}^j$, and ${}^{s(j)}\boldsymbol{\theta}^j$ are accurately obtained through calibration.

All filters are initialized with the parameters listed in Table II, and their error covariance matrices are constructed as described in Appendix D. To ensure a fair comparison, each filter is evaluated over 100 trials using the same ground-truth trajectory, with initial states randomly sampled from the IterIEKF relative representation distribution.

Figure 3 shows the mean and one standard deviation of the rotational, velocity, and position errors for each of the four IMUs mounted on the robot. For consistency, all errors are expressed using the absolute representation of the IterIEKF. Because rotation around the gravity vector is unobservable, this component is corrected at every timestep in all filters prior to evaluation.

Several preliminary observations are in order. First, the orientation error of the first IMU remains zero throughout the sequence. This follows from the manual correction in the unobservable direction: since the joint axis $\boldsymbol{\theta}^1$ is collinear with the gravity vector $\mathbf{g}$, the orientation error of IMU 1 lies entirely within this unobservable subspace. Then, the initial velocity error is zero for all filters. This follows from the ground-truth trajectory, which begins at rest with zero velocity. As a result, all IMUs have zero angular velocity at initialization, and no uncertainty arises in the initial velocity under the chosen covariance initialization.

With these preliminaries in mind, we now compare the performance of the different filters. The IterIEKF in relative representation consistently outperforms all alternatives, regardless of IMU or error type. It converges much faster than the other filters and exhibits almost no variance across estimates,

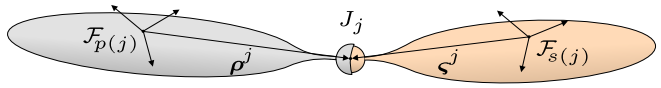
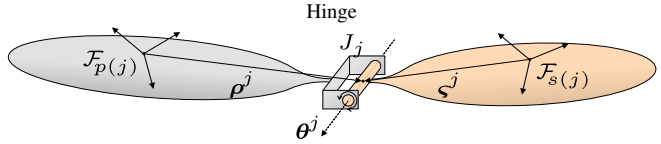
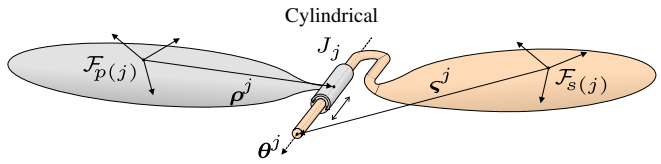
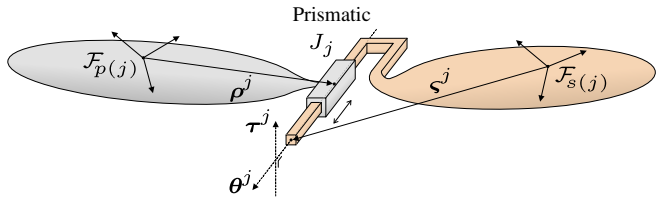
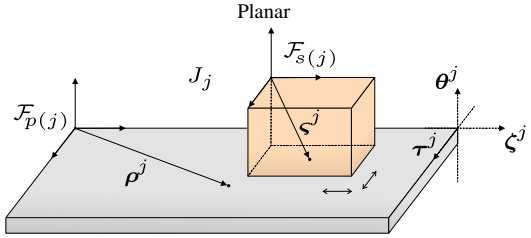
| Joint type | Constraints in invariant form |
|---|---|
| Spherical  | $\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\rho}^j \\ 0 \\ 1 \end{bmatrix}$ |
| Hinge  | $\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\rho}^j \\ 0 \\ 1 \end{bmatrix}$<br>$\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix}$ |
| Cylindrical  | $\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\rho}^j \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{n}_k^j \\ 0 \\ 0 \end{bmatrix}$<br>$\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix}$<br>$\mathbf{n}_k^j \sim \mathcal{N}(\mathbf{0}_{3\times 1}, \mathbf{N}_k^j)$<br>$\mathbf{N}_k^j = \sigma({}^{p(j)}\boldsymbol{\theta}^j)({}^{p(j)}\boldsymbol{\theta}^j)^T$ |
| Prismatic  | $\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\rho}^j \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{n}_k^j \\ 0 \\ 0 \end{bmatrix}$<br>$\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix}$<br>$\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\tau}^j \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\tau}^j \\ 0 \\ 0 \end{bmatrix}$<br>$\mathbf{n}_k^j \sim \mathcal{N}(\mathbf{0}_{3\times 1}, \mathbf{N}_k^j)$<br>$\mathbf{N}_k^j = \sigma({}^{p(j)}\boldsymbol{\theta}^j)({}^{p(j)}\boldsymbol{\theta}^j)^T$ |
| Planar  | $\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\rho}^j \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} \mathbf{n}_k^j \\ 0 \\ 0 \end{bmatrix}$<br>$\left(\mathbf{X}_k\right)_j \begin{bmatrix} {}^{s(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} {}^{p(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix}$<br>$\mathbf{n}_k^j \sim \mathcal{N}(\mathbf{0}_{3\times 1}, \mathbf{N}_k^j)$<br>$\mathbf{N}_k^j = \sigma[{}^{p(j)}\boldsymbol{\tau}^j \quad {}^{p(j)}\boldsymbol{\zeta}^j][{}^{p(j)}\boldsymbol{\tau}^j \quad {}^{p(j)}\boldsymbol{\zeta}^j]^T$ |

TABLE I

COMMON JOINT CONSTRAINTS EXPRESSED USING THE RELATIVE $L$-EXTENDED POSE. THE VECTORS ${}^{p(j)}\boldsymbol{\rho}^j$, ${}^{s(j)}\boldsymbol{\varsigma}^j$, ${}^{p(j)}\boldsymbol{\theta}^j$, ${}^{s(j)}\boldsymbol{\theta}^j$, ${}^{p(j)}\boldsymbol{\tau}^j$, ${}^{s(j)}\boldsymbol{\tau}^j$, ${}^{p(j)}\boldsymbol{\zeta}^j \in \mathbb{R}^3$ ARE KNOWN FROM CALIBRATION. THE ELEMENT $(\mathbf{X}_k)_j \in SE_2(3)$ DENOTES THE $j^{\text{th}}$ DIAGONAL BLOCK OF $\mathbf{X}_k \in SE_2^L(3)$.

| Parameter | Value |
|---|---|
| Orientation error around joint axis (std.) | $\pi/6\,\text{rad}$ |
| Pos. constraint noise along each axis (std.) | $10^{-2.5}\,\text{m}$ |
| Rot. constraint noise along each axis (std.) | $10^{-2.5}\,\text{rad}$ |
| Gyroscope noise along each axis (std.) | $10^{-2}\,\text{rad}\,\text{s}^{-1}$ |
| Accelerometer noise along each axis (std.) | $10^{-1}\,\text{m}\,\text{s}^{-2}$ |
| IMU frequency | $100\,\text{Hz}$ |

TABLE II

FILTER INITIALIZATION AND NOISE PARAMETERS FOR THE UR5E EXPERIMENT.

indicating robust convergence. By contrast, the IterIEKF in absolute representation and the other filters show broadly similar behavior, with relatively slow average convergence in both orientation and position errors across all IMUs. Their velocity error even increases sharply at the beginning before gradually decreasing. In comparison, the IterIEKF in relative representation displays only a small transient peak in velocity error, after which it remains close to zero. This peak arises from non-zero orientation and position errors: although the filter encodes no uncertainty in the relative velocity between adjacent bodies, converting from relative to absolute representation causes errors in relative rotation and position to propagate into the absolute velocity estimate.

### *B. Human leg extended pose estimation*

The second task involves estimating the extended pose of a human leg during a forward lunge exercise. Trigno IMUs (Delsys) were placed on the participant's thigh and shank.

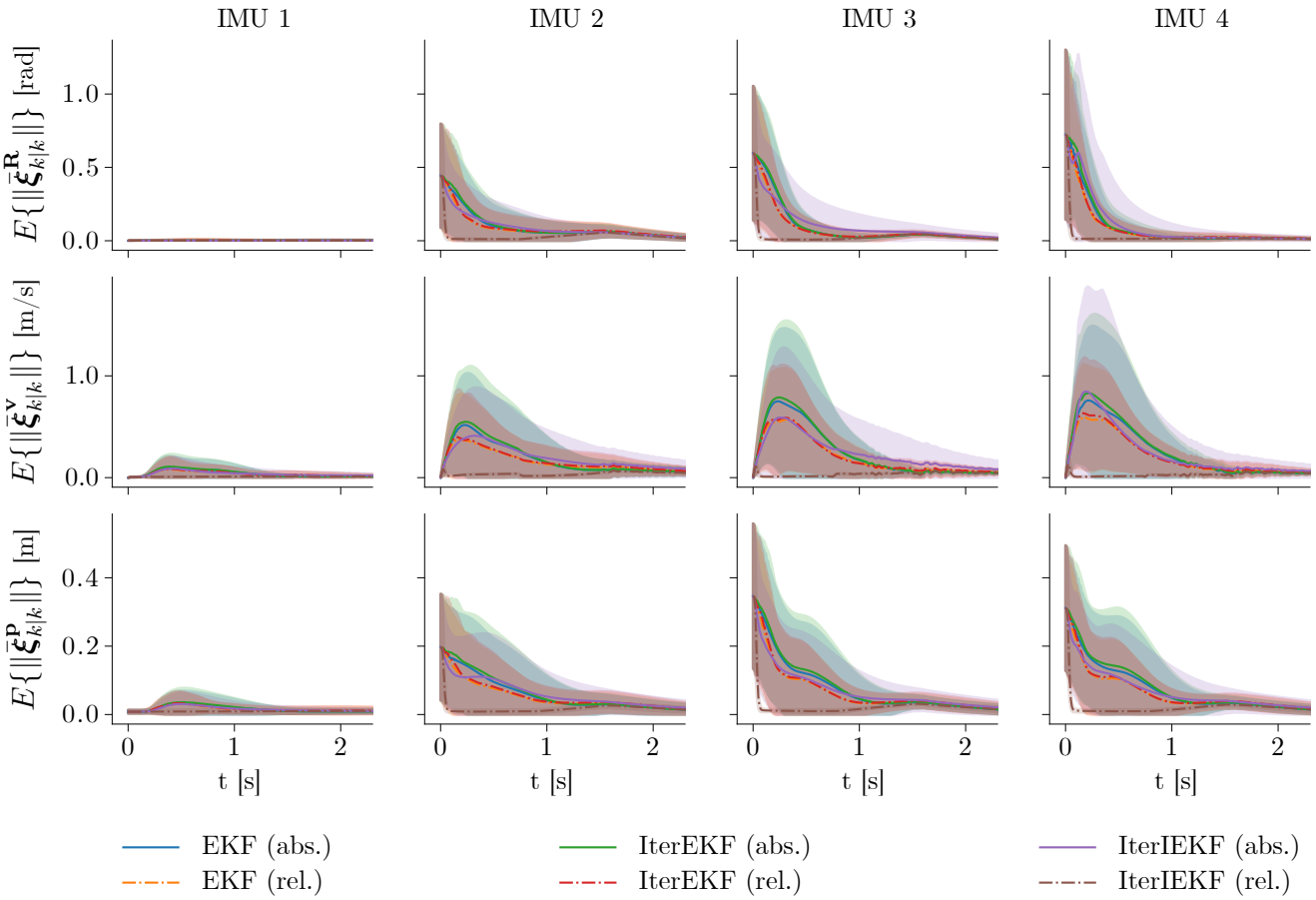



Fig. 3. Mean and one standard deviation of the rotational, velocity, and position errors over time, computed from 100 estimations of the extended pose using the four IMUs mounted on the UR5e robot.

Ground-truth pose estimates for each IMU were obtained using a Qualisys motion capture system with nine cameras. Reflective markers were positioned on the medial and lateral malleoli, the medial and lateral femoral epicondyles, the greater trochanter, the right and left anterior superior iliac spines, and the right and left posterior superior iliac spines. In addition, a frame equipped with reflective markers was rigidly attached to each IMU in order to retrieve its absolute position and orientation. The setup is depicted in Figure 4

The ankle and knee joint centers were defined as the midpoints between the medial and lateral malleoli markers and between the medial and lateral epicondyle markers, respectively. Both joints were modeled as hinges. The ankle joint axis was defined as the average direction of the line connecting the medial and lateral malleoli over a calibration motion sequence, and the knee joint axis as the average direction of the line connecting the medial and lateral epicondyles over the same sequence. The hip joint center was estimated using the Harrington regression equations based on pelvic width, pelvic depth, and leg length.

All filters are initialized with the parameters listed in Table III, and their error covariance matrices are defined as described in Appendix D. We perform 100 estimations of the same ground-truth trajectory, each initialized with a random

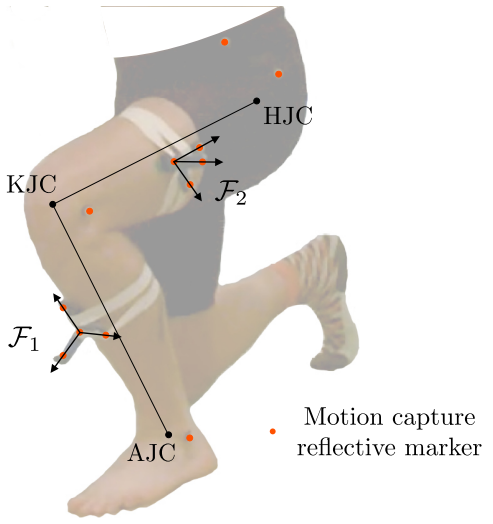



Fig. 4. Forward lunge exercise with IMUs mounted on the thigh and shank. The ankle and knee are modeled as hinge joints, with the ankle assumed fixed in the inertial frame. Ground-truth trajectories of the hip, knee, and ankle joint centers (HJC, KJC, and AJC) are computed from eight motion capture reflective markers placed on the body. Ground-truth IMU position and orientation trajectories are obtained using 3D-printed frames equipped with motion capture reflective markers.

sample from the initial state distribution of the IterIEKF in relative representation, ensuring that all filters operate under comparable conditions. IMU biases are optimized beforehand using the ground-truth orientation and position trajectories and subsequently subtracted from the recorded IMU outputs. Throughout the sequence, the ankle joint center is assumed fixed in the inertial frame, which, together with the hinge joint assumption, makes the estimation problem fully observable. The ankle is designated as joint $J_1$, and the knee as joint $J_2$. The vectors ${}^{p(j)}\boldsymbol{\rho}^j$, ${}^{s(j)}\boldsymbol{\varsigma}^j$, ${}^{p(j)}\boldsymbol{\theta}^j$, and ${}^{s(j)}\boldsymbol{\theta}^j$ for $j = 1, 2$ are obtained from motion-capture ground truth. Because soft-tissue artifacts induce small IMU displacements relative to the ankle and knee joints, these vectors are actually not constant over the entire sequence. While this has little effect on position constraints, it is more problematic for rotation constraints: even small errors in ${}^{p(j)}\boldsymbol{\theta}^j$ or ${}^{s(j)}\boldsymbol{\theta}^j$ are amplified by the lever arm between the joint and the body frame, leading to larger position estimation errors. To mitigate this effect, the rotation constraints are relaxed by increasing their noise standard deviation relative to the previous experiment.

| Parameter | Value |
|---|---|
| Orientation error around joint axis (std.) | $\pi/6\text{rad}$ |
| Pos. constraint noise along each axis (std.) | $10^{-2.5}\text{m}$ |
| Rot. constraint noise along each axis (std.) | $10^{-0.5}\text{rad}$ |
| Gyroscope noise along each axis (std.) | $8 \cdot 10^{-3}\text{rad}\,\text{s}^{-1}$ |
| Accelerometer noise along each axis (std.) | $5 \cdot 10^{-2}\text{m}\,\text{s}^{-2}$ |
| IMU frequency | 200Hz |

TABLE III
FILTER INITIALIZATION AND NOISE PARAMETERS FOR THE HUMAN LEG EXPERIMENT.

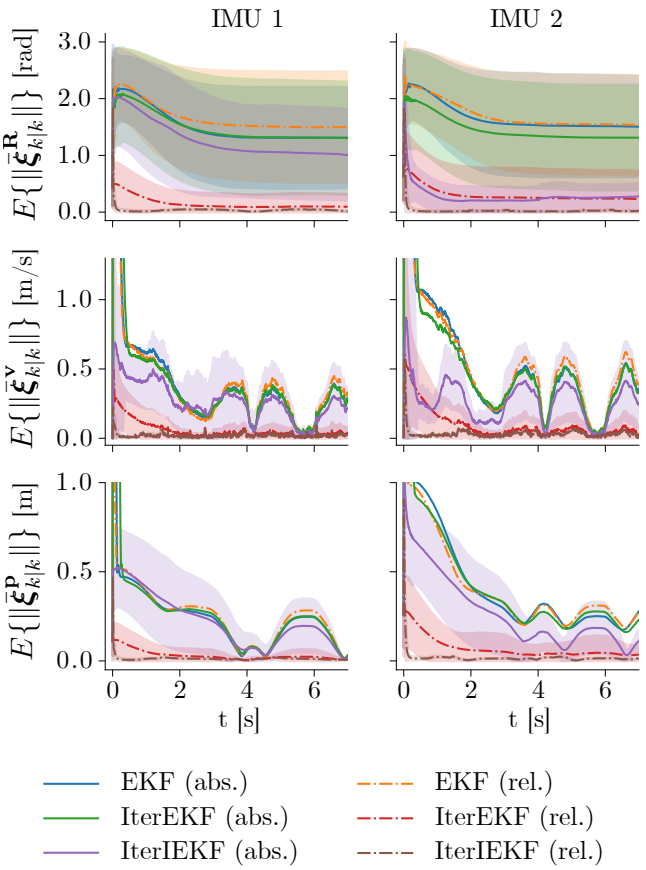



Fig. 5. Mean and one standard deviation of the orientation, velocity, and position errors over time, computed from 100 estimations using IMUs placed on the shank and thigh.

Figure 5 presents the mean and one standard deviation of the rotational, velocity, and position errors for the two IMUs placed on the leg. As before, all errors are expressed in the absolute representation of the IterIEKF.

These results corroborate the previous experiment: the IterIEKF in relative representation consistently outperforms all alternatives. It converges rapidly and shows negligible variability across runs, regardless of the IMU or the estimated quantity. Compared to the UR5e experiment, the transient peak in velocity error is markedly larger for both IMUs, likely due to the larger initial rotation error in this setup.

Among the remaining filters, the IterEKF in relative representation performs second best. Its convergence occurs in two phases—a rapid initial improvement over the first few time steps, followed by a much slower progression. Nonetheless, it never matches the accuracy of the IterIEKF in relative representation, particularly for orientation estimates of both IMUs and for the position of the thigh-mounted IMU, where the average error remains noticeably above zero. In addition, its estimates exhibit significantly greater variance.

The other four filters perform markedly worse. They show no clear convergence and maintain relatively high variance across estimates. All struggle with velocity estimation during motion phases, as evidenced by three distinct peaks in the velocity error that coincide with moments of participant movement. Within this group, the IterIEKF in absolute representation achieves slightly better results than the others, particularly in estimating the extended pose of the thigh-mounted IMU, though it still lags far behind the relative representation filters.

The performance can also be evaluated by examining how accurately the positions of the ankle, knee, and hip joint centers are tracked over time using only the IMU outputs and kinematic constraints. Figure 6 reports the mean and standard deviation of the position error for these three joints, computed over the 100 estimations. It should be noted that the knee position is inferred from the position and orientation of the second IMU, which was mounted on the participant's thigh.

The two bumps observed in the ankle position error for all filters arise from the fact that the ankle joint center is not perfectly fixed in the inertial frame, contrary to the initial assumption. With this in mind, all filters show broadly similar behavior in estimating the ankle joint center, though the IterIEKF in relative representation stands out: its error decreases directly toward zero, whereas the other filters display an initial peak of varying magnitude before converging.

For the knee and hip joint centers, the IterIEKF in relative representation again achieves the best performance, characterized by very fast convergence and minimal variance across estimates. The errors do not converge to zero, as soft tissue artifacts remain unmitigated and the hinge joint assumption is only an approximation. Nevertheless, these effects appear to

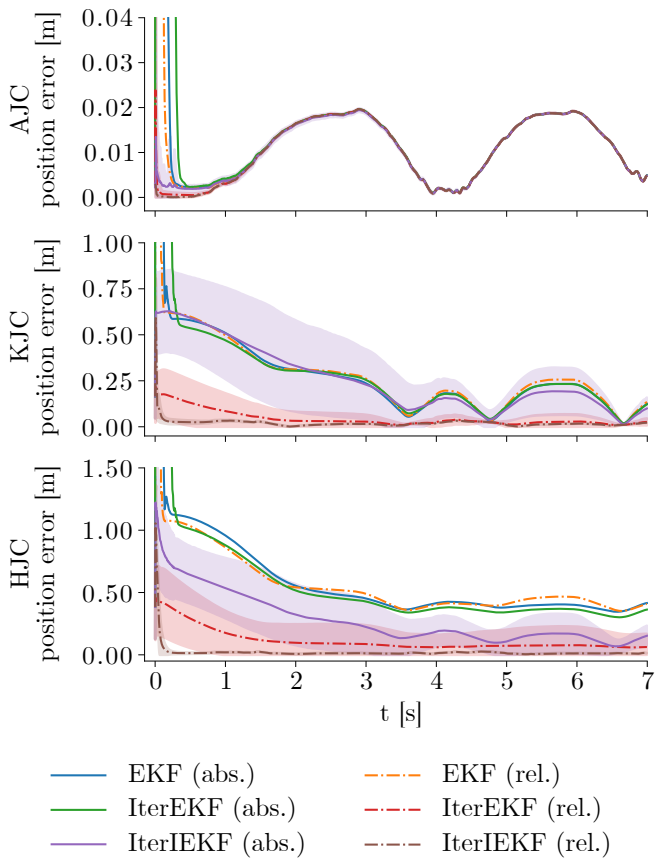



Fig. 6. Mean and standard deviation of position errors for the ankle (AJC), knee (KJC), and hip (HJC) joint centers over time, computed from 100 estimations with IMUs mounted on the shank and thigh.

have only a minor impact on overall accuracy. As previously observed, the IterEKF in relative representation ranks second, showing much slower convergence and higher variance.

The four remaining filters perform substantially worse, with no meaningful convergence. Among them, the IterIEKF in absolute representation achieves the lowest errors, but the residual errors remain relatively large. Finally, it is clear that estimation errors increase progressively deeper into the kinematic chain: errors are smallest near the root and accumulate with distance from this fixed reference point, an expected outcome given the propagation of uncertainty along the chain.

## V. Conclusion

This work introduced a Lie group representation of the extended pose for rigid-body systems with a kinematic tree structure, referred to as the relative $L$-extended pose. When each body is equipped with an IMU, this representation exhibits group-affine dynamics. In addition, it enables many joint kinematic constraints to be expressed in invariant form, allowing them to be seamlessly incorporated as (potentially noise-free) pseudo-measurements in the filtering process. By satisfying these two conditions, the approach preserves the remarkable convergence properties of the invariant framework.

The proposed approach was validated in two experimental setups: estimating the extended pose of a UR5e robot and of a human leg performing forward lunges, both instrumented with IMUs. In both cases, the IterIEKF based on the relative $L$-extended pose consistently outperformed its counterpart using absolute poses, as well as the corresponding EKF and IterEKF formulations. It achieved faster convergence and substantially lower estimation variance. These results further demonstrate the superiority of the IterIEKF over the EKF and IterEKF when measurement noise is negligible, and emphasize the importance of adopting a state representation—such as the relative $L$-extended pose—that enables kinematic constraints to be expressed as pseudo-measurements in invariant form within the invariant filtering framework.

Overall, this work establishes a principled framework for invariant extended-pose estimation in rigid-body systems. Future work will focus on extending the framework to global human pose estimation with IMUs, enabling applications in exoskeleton control and programming-by-demonstration, including pick-and-place learning and gait imitation in humanoid robots. We also plan to investigate motion-dependent compensation strategies to mitigate the effect of soft-tissue artifacts, which remain a major source of error in IMU-based human motion capture. In parallel, we aim to integrate vision-based measurements with inertial data. Such sensor fusion could exploit the complementary strengths of the two modalities, with IMUs providing robust short-term dynamics and cameras offering drift-free global information, thereby advancing markerless motion capture toward greater accuracy, robustness, and practical usability.

## References

[1] M. Field, Z. Pan, D. Stirling, and F. Naghdy, "Human motion capture sensors and analysis in robotics," *Industrial Robot: An International Journal*, vol. 38, no. 2, pp. 163–171, 2011.

[2] M. Á. Naya, E. Sanjurjo, A. J. Rodríguez, and J. Cuadrado, "Kalman filters based on multibody models: linking simulation and real world. a comprehensive review," *Multibody System Dynamics*, vol. 58, no. 3, pp. 479–521, 2023.

[3] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[4] R. Pastorino, D. Richiedei, J. Cuadrado, and A. Trevisani, "State estimation using multibody models and non-linear Kalman filters," *International Journal of Non-Linear Mechanics*, vol. 53, pp. 83–90, 2013.

[5] F. E. Xavier, G. Burger, M. Pétriaux, J.-E. Deschaud, and F. Goulette, "Multi-imu proprioceptive state estimator for humanoid robots," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 10 880–10 887.

[6] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, "State estimation for legged robots-consistent fusion of leg kinematics and IMU," *Robotics*, vol. 17, pp. 17–24, 2013.

[7] N. Rotella, M. Bloesch, L. Righetti, and S. Schaal, "State estimation for a humanoid robot," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 952–958.

[8] A. Filippeschi, N. Schmitz, M. Miezal, G. Bleser, E. Ruffaldi, and D. Stricker, "Survey of motion tracking methods based on inertial sensors: A focus on upper limb human motion," *Sensors*, vol. 17, no. 6, p. 1257, 2017.

[9] D. Roetenberg, H. Luinge, P. Slycke *et al.*, "Xsens MVN: Full 6DOF human motion tracking using miniature inertial sensors," *Xsens Motion Technologies BV, Tech. Rep*, vol. 1, no. 2009, pp. 1–7, 2009.

[10] X. Yun and E. R. Bachmann, "Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking," *IEEE transactions on Robotics*, vol. 22, no. 6, pp. 1216–1227, 2006.

[11] T. L. Baldi, F. Farina, A. Garulli, A. Giannitrapani, and D. Prattichizzo, "Upper body pose estimation using wearable inertial sensors and multiplicative Kalman filter," *IEEE Sensors Journal*, vol. 20, no. 1, pp. 492–500, 2019.

[12] Y. Tian, H. Wei, and J. Tan, "An adaptive-gain complementary filter for real-time human motion tracking with MARG sensors in free-living environments," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 2, pp. 254–264, 2012.

[13] Z.-Q. Zhang and J.-K. Wu, "A novel hierarchical information fusion method for three-dimensional upper limb motion estimation," *IEEE transactions on instrumentation and measurement*, vol. 60, no. 11, pp. 3709–3719, 2011.

[14] M. Kok, J. D. Hol, and T. B. Schön, "An optimization-based approach to human body motion capture using inertial sensors," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 79–85, 2014.

[15] S. Bonnabel and P. Rouchon, "On invariant observers," *Control and observer design for nonlinear finite and infinite dimensional systems*, pp. 53–65, 2005.

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

[17] R. Mahony and J. Trumpf, "Equivariant filter design for kinematic systems on Lie groups," *IFAC-PapersOnLine*, vol. 54, no. 9, pp. 253–260, 2021.

[18] A. Barrau and S. Bonnabel, "The invariant extended Kalman filter as a stable observer," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1797–1812, 2016.

[19] ——, "Invariant Kalman filtering," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, pp. 237–257, 2018.

[20] R. Mahony and T. Hamel, "A geometric nonlinear observer for simultaneous localisation and mapping," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 2408–2415.

[21] P. van Goor, R. Mahony, T. Hamel, and J. Trumpf, "A geometric observer design for visual localisation and mapping," in *2019 IEEE 58th Conference on Decision and Control (CDC)*. IEEE, 2019, pp. 2543–2549.

[22] R. Mahony, T. Hamel, and J. Trumpf, "An homogeneous space geometry for simultaneous localisation and mapping," *Annual Reviews in Control*, vol. 51, pp. 254–267, 2021.

[23] N. van Der Laan, M. Cohen, J. Arsenault, and J. R. Forbes, "The invariant Rauch-Tung-Striebel smoother," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5067–5074, 2020.

[24] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended Kalman filtering for robot state estimation," *The International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020.

[25] N. Pavlasek, A. Walsh, and J. R. Forbes, "Invariant extended kalman filtering using two position receivers for extended pose estimation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 5582–5588.

[26] 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)*, Sep. 2017, pp. 1578–1585.

[27] S. Heo and C. G. Park, "Consistent EKF-based visual-inertial odometry on matrix Lie group," *IEEE Sensors Journal*, vol. 18, no. 9, pp. 3780–3788, May 2018.

[28] A. Barrau, "Non-linear state error based extended Kalman filters with applications to navigation," Ph.D. dissertation, Mines Paristech, 2015.

[29] A. Barrau and S. Bonnabel, "The geometry of navigation problems," *IEEE Transactions on Automatic Control*, vol. 68, no. 2, pp. 689–704, 2023.

[30] S. Goffin, S. Bonnabel, O. Brüls, and P. Sacré, "Invariant Kalman filtering with noise-free pseudo-measurements," in *2023 62nd IEEE Conference on Decision and Control (CDC)*, 2023, pp. 8665–8671.

[31] M. Brossard, A. Barrau, P. Chauchat, and S. Bonnabel, "Associating uncertainty to extended poses for on Lie group IMU preintegration with rotating earth," *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 998–1015, 2021.

[32] G. Bleser, D. Damen, A. Behera, G. Hendeby, K. Mura, M. Miezal, A. Gee, N. Petersen, G. Maçães, H. Domingues *et al.*, "Cognitive learning, monitoring and assistance of industrial workflows using egocentric sensor networks," *PloS one*, vol. 10, no. 6, p. e0127769, 2015.

[33] M. Miezal, G. Bleser, N. Schmitz, and D. Stricker, "A generic approach to inertial tracking of arbitrary kinematic chains," in *Proceedings of the 8th international conference on body area networks*, 2013, pp. 189–192.

[34] T. Seel, J. Raisch, and T. Schauer, "IMU-based joint angle measurement for gait analysis," *Sensors*, vol. 14, no. 4, pp. 6891–6909, 2014.

[35] S. Goffin, A. Barrau, S. Bonnabel, O. Brüls, and P. Sacré, "Iterated invariant extended kalman filter (iteriekf)," *arXiv preprint arXiv:2404.10665*, 2024.

[36] R. Featherstone, *Rigid body dynamics algorithms*. Springer, 2008.

[37] B. C. Hall, "Lie groups, lie algebras, and representations," in *Quantum Theory for Mathematicians*. Springer, 2013, pp. 333–366.

[38] G. S. Chirikjian, *Stochastic Models, Information Theory, and Lie Groups, Volume 1: Classical Results and Geometric Methods*. Springer Science & Business Media, 2009.

[39] ——, *Stochastic models, information theory, and Lie groups, volume 2: Analytic methods and modern applications*. Springer Science & Business Media, 2011, vol. 2.


## APPENDIX A
## RIGID-BODY SYSTEMS

As defined in [36], a rigid-body system is a set of rigid bodies connected by joints that constrain their relative motion. The system topology is described by a connectivity graph, where nodes represent bodies, edges represent joints, and exactly one node corresponds to a fixed body, known as the base. All other nodes represent movable bodies. The connectivity graph is assumed to be connected.

Kinematic trees are rigid-body systems whose connectivity graph is a tree. To facilitate identification of bodies and joints, we adopt the regular indexing scheme:

1) assign the index 0 to the node representing the fixed base,
2) index the remaining nodes from 1 to $L$ so that each node has a higher index than its parent,
3) index the edges from 1 to $L$, with edge $j$ connecting node $j$ to its parent,
4) assign each body and joint the same index as their corresponding node and edge, respectively.

Each body is associated with a frame $\mathcal{F}_i$ labeled by its body index. The base frame $\mathcal{F}_0$ serves as the inertial reference frame.

Each joint $J_j$ connects two bodies. The body with the lower index is referred to as the predecessor, and the other as the successor. Their indices are respectively denoted by $p(j)$ and $s(j)$. We denote by $\boldsymbol{\rho}^j$ and $\boldsymbol{\varsigma}^j$ the vectors from the frames $\mathcal{F}_{p(j)}$ and $\mathcal{F}_{s(j)}$ to joint $j$.

We define the number of relative degrees of freedom as the dimension of the configuration space describing the relative pose (position + orientation) of two rigid bodies. Two unconstrained rigid bodies have thus six relative degrees of freedom. Holonomic constraints imposed by joints reduce this number by the number of independent constraint equations. In contrast, nonholonomic constraints restrict only the set of admissible instantaneous relative velocities; they reduce the dimensionality of the allowable velocity space at each configuration but do not reduce the number of relative degrees of freedom because they do not reduce the dimension of the configuration space itself.

## APPENDIX B
## THE INVARIANT EXTENDED KALMAN FILTER

Let $G \subseteq GL(N;\mathbb{R})$ be a matrix Lie group with associated Lie algebra $\mathfrak{g}$, where $\dim \mathfrak{g} = n$. We denote by

$$\mathcal{L}_{\mathfrak{g}}(\cdot) : \mathbb{R}^n \to \mathfrak{g} \tag{21}$$

the bijective linear map that identifies $\mathfrak{g}$ to $\mathbb{R}^n$. Using this identification, the Lie group exponential map is defined as

$$\exp_G(\cdot) \coloneqq \exp_m(\mathcal{L}_{\mathfrak{g}}(\cdot)), \tag{22}$$

where $\exp_m$ is the standard matrix exponential. We denote by $\log_G(\cdot)$ the logarithm map that satisfies $\log_G(\exp_G(\boldsymbol{\chi})) = \boldsymbol{\chi}$ for all $\boldsymbol{\chi}$ such that $\|\boldsymbol{\chi}\| < \ln(2)$, where $\|\cdot\|$ is the Hilbert–Schmidt norm. When the context is clear, we omit the subscript and write $\mathcal{L}(\cdot)$, $\exp(\cdot)$, and $\log(\cdot)$ for brevity. See [37] for more details about Lie groups and Lie algebras.

Consider the discrete-time dynamical system evolving on $G$, described by

$$\boldsymbol{\chi}_{k+1} = \mathbf{f}(\boldsymbol{\chi}_k, \mathbf{u}_k, \mathbf{w}_k), \tag{23a}$$
$$\mathbf{y}_k = \mathbf{h}(\boldsymbol{\chi}_k) + \mathbf{n}_k, \tag{23b}$$

where $\boldsymbol{\chi}_k \in G$ is the state of the system at time $k$, $\mathbf{u}_k \in \mathbb{R}^b$ is the system input, $\mathbf{y}_k \in \mathbb{R}^m$ is the measurement output, and $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}_{o\times 1}, \mathbf{Q}_k)$ and $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}_{m\times 1}, \mathbf{N}_k)$ denote the process and measurement noise, respectively. The dynamics function $\mathbf{f} : G \times \mathbb{R}^b \times \mathbb{R}^o \to G$ is assumed to be nonlinear.

The theoretical foundation of the invariant filtering framework relies on the assumption that the system dynamics are group affine, meaning that the function $\mathbf{f}(\boldsymbol{\chi}, \mathbf{u}, \mathbf{w})$ admits the form $\bar{\mathbf{f}}(\boldsymbol{\chi}, \mathbf{u})\mathbf{g}(\mathbf{w})$ as a first order approximation in $\mathbf{w}$, with $\bar{\mathbf{f}}$ satisfying Equation (3). Under this condition, the next two theorems demonstrate that, by appropriately defining the estimation error as left- or right-invariant, a broad class of nonlinear systems give rise to linear error dynamics. In the following, the notation $\hat{\boldsymbol{\chi}}_{k|l}$ denotes the estimate of state $\boldsymbol{\chi}_k$ taking into account measurements up to time $l$.

**Definition 3** (Left- and right-invariant errors)**.** *The left- and right-invariant errors between a state $\boldsymbol{\chi}_k$ and its estimation $\hat{\boldsymbol{\chi}}_{k|l}$ are*

$$\boldsymbol{\eta}^L_{k|l} = \hat{\boldsymbol{\chi}}^{-1}_{k|l}\boldsymbol{\chi}_k \qquad \textit{(left-invariant),} \tag{24a}$$
$$\boldsymbol{\eta}^R_{k|l} = \boldsymbol{\chi}_k\hat{\boldsymbol{\chi}}^{-1}_{k|l} \qquad \textit{(right-invariant).} \tag{24b}$$

**Theorem 3** (Fundamental property of invariant filtering [19])**.** *In the absence of process noise, group-affine dynamics $\mathbf{f}$—i.e., satisfying* (3)*—yield a state-independent trajectory for the left- and right-invariant errors:*

$$\boldsymbol{\eta}^L_{k+1|l} = \mathbf{g}(\boldsymbol{\eta}^L_{k|l}, \mathbf{u}_k), \quad \mathbf{g}(\boldsymbol{\eta}, \mathbf{u}) = \mathbf{f}(\mathbf{I}, \mathbf{u})^{-1}\mathbf{f}(\boldsymbol{\eta}, \mathbf{u}),$$
$$\boldsymbol{\eta}^R_{k+1|l} = \mathbf{g}(\boldsymbol{\eta}^R_{k|l}, \mathbf{u}_k), \quad \mathbf{g}(\boldsymbol{\eta}, \mathbf{u}) = \mathbf{f}(\boldsymbol{\eta}, \mathbf{u})\mathbf{f}(\mathbf{I}, \mathbf{u})^{-1}.$$

*Moreover, in this case, for each $\mathbf{u}_k \in \mathbb{R}^b$, there exists $\mathbf{F}_k \in \mathbb{R}^{n\times n}$ such that, for all $\boldsymbol{\xi} \in \mathbb{R}^n$,*

$$\mathbf{g}(\exp(\boldsymbol{\xi}), \mathbf{u}_k) = \exp(\mathbf{F}_k\boldsymbol{\xi}), \tag{26}$$

*meaning that the function $\mathbf{g}(\cdot, \mathbf{u}_k)$ is wholly encoded in $\mathbf{F}_k$.*

This last result suggests that, in the absence of process noise, the linearized error $\boldsymbol{\xi}_{k|l} \coloneqq \log(\boldsymbol{\eta}_{k|l})$ evolves linearly according to

$$\boldsymbol{\xi}_{k+1|l} = \mathbf{F}_k\boldsymbol{\xi}_{k|l}. \tag{27}$$

**Theorem 4** (Log-Linear property of the error [18])**.** *In the case of deterministic dynamics, consider the left- or right-invariant error $\boldsymbol{\eta}_{k|l}$ between two trajectories, which may be arbitrarily far apart. Suppose the initial error is given by $\boldsymbol{\eta}_{0|l} = \exp(\boldsymbol{\xi}_{0|l})$ for some $\boldsymbol{\xi}_{0|l} \in \mathbb{R}^n$. Then, for all $k \geq 0$,*

$$\boldsymbol{\eta}_{k|l} = \exp(\boldsymbol{\xi}_{k|l}). \tag{28}$$

*In other words, the nonlinear error $\boldsymbol{\eta}_{k|l}$ evolves entirely through its corresponding linear variable $\boldsymbol{\xi}_{k|l}$, which satisfies the linear time-varying Equation* (27)*.*

The invariant framework assumes that the system state is distributed according to a concentrated Gaussian on the Lie group $G$, a concept first introduced, to the best of our knowledge, in [38], [39]. In this setting, the left- or right-invariant error takes the form

$$\boldsymbol{\eta}_{k|l} = \exp(\boldsymbol{\xi}_{k|l}), \qquad \boldsymbol{\xi}_{k|l} \sim \mathcal{N}(\mathbf{0}, \mathbf{P}_{k|l}). \tag{29}$$

Under this assumption and with deterministic dynamics, the log-linear property enables exact propagation of $\mathbf{P}_{k|l}$.

**Remark 3.** *Deterministic dynamics are uncommon in practice. In the presence of process noise, Equation* (27) *becomes*

$$\begin{aligned}\boldsymbol{\xi}_{k+1|l} =& \mathbf{F}_k\boldsymbol{\xi}_{k|l} + \mathbf{G}_k\mathbf{w}_k \\ &+ \mathcal{O}(\|\boldsymbol{\xi}_{k|l}\|^2, \|\mathbf{w}_k\|^2, \|\boldsymbol{\xi}_{k|l}\|\|\mathbf{w}_k\|),\end{aligned} \tag{30}$$

*where the Jacobian $\mathbf{G}_k$ typically depends on the state estimate $\hat{\boldsymbol{\chi}}_{k|l}$, while $\mathbf{F}_k$ does not. This yields more consistent and robust covariance propagation than in the standard EKF, where all Jacobians are state-dependent.*

At the update stage, the IEKF refines its estimate using the newly received measurement $\mathbf{y}_k$, which may be expressed in either left- or right-invariant form:

$$\mathbf{y}_k = \boldsymbol{\chi}_k\mathbf{d} + \mathbf{n}_k, \qquad \text{(left-invariant)} \tag{31a}$$
$$\mathbf{y}_k = \boldsymbol{\chi}_k^{-1}\mathbf{d} + \mathbf{n}_k, \qquad \text{(right-invariant)} \tag{31b}$$

with $\mathbf{d} \in \mathbb{R}^N$. The corresponding innovation is defined as

$$\mathbf{z}_k = \hat{\boldsymbol{\chi}}^{-1}_{k|k-1}\mathbf{y}_k - \mathbf{d}, \qquad \text{(left-invariant)} \tag{32a}$$
$$\mathbf{z}_k = \hat{\boldsymbol{\chi}}_{k|k-1}\mathbf{y}_k - \mathbf{d}, \qquad \text{(right-invariant)} \tag{32b}$$

which expands to $\mathbf{z}_k = \exp(\pm\boldsymbol{\xi}_{k|k-1})\mathbf{d} - \mathbf{d} + \hat{\boldsymbol{\chi}}^{\mp 1}_{k|k-1}\mathbf{n}_k$ with the upper sign for the left-invariant case and the lower for the right-invariant case. This expression can be linearized around $\boldsymbol{\xi}_{k|k-1} = \mathbf{0}$, yielding a measurement Jacobian $\mathbf{H}$ that is, once again, independent of $\hat{\boldsymbol{\chi}}_{k|k-1}$. This structural property endows the IEKF with significantly better convergence behavior than the standard EKF, in which the measurement Jacobian $\mathbf{H}_k$ typically depends on the current state estimate.

## APPENDIX C
## THE MATRIX LIE GROUP OF EXTENDED POSES

Many applications in robotics and navigation require estimating the orientation, velocity, and position of a rigid body in three-dimensional space. This state can be represented as an element of the matrix Lie group

$$SE_2(3) \coloneqq \left\{ \left[\begin{array}{c|cc} \mathbf{R} & \mathbf{v} & \mathbf{p} \\ \hline \mathbf{0}_{2\times 3} & \multicolumn{2}{c}{\mathbf{I}_2} \end{array}\right] \middle| \begin{array}{c} \mathbf{R} \in SO(3) \\ \mathbf{v}, \mathbf{p} \in \mathbb{R}^3 \end{array} \right\}, \tag{33}$$

where group composition and inversion are given by matrix multiplication and inversion, respectively. The associated Lie algebra $\mathfrak{se}_2(3)$ is identified with $\mathbb{R}^9$ via the linear map

$$\mathcal{L}_{\mathfrak{se}_2(3)}\left(\begin{bmatrix}\boldsymbol{\phi}\\ \boldsymbol{\nu}\\ \boldsymbol{\rho}\end{bmatrix}\right) := \left[\begin{array}{c|cc}(\boldsymbol{\phi})_\times & \boldsymbol{\nu} & \boldsymbol{\rho}\\ \hline \mathbf{0}_{2\times 3} & \multicolumn{2}{c}{\mathbf{0}_{2\times 2}}\end{array}\right] \in \mathfrak{se}_2(3), \tag{34}$$

where $\boldsymbol{\phi}, \boldsymbol{\nu}, \boldsymbol{\rho} \in \mathbb{R}^3$ and $(\boldsymbol{\phi})_\times$ is the skew-symmetric matrix associated with the vector cross product in $\mathbb{R}^3$.

Given $\boldsymbol{\chi} \in SE_2(3)$, the adjoint of $\boldsymbol{\chi}$ is defined as

$$\mathbf{Ad}_{\boldsymbol{\chi}} = \begin{bmatrix}\mathbf{R} & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3}\\ (\mathbf{v})_\times \mathbf{R} & \mathbf{R} & \mathbf{0}_{3\times 3}\\ (\mathbf{p})_\times \mathbf{R} & \mathbf{0}_{3\times 3} & \mathbf{R}\end{bmatrix}, \tag{35}$$

such that $\boldsymbol{\chi}\exp(\boldsymbol{\xi})\boldsymbol{\chi}^{-1} = \exp(\mathbf{Ad}_{\boldsymbol{\chi}}\boldsymbol{\xi})$ for all $\boldsymbol{\xi} \in \mathbb{R}^9$.

Consider a rigid body with an attached frame $\mathcal{F}_i$. The discrete-time dynamics of its extended pose, expressed in the inertial frame $\mathcal{F}_0$, follow from integrating the angular velocity ${}^i\boldsymbol{\omega}_k^i$ and double integrating the linear acceleration ${}^i\mathbf{a}_k^i$ measured by an IMU fixed to $\mathcal{F}_i$. Under the flat Earth assumption and neglecting IMU biases, this yields [31]

$$\begin{aligned}{}^0\boldsymbol{\chi}_{k+1}^{0i} &= \mathbf{f}({}^0\boldsymbol{\chi}_k^{0i}, \mathbf{u}_k^i, \mathbf{w}_k^i),\\ &= \boldsymbol{\Gamma}\boldsymbol{\Phi}({}^0\boldsymbol{\chi}_k^{0i})\boldsymbol{\Upsilon}(\mathbf{u}_k^i)\exp(\mathbf{G}_k^i\mathbf{w}_k^i),\end{aligned} \tag{36}$$

where $\mathbf{u}_k^i = ({}^i\boldsymbol{\omega}_k^i, {}^i\mathbf{a}_k^i)$, and where $\mathbf{w}_k^i = (\mathbf{w}_k^{i,\boldsymbol{\omega}}, \mathbf{w}_k^{i,\mathbf{a}})$ stacks the gyroscope and accelerometer noises. The last expression is a first order approximation in $\mathbf{w}_k^i$, with

$$\boldsymbol{\Gamma} = \left[\begin{array}{c|cc}\mathbf{I}_3 & {}^0\mathbf{g}dt & {}^0\mathbf{g}\frac{dt^2}{2}\\ \hline \mathbf{0}_{2\times 3} & \multicolumn{2}{c}{\mathbf{I}_2}\end{array}\right], \tag{37}$$

$$\boldsymbol{\Phi}(\boldsymbol{\chi}) = \left[\begin{array}{c|cc}\mathbf{R} & \mathbf{v} & \mathbf{p}+\mathbf{v}dt\\ \hline \mathbf{0}_{2\times 3} & \multicolumn{2}{c}{\mathbf{I}_2}\end{array}\right], \tag{38}$$

$$\boldsymbol{\Upsilon}(\mathbf{u}) = \left[\begin{array}{c|cc}\exp_{SO(3)}(\boldsymbol{\omega}dt) & \mathbf{a}dt & \mathbf{a}\frac{dt^2}{2}\\ \hline \mathbf{0}_{2\times 3} & \multicolumn{2}{c}{\mathbf{I}_2}\end{array}\right], \tag{39}$$

$$\mathbf{G}_k^i = \begin{bmatrix}\mathcal{J}_{-\,{}^i\boldsymbol{\omega}_k^i dt} & \mathbf{0}_{3\times 3} & \\ \mathbf{0}_{3\times 3} & \exp_{SO(3)}(-{}^i\boldsymbol{\omega}_k^i dt)dt & \\ \mathbf{0}_{3\times 3} & \exp_{SO(3)}(-{}^i\boldsymbol{\omega}_k^i dt)\frac{dt^2}{2} & \end{bmatrix}, \tag{40}$$

where $\mathcal{J}_{\boldsymbol{\phi}}$ is the left-Jacobian of $SO(3)$ evaluated at $\boldsymbol{\phi}$.

## Appendix D
## Covariance initialization with hinge constraints

In a kinematic tree, it is generally more natural to represent the uncertainty in the pose of a body relative to its parent rather than to the frame $\mathcal{F}_0$. This way, the covariance matrix in the relative representation exhibits a block-diagonal structure.

The constraints imposed by hinge joints can be found in Table I. The initial state $(\hat{\mathbf{X}}_{0|0})_j$ and the covariance matrix $(\mathbf{P}_{0|0}^{\text{IterIEKF}})_j$ must be consistent with these constraints. Assuming it is the case for the initial state and recalling that

$$(\mathbf{X}_0)_j = (\hat{\mathbf{X}}_{0|0})_j \exp_{SE_2(3)}\left((\boldsymbol{\xi}_{0|0})_j\right), \tag{41}$$

a first-order approximation of the considered constraints in $(\boldsymbol{\xi}_{0|0})_j$ yields:

$$(\boldsymbol{\xi}_{0|0}^{\mathbf{R}})_j \approx \alpha \cdot {}^{s(j)}\boldsymbol{\theta}^j, \tag{42}$$

$$(\boldsymbol{\xi}_{0|0}^{\mathbf{p}})_j \approx \left({}^{s(j)}\boldsymbol{\varsigma}^j\right)_\times (\boldsymbol{\xi}_{0|0}^{\mathbf{R}})_j, \tag{43}$$

where $\alpha \in \mathbb{R}$ and where $(\boldsymbol{\xi}_{0|0}^{\mathbf{R}})_j$, $(\boldsymbol{\xi}_{0|0}^{\mathbf{p}})_j \in \mathbb{R}^3$ denote the sub-vectors of $(\boldsymbol{\xi}_{0|0})_j$ corresponding to rotation and position, respectively. The same reasoning applied to the time-derivative of the position constraint gives the corresponding velocity relation

$$(\boldsymbol{\xi}_{0|0}^{\mathbf{v}})_j \approx \left(\left({}^{s(j)}\boldsymbol{\omega}_0^j\right)_\times {}^{s(j)}\boldsymbol{\varsigma}^j\right)_\times (\boldsymbol{\xi}_{0|0}^{\mathbf{R}})_j. \tag{44}$$

This yields the following diagonal blocks:

$$(\mathbf{P}_{0|0}^{\text{IterIEKF}})_j = \begin{bmatrix}\mathbf{A}_{00} & \mathbf{A}_{01} & \mathbf{A}_{02}\\ \mathbf{A}_{01}^T & \mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{02}^T & \mathbf{A}_{12}^T & \mathbf{A}_{22}\end{bmatrix}, \tag{45}$$

with block entries

$$\mathbf{A}_{00} = \sigma_\phi^2({}^{s(j)}\boldsymbol{\theta}^j)({}^{s(j)}\boldsymbol{\theta}^j)^T, \tag{46}$$

$$\mathbf{A}_{11} = -\left({}^{s(j)}\mathbf{r}^j\right)_\times \mathbf{A}_{00}\left({}^{s(j)}\mathbf{r}^j\right)_\times, \tag{47}$$

$$\mathbf{A}_{22} = -\left({}^{s(j)}\boldsymbol{\varsigma}^j\right)_\times \mathbf{A}_{00}\left({}^{s(j)}\boldsymbol{\varsigma}^j\right)_\times, \tag{48}$$

$$\mathbf{A}_{01} = -\mathbf{A}_{00}\left({}^{s(j)}\mathbf{r}^j\right)_\times, \tag{49}$$

$$\mathbf{A}_{02} = -\mathbf{A}_{00}\left({}^{s(j)}\boldsymbol{\varsigma}^j\right)_\times, \tag{50}$$

$$\mathbf{A}_{12} = -\left({}^{s(j)}\mathbf{r}^j\right)_\times \mathbf{A}_{00}\left({}^{s(j)}\boldsymbol{\varsigma}^j\right)_\times, \tag{51}$$

where ${}^{s(j)}\mathbf{r}^j = ({}^{s(j)}\boldsymbol{\omega}_0^{s(j)})_\times {}^{s(j)}\boldsymbol{\varsigma}^j$ and where $\sigma_\phi$ denotes the standard deviation of the rotational error around $\boldsymbol{\theta}^j$.

Consider joints $i$ and $j$ such that $s(i) = p(j)$. The IterIEKF covariance matrix in the absolute representation is obtained by recalling that

$$\begin{aligned}(\mathbf{X}_0)_j &= \left({}^0\boldsymbol{\chi}_0^{0p(j)}\right)^{-1}\left({}^0\boldsymbol{\chi}_0^{0s(j)}\right),\\ &= (\hat{\mathbf{X}}_{0|0})_j \exp_{SE_2(3)}\left((\boldsymbol{\xi}_{0|0})_j\right),\end{aligned} \tag{52}$$

$${}^0\boldsymbol{\chi}_0^{0s(j)} = {}^0\hat{\boldsymbol{\chi}}_0^{0s(j)} \exp_{SE_2(3)}\left((\bar{\boldsymbol{\xi}}_{0|0})_j\right), \tag{53}$$

$${}^0\boldsymbol{\chi}_0^{0p(j)} = {}^0\hat{\boldsymbol{\chi}}_0^{0p(j)} \exp_{SE_2(3)}\left((\bar{\boldsymbol{\xi}}_{0|0})_i\right), \tag{54}$$

Substituting the last two expressions into the first one and applying a first-order approximation of the BCH formula yields:

$$(\boldsymbol{\xi}_{0|0})_j \approx (\bar{\boldsymbol{\xi}}_{0|0})_j - \mathbf{Ad}_{(\hat{\mathbf{X}}_{0|0})_j^{-1}}(\bar{\boldsymbol{\xi}}_{0|0})_i, \tag{55}$$

where $\mathbf{Ad}_{\mathbf{X}}$ denotes the adjoint matrix of $\mathbf{X}$. This relation allows us to construct recursively the covariance matrix in absolute representation block by block as

$$\begin{aligned}(\bar{\mathbf{P}}_{0|0}^{\text{IterIEKF}})_{kl} &= (\mathbf{P}_{0|0}^{\text{IterIEKF}})_{kl}\\ &+ \mathbf{Ad}_{(\hat{\mathbf{X}}_{0|0})_k^{-1}}(\bar{\mathbf{P}}_{0|0}^{\text{IterIEKF}})_{ml}\\ &+ (\bar{\mathbf{P}}_{0|0}^{\text{IterIEKF}})_{kn}\mathbf{Ad}_{(\hat{\mathbf{X}}_{0|0})_l^{-1}}^T\\ &+ \mathbf{Ad}_{(\hat{\mathbf{X}}_{0|0})_k^{-1}}(\bar{\mathbf{P}}_{0|0}^{\text{IterIEKF}})_{mn}\mathbf{Ad}_{(\hat{\mathbf{X}}_{0|0})_l^{-1}}^T,\end{aligned} \tag{56}$$

with $p(k) = s(m)$ and $p(l) = s(n)$, ensuring $m < k$ and $n < l$. Note that when $k = 0$ (resp. $l = 0$), the index $m$ (resp. $n$) does not exist, and the corresponding terms in the above expression is replaced by $\mathbf{0}_{9\times 9}$.

For the EKF and IterEKF, the following first-order approximations hold:

$$(\mathbf{e}_{0|0})_j \approx \underbrace{\begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \hat{\mathbf{R}}_{k|l}^{p(j)s(j)} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \hat{\mathbf{R}}_{k|l}^{p(j)s(j)} \end{bmatrix}}_{(\mathbf{B})_j} (\boldsymbol{\xi}_{k|l})_j, \tag{57}$$

$$(\bar{\mathbf{e}}_{0|0})_j \approx \underbrace{\begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \hat{\mathbf{R}}_{k|l}^{0s(j)} & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \hat{\mathbf{R}}_{k|l}^{0s(j)} \end{bmatrix}}_{(\bar{\mathbf{B}})_j} (\bar{\boldsymbol{\xi}}_{k|l})_j. \tag{58}$$

Thus, to first order,

$$\mathbf{P}_{0|0}^{\text{EKF}} = \mathbf{B}\mathbf{P}_{0|0}^{\text{IterIEKF}}\mathbf{B}^T, \tag{59}$$

$$\bar{\mathbf{P}}_{0|0}^{\text{EKF}} = \bar{\mathbf{B}}\bar{\mathbf{P}}_{0|0}^{\text{IterIEKF}}\bar{\mathbf{B}}^T, \tag{60}$$

$$\mathbf{B} = \operatorname{diag}\left((\mathbf{B})_1, \ldots, (\mathbf{B})_L\right), \tag{61}$$

$$\bar{\mathbf{B}} = \operatorname{diag}\left((\bar{\mathbf{B}})_1, \ldots, (\bar{\mathbf{B}})_L\right). \tag{62}$$

## 7.3 Epilogue

Beyond the specific experimental scenarios considered in this work, the proposed formulation provides a general modeling and estimation framework for rigid-body systems equipped with inertial sensors. It naturally accommodates kinematic tree structures and a wide range of joint types, while avoiding the singularities and inconsistencies associated with minimal-coordinate or purely Euclidean representations. As such, it offers a unified geometric foundation for extended-pose estimation in articulated rigid-body systems.

That said, several aspects still warrant further investigation. In particular, the practical impact of deliberately introducing measurement noise to relax translational constraints along selected directions was not evaluated in the paper. To provide initial evidence that this heuristic is sensible in practice, we consider a simple simulation consisting of a single IterIEKF update, using the modified IterIEKF described in Remark 1 of the previous paper. The goal is to enforce the constraints imposed by a cylindrical, prismatic, or planar joint on the relative pose $\boldsymbol{\chi}_k = (\mathbf{X}_k)_j$ between the two bodies $s(j)$ and $p(j)$ of a rigid-body system. We adopt the left-invariant formulation and the constraint forms reported in Table I of the previous paper. Assume we are at time $k$, just before the update associated with the joint constraint, and that the prior is characterized by

$$\hat{\boldsymbol{\chi}}_{k|k-1} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 1 & 0 \\ \mathbf{0} & 0 & 1 \end{bmatrix}, \qquad \boldsymbol{\Sigma}_{k|k-1} = \mathrm{block\_diag}((\tfrac{\pi}{6})^2\mathbf{I}, \mathbf{I}, \mathbf{I}). \tag{7.1}$$

Our aim is not to drive the estimate toward a particular configuration, but rather to verify that the posterior distribution is consistent with the constraint enforced at the update stage. Consider the pseudo-measurement model associated with the constraint,

$$\mathbf{y}_k = \boldsymbol{\chi}_k \mathbf{d}_k + \mathbf{n}_k, \tag{7.2}$$

where $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ and $\mathbf{N}_k$ is rank-deficient. We assess the compatibility of the posterior with this constraint by drawing $10{,}000$ samples from the concentrated Gaussian

$$\boldsymbol{\chi}_k = \hat{\boldsymbol{\chi}}_{k|k} \exp(\boldsymbol{\xi}_{k|k}), \quad \boldsymbol{\xi}_{k|k} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{k|k}), \tag{7.3}$$

where $\exp = \exp_{SE_2(3)} \circ \mathcal{L}_{\mathfrak{se}_2(3)}$. For each sample $\bar{\boldsymbol{\chi}}_{k|k}$, we compute the residual

$$\bar{\mathbf{r}}_{k|k} = \mathbf{y}_k - \bar{\boldsymbol{\chi}}_{k|k} \mathbf{d}_k. \tag{7.4}$$

We then examine the empirical distribution of these residuals. For the posterior to

be compliant with the constraint, the residual components orthogonal to $\mathrm{Im}(\mathbf{N}_k)$ should be concentrated around zero, and any remaining variability should primarily lie in $\mathrm{Im}(\mathbf{N}_k)$.

**Cylindrical joint** Cylindrical joints yield the following constraints

$$\mathbf{\Pi}\boldsymbol{\chi}_k \begin{bmatrix} {}^{s(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix} = {}^{p(j)}\boldsymbol{\theta}^j, \tag{7.5}$$

$$\mathbf{\Pi}\boldsymbol{\chi}_k \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = {}^{p(j)}\boldsymbol{\rho}^j + \mathbf{n}_k, \tag{7.6}$$

where the noise term satisfies $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$, with $\mathbf{N}_k = 3^2({}^{p(j)}\boldsymbol{\theta}^j)({}^{p(j)}\boldsymbol{\theta}^j)^T$. The matrix $\mathbf{\Pi} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}$ is used to discard the last two components of each pseudo-measurement, since they do not carry information about the state. We choose $\boldsymbol{\theta}^j$ as the unit vector aligned with the $z$-axis of the frame $\mathcal{F}_{p(j)}$, so that ${}^{p(j)}\boldsymbol{\theta}^j$ has a single nonzero entry, namely its third component. The vector ${}^{s(j)}\boldsymbol{\theta}^j$ is obtained by applying a random rotation matrix to ${}^{p(j)}\boldsymbol{\theta}^j$. The vectors ${}^{s(j)}\boldsymbol{\varsigma}^j$ and ${}^{p(j)}\boldsymbol{\rho}^j$ are chosen as random unit vectors. The reader is referred to Table I of the previous paper for an illustration of the cylindrical joint and the associated vectors.

We denote by $\bar{\mathbf{r}}^{\mathbf{R}}_{k|k}, \bar{\mathbf{r}}^{\mathbf{p}}_{k|k} \in \mathbb{R}^3$ the residuals associated with the rotation and position constraints, respectively. The empirical mean and standard deviation of each component are reported in Table 7.1. The residuals for the rotation constraint are essentially zero, which indicates that the posterior distribution is consistent with the rotational constraint. For the position constraint, the residuals exhibit nonzero mean and nonzero standard deviation only in their third component, showing that the constraint is relaxed along the $z$-axis of $\mathcal{F}_{p(j)}$. This is precisely the direction spanned by ${}^{p(j)}\boldsymbol{\theta}^j$, as intended. The posterior distribution therefore remains consistent with the imposed position constraint.

Finally, note that the standard deviation of the positional residual along the $z$-axis is close to $1$, even though we set a standard deviation of $3\,\mathrm{m}$ in the measurement noise $\mathbf{n}_k$. This is a consequence of the prior uncertainty: since we set $(\mathbf{\Sigma}_{k|k-1})_{9,9} = 1$, the update cannot increase uncertainty, so $(\mathbf{\Sigma}_{k|k})_{9,9} \leq (\mathbf{\Sigma}_{k|k-1})_{9,9}$. This, in turn, limits the dispersion of the residuals along the unconstrained direction.

| | | $i=1$ | $i=2$ | $i=3$ |
|---|---|---|---|---|
| $(\bar{\mathbf{r}}^{\mathbf{R}}_{k\|k})_i$ | [rad] | $(0.0 \pm 3.4)10^{-4}$ | $(0.0 \pm 2.7)10^{-4}$ | $(0.4 \pm 1.0)10^{-7}$ |
| $(\bar{\mathbf{r}}^{\mathbf{p}}_{k\|k})_i$ | [m] | $(0.0 \pm 3.1)10^{-4}$ | $(0.0 \pm 4.1)10^{-4}$ | $(-1.3 \pm 0.9)$ |

**Table 7.1:** Empirical mean and standard deviation of each component of the residuals associated with the orientation and position constraints of a cylindrical joint, computed from 10,000 samples.

**Prismatic joint** A prismatic joint yields the constraints

$$\mathbf{\Pi}\boldsymbol{\chi}_k \begin{bmatrix} {}^{s(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix} = {}^{p(j)}\boldsymbol{\theta}^j, \tag{7.7}$$

$$\mathbf{\Pi}\boldsymbol{\chi}_k \begin{bmatrix} {}^{s(j)}\boldsymbol{\tau}^j \\ 0 \\ 0 \end{bmatrix} = {}^{p(j)}\boldsymbol{\tau}^j, \tag{7.8}$$

$$\mathbf{\Pi}\boldsymbol{\chi}_k \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = {}^{p(j)}\boldsymbol{\rho}^j + \mathbf{n}_k, \tag{7.9}$$

where $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$ with $\mathbf{N}_k = 3^2, {}^{p(j)}\boldsymbol{\theta}^j({}^{p(j)}\boldsymbol{\theta}^j)^T$. We choose $\boldsymbol{\theta}^j$ and $\boldsymbol{\tau}^j$ as the unit vectors aligned with the $z$-axis and $x$-axis of the frame $\mathcal{F}_{p(j)}$, respectively, so that ${}^{p(j)}\boldsymbol{\theta}^j$ and ${}^{p(j)}\boldsymbol{\tau}^j$ each have a single nonzero entry (the third component for ${}^{p(j)}\boldsymbol{\theta}^j$ and the first component for ${}^{p(j)}\boldsymbol{\tau}^j$). The vectors ${}^{s(j)}\boldsymbol{\theta}^j$ and ${}^{s(j)}\boldsymbol{\tau}^j$ are obtained by applying the same random rotation matrix to ${}^{p(j)}\boldsymbol{\theta}^j$ and ${}^{p(j)}\boldsymbol{\tau}^j$. The remaining vectors ${}^{s(j)}\boldsymbol{\varsigma}^j$ and ${}^{p(j)}\boldsymbol{\rho}^j$ are chosen as random unit vectors. The reader is referred to Table I of the previous paper for an illustration of the prismatic joint and the associated vectors.

We denote by $\bar{\mathbf{r}}^{\mathbf{R},1}_{k|k}, \bar{\mathbf{r}}^{\mathbf{R},2}_{k|k} \in \mathbb{R}^3$ the residuals associated with the first and second rotation constraints, and by $\bar{\mathbf{r}}^{\mathbf{p}}_{k|k} \in \mathbb{R}^3$ the residuals associated with the position constraint. The empirical mean and standard deviation of each component are reported in Table 7.2. The residuals for both rotational constraints are essentially zero, indicating that the posterior distribution is consistent with the rotational constraints. The residuals associated with the position constraint exhibit relaxation along the $z$-axis of $\mathcal{F}_{p(j)}$, which coincides with the direction spanned by ${}^{p(j)}\boldsymbol{\theta}^j$, as intended.

| | | $i=1$ | $i=2$ | $i=3$ |
|---|---|---|---|---|
| $(\bar{\mathbf{r}}^{\mathbf{R},1}_{k\|k})_i$ | [rad] | $(0.0 \pm 2.5)10^{-4}$ | $(0.0 \pm 3.0)10^{-4}$ | $(1.0 \pm 8.9)10^{-8}$ |
| $(\bar{\mathbf{r}}^{\mathbf{R},2}_{k\|k})_i$ | [rad] | $(2.4 \pm 9.0)10^{-8}$ | $(0.0 \pm 3.1)10^{-4}$ | $(0.0 \pm 2.5)10^{-4}$ |
| $(\bar{\mathbf{r}}^{\mathbf{p}}_{k\|k})_i$ | [m] | $(0.0 \pm 3.5)10^{-4}$ | $(0.0 \pm 4.5)10^{-4}$ | $(-1.3 \pm 1.0)$ |

**Table 7.2:** Empirical mean and standard deviation of each component of the residuals associated with the orientation and position constraints of a prismatic joint, computed from 10,000 samples.

**Planar joint** A planar joint yields the constraints

$$\mathbf{\Pi}\boldsymbol{\chi}_k \begin{bmatrix} {}^{s(j)}\boldsymbol{\theta}^j \\ 0 \\ 0 \end{bmatrix} = {}^{p(j)}\boldsymbol{\theta}^j, \tag{7.10}$$

$$\mathbf{\Pi}\boldsymbol{\chi}_k \begin{bmatrix} {}^{s(j)}\boldsymbol{\varsigma}^j \\ 0 \\ 1 \end{bmatrix} = {}^{p(j)}\boldsymbol{\rho}^j + \mathbf{n}_k, \tag{7.11}$$

where the noise term satisfies $\mathbf{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{N}_k)$, with

$$\mathbf{N}_k = 3^2 \begin{bmatrix} {}^{p(j)}\boldsymbol{\tau}^j & {}^{p(j)}\boldsymbol{\zeta}^j \end{bmatrix} \begin{bmatrix} {}^{p(j)}\boldsymbol{\tau}^j & {}^{p(j)}\boldsymbol{\zeta}^j \end{bmatrix}^T. \tag{7.12}$$

We choose $\boldsymbol{\tau}^j$, $\boldsymbol{\zeta}^j$ and $\boldsymbol{\theta}^j$ as the unit vectors aligned with the $x$-axis, $y-$axis and $z$-axis of the frame $\mathcal{F}_{p(j)}$, respectively. The vectors ${}^{s(j)}\boldsymbol{\tau}^j$, ${}^{s(j)}\boldsymbol{\zeta}^j$ and ${}^{s(j)}\boldsymbol{\theta}^j$ are obtained by applying the same random rotation matrix to ${}^{p(j)}\boldsymbol{\tau}^j$, ${}^{p(j)}\boldsymbol{\zeta}^j$ and ${}^{p(j)}\boldsymbol{\theta}^j$. The remaining vectors ${}^{s(j)}\boldsymbol{\varsigma}^j$ and ${}^{p(j)}\boldsymbol{\rho}^j$ are chosen as random unit vectors. The reader is referred to Table I of the previous paper for an illustration of the planar joint and the associated vectors.

The empirical mean and standard deviation of each residual component are reported in Table 7.3. The results show that the residuals associated with the rotation constraint are essentially zero, whereas the residuals associated with the position constraint exhibit relaxation along the directions spanned by ${}^{p(j)}\boldsymbol{\tau}^j$ and ${}^{p(j)}\boldsymbol{\zeta}^j$, as intended. To further assess the distribution of these residuals, Figure 7.1 shows a corner plot of the empirical distribution of the residuals associated with the position constraint. The residuals appear approximately Gaussian in the $xy$-plane of the frame $\mathcal{F}_{p(j)}$, with a planar covariance that is approximately isotropic.

Overall, these simulations suggest that the proposed noise-relaxation heuristic

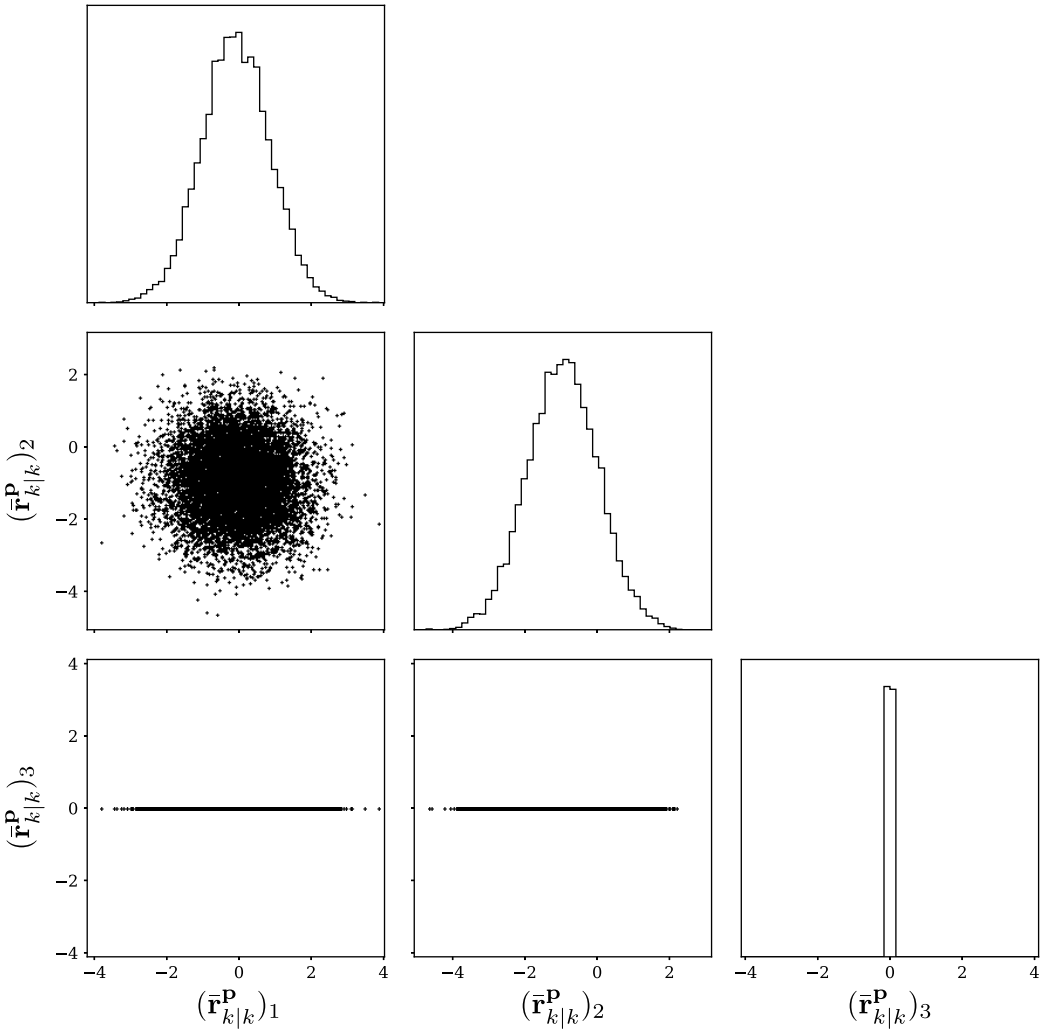



**Figure 7.1:** Corner plot of the empirical distribution of the residuals associated with the planar joint position constraint, computed from 10,000 samples.

| | | $i=1$ | $i=2$ | $i=3$ |
|---|---|---|---|---|
| $(\bar{\mathbf{r}}^{\mathbf{R}}_{k\|k})_i$ | [rad] | $(0.1 \pm 2.6)10^{-4}$ | $(0.0 \pm 3.1)10^{-4}$ | $(-1.6 \pm 9.9)10^{-8}$ |
| $(\bar{\mathbf{r}}^{\mathbf{p}}_{k\|k})_i$ | [m] | $(-0.9 \pm 9.5)10^{-1}$ | $(-9.5 \pm 9.7)10^{-1}$ | $(0.0 \pm 2.3)10^{-4}$ |

**Table 7.3:** Empirical mean and standard deviation of each component of the residuals associated with the orientation and position constraints of a planar joint, computed from 10,000 samples.

behaves as intended in this setting. They provide qualitative evidence, but they do not constitute a formal guarantee, nor do they cover the full range of configurations and noise regimes. A more systematic analysis is left for future work.

The following chapter places the thesis results in a broader perspective, discusses their limitations, and outlines directions for future research.

# 8 Synthesis and perspectives

This thesis set out to extend the invariant Kalman filtering framework to the problem of pose estimation for articulated rigid-body systems. Motivated by applications in human–robot interaction, rehabilitation, and wearable robotics, the objective was to move beyond single rigid-body models and address the structural complexity of multibody systems without sacrificing the favorable convergence properties of invariant filtering. The manuscript followed a progressive development, moving from constraint handling, to globally consistent updates in the low-noise regime, and finally to multibody state representations.

The first contribution addressed the problem of handling state equality constraints within the Kalman filtering process. By interpreting such constraints as noise-free pseudo-measurements and identifying the challenges associated with their enforcement, this thesis clarified both the potential and the limitations of imposing hard constraints in an invariant filtering setting, as required for the treatment of kinematic constraints in rigid-body systems.

Building on this foundation, the second contribution addressed a critical shortcoming of the IEKF in the low-noise regime, namely the inability of the update stage to ensure global consistency of the state estimate with noise-free measurements. By introducing the IterIEKF, this thesis showed how Gauss–Newton relinearization makes it possible, under suitable conditions, to recover key properties reminiscent of the linear Kalman filter. In particular, the IterIEKF was shown to enforce noise-free information globally on the Lie group, providing a sound theoretical and algorithmic basis for handling exact constraints.

The final contribution brought these methodological advances together and achieved the primary objective of the thesis. The proposed relative L-extended pose representation exploits the Lie group structure of rigid-body motion, admits group-affine dynamics when inertial measurements are available, and allows a broad class of joint constraints to be expressed in invariant form. Combined with the IterIEKF, it enables consistent and efficient extended-pose estimation in multibody systems, as demonstrated on both robotic and biomechanical experimental setups.

Beyond the specific algorithms and representations introduced, this work highlights the importance of respecting geometry, not only in the state space, but also in the way constraints and measurements are modeled and enforced. The results suggest that many of the favorable properties of invariant filtering, previously

limited to single rigid bodies, can be retained in more complex systems, provided that the underlying modeling choices are made with care.

## 8.1 Current limitations

This thesis has several limitations, starting with the IterIEKF. Its theoretical guarantees are conditioned on the convergence of the Gauss–Newton iterative scheme used during the update stage. In practice, this method may be sensitive to initialization and to strongly nonlinear measurement models. Although no Gauss–Newton divergence was observed in the simulations and experiments reported here, this does not in itself guarantee robust behavior across all operating conditions.

A second limitation of the IterIEKF is the additional computational cost induced by its iterative update. Although our simulations indicate that most updates converge in 2 to 3 Gauss–Newton iterations, this overhead may still be significant in time-critical applications. The cost also increases with the number of (pseudo-)measurements processed at each time step, which can be large in articulated systems with many kinematic constraints. As a result, scalability and real-time performance are not guaranteed by the current formulation.

At a broader level, the proposed invariant filtering framework relies on sensing and modeling assumptions that may be restrictive in practice. In particular, it assumes access to angular velocity and specific force measurements for each body, which typically requires one IMU per segment, as well as accurate time synchronization across sensors. The impact of imperfect synchronization was not investigated. Moreover, IMU biases were not modeled: in the real-world experiments, the measurements were pre-processed to mitigate biases, while the simulations assumed bias-free sensors. As a result, the reported performance may not fully reflect practical online use, where the filter must estimate the state while simultaneously accounting for biases and synchronization errors.

The proposed framework is currently limited to systems with a kinematic tree structure. This is particularly problematic in robotics and biomechanics, where closed chains are common. In addition, environmental contacts may create kinematic loops, leading to hybrid dynamics with time-varying constraints that appear and disappear over time. This thesis does not address such switching constraints.

Another important practical limitation of the framework is its reliance on calibration. In particular, expressing joint constraints requires accurate knowledge of each IMU's position and orientation relative to the corresponding body segment. This makes the method sensitive to mounting errors and calibration drift, and conflicts with the goal of an easy-to-deploy inertial motion estimation setup.

Regarding human motion tracking, this thesis does not address several effects that are known to be challenging in practice. In particular, soft tissue artifacts are not modeled, and the joint models considered remain simplified, for example hinge models for the knee and ankle. The impact of these modeling simplifications on estimation accuracy and robustness has not been assessed.

Finally, broader experimental validation is needed to properly assess the proposed invariant filtering framework for rigid-body systems. In particular, the relaxation noise mechanism used to express kinematic constraints associated with cylindrical, prismatic, and planar joints has so far been evaluated only on a single simulation scenario. For this reason, its robustness and practical range of applicability remain unclear.

## 8.2 Future perspectives

Future work can be organized around three themes: strengthening the robustness and efficiency of the IterIEKF, relaxing sensing and calibration assumptions, and broadening experimental validation.

A first priority is to characterize and mitigate failure modes of the IterIEKF update, in particular those associated with divergence of the Gauss–Newton iterations. Incorporating standard safeguards from iterative estimation, such as damping strategies, trust-region methods, and adaptive stopping criteria, should improve robustness in challenging conditions. In parallel, improving computational efficiency is essential for time-critical applications. This includes developing strategies to handle large numbers of (pseudo-)measurements more efficiently, for instance through constraint selection, scheduling, or partial updates. Benchmarking runtime and estimation accuracy as a function of system size and constraint count would help identify an appropriate trade-off between accuracy and computational cost.

From a sensing perspective, an important extension is to explicitly model and estimate IMU biases and synchronization errors within the filtering process. While biases are commonly included in the state and estimated online in navigation and robotics, doing so can raise observability issues and may also compromise the group-affine structure exploited in the prediction stage. Understanding how to incorporate these effects while preserving, as much as possible, the benefits of invariance is therefore a key next step.

On the modeling side, extending the framework to closed-chain systems remains an open challenge. This includes developing principled mechanisms to handle kinematic loop closure while preserving consistency of the estimated state. Another important extension is to handle contact-induced events, which would broaden

the range of applications of the proposed method. In this regard, contact-aided invariant filtering has proven effective in related settings [49], and adapting these ideas to the rigid-body system formulation proposed here appears promising.

Another practical direction is to reduce sensitivity to calibration. Incorporating sensor-to-segment placement parameters into the estimation problem, or estimating them online, would improve robustness to calibration errors and facilitate application of the framework in less controlled settings.

Regarding human motion tracking, it would be valuable to investigate whether richer biomechanical joint models can produce kinematic constraints that remain expressible in invariant form. When this is not the case, an important question is whether the resulting gains in accuracy justify losing the invariant structure of the constraints. In addition, quantifying soft tissue artifacts and developing compensation methods could lead to substantial improvements in estimation accuracy.

Finally, broader validation is required to assess the robustness and practical range of applicability of the proposed framework. This includes testing across a wider set of motions, evaluating the relaxation noise mechanism beyond the single scenario considered in this thesis, and conducting systematic comparisons with alternative motion estimation methods. In particular, comparisons with established commercial pipelines such as Xsens inertial motion capture systems, which are widely regarded as an industry standard, would help better quantify the practical performance of the proposed approach.

# Bibliography


[1] H. Zhou and H. Hu, **Human motion tracking for rehabilitation—a survey**, *Biomedical Signal Processing and Control*, vol. 3, no. 1, 1–18. DOI: 10.1016/j.bspc.2007.09.001. [Online]. Available: https://doi.org/10.1016/j.bspc.2007.09.001.

[2] C. Zhang, G. Liu, C. Li, J. Zhao, H. Yu, and Y. Zhu, **Development of a lower limb rehabilitation exoskeleton based on real-time gait detection and gait tracking**, *Advances in Mechanical Engineering*, vol. 8, no. 1. DOI: 10.1177/1687814015627982. [Online]. Available: https://doi.org/10.1177/1687814015627982.

[3] P. T. Chinmilli, S. Redkar, W. Zhang, and T. Sugar, **A review on wearable inertial tracking based human gait analysis and control strategies of lower-limb exoskeletons**, *International Robotics & Automation Journal*, vol. 3, no. 7, 398–415. DOI: 10.15406/iratj.2017.03.00080. [Online]. Available: https://doi.org/10.15406/iratj.2017.03.00080.

[4] M. Field, Z. Pan, D. Stirling, and F. Naghdy, **Human motion capture sensors and analysis in robotics**, *Industrial Robot*, vol. 38, no. 2, 163–171. DOI: 10.1108/01439911111106372. [Online]. Available: https://doi.org/10.1108/01439911111106372.

[5] S. L. Colyer, M. Evans, D. P. Cosker, and A. I. T. Salo, **A review of the evolution of vision-based motion analysis and the integration of advanced computer vision methods towards developing a markerless system**, *Sports Medicine - Open*, vol. 4, no. 1, 24. DOI: 10.1186/s40798-018-0139-y. [Online]. Available: https://doi.org/10.1186/s40798-018-0139-y.

[6] E. van der Kruk and M. M. Reijne, **Accuracy of human motion capture systems for sport applications; state-of-the-art review**, *European Journal of Sport Science*, vol. 18, no. 6, 806–819. DOI: 10.1080/17461391.2018.1463397. [Online]. Available: https://doi.org/10.1080/17461391.2018.1463397.

[7] X. Wei, P. Zhang, and J. Chai, **Accurate realtime full-body motion capture using a single depth camera**, *ACM Transactions on Graphics*, vol. 31, no. 6, 188:1–188:12. DOI: 10.1145/2366145.2366207. [Online]. Available: https://doi.org/10.1145/2366145.2366207.

[8] T. Yu, K. Guo, F. Xu, Y. Dong, Z. Su, J. Zhao, J. Li, Q. Dai, and Y. Liu, **Bodyfusion: Real-time capture of human motion and surface geometry using a single depth camera**, in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy: IEEE, 2017, 910–919. DOI: 10.1109/ICCV.2017.104.

[9] S. L. Dockstader and A. M. Tekalp, **Multiple camera tracking of interacting and occluded human motion**, *Proceedings of the IEEE*, vol. 89, no. 10, 1441–1455. DOI: 10.1109/5.959340. [Online]. Available: https://doi.org/10.1109/5.959340.

[10] G. Nagymáté and R. M. Kiss, **Application of OptiTrack motion capture systems in human movement analysis: A systematic literature review**, *Recent Innovations in Mechatronics*, vol. 5, no. 1, 1–9. DOI: 10.17667/riim.2018.1/13. [Online]. Available: https://doi.org/10.17667/riim.2018.1/13.

[11] C. Gu, W. Lin, X. He, L. Zhang, and M. Zhang, **IMU-based motion capture system for rehabilitation applications: A systematic review**, *Biomimetic Intelligence and Robotics*, vol. 3, no. 2, 100097. DOI: 10.1016/j.birob.2023.100097. [Online]. Available: https://doi.org/10.1016/j.birob.2023.100097.

[12] R. E. Kalman, **A new approach to linear filtering and prediction problems**, *Journal of Basic Engineering*, vol. 82, no. 1, 35–45.

[13] M. Á. Naya, E. Sanjurjo, A. J. Rodríguez, and J. Cuadrado, **Kalman filters based on multibody models: Linking simulation and real world. a comprehensive review**, *Multibody System Dynamics*, vol. 58, no. 3, 479–521. DOI: 10.1007/s11044-023-09893-w. [Online]. Available: https://doi.org/10.1007/s11044-023-09893-w.

[14] R. Pastorino, D. Richiedei, J. Cuadrado, and A. Trevisani, **State estimation using multibody models and non-linear Kalman filters**, *International Journal of Non-Linear Mechanics*, vol. 53, 83–90. DOI: 10.1016/j.ijnonlinmec.2013.01.016. [Online]. Available: https://doi.org/10.1016/j.ijnonlinmec.2013.01.016.

[15] M. Miezal, G. Bleser, N. Schmitz, and D. Stricker, **A generic approach to inertial tracking of arbitrary kinematic chains**, in *Proceedings of the 8th International Conference on Body Area Networks (BodyNets 2013)*, ICST, 2013, 189–192. DOI: 10.4108/icst.bodynets.2013.253608. [Online]. Available: https://doi.org/10.4108/icst.bodynets.2013.253608.

[16] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, **State estimation for legged robots - consistent fusion of leg kinematics and IMU**, in *Robotics: Science and Systems VIII*, The MIT Press, 2013, 17–24. DOI: 10.7551/mitpress/9816.003.0008. [Online]. Available: https://doi.org/10.7551/mitpress/9816.003.0008.

[17] N. Rotella, M. Bloesch, L. Righetti, and S. Schaal, **State estimation for a humanoid robot**, in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2014, 952–958. DOI: 10.1109/IROS.2014.6942674. [Online]. Available: https://doi.org/10.1109/IROS.2014.6942674.

[18] F. E. Xavier, G. Burger, M. Pétriaux, J.-E. Deschaud, and F. Goulette, **Multi-IMU proprioceptive state estimator for humanoid robots**, in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Detroit, MI, USA, 2023, 10880–10887. DOI: 10.1109/IROS55552.2023.10341849. [Online]. Available: https://doi.org/10.1109/IROS55552.2023.10341849.

[19] J. S. Lora-Millán, A. F. Hidalgo, and E. Rocon, **An IMUs-based extended Kalman filter to estimate gait lower limb sagittal kinematics for the control of wearable robotic devices**, *IEEE Access*, vol. 9, 144540–144554. DOI: 10.1109/ACCESS.2021.3122160. [Online]. Available: https://doi.org/10.1109/ACCESS.2021.3122160.

[20] A. Filippeschi, N. Schmitz, M. Miezal, G. Bleser, E. Ruffaldi, and D. Stricker, **Survey of motion tracking methods based on inertial sensors: A focus on upper limb human motion**, *Sensors*, vol. 17, no. 6, 1257. DOI: 10.3390/s17061257. [Online]. Available: https://doi.org/10.3390/s17061257.

[21] D. Roetenberg, H. Luinge, and P. Slycke, **Xsens MVN: Full 6DOF human motion tracking using miniature inertial sensors**, Xsens Technologies B.V., Enschede, The Netherlands, Technical Report 1, 2009, 1–7. [Online]. Available: https://aferro.dynu.net/blender_mvnx/6dof_paper_xsens.pdf.

[22] X. Yun and E. R. Bachmann, **Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking**, *IEEE Transactions on Robotics*, vol. 22, no. 6, 1216–1227. DOI: 10.1109/TRO.2006.886270. [Online]. Available: https://doi.org/10.1109/TRO.2006.886270.

[23] T. Lisini Baldi, F. Farina, A. Garulli, A. Giannitrapani, and D. Prattichizzo, **Upper body pose estimation using wearable inertial sensors and multiplicative Kalman filter**, *IEEE Sensors Journal*, vol. 20, no. 1, 492–500. DOI: 10.1109/JSEN.2019.2940612. [Online]. Available: https://doi.org/10.1109/JSEN.2019.2940612.

[24] Y. Tian, H. Wei, and J. Tan, **An adaptive-gain complementary filter for real-time human motion tracking with MARG sensors in free-living environments**, *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 2, 254–264, ISSN: 1534-4320. DOI: 10.1109/TNSRE.2012.2205706. [Online]. Available: https://doi.org/10.1109/TNSRE.2012.2205706.

[25] A. Gallagher, Y. Matsuoka, and W.-T. Ang, **An efficient real-time human posture tracking algorithm using low-cost inertial and magnetic sensors**, in *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, IEEE/RSJ, IEEE, 2004, 2967–2972. DOI: 10.1109/IROS.2004.1389860. [Online]. Available: https://doi.org/10.1109/IROS.2004.1389860.

[26] N. Miller, O. C. Jenkins, M. Kallmann, and M. J. Mataric, **Motion capture from inertial sensing for untethered humanoid teleoperation**, in *4th IEEE/RAS International Conference on Humanoid Robots (Humanoids 2004)*, vol. 2, IEEE/RAS, IEEE, 2004, 547–565. DOI: 10.1109/ICHR.2004.1442670. [Online]. Available: https://doi.org/10.1109/ICHR.2004.1442670.

[27] Z.-Q. Zhang and J.-K. Wu, **A novel hierarchical information fusion method for three-dimensional upper limb motion estimation**, *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 11, 3709–3719. DOI: 10.1109/TIM.2011.2135070. [Online]. Available: https://doi.org/10.1109/TIM.2011.2135070.

[28] M. Kok, J. D. Hol, and T. B. Schön, **An optimization-based approach to human body motion capture using inertial sensors**, in, vol. 47, ser. IFAC Proceedings Volumes 3, Elsevier, 2014, 79–85. DOI: 10.3182/20140824-6-ZA-1003.02252. [Online]. Available: https://doi.org/10.3182/20140824-6-ZA-1003.02252.

[29] M. Miezal, B. Taetz, and G. Bleser, **On inertial body tracking in the presence of model calibration errors**, *Sensors*, vol. 16, no. 7, 1132. DOI: 10.3390/s16071132. [Online]. Available: https://doi.org/10.3390/s16071132.

[30] R. Mahony and J. Trumpf, **Equivariant filter design for kinematic systems on Lie groups**, *IFAC-PapersOnLine*, vol. 54, no. 9, 253–260. DOI: 10.1016/j.ifacol.2021.06.148. [Online]. Available: https://doi.org/10.1016/j.ifacol.2021.06.148.

[31] R. Mahony and T. Hamel, **A geometric nonlinear observer for simultaneous localisation and mapping**, in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, IEEE, IEEE, 2017, 2408–2415. DOI: 10.1109/CDC.2017.8264002. [Online]. Available: https://doi.org/10.1109/CDC.2017.8264002.

[32] R. Mahony, T. Hamel, and J. Trumpf, **An homogeneous space geometry for simultaneous localisation and mapping**, *Annual Reviews in Control*, vol. 51, 254–267. DOI: 10.1016/j.arcontrol.2021.04.012. [Online]. Available: https://doi.org/10.1016/j.arcontrol.2021.04.012.

[33] P. van Goor, R. Mahony, T. Hamel, and J. Trumpf, **A geometric observer design for visual localisation and mapping**, in *2019 IEEE 58th Conference on Decision and Control (CDC)*, IEEE, IEEE, 2019, 2543–2549. DOI: 10.1109/CDC40024.2019.9029435. [Online]. Available: https://doi.org/10.1109/CDC40024.2019.9029435.

[34] N. van der Laan, M. Cohen, J. Arsenault, and J. R. Forbes, **The invariant Rauch–Tung–Striebel smoother**, *IEEE Robotics and Automation Letters*, vol. 5, no. 4, 5067–5074. DOI: 10.1109/LRA.2020.3005132. [Online]. Available: https://doi.org/10.1109/LRA.2020.3005132.

[35] R. Mahony, T. Hamel, and J.-M. Pflimlin, **Nonlinear complementary filters on the special orthogonal group**, *IEEE Transactions on Automatic Control*, vol. 53, no. 5, 1203–1218. DOI: 10.1109/TAC.2008.923738. [Online]. Available: https://doi.org/10.1109/TAC.2008.923738.

[36] S. Bonnabel and P. Rouchon, "On invariant observers," in *Control and Observer Design for Nonlinear Finite and Infinite Dimensional Systems*, ser. Lecture Notes in Control and Information Sciences, T. Meurer, K. Graichen, and E. D. Gilles, Eds., vol. 322, Springer, 2005, 53–65. DOI: 10.1007/11529798_4. [Online]. Available: https://doi.org/10.1007/11529798_4.

[37] A. Barrau, **Non-linear state error based extended Kalman filters with applications to navigation**, Ph.D. dissertation, MINES ParisTech, Paris, France, 2015.

[38] A. Barrau and S. Bonnabel, **The invariant extended Kalman filter as a stable observer**, *IEEE Transactions on Automatic Control*, vol. 62, no. 4, 1797–1812. DOI: 10.1109/TAC.2016.2594085.

[39] A. Barrau and S. Bonnabel, **Invariant Kalman filtering**, *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, no. 1, 237–257. DOI: 10.1146/annurev-control-060117-105010.

[40] S. Bonnabel, P. Martin, and E. Salaün, **Invariant extended Kalman filter: Theory and application to a velocity-aided attitude estimation problem**, in *Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with the 2009 28th Chinese Control Conference (CCC)*, IEEE, IEEE, 2009, 1297–1304. DOI: 10.1109/CDC.2009.5400372. [Online]. Available: https://doi.org/10.1109/CDC.2009.5400372.

[41] A. Barrau and S. Bonnabel, **Intrinsic filtering on Lie groups with applications to attitude estimation**, *IEEE Transactions on Automatic Control*, vol. 60, no. 2, 436–449. DOI: 10.1109/TAC.2014.2342911. [Online]. Available: https://doi.org/10.1109/TAC.2014.2342911.

[42] E. R. Potokar, K. Norman, and J. G. Mangelson, **Invariant extended Kalman filtering for underwater navigation**, *IEEE Robotics and Automation Letters*, vol. 6, no. 3, 5792–5799. DOI: 10.1109/LRA.2021.3085167. [Online]. Available: https://doi.org/10.1109/LRA.2021.3085167.

[43] 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)*, IEEE/RSJ, IEEE, 2017, 1578–1585. DOI: 10.1109/IROS.2017.8205965. [Online]. Available: https://doi.org/10.1109/IROS.2017.8205965.

[44] A. Barrau and S. Bonnabel, **The geometry of navigation problems**, *IEEE Transactions on Automatic Control*, vol. 68, no. 2, 689–704. DOI: 10.1109/TAC.2022.3144328.

[45] M. Brossard, S. Bonnabel, and A. Barrau, **Invariant Kalman filtering for visual inertial SLAM**, in *2018 21st International Conference on Information Fusion (FUSION)*, IEEE, IEEE, 2018, 2021–2028. DOI: 10.23919/ICIF.2018.8455807. [Online]. Available: https://doi.org/10.23919/ICIF.2018.8455807.

[46] Y. Song, Z. Zhang, J. Wu, Y. Wang, L. Zhao, and S. Huang, **A right invariant extended Kalman filter for object based SLAM**, *IEEE Robotics and Automation Letters*, vol. 7, no. 2, 1316–1323. DOI: 10.1109/LRA.2021.3139370. [Online]. Available: https://doi.org/10.1109/LRA.2021.3139370.

[47] S. Heo and C. G. Park, **Consistent EKF-based visual-inertial odometry on matrix Lie group**, *IEEE Sensors Journal*, vol. 18, no. 9, 3780–3788. DOI: 10.1109/JSEN.2018.2808330. [Online]. Available: https://doi.org/10.1109/JSEN.2018.2808330.

[48] N. Pavlasek, A. Walsh, and J. R. Forbes, **Invariant extended Kalman filtering using two position receivers for extended pose estimation**, in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021, 5582–5588. DOI: 10.1109/ICRA48506.2021.9561150. [Online]. Available: https://doi.org/10.1109/ICRA48506.2021.9561150.

[49] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, **Contact-aided invariant extended Kalman filtering for robot state estimation**, *The International Journal of Robotics Research*, vol. 39, no. 4, 402–430. DOI: 10.1177/0278364919894385. [Online]. Available: https://doi.org/10.1177/0278364919894385.

[50] D. Youm, H. Oh, S. Choi, H. Kim, S. Jeon, and J. Hwangbo, **Legged robot state estimation with invariant extended Kalman filter using neural measurement network**, in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, IEEE, 2025, 670–676. DOI: 10.1109/ICRA55743.2025.11127971. [Online]. Available: https://doi.org/10.1109/ICRA55743.2025.11127971.

[51] K.-H. Kim, D. Ahn, D.-h. Lee, J. Yoon, and D. J. Hyun, **Adaptive invariant extended Kalman filter for legged robot state estimation**, in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE/RSJ, IEEE, 2025, 3063–3068. DOI: 10.1109/IROS60139.2025.11247138. [Online]. Available: https://doi.org/10.1109/IROS60139.2025.11247138.

[52] D. P. Bertsekas and J. N. Tsitsiklis, **Introduction to Probability**, 2nd ed. Belmont, MA: Athena Scientific, 2008.

[53] A. Papoulis, **Random Variables and Stochastic Processes**, 1st ed. New York: McGraw-Hill, 1965.

[54] J. K. Blitzstein and J. Hwang, **Introduction to Probability**, 2nd ed. New York: Chapman and Hall/CRC, 2019.

[55] T. D. Barfoot, **State Estimation for Robotics**. Cambridge, UK: Cambridge University Press, 2024.

[56] G. L. Smith, S. F. Schmidt, and L. A. McGee, **Application of Statistical Filter Theory to the Optimal Estimation of Position and Velocity on Board a Circumlunar Vehicle** (NASA Special Publication SP-135). Washington, DC: National Aeronautics and Space Administration, 1962.

[57] L. A. McGee and S. F. Schmidt, **Discovery of the Kalman filter as a practical tool for aerospace and industry**, NASA Ames Research Center, Moffett Field, CA, USA, NASA Technical Memorandum NASA-TM-86847, 1985. [Online]. Available: https://ntrs.nasa.gov/citations/19860003843.

[58] P. Swerling, **First-order error propagation in a stagewise smoothing procedure for satellite observations**, RAND Corporation, Santa Monica, CA, USA, Research Memorandum RM-2329, 1959. [Online]. Available: https://www.rand.org/pubs/research_memoranda/RM2329.html.

[59] T. N. Thiele, "Om anvendelse af mindste kvadraters methode i nogle tilfælde, hvor en komplikation af visse slags uensartede tilfældige fejlkilder giver fejlene en "systematisk" karakter," Danish, in *Det Kongelige Danske Videnskabernes Selskabs Skrifter*, ser. 5. Række, naturvidenskabelig og mathematisk Afdeling 5, vol. 12, Kjøbenhavn: B. Lunos Kgl. Hof-Bogtrykkeri, 1880, 381–408.

[60] A. Hald, **T. N. Thiele's contributions to statistics**, *International Statistical Review / Revue Internationale de Statistique*, vol. 49, no. 1, 1–20. DOI: 10.2307/1403034. [Online]. Available: https://www.jstor.org/stable/1403034.

[61] R. E. Kalman and R. S. Bucy, **New results in linear filtering and prediction theory**, *Journal of Basic Engineering*, vol. 83, no. 1, 95–108. DOI: 10.1115/1.3658902. [Online]. Available: https://doi.org/10.1115/1.3658902.

[62] H. W. Sorenson, "Kalman filtering techniques," in, ser. Advances in Control Systems, C. Leondes, Ed., vol. 3, Elsevier, 1966, 219–292. DOI: https://doi.org/10.1016/B978-1-4831-6716-9.50010-2. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9781483167169500102.

[63] H. W. Sorenson, **Least-squares estimation: From Gauss to Kalman**, *IEEE Spectrum*, vol. 7, no. 7, 63–68. DOI: 10.1109/MSPEC.1970.5213471. [Online]. Available: https://doi.org/10.1109/MSPEC.1970.5213471.

[64] G. Welch and G. Bishop, **An introduction to the Kalman filter**, University of North Carolina at Chapel Hill, Department of Computer Science, Chapel Hill, NC, USA, Technical Report TR 95-041, 1995. [Online]. Available: https://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf.

[65] G. Bishop and G. Welch, **An introduction to the Kalman filter**, in *ACM SIGGRAPH 2001 Courses*, Course 8, Los Angeles, CA, USA: ACM, 2001, 1–41. [Online]. Available: https://history.siggraph.org/learning/an-introduction-to-the-kalman-filter-by-welch-and-bishop/.

[66] P. S. Maybeck, "The Kalman filter: An introduction to concepts," in *Autonomous Robot Vehicles*, I. J. Cox and G. T. Wilfong, Eds., New York, NY, USA: Springer, 1990, 194–204. DOI: 10.1007/978-1-4613-8997-2_15. [Online]. Available: https://doi.org/10.1007/978-1-4613-8997-2_15.

[67] S. J. Julier and J. K. Uhlmann, **A counter example to the theory of simultaneous localization and map building**, in *Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, IEEE, Seoul, South Korea, 2001, 4238–4243. DOI: 10.1109/ROBOT.2001.933280.

[68] J. A. Castellanos, J. Neira, and J. D. Tardós, **Limits to the consistency of EKF-based SLAM**, *IFAC Proceedings Volumes*, vol. 37, no. 8, 716–721. DOI: 10.1016/S1474-6670(17)32063-3.

[69] S. Huang and G. Dissanayake, **Convergence and consistency analysis for extended Kalman filter based SLAM**, *IEEE Transactions on Robotics*, vol. 23, no. 5, 1036–1049. DOI: 10.1109/TRO.2007.903811.

[70] R. P. Wishner, J. A. Tabaczynski, and M. Athans, **A comparison of three non-linear filters**, *Automatica*, vol. 5, no. 4, 487–496. DOI: 10.1016/0005-1098(69)90110-1. [Online]. Available: https://doi.org/10.1016/0005-1098(69)90110-1.

[71] A. H. Jazwinski, **Stochastic Processes and Filtering Theory** (Dover Books on Electrical Engineering). Mineola, NY: Dover Publications, 2007, Unabridged republication of the 1970 original (Academic Press)., ISBN: 978-0-486-46274-5.

[72] B. M. Bell and F. W. Cathey, **The iterated Kalman filter update as a Gauss–Newton method**, *IEEE Transactions on Automatic Control*, vol. 38, no. 2, 294–297. DOI: 10.1109/9.250476. [Online]. Available: https://doi.org/10.1109/9.250476.

[73] C. F. Gauss, **Theoria motus corporum coelestium in sectionibus conicis solem ambientium**, Latin. Hamburgi: Sumptibus Frid. Perthes et I. H. Besser, 1809. [Online]. Available: https://archive.org/details/bub_gb_ORUOAAAAQAAJ.

[74] J. J. Moré and D. C. Sorensen, **Newton's method**, Argonne National Laboratory, Argonne, IL, USA, Tech. Rep. ANL-82-8, 1982. DOI: 10.2172/5326201. [Online]. Available: https://www.osti.gov/biblio/5326201.

[75] J. Dennis John E. and R. B. Schnabel, **Numerical Methods for Unconstrained Optimization and Nonlinear Equations** (Classics in Applied Mathematics). Philadelphia, PA: SIAM, 1996, Corrected reprint of the 1983 Prentice-Hall edition., ISBN: 978-0-89871-364-0. DOI: 10.1137/1.9781611971200. [Online]. Available: https://doi.org/10.1137/1.9781611971200.

[76] P. van Goor, T. Hamel, and R. Mahony, **Equivariant filter (EqF)**, *IEEE Transactions on Automatic Control*, vol. 68, no. 6, 3501–3512. DOI: 10.1109/TAC.2022.3194094. [Online]. Available: https://doi.org/10.1109/TAC.2022.3194094.

[77] P. van Goor, T. Hamel, and R. Mahony, **Equivariant filter (EqF): A general filter design for systems on homogeneous spaces**, in *2020 59th IEEE Conference on Decision and Control (CDC)*, IEEE, IEEE, 2020, 5401–5408. DOI: 10.1109/CDC42340.2020.9303813. [Online]. Available: https://doi.org/10.1109/CDC42340.2020.9303813.

[78] A. Fornasier, Y. Ge, P. van Goor, R. Mahony, and S. Weiss, **Equivariant symmetries for inertial navigation systems**, *Automatica*, vol. 181, 112495. DOI: 10.1016/j.automatica.2025.112495. [Online]. Available: https://doi.org/10.1016/j.automatica.2025.112495.

[79] A. Fornasier, Y. Ge, P. van Goor, M. Scheiber, A. Tridgell, R. Mahony, and S. Weiss, **An equivariant approach to robust state estimation for the ArduPilot autopilot system**, in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, IEEE, 2024, 11956–11962. DOI: 10.1109/ICRA57147.2024.10611108. [Online]. Available: https://doi.org/10.1109/ICRA57147.2024.10611108.

[80] P. van Goor and R. Mahony, **EqVIO: An equivariant filter for visual-inertial odometry**, *IEEE Transactions on Robotics*, vol. 39, no. 5, 3567–3585. DOI: 10.1109/TRO.2023.3289587. [Online]. Available: https://doi.org/10.1109/TRO.2023.3289587.

[81] P. van Goor, R. Mahony, T. Hamel, and J. Trumpf, **Constructive observer design for visual simultaneous localisation and mapping**, *Automatica*, vol. 132, 109803. DOI: 10.1016/j.automatica.2021.109803. [Online]. Available: https://doi.org/10.1016/j.automatica.2021.109803.

[82] Y. Ge, A. Pearce, P. van Goor, and R. Mahony, **Equivariant filter design for range-only SLAM**, *arXiv preprint arXiv:2503.03973*. DOI: 10.48550/arXiv.2503.03973. [Online]. Available: https://arxiv.org/abs/2503.03973.

[83] E. J. Lefferts, F. L. Markley, and M. D. Shuster, **Kalman filtering for spacecraft attitude estimation**, *Journal of Guidance, Control, and Dynamics*, vol. 5, no. 5, 417–429. DOI: 10.2514/3.56190. [Online]. Available: https://doi.org/10.2514/3.56190.

[84] F. Li and L. Chang, **MEKF with navigation frame attitude error parameterization for INS/GPS**, *IEEE Sensors Journal*, vol. 20, no. 3, 1536–1549. DOI: 10.1109/JSEN.2019.2947456. [Online]. Available: https://doi.org/10.1109/JSEN.2019.2947456.

[85] P. Martin and E. Salaün, **Generalized multiplicative extended Kalman filter for aided attitude and heading reference system**, in *AIAA Guidance, Navigation, and Control Conference*, American Institute of Aeronautics and Astronautics (AIAA), 2010, 8300. DOI: 10.2514/6.2010-8300. [Online]. Available: https://doi.org/10.2514/6.2010-8300.

[86] M. Ghobadi, P. Singla, and E. T. Esfahani, **Robust attitude estimation from uncertain observations of inertial sensors using covariance inflated multiplicative extended Kalman filter**, *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 1, 209–217. DOI: 10.1109/TIM.2017.2761230. [Online]. Available: https://doi.org/10.1109/TIM.2017.2761230.

[87] G. Bourmaud, R. Mégret, M. Arnaudon, and A. Giremus, **Continuous-discrete extended Kalman filter on matrix Lie groups using concentrated Gaussian distributions**, *Journal of Mathematical Imaging and Vision*, vol. 51, no. 2, 209–228. DOI: 10.1007/s10851-014-0517-0. [Online]. Available: https://doi.org/10.1007/s10851-014-0517-0.

[88] G. Bourmaud, R. Mégret, A. Giremus, and Y. Berthoumieu, **Discrete extended Kalman filter on Lie groups**, in *21st European Signal Processing Conference (EUSIPCO 2013)*, IEEE, 2013, 1–5. [Online]. Available: https://www.eurasip.org/Proceedings/Eusipco/Eusipco2013/papers/1569743485.pdf.

[89] G. Bourmaud, R. Mégret, A. Giremus, and Y. Berthoumieu, **From intrinsic optimization to iterated extended Kalman filtering on Lie groups**, *Journal of Mathematical Imaging and Vision*, vol. 55, no. 3, 284–303. DOI: 10.1007/s10851-015-0622-8. [Online]. Available: https://doi.org/10.1007/s10851-015-0622-8.

[90] M. Brossard, S. Bonnabel, and J.-P. Condomines, **Unscented Kalman filtering on Lie groups**, in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE/RSJ, IEEE, 2017, 2485–2491. DOI: 10.1109/IROS.2017.8206066. [Online]. Available: https://doi.org/10.1109/IROS.2017.8206066.

[91] M. Brossard, S. Bonnabel, and A. Barrau, **Unscented Kalman filter on Lie groups for visual inertial odometry**, in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE/RSJ, IEEE, 2018, 649–655. DOI: 10.1109/IROS.2018.8593627. [Online]. Available: https://doi.org/10.1109/IROS.2018.8593627.

[92] B. C. Hall, **Lie Groups, Lie Algebras, and Representations: An Elementary Introduction** (Graduate Texts in Mathematics), 2nd ed. Cham: Springer, 2015, vol. 222, ISBN: 978-3-319-13466-6. DOI: 10.1007/978-3-319-13467-3.

[93] J. E. Humphreys, **Introduction to Lie Algebras and Representation Theory** (Graduate Texts in Mathematics). New York: Springer, 2012, vol. 9.

[94] M. Brossard, A. Barrau, P. Chauchat, and S. Bonnabel, **Associating uncertainty to extended poses for on Lie group IMU preintegration with rotating Earth**, *IEEE Transactions on Robotics*, vol. 38, no. 2, 998–1015.

[95] G. S. Chirikjian, **Stochastic Models, Information Theory, and Lie Groups, Volume 1: Classical Results and Geometric Methods** (Applied and Numerical Harmonic Analysis). New York, NY, USA: Springer Science & Business Media, 2009, ISBN: 978-0-8176-4802-2. DOI: 10.1007/978-0-8176-4803-9.

[96] Y. Wang and G. S. Chirikjian, **Error propagation on the Euclidean group with applications to manipulator kinematics**, *IEEE Transactions on Robotics*, vol. 22, no. 4, 591–602. DOI: 10.1109/TRO.2006.878978.

[97] K. C. Wolfe, M. Mashner, and G. S. Chirikjian, **Bayesian fusion on Lie groups**, *Journal of Algebraic Statistics*, vol. 2, no. 1, 75–97.

[98] S. Goffin, S. Bonnabel, O. Brüls, and P. Sacré, **Invariant Kalman filtering with noise-free pseudo-measurements**, in *2023 62nd IEEE Conference on Decision and Control (CDC)*, IEEE, 2023, 8665–8671. DOI: 10.1109/CDC49753.2023.10383262.

[99] S. Goffin, A. Barrau, S. Bonnabel, O. Brüls, and P. Sacré, **Iterated invariant extended Kalman filter (IterIEKF)**, *IEEE Transactions on Automatic Control*, vol. PP, no. 99, Early Access, 1–8. DOI: 10.1109/TAC.2025.3637661.