Article

The International Journal of Robotics Research
2019, Vol. 39(4) 402–430

DOI: 10.1177/0278364919894385
journals.sagepub.com/home/ijr

Contact-aided invariant extended Kalman filtering for robot state estimation

Ross Hartley, Maani Ghaffari, Ryan M Eustice and Jessy W Grizzle

Abstract

Legged robots require knowledge of pose and velocity in order to maintain stability and execute walking paths. Current solutions either rely on vision data, which is susceptible to environmental and lighting conditions, or fusion of kinematic and contact data with measurements from an inertial measurement unit (IMU). In this work, we develop a contact-aided invariant extended Kalman filter (InEKF) using the theory of Lie groups and invariant observer design. This filter combines contact-inertial dynamics with forward kinematic corrections to estimate pose and velocity along with all current contact points. We show that the error dynamics follows a log-linear autonomous differential equation with several important consequences: (a) the observable state variables can be rendered convergent with a domain of attraction that is independent of the system's trajectory; (b) unlike the standard EKF, neither the linearized error dynamics nor the linearized observation model depend on the current state estimate, which (c) leads to improved convergence properties and (d) a local observability matrix that is consistent with the underlying nonlinear system. Furthermore, we demonstrate how to include IMU biases, add/remove contacts, and formulate both world-centric and robo-centric versions. We compare the convergence of the proposed InEKF with the commonly used quaternion-based extended Kalman filter (EKF) through both simulations and experiments on a Cassie-series bipedal robot. Filter accuracy is analyzed using motion capture, while a LiDAR mapping experiment provides a practical use case. Overall, the developed contact-aided InEKF provides better performance in comparison with the quaternion-based EKF as a result of exploiting symmetries present in system.

Keywords

Invariant extended Kalman filter, invariant observer, Kalman filter, legged robots, inertial navigation, contact-aided navigation, state estimation, SLAM, Cassie robot, Lie groups

1. Introduction

Legged robots have the potential to transform the logistics and package delivery industries, become assistants in our homes, and aide in search and rescue (Clifford, 2018). In order to develop motion planning algorithms and robust feedback controllers for these tasks, accurate estimates of the robot's state are needed. Some states, such as joint angles, can be directly measured using encoders, while other states, such as the robot's pose and velocity, require additional sensors. Most legged robots are equipped with an inertial measurement unit (IMU) that can measure linear acceleration and angular velocity, albeit with noise and bias perturbations. Consequently, nonlinear observers are typically used to fuse leg odometry and inertial measurements to infer trajectory, velocity, and calibration parameters (Benallegue and Lamiraux, 2015; Eljaik et al., 2015; Kuindersma et al., 2016; Rotella et al., 2014). In view of a practical solution, designing a globally convergent observer is sacrificed for one with at best local properties, such as the extended Kalman filter (EKF) (Grizzle and Song, 1995; Krener, 2003; Sola, 2017; Trawny and Roumeliotis, 2005). This EKF-based approach is computationally efficient and easily customizable, allowing successful implementation on a number of legged robots with rigorous real-time performance requirements (Bloesch et al., 2012; Bloesch, 2017; Fallon et al., 2014; Nobili et al., 2017).

Accurate pose estimation can be combined with visual data to build maps of the environment (Fankhauser et al., 2014). These maps can be used in gait selection to improve the stability of a robot while walking on uneven terrains and as a basis for high-level motion planning. Although there have been many recent advancements in visual–inertial

Robotics Institute and College of Engineering, University of Michigan, Ann Arbor, MI, USA

Corresponding author:
Ross Hartley, Robotics Institute, University of Michigan, Ann Arbor, MI 48109, USA.
Email: rosshart@umich.edu

odometry and simultaneous localization and mapping (SLAM) (Forster et al., 2017; Huai and Huang, 2018; Qin et al., 2018), these algorithms often rely on visual data for pose estimation. This means that the observer (and ultimately the feedback controller) can be adversely affected by rapid changes in lighting as well as the operating environment. It is therefore beneficial to develop a low-level state estimator that fuses data only from proprioceptive sensors to form accurate high-frequency state estimates. This approach was taken by Bloesch et al. (2012) when developing a quaternion-based extended Kalman filter (QEKF) that combines inertial, contact, and kinematic data to estimate the robot's base pose, velocity, and a number of contact states. In this article, we expand upon these ideas to develop an invariant extended Kalman filter (InEKF) that has improved convergence and consistency properties allowing for a more robust observer that is suitable for long-term autonomy.

The theory of invariant observer design is based on the estimation error being invariant under the action of a matrix Lie group (Aghannan and Rouchon, 2002; Bonnabel et al., 2009), which has recently led to the development of the InEKF[1] (Barrau, 2015; Barrau and Bonnabel, 2017, 2018; Bonnabel, 2007) with successful applications and promising results in SLAM (Barrau, 2015; Zhang et al., 2017) and aided inertial navigation systems (INSs) (Barczyk and Lynch, 2011, 2013; Barrau, 2015; Wu et al., 2017). The invariance of the estimation error with respect to a Lie group action is referred to as a symmetry of the system (Barczyk and Lynch, 2013). Summarized briefly, Barrau and Bonnabel (2017) showed that if the state is defined on a Lie group, and the dynamics satisfy a particular "group affine" property, then the symmetry leads to the estimation error satisfying a "log-linear" autonomous differential equation on the Lie algebra. In the deterministic case, this linear system can be used to exactly recover the estimated state of the nonlinear system as it evolves on the group. The log-linear property therefore allows the design of a nonlinear observer or state estimator with strong convergence properties.

1.1. Contribution

In this article, we derive an invariant extended Kalman filter (InEKF) for a system containing IMU and contact sensor dynamics, with forward kinematic correction measurements. We show that the defined deterministic system satisfies the "group affine" property, and therefore, the error dynamics is exactly log-linear. In practice, with the addition of sensor noise and IMU bias, this log-linear error system is only approximate; we show, however, that in many cases the proposed InEKF is still preferred over standard quaternion-based extended Kalman filters (QEKFs) owing to superior convergence and consistency properties. We demonstrate the utility and accuracy of the developed observer through a series of LiDAR mapping experiments performed on a Cassie-series biped robot.

In summary, this work makes the following contributions:

1. derivation of a continuous-time right-invariant EKF for an IMU/contact process model with a forward kinematic measurement model; the observability analysis is also presented;
2. state augmentation with IMU biases;
3. evaluations of the derived observer in simulation and hardware experiments using a 3D bipedal robot;
4. alternative derivation of the observer using a left-invariant error definition;
5. detailed explanation of the connection between the invariant error choice and the world-centric and robo-centric estimator formulations;
6. equations for the analytical discretization of the proposed observers; and
7. development of an open-source C++ library for aided-inertial navigation using the InEKF https://github.com/RossHartley/invariant-ekf.

1.2. Outline

The remainder of this article is organized as follows. Background and related work are given in Section 2. Section 3 provides the necessary preliminary material needed for understanding the InEKF formulation, which is motivated by an example from attitude dynamics in Section 4. Section 5 provides the derivation of a right-invariant extended Kalman filter (RIEKF) for contact-inertial navigation with a right-invariant forward kinematic measurement model. In Section 6, we present simulation results comparing the convergence properties with a state-of-the-art QEKF. Section 7 discusses the state augmentation of the previously derived InEKF with IMU bias. The consequences of adding and removing of contact points in the estimator are described in Section 8. Experimental evaluations on a 3D biped robot, shown in Figure 1, are presented in Section 9 along with an application towards LiDAR-based terrain mapping. Section 10 provides an alternative derivation of the proposed observer using the left-invariant dynamics along with an explanation of how to easily switch between the two formulations. Section 11 details how these equations can be modified to create a "robot-centric" estimator. Section 12 itemizes additional sensor measurements that fit within the InEKF framework and describes the relation between the developed filter and landmark-based simultaneous localization and mapping (SLAM). Finally, Section 13 concludes the article and suggests future directions. Details about time-discretization and useful Lie group expressions are given in the appendix.

2. Background and related work

In this section, we first review the Kalman filtering literature to locate the proposed state estimator within the related

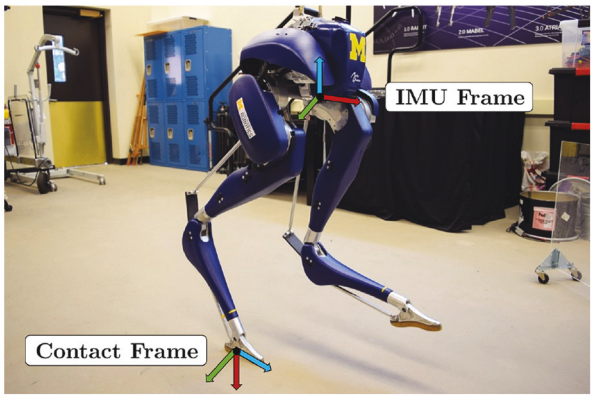

Fig. 1. A Cassie-series biped robot is used for both simulation and experimental results. The robot was developed by Agility Robotics and has 20 degrees of freedom, 10 actuators, joint encoders, and an inertial measurement unit (IMU). The contact and IMU frames used in this work are depicted above.

work. Then, we review the state estimation techniques for humanoid and legged robots.

2.1. Kalman filtering

Filtering methods involve estimating the robot's current state (and potentially landmarks) using the set of all measurements up to the current time (Anderson and Moore, 1979; Grewal, 2011; Smith and Singh, 2006). When the process model and measurements are linear, and the noise is white and Gaussian, Kalman filtering (Kalman, 1960) provides an optimal method (minimum mean squared error) for state estimation. The general process for Kalman filtering involves two phases, propagation and correction. The state is typically represented using a Gaussian random vector, which is parameterized by a mean and a covariance. During the propagation phase, the previous state and covariance estimate are propagated forward in time using the system dynamics (alternatively known as the process model). When a measurement is obtained, the state and covariance estimate are corrected using the measurement model along with an associated measurement noise covariance.

Although the Kalman filter provides a method for optimal linear filtering, most practical mobile robots have nonlinear system dynamics, and many useful sensor models are also nonlinear. For these cases, an EKF can be designed, which utilizes Taylor series expansions to linearize the process and measurement models around the current state estimate (Thrun et al., 2005). Owing to its low computational complexity and accurate performance, the EKF quickly became the de facto standard of nonlinear filtering for many mobile robot applications, including wheeled vehicles, drones, and legged robots (Bloesch et al., 2012; Brossard et al., 2019; Sebesta and Boizot, 2014). The EKF has also been proposed to solve the SLAM problem (Smith et al., 1990). However, because the nonlinear system is linearized about the current state estimate, the EKF is, at best, only a locally stable observer (Krener, 2003; Song and Grizzle, 1992). The local convergence proofs are based on Lipschitz bounds of the nonlinear terms in a model, and hence "the more nonlinear a system is, the worse an EKF may perform." Importantly, if the state estimate is initialized poorly, it is possible for the filter to diverge. In addition, because an EKFs uses a systems linearization about the current estimate, states that are unobservable can spuriously be treated as observable by the filter. While this can be mitigated through the use of an observability-constrained EKF developed by Huang et al. (2010), it cannot be altogether avoided.

For many systems, the *error-state* (or *indirect*) EKF offers superior performance to the standard (total state or direct) form. As the name implies, the error-state extended Kalman filter (ErEKF) is formulated using the errors, such as pose and velocity errors, as the filter variables, whereas the standard EKF tracks the states themselves (pose and velocity directly) (Madyastha et al., 2011; Roumeliotis et al., 1999; Sola, 2017; Trawny and Roumeliotis, 2005). Under small noise assumptions, this leads to linear error dynamics which are then used for covariance propagation in the error-state filter. The measurement model is also rewritten with respect to these errors. Although the error dynamics are linear in the error variables, they may still depend on the current state estimate, which if initialized poorly, will degrade the performance of the filter. However, the approximately linear nature of the error dynamics may respect the linear assumptions of the original Kalman filter better than the underlying system dynamics, which can lead to improved performance (Madyastha et al., 2011).

Perhaps the most important feature of the ErEKF is the ability to circumvent dynamic modeling (Roumeliotis et al., 1999). This is done by replacing the potentially complicated process model with a relatively simple IMU integration model (also known as *strapdown* modeling) (Merhav and Koifman, 1991; Titterton et al., 2004;Woodman, 2007). Essentially, the IMU's angular velocity and linear acceleration measurements are integrated to propagate the state estimate, whereas the covariance is propagated using the error dynamics. Additional (independent) sensor measurements will correct the estimated error, which can then be used to update the state estimate. Using this method, there is no longer a need to formulate complicated, platform-specific dynamics models, which may require a large number of state variables and is likely to be exceedingly nonlinear. The "strapdown" ErEKFs has proven to yield highly accurate results (even with a low-cost IMUs) and continues to form the basis of many inertial navigation systems (INSs) (Barshan and Durrant-Whyte, 1995; Bloesch et al., 2012; Lefferts et al., 1982; Sola, 2017; Trawny and Roumeliotis, 2005).

In the standard formulation of Kalman filtering theory, the system evolves on Euclidean spaces. However, in many cases, the state variables we are interested in lie on a *manifold*. For example, the orientation of a 3D rigid body is

represented by an element of the *special orthogonal group*, SO(3). This matrix Lie group is defined by the set of orthogonal 3×3 matrices with a determinant of one. The matrix contains nine variables, whereas the dimension of the manifold is only three. One common approach is to parameterize SO(3) using local coordinates such as three Euler or Tait–Bryan angles (Gebre-Egziabher et al., 2004; Setoodeh et al., 2004). This allows the standard ErEKF equations to be applied; however, these local parameterizations are often arbitrary (and therefore confusing) and contain singularities (the well-known Gimbal lock problem). Alternatively, it is possible to represent 3D orientation using quaternions, which are a four-dimensional double cover (a two to one diffeomorphism) of SO(3). Using quaternions eliminates the singularities; however, modifications to the standard ErEKF equations have to be made (Sola, 2017; Trawny and Roumeliotis, 2005). In brief, while the orientation is represented by a four-dimensional quaternion, the orientation error has to be defined by a $3-$vector (in the Lie algebra of SO(3)) and the associated covariance by a 3×3 matrix in order to prevent degeneracy. In addition, the orientation corrections are done through quaternion multiplication instead of vector addition. This QEKF[2] has been well studied and implemented on a number of platforms, ranging from spacecraft (Lefferts et al., 1982; Murrell, 1978) to legged robots (Bloesch et al., 2012; Fallon et al., 2014; Rotella et al., 2014).

It turns out, many useful robot states can be characterized using matrix Lie groups. Examples include 2D orientation, SO(2), 3D orientation, SO(3), and 3D pose (orientation and position), SE(3). If the state to be estimated is a matrix Lie group, it is possible to further improve the EKF filtering approach. Bourmaud et al. (2015, 2013) developed versions of both discrete and continuous-time EKFs for systems where the state dynamics and measurements evolve on matrix Lie groups. In these formulations, noise is represented as a *concentrated Gaussian on a Lie group* (Wang and Chirikjian, 2006, 2008), which is a generalization of the multivariate Gaussian distribution. In essence, noise is represented as a Gaussian in the tangent space about a point on the manifold. This noise is then mapped to the Lie group through the use of the group's exponential map, resulting in a decidedly non-Gaussian distribution on the manifold. An improved state estimate is obtained owing to the filter taking into account the geometry and structure of the problem (Bourmaud et al., 2015).

Most recently, a new type of EKF has been developed that is rooted in the theory of invariant observer design, in which the estimation error is invariant under the action of a Lie group (Aghannan and Rouchon, 2002; Bonnabel et al., 2009). This invariance is referred to as the *symmetries* of the system (Barczyk and Lynch, 2013). This work led to the development of the InEKF (Barrau, 2015; Barrau and Bonnabel, 2017, 2018; Bonnabel, 2007), with successful applications and promising results in SLAM (Barrau, 2015; Zhang et al., 2017) and aided INSs (Barczyk and Lynch, 2011, 2013; Barrau, 2015; Bonnable et al., 2009; Wu et al., 2017). Similar to the above-mentioned EKFs on matrix Lie groups, the state is again represented as a matrix Lie group and the noise as a concentrated Gaussian on the group.

However, the InEKF exploits available system symmetries to further improve filtering results. The culminating result of the InEKF states that if a system satisfies a "group-affine" property, the estimation error satisfies a "log-linear" autonomous differential equation on the Lie algebra of the corresponding Lie group (Barrau, 2015; Barrau and Bonnabel, 2018). In other words, the system linearization does not depend upon the estimated states. Therefore, one can design a nonlinear state estimator with strong convergence properties. Surprisingly, many mobile robot state estimation problems can be solved within the InEKF framework. This includes attitude estimation (Bonnabel, 2007), inertial odometry (Barrau, 2015; Barrau and Bonnabel, 2018), velocity-aided inertial navigation (Bonnable et al., 2009), landmark-aided navigation (Barrau, 2015), GPS and magnetometer-aided navigation (Barczyk and Lynch, 2011), and even EKF-based SLAM. In this article, we extend this class of solutions to contact-aided inertial navigation (Hartley et al., 2018b), where forward kinematics is used to correct an inertial and contact-based process model. This approach successfully allows an InEKF to be used for legged robot state estimation.

2.2. Legged robot and humanoid state estimation

Legged robots are a subclass of mobile robots that locomote through direct and switching contact with the environment. These robots typically contain proprioceptive sensors, such as IMUs, joint encoders, and contact sensors. In addition, some legged robots, especially humanoids, also have access to exteroceptive sensors, namely cameras and LiDARs. As with all mobile robots, state estimation for legged robots is critical for mapping, planning, designing feedback controllers, and developing general autonomy. In this section, an overview of notable techniques for legged robot state estimation is given.

The simplest approach for estimation of a legged robot's spatial location and velocity is kinematic dead-reckoning, otherwise known as kinematic odometry. This involves estimating relative transformations using only kinematic and contact measurements. In particular, encoder measurements and the kinematics model are used to track the position, orientation, and velocity of the robot's base frame based on the assumption that a stance foot remains fixed to the ground. Although this method can be easily implemented, the state estimate is typically noisy owing to kinematic modeling errors, encoder noise, and foot slip (Roston and Krotkov, 1991). When only one foot is in contact with the ground, this "static contact assumption" may be violated. For example, if the robot has point feet, the stance foot position may remain fixed, but the foot orientation is free to rotate (without changing the joint angles). Therefore, a gyroscope is

often used to provide angular velocity measurements which allows the robot's body velocity to be recovered.

Alternatively, if the terrain is known a priori and at least three non-collinear point feet are on the ground, Lin et al. (2005) showed that the robot's instantaneous base pose can be computed through kinematics. These kinematic-based methods have been implemented on a number of legged robots including a planar one-legged hopper (Hodgins, 1989), the CMU Ambler hexapod (Roston and Krotkov, 1991), the RHex hexapod (Lin et al., 2005), and the biped robot MARLO (Da et al., 2016, 2017). However, owing to the high levels of noise coming from encoders and foot slip, the velocity estimate typically needs to be heavily filtered before becoming usable in the feedback controller (Hartley et al., 2017). In addition, this noise causes the position and orientation estimates to drift substantially rendering the estimator impractical for mapping and autonomy tasks.

Fortunately, legged robots are often equipped with additional sensors such as IMUs, GPS, cameras, or LiDARs, which can provide independent, noisy odometry measurements. Much of the literature on legged robot state estimation focuses on fusing these measurements (potentially with kinematic odometry) using filtering and smoothing methods. Singh et al. (2006) combined inertial measurements with optical flow measurements in a four-phase hybrid EKF. This required explicit dynamic modeling of the robot in flight, landing, stance, and thrust phases. Lin et al. (2006) took a similar model-based approach and used an EKF to fuse kinematic information with IMU measurements to estimate the state of a hexapod. Cobano et al. (2008) developed an EKF that fuses kinematic odometry and magnetometer readings with position measurements from a GPS to localize a SILO4 quadruped outdoors. This implementation fixes the issues with unbounded drift, but cannot operate in GPS-denied environments. If a prior terrain map is known, Chitta et al. (2007) showed that it is possible to solve the localization problem for legged robots using only proprioceptive sensors and a particle filter. The key idea was that if the robot "senses" a terrain change through kinematics, then this limits the potential locations the robot can be in a known map. The method was demonstrated on the LittleDog quadruped.

A breakthrough came in 2012 when Bloesch et al. (2012) combined inertial and kinematic measurements in an observability-constrained ErEKF using the strapdown IMU modeling approach. In this work, no a priori knowledge of the terrain is assumed, and the IMU integration model completely eliminates the need for dynamic modeling of the robot. Therefore, the derived filter equations are general enough to be used on any legged robot. The key idea was to augment the state vector with the set of all foot positions currently in contact with the environment. During the prediction phase, the foot contact dynamics are assumed to be Brownian motion, which can account for some foot slippage. In the correction phase, forward kinematic position measurements are used to correct the estimated state. This work was conducted on the StarlETH quadruped robot. If the stance feet orientations also remain constant, as is the case for many humanoids, Rotella et al. (2014) showed that this ErEKF can be extended to allow forward kinematic orientation measurements. The same group also formulated a similar unscented Kalman filter that uses forward kinematic velocity measurements to correct inertial predictions and to accurately detect foot slip (Bloesch et al., 2013). A detailed analysis of these filtering techniques combined with methods for incorporating computer vision can be found in Bloesch (2017). Owing to the complexity involved in accurately formulating dynamic models, many groups have since adopted this IMU motion model approach to legged robot state estimation (Fallon et al., 2014; Hartley et al., 2018b; Xinjilefu et al., 2014).

This combined inertial and kinematic filtering approach yields an estimate of the robot's base pose and velocity. However, some legged robots require additional states to be estimated. Hwangbo et al. (2016) formulated a probabilistic contact estimator for cases when contact sensors are unavailable. Xinjilefu et al. (2014) developed a decoupled EKF that is able to estimate the full state of the humanoid robot ATLAS, including base states, joint angles, and joint velocities. Using proprioceptive sensing only, Bloesch et al. (2012) proved that the absolute positions and yaw angles are unobservable. Thus, over time, estimates of these quantities will drift unboundedly. This is unacceptable for global mapping and planning algorithms; however, local elevation maps can still be obtained (Fankhauser et al., 2014). Fallon et al. (2014) proposed a method for drift-free state estimation for the humanoid ATLAS. In their implementation, inertial and kinematic measurements were fused to yield accurate odometry. Point cloud data from a LiDAR sensor was used with a particle filter to localize the robot into a pre-built map. This approach provided corrections of position and yaw to obtain a drift-free estimate of the state. Nobili et al. (2017) took a similar approach but used the iterative closest point (ICP) algorithm to perform LiDAR-based point cloud matching. The algorithm was tested on the HyQ quadruped robot.

In this article, we propose using an InEKF to estimate the base pose and velocity of a general legged robot. The approach we take is most similar to Bloesch et al. (2012), however we model the entire state as a single matrix Lie group as opposed to a decoupled state approach. This allows us to take advantage of the geometry and symmetry of the estimation problem to formulate autonomous error dynamics. In addition, with our formulation, the linearizations are independent of the state estimate resulting in improved convergence properties, especially for poor state initializations. We formulate both world-centric and robocentric state estimators highlighting the relation between the left- and right-invariant error dynamics. In addition, we provide exact analytical time-discretizations of both filters. The implemented filters can be run at high speeds ($>$2,000 Hz) and can be directly used for accurate local

odometry. We demonstrate this idea through a LiDAR terrain mapping application on a Cassie-series biped robot.

3. Theoretical background and preliminaries

We assume a matrix Lie group (Chirikjian, 2011; Hall, 2015) denoted $\mathcal{G}$ and its associated Lie Algebra denoted $\mathfrak{g}$. If elements of $\mathcal{G}$ are $n \times n$ matrices, then so are elements of $\mathfrak{g}$. When doing calculations, it is very convenient to let

$$(\cdot)^\wedge : \mathbb{R}^{\dim \mathfrak{g}} \to \mathfrak{g}$$

be the linear map that takes elements of the tangent space of $\mathcal{G}$ at the identity to the corresponding matrix representation so that the *exponential map of the Lie group*, $\exp : \mathbb{R}^{\dim \mathfrak{g}} \to \mathcal{G}$, is computed by

$$\exp(\boldsymbol{\xi}) = \exp_m(\boldsymbol{\xi}^\wedge)$$

where $\exp_m(\cdot)$ is the usual exponential of $n \times n$ matrices.

A process dynamics evolving on the Lie group with state at time t, $\mathbf{X}_t \in \mathcal{G}$, is denoted by

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{X}_t = f_{u_t}(\mathbf{X}_t)$$

and $\bar{\mathbf{X}}_t$ is used to denote an estimate of the state. The state estimation error is defined using right or left multiplication of $\mathbf{X}_t^{-1}$ as follows.

Definition 1 (Left- and right-invariant error)**.** The right- and left-invariant errors between two trajectories $\mathbf{X}_t$ and $\bar{\mathbf{X}}_t$ are

$$\begin{aligned} \boldsymbol{\eta}_t^r &= \bar{\mathbf{X}}_t \mathbf{X}_t^{-1} = (\bar{\mathbf{X}}_t \mathbf{L})(\mathbf{X}_t \mathbf{L})^{-1} \quad \text{(right-invariant)} \\ \boldsymbol{\eta}_t^l &= \mathbf{X}_t^{-1} \bar{\mathbf{X}}_t = (\mathbf{L}\bar{\mathbf{X}}_t)^{-1}(\mathbf{L}\mathbf{X}_t), \quad \text{(left-invariant)} \end{aligned} \tag{1}$$

where $\mathbf{L}$ is an arbitrary element of the group.

The following two theorems are the fundamental results for deriving an InEKF and show that by correct parametrization of the error variable, a wide range of nonlinear problems can lead to linear error equations.

Theorem 1 (Autonomous error dynamics (Barrau and Bonnabel, 2017))**.** *A system is group affine if the dynamics, $f_{u_t}(\cdot)$, satisfies*

$$f_{u_t}(\mathbf{X}_1\ \mathbf{X}_2) = f_{u_t}(\mathbf{X}_1)\mathbf{X}_2 + \mathbf{X}_1 f_{u_t}(\mathbf{X}_2) - \mathbf{X}_1 f_{u_t}(\boldsymbol{I}_d)\mathbf{X}_2 \tag{2}$$

for all $t > 0$ and $\mathbf{X}_1, \mathbf{X}_2 \in \mathcal{G}$. Furthermore, if this condition is satisfied, the right- and left-invariant error dynamics are trajectory independent and satisfy

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\eta}_t^r &= g_{u_t}(\boldsymbol{\eta}_t^r) \quad \textit{where} \quad g_{u_t}(\boldsymbol{\eta}^r) = f_{u_t}(\boldsymbol{\eta}^r) - \boldsymbol{\eta}^r f_{u_t}(\boldsymbol{I}_d) \\ \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\eta}_t^l &= g_{u_t}(\boldsymbol{\eta}_t^l) \quad \textit{where} \quad g_{u_t}(\boldsymbol{\eta}^l) = f_{u_t}(\boldsymbol{\eta}^l) - f_{u_t}(\boldsymbol{I}_d)\boldsymbol{\eta}^l \end{aligned}$$

In the above, $\boldsymbol{I}_d \in \mathcal{G}$ denotes the group identity element; to avoid confusion, we use $\mathbf{I}$ for a 3×3 identity matrix, and $\mathbf{I}_n$ for the $n \times n$ case. The following statements hold for both the left- and right-invariant error dynamics.

Define $\mathbf{A}_t$ to be a $\dim\mathfrak{g} \times \dim\mathfrak{g}$ matrix satisfying

$$g_{u_t}(\exp(\boldsymbol{\xi})) \triangleq (\mathbf{A}_t \boldsymbol{\xi})^\wedge + \mathcal{O}(||\boldsymbol{\xi}||^2)$$

For all $t \geq 0$, let $\boldsymbol{\xi}_t$ be the solution of the linear differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\xi}_t = \mathbf{A}_t \boldsymbol{\xi}_t \tag{3}$$

Theorem 2 (Log-linear property of the error (Barrau and Bonnabel, 2017))**.** *Consider the right-invariant error, $\boldsymbol{\eta}_t$, between two trajectories (possibly far apart). For arbitrary initial error $\boldsymbol{\xi}_0 \in \mathbb{R}^{\dim\mathfrak{g}}$, if $\boldsymbol{\eta}_0 = \exp(\boldsymbol{\xi}_0)$, then for all $t \geq 0$,*

$$\boldsymbol{\eta}_t = \exp(\boldsymbol{\xi}_t)$$

that is, the nonlinear estimation error $\boldsymbol{\eta}_t$ can be exactly recovered from the time-varying linear differential equation (3)*.*

This theorem states that (3) is not the typical Jacobian linearization along a trajectory because the (left- or) right-invariant error on the Lie group can be exactly recovered from its solution. This result is of major importance for the propagation (prediction) step of the InEKF (Barrau and Bonnabel, 2017), where in the deterministic case, the log-linear error dynamics allows for exact covariance propagation.

Remark 1. This indirect way of expressing the Jacobian of g_{u_t} is from (Barrau and Bonnabel, 2017); it is used because we are working with a matrix Lie group viewed as an embedded submanifold of a set of $n \times n$ matrices.

During the correction step of a Kalman filter, the error is updated using incoming sensor measurements. If these observations take a particular form, then the linearized observation model and the innovation will also be autonomous (Barrau and Bonnabel, 2017). This happens when the measurement, $\mathbf{Y}_t$, can be written as either

$$\begin{aligned} \mathbf{Y}_t &= \mathbf{X}_t \mathbf{b} + \mathbf{V}_t \quad \text{(left-invariant observation)} \quad \text{or} \\ \mathbf{Y}_t &= \mathbf{X}_t^{-1} \mathbf{b} + \mathbf{V}_t \quad \text{(right-invariant observation)} \end{aligned} \tag{4}$$

where $\mathbf{b}$ is a constant vector and $\mathbf{V}_t$ is a vector of Gaussian noise.

The adjoint representation plays a key role in the theory of Lie groups and through this linear map we can capture the non-commutative structure of a Lie group.

Definition 2 (The adjoint map (Hall, 2015: 63))**.** Let $\mathcal{G}$ be a matrix Lie group with Lie algebra $\mathfrak{g}$. For any $\mathbf{X} \in \mathcal{G}$ the adjoint map, $\mathrm{Ad}_{\mathbf{X}} : \mathfrak{g} \to \mathfrak{g}$, is a linear map defined as $\mathrm{Ad}_{\mathbf{X}}(\boldsymbol{\xi}^\wedge) = \mathbf{X}\boldsymbol{\xi}^\wedge\mathbf{X}^{-1}$. Furthermore, we denote the matrix representation of the adjoint map by $\mathrm{Ad}_{\mathbf{X}}$.

For more details on the material discussed above, along with the theory and proofs about the InEKF, we refer the reader to Barrau (2015); Barrau and Bonnabel (2017, 2018).

4. A motivating example: 3D orientation propagation

Suppose we are interested in estimating the 3D orientation of a rigid body given angular velocity measurements in the body frame, $\tilde{\boldsymbol{\omega}}_t \triangleq \text{vec}(\omega_x, \omega_y, \omega_z) \in \mathbb{R}^3$. This type of measurement can be easily obtained from a gyroscope.

There are several different parameterizations of SO(3); Euler angles, quaternions, and rotation matrices being the most common. If we let $\mathbf{q}_t \triangleq \text{vec}(q_x, q_y, q_z)$ be a vector of Euler angles using the $R_z R_y R_x$ convention, then the orientation dynamics can be expressed as (Diebel, 2006)

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix} = \begin{bmatrix} 1 & \sin(q_x)\tan(q_y) & \cos(q_x)\tan(q_y) \\ 0 & \cos(q_x) & -\sin(q_x) \\ 0 & \sin(q_x)\sec(q_y) & \cos(q_x)\sec(q_y) \end{bmatrix}\begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix}$$

Let $\delta\mathbf{q}_t \triangleq \mathbf{q}_t - \bar{\mathbf{q}}_t \in \mathbb{R}^3$ be the error between the true and estimated Euler angles. The error dynamics can be written as a nonlinear function of the error variable, the inputs, and the state

$$\frac{\mathrm{d}}{\mathrm{d}t}\delta\mathbf{q}_t = g(\delta\mathbf{q}_t, \tilde{\boldsymbol{\omega}}_t, \mathbf{q}_t)$$

In order to propagate the covariance in an EKF, we need to linearize the error dynamics at the current state estimate, $\mathbf{q}_t = \bar{\mathbf{q}}_t$ (i.e., zero error). This leads to a linear error dynamics of the form:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\delta\mathbf{q}_t &\approx \begin{bmatrix} 0 & (\omega_z\bar{c}_x + \omega_y\bar{s}_x)/\bar{c}_y^2 & \bar{t}_y(\omega_y\bar{c}_x - \omega_z\bar{s}_x) \\ 0 & 0 & \omega_z\bar{c}_x + \omega_y\bar{s}_x \\ 0 & (\bar{s}_y(\omega_z\bar{c}_x + \omega_y\bar{s}_x))/\bar{c}_y^2 & (\omega_y\bar{c}_x - \omega_z\bar{s}_x)/\bar{c}_y \end{bmatrix}\delta\mathbf{q}_t \\ &\triangleq \mathbf{A}(\tilde{\boldsymbol{\omega}}_t, \bar{\mathbf{q}}_t)\,\delta\mathbf{q}_t\end{aligned}$$

where $\bar{c}_x$, $\bar{s}_x$, and $\bar{t}_x$ are shorthand for $\cos(\bar{q}_x)$, $\sin(\bar{q}_x)$, and $\tan(\bar{q}_x)$. The linear dynamics matrix clearly depends on the estimated angles, therefore bad estimates will affect the accuracy of the linearization and ultimately the performance and consistency of the filter.

Now instead, let us use a rotation matrix to represent the 3D orientation, $\mathbf{R}_t \in \text{SO}(3)$. The dynamics can now be simply expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t = \mathbf{R}_t(\tilde{\boldsymbol{\omega}}_t)_\times$$

where $(\cdot)_\times$ denotes a 3×3 skew-symmetric matrix. If we define the error between the true and estimated orientation as $\boldsymbol{\eta}_t \triangleq \mathbf{R}_t^\mathsf{T}\bar{\mathbf{R}}_t \in \text{SO}(3)$, then the (left-invariant) error dynamics becomes

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\eta}_t &= \mathbf{R}_t^\mathsf{T}\bar{\mathbf{R}}_t(\tilde{\boldsymbol{\omega}}_t)_\times + \left(\mathbf{R}_t(\tilde{\boldsymbol{\omega}}_t)_\times\right)^\mathsf{T}\bar{\mathbf{R}}_t \\ &= \mathbf{R}_t^\mathsf{T}\bar{\mathbf{R}}_t(\tilde{\boldsymbol{\omega}}_t)_\times - (\tilde{\boldsymbol{\omega}}_t)_\times\mathbf{R}_t^\mathsf{T}\bar{\mathbf{R}}_t \\ &= \boldsymbol{\eta}_t(\tilde{\boldsymbol{\omega}}_t)_\times - (\tilde{\boldsymbol{\omega}}_t)_\times\boldsymbol{\eta}_t \\ &= g(\boldsymbol{\eta}_t, \tilde{\boldsymbol{\omega}}_t)\end{aligned} \tag{5}$$

Using this particular choice of state and error variable yields an autonomous error dynamics function (independent of the state directly). As SO(3) is a Lie group, we can look at the dynamics of a redefined error that resides in the tangent space, $\boldsymbol{\eta}_t \triangleq \exp(\boldsymbol{\xi}_t)$.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}(\exp(\boldsymbol{\xi}_t)) &= \exp(\boldsymbol{\xi}_t)(\tilde{\boldsymbol{\omega}}_t)_\times - (\tilde{\boldsymbol{\omega}}_t)_\times\exp(\boldsymbol{\xi}_t) \\ \frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{I} + (\boldsymbol{\xi}_t)_\times\right) &\approx \left(\mathbf{I} + (\boldsymbol{\xi}_t)_\times\right)(\tilde{\boldsymbol{\omega}}_t)_\times - (\tilde{\boldsymbol{\omega}}_t)_\times\left(\mathbf{I} + (\boldsymbol{\xi}_t)_\times\right) \\ \implies \frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{\xi}_t)_\times &= (\boldsymbol{\xi}_t)_\times(\tilde{\boldsymbol{\omega}}_t)_\times - (\tilde{\boldsymbol{\omega}}_t)_\times(\boldsymbol{\xi}_t)_\times \\ &= -(\tilde{\boldsymbol{\omega}}_t)_\times(\boldsymbol{\xi}_t)_\times \\ \implies \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\xi}_t &= (\tilde{\boldsymbol{\omega}}_t)_\times\boldsymbol{\xi}_t\end{aligned}$$

After making a first-order approximation of the exponential map, the tangent space error dynamics become linear. In addition, this linear system only depends on the error, not the state estimate directly. In other words, incorrect state estimates will not affect the accuracy of the linearization, which leads to better accuracy and consistency of the filter. For SO(3), this effect is well studied and has been leveraged to design the commonly used QEKFs,[3] (Sola, 2017; Trawny and Roumeliotis, 2005). However, the extension to general matrix Lie groups, called the InEKF, was only recently developed by Barrau and Bonnabel (2017).

In the above example, we utilized the first-order approximation for the exponential map of SO(3); $\exp(\boldsymbol{\xi}_t) \approx \mathbf{I} + (\boldsymbol{\xi}_t)_\times$. In general, one may ask how much accuracy is lost when making this approximation. The surprising result by Barrau and Bonnabel (2017) is that this linearization is, in fact, exact. This is the basis of Theorem 2. If the initial error is known, the nonlinear error dynamics can be exactly recovered from this linear system.

Let us demonstrate this theorem for the SO(3) example. Let $\boldsymbol{\eta}_0 = \exp(\boldsymbol{\xi}_0)$ be the initial left invariant error. We can show that $\boldsymbol{\eta}_t = \mathbf{R}_t^\mathsf{T}\boldsymbol{\eta}_0\mathbf{R}_t$ is a solution to the error dynamics equation (5) through differentiation:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\eta}_t &= \mathbf{R}_t^\mathsf{T}\boldsymbol{\eta}_0\mathbf{R}_t(\tilde{\boldsymbol{\omega}}_t)_\times - (\tilde{\boldsymbol{\omega}}_t)_\times\mathbf{R}_t^\mathsf{T}\boldsymbol{\eta}_0\mathbf{R}_t \\ &= \boldsymbol{\eta}_t(\tilde{\boldsymbol{\omega}}_t)_\times - (\tilde{\boldsymbol{\omega}}_t)_\times\boldsymbol{\eta}_t\end{aligned}$$

Once we replace the group error with the tangent space error and use the group's adjoint definition to shift the rotation inside the exponential, we arrive at a simple expression for the tangent space error:

$$\begin{aligned}\boldsymbol{\eta}_t &= \mathbf{R}_t^\mathsf{T}\boldsymbol{\eta}_0\mathbf{R}_t \\ \implies \exp(\boldsymbol{\xi}_t) &= \mathbf{R}_t^\mathsf{T}\exp(\boldsymbol{\xi}_0)\mathbf{R}_t = \exp(\mathbf{R}_t^\mathsf{T}\boldsymbol{\xi}_0) \\ \implies \boldsymbol{\xi}_t &= \mathbf{R}_t^\mathsf{T}\boldsymbol{\xi}_0\end{aligned}$$

Our final log-linear error dynamics can now be obtained by differentiating this new error expression:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\xi}_t &= -(\tilde{\boldsymbol{\omega}}_t)_\times\mathbf{R}_t^\mathsf{T}\boldsymbol{\xi}_0 \\ &= -(\tilde{\boldsymbol{\omega}}_t)_\times\boldsymbol{\xi}_t\end{aligned}$$

Again, this result shows that if the initial error is known, the nonlinear error dynamics can be exactly recovered from this linear system. In this work, we leverage these ideas to develop a contact-aided inertial observer for legged robots.

5. $SE_{N+2}(3)$ continuous right-invariant EKF

In this section, we derive a right-invariant extended Kalman filter (RIEKF) using IMU and contact motion models with corrections made through forward kinematic measurements. This RIEKF can be used to estimate the state of a robot that has an arbitrary (finite) number of points in contact with a static environment. While the filter is particularly useful for legged robots, the same theory can be applied for manipulators as long as the contact assumptions (presented in Section 5.2) are verified.

In order to be consistent with the standard InEKF theory, IMU biases are neglected for now. Section 7 provides a method for reintroducing the bias terms.

5.1. State representation

As with typical aided inertial navigation, we wish to estimate the orientation, velocity, and position of the IMU (body) in the world frame (Forster et al., 2017; Lupton and Sukkarieh, 2012; Yang and Shen, 2017). These states are represented by $\mathbf{R}_{\mathrm{WB}}(t)$, ${}_{\mathrm{W}}\mathbf{v}_{\mathrm{B}}(t)$, and ${}_{\mathrm{W}}\mathbf{p}_{\mathrm{WB}}(t)$, respectively. In addition, we append the position of all contact points (in the world frame), ${}_{\mathrm{W}}\mathbf{p}_{\mathrm{WC}_i}(t)$, to the list of state variables. This is similar to the approach taken in Bloesch et al. (2012); Bloesch (2017).

The above collection of state variables forms a matrix Lie group, $\mathcal{G}$. Specifically, for N contact points, $\mathbf{X}_t \in \mathrm{SE}_{N+2}(3)$ can be represented by the following matrix:

$$\mathbf{X}_t \triangleq \begin{bmatrix} \mathbf{R}_{\mathrm{WB}}(t) & {}_{\mathrm{W}}\mathbf{v}_{\mathrm{B}}(t) & {}_{\mathrm{W}}\mathbf{p}_{\mathrm{WB}}(t) & {}_{\mathrm{W}}\mathbf{p}_{\mathrm{WC}_1}(t) & \cdots & {}_{\mathrm{W}}\mathbf{p}_{\mathrm{WC}_N}(t) \\ \mathbf{0}_{1,3} & 1 & 0 & 0 & \cdots & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 & \cdots & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

This Lie group is an extension of $\mathrm{SE}(3)$ and has been previously used to solve inertial navigation (Barrau, 2015) and SLAM problems (Barrau and Bonnabel, 2015; Bonnabel, 2012). In fact, the estimators derived in this work have a connection to the landmark-based SLAM problem. This connection is detailed later in Section 12.

Because the process and measurements models for each contact point, ${}_{\mathrm{W}}\mathbf{p}_{\mathrm{WC}_i}(t)$, are identical, without loss of generality, we will derive all further equations assuming only a single contact point. Furthermore, for the sake of readability, we introduce the following shorthand notation:

$$\mathbf{X}_t \triangleq \begin{bmatrix} \mathbf{R}_t & \mathbf{v}_t & \mathbf{p}_t & \mathbf{d}_t \\ \mathbf{0}_{1,3} & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{u}_t = \begin{bmatrix} {}_{\mathrm{B}}\tilde{\boldsymbol{\omega}}_{\mathrm{WB}}(t) \\ {}_{\mathrm{B}}\tilde{\mathbf{a}}_{\mathrm{WB}}(t) \end{bmatrix} \triangleq \begin{bmatrix} \tilde{\boldsymbol{\omega}}_t \\ \tilde{\mathbf{a}}_t \end{bmatrix}$$

where the input $\mathbf{u}_t$ is formed from the angular velocity and linear acceleration measurements coming from the IMU. It is important to note that these measurements are taken in the body (or IMU) frame. The Lie algebra of $\mathcal{G}$, denoted by $\mathfrak{g}$, is an $(N+5)$-dimensional square matrix. We use the *hat* operator, $(\cdot)^\wedge : \mathbb{R}^{3N+9} \to \mathfrak{g}$, to map a vector to the corresponding element of the Lie algebra. In the case of a single contact, for example, this function is defined by

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} (\boldsymbol{\xi}^R)_\times & \boldsymbol{\xi}^v & \boldsymbol{\xi}^p & \boldsymbol{\xi}^d \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix}, \tag{6}$$

where $(\cdot)_\times$ denotes a 3×3 skew-symmetric matrix. The inverse operation is defined using the *wedge* operator, $(\cdot)^\vee : \mathfrak{g} \to \mathbb{R}^{3N+9}$. The matrix representation of the adjoint is given by

$$\mathrm{Ad}_{\mathbf{X}_t} = \begin{bmatrix} \mathbf{R} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ (\mathbf{v}_t)_\times \mathbf{R}_t & \mathbf{R}_t & \mathbf{0} & \mathbf{0} \\ (\mathbf{p}_t)_\times \mathbf{R}_t & \mathbf{0} & \mathbf{R}_t & \mathbf{0} \\ (\mathbf{d}_t)_\times \mathbf{R}_t & \mathbf{0} & \mathbf{0} & \mathbf{R}_t \end{bmatrix} \tag{7}$$

A closed-form expression for the exponential map of $\mathrm{SE}_{N+2}(3)$ is given in Appendix B.

5.2. Continuous-time system dynamics

The IMU measurements are modeled as being corrupted by additive Gaussian white noise processes, per

$$\begin{aligned} \tilde{\boldsymbol{\omega}}_t &= \boldsymbol{\omega}_t + \mathbf{w}_t^g, & \mathbf{w}_t^g &\sim \mathcal{GP}\left(\mathbf{0}_{3,1}, \boldsymbol{\Sigma}^g \delta(t-t')\right) \\ \tilde{\mathbf{a}}_t &= \mathbf{a}_t + \mathbf{w}_t^a, & \mathbf{w}_t^a &\sim \mathcal{GP}\left(\mathbf{0}_{3,1}, \boldsymbol{\Sigma}^a \delta(t-t')\right) \end{aligned}$$

where $\mathcal{GP}$ denotes a Gaussian process and $\delta(t-t')$ denotes the Dirac delta function. These are explicit measurements coming directly from a physical sensor. In contrast, the velocity of the contact point is implicitly *inferred* through a contact sensor; specifically, when a binary sensor indicates contact, the position of the contact point is *assumed to remain fixed in the world frame*, i.e., the measured velocity is zero. In order to accommodate potential slippage, the measured velocity is assumed to be the actual velocity plus white Gaussian noise, namely

$${}_{\mathrm{W}}\tilde{\mathbf{v}}_{\mathrm{C}} = \mathbf{0}_{3,1} = {}_{\mathrm{C}}\mathbf{v}_{\mathrm{C}} + \mathbf{w}_t^v, \quad \mathbf{w}_t^v \sim \mathcal{GP}\left(\mathbf{0}_{3,1}, \boldsymbol{\Sigma}^v \delta(t-t')\right)$$

Using the IMU and contact measurements, the individual terms of the system dynamics can be written as

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t &= \mathbf{R}_t(\tilde{\boldsymbol{\omega}}_t - \mathbf{w}_t^g)_\times \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}_t &= \mathbf{R}_t(\tilde{\mathbf{a}}_t - \mathbf{w}_t^a) + \mathbf{g} \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{p}_t &= \mathbf{v}_t \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{d}_t &= \mathbf{R}_t\, \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)(-\mathbf{w}_t^v) \end{aligned} \tag{8}$$

where $\mathbf{g}$ is the gravity vector and $\boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)$ is the measured orientation of the contact frame with respect to the IMU frame as computed through encoder measurements, $\tilde{\boldsymbol{\alpha}}_t \in \mathbb{R}^M$, and forward kinematics. Therefore, $\mathbf{R}_t\, \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)$ is a rotation matrix that transforms a vector from the contact frame to the world frame.

In matrix form, the dynamics can be expressed as

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{X}_t &= \begin{bmatrix} \mathbf{R}_t(\tilde{\boldsymbol{\omega}}_t)_\times & \mathbf{R}_t\tilde{\mathbf{a}}_t+\mathbf{g} & \mathbf{v}_t & \mathbf{0}_{3,1} \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix} \\ &\quad - \begin{bmatrix} \mathbf{R}_t & \mathbf{v}_t & \mathbf{p}_t & \mathbf{d}_t \\ \mathbf{0}_{1,3} & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} (\mathbf{w}_t^g)_\times & \mathbf{w}_t^a & \mathbf{0}_{3,1} & \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^v \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix} \\ &\triangleq f_{u_t}(\mathbf{X}_t) - \mathbf{X}_t\mathbf{w}_t^\wedge \end{aligned} \tag{9}$$

with $\mathbf{w}_t \triangleq \mathrm{vec}(\mathbf{w}_t^g,\ \mathbf{w}_t^a,\ \mathbf{0}_{3,1},\ \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^v)$. The deterministic system dynamics, $f_{u_t}(\cdot)$, can be shown to satisfy the group affine property (2). Therefore, following Theorem 1, the left- and right-invariant error dynamics will evolve independently of the system's state.

Using Theorem 1, the right-invariant error dynamics is

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\eta}_t^r &= f_{u_t}(\boldsymbol{\eta}_t^r) - \boldsymbol{\eta}_t^r f_{u_t}(\boldsymbol{I}_d) + (\bar{\mathbf{X}}_t\mathbf{w}_t^\wedge\bar{\mathbf{X}}_t^{-1})\,\boldsymbol{\eta}_t^r \\ &\triangleq g_{u_t}(\boldsymbol{\eta}_t^r) + \bar{\mathbf{w}}_t^\wedge\boldsymbol{\eta}_t^r \end{aligned} \tag{10}$$

where the second term arises from the additive noise. The derivation follows the results in Barrau and Bonnabel (2017) and is not repeated here.

Furthermore, Theorem 2 specifies that the invariant error satisfies a log-linear property. Namely, if $\mathbf{A}_t$ is defined by $g_{u_t}(\exp(\boldsymbol{\xi})) \triangleq (\mathbf{A}_t\boldsymbol{\xi})^\wedge + \mathcal{O}(||\boldsymbol{\xi}||^2)$, then the log of the invariant error, $\boldsymbol{\xi} \in \mathbb{R}^{\dim\mathfrak{g}}$, approximately satisfies[4] the linear system

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\xi}_t &= \mathbf{A}_t^r\boldsymbol{\xi}_t + \bar{\mathbf{w}}_t = \mathbf{A}_t\boldsymbol{\xi}_t + \mathrm{Ad}_{\bar{\mathbf{X}}_t}\mathbf{w}_t \\ \boldsymbol{\eta}_t^r &= \exp(\boldsymbol{\xi}_t) \end{aligned} \tag{11}$$

To compute the matrix $\mathbf{A}_t$, we linearize the invariant error dynamics, $g_{u_t}(\cdot)$, using the first-order approximation $\boldsymbol{\eta}_t^r = \exp(\boldsymbol{\xi}^t) \approx \boldsymbol{I}_d + \boldsymbol{\xi}_t^\wedge$ to yield

$$\begin{aligned} & g_{u_t}(\exp(\boldsymbol{\xi}_t)) \approx \\ & \begin{bmatrix} \left(\mathbf{I}+(\boldsymbol{\xi}_t^R)_\times\right)(\tilde{\boldsymbol{\omega}}_t)_\times & \left(\mathbf{I}+(\boldsymbol{\xi}_t^R)_\times\right)\tilde{\mathbf{a}}_t+\mathbf{g} & \boldsymbol{\xi}_t^v & \mathbf{0}_{3,1} \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix} \\ & - \begin{bmatrix} \mathbf{I}+(\boldsymbol{\xi}_t^R)_\times & \boldsymbol{\xi}_t^v & \boldsymbol{\xi}_t^p & \boldsymbol{\xi}_t^d \\ \mathbf{0}_{3,1} & 1 & 0 & 0 \\ \mathbf{0}_{3,1} & 0 & 1 & 0 \\ \mathbf{0}_{3,1} & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} (\tilde{\boldsymbol{\omega}}_t)_\times & \tilde{\mathbf{a}}+\mathbf{g} & \mathbf{0}_{3,1} & \mathbf{0}_{3,1} \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix} \\ & = \begin{bmatrix} \mathbf{0}_{3,3} & (\mathbf{g})_\times\,\boldsymbol{\xi}_t^R & \boldsymbol{\xi}_t^v & \mathbf{0}_{3,1} \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{3,1} \\ (\mathbf{g})_\times\,\boldsymbol{\xi}_t^R \\ \boldsymbol{\xi}_t^v \\ \mathbf{0}_{3,1} \end{bmatrix}^\wedge \end{aligned} \tag{12}$$

With the above, we can express the prediction step of the RIEKF. The state estimate, $\bar{\mathbf{X}}_t$, is propagated though the deterministic system dynamics, whereas the covariance matrix, $\mathbf{P}_t$, is computed using the Riccati equation (Maybeck, 1982), namely,

$$\frac{\mathrm{d}}{\mathrm{d}t}\bar{\mathbf{X}}_t = f_{u_t}(\bar{\mathbf{X}}_t) \quad \text{and} \quad \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{P}_t = \mathbf{A}_t\mathbf{P}_t + \mathbf{P}_t\mathbf{A}_t^\mathsf{T} + \bar{\mathbf{Q}}_t \tag{13}$$

where the matrices $\mathbf{A}_t$ and $\bar{\mathbf{Q}}_t$ are obtained from (12) and (11),

$$\mathbf{A}_t = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ (\mathbf{g})_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \text{ and } \bar{\mathbf{Q}}_t = \mathrm{Ad}_{\bar{\mathbf{X}}_t}\mathrm{Cov}\left(\mathbf{w}_t\right)\mathrm{Ad}_{\bar{\mathbf{X}}_t}^\mathsf{T} \tag{14}$$

Remark 2. For the right-invariant case, expression (14), $\mathbf{A}_t$ is time-invariant and the time subscript could be dropped. However, in general it can be time-varying, therefore, we use $\mathbf{A}_t$ throughout the article.

5.3. Right-invariant forward kinematic measurement model

Let $\boldsymbol{\alpha}_t \in \mathbb{R}^M$ denote the vector of joint positions (prismatic or revolute) between the body and the contact point. We assume that the encoder measurements at time t are corrupted by additive white Gaussian noise.

$$\tilde{\boldsymbol{\alpha}}_t = \boldsymbol{\alpha}_t + \mathbf{w}_t^\alpha, \quad \mathbf{w}_t^\alpha \sim \mathcal{N}(\mathbf{0}_{M,1}, \boldsymbol{\Sigma}^\alpha) \tag{15}$$

Using forward kinematics, we measure the relative position of the contact point with respect to the body,

$${}_\mathrm{B}\mathbf{p}_\mathrm{BC}(t) \triangleq \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t - \mathbf{w}_t^\alpha) \approx \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) - \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha \tag{16}$$

where $\mathbf{J}_p$ denotes the analytical Jacobian of the forward kinematics function. Using the state variables, the forward-kinematics position measurement becomes

$$\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) = \mathbf{R}_t^\mathsf{T}(\mathbf{d}_t - \mathbf{p}_t) + \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha \tag{17}$$

Re-written in matrix form, this measurement has the right-invariant observation form (4),

$$\underbrace{\begin{bmatrix} \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) \\ 0 \\ 1 \\ -1 \end{bmatrix}}_{\mathbf{Y}_t} = \underbrace{\begin{bmatrix} \mathbf{R}_t^\mathsf{T} & -\mathbf{R}_t^\mathsf{T}\mathbf{v}_t & -\mathbf{R}_t^\mathsf{T}\mathbf{p}_t & -\mathbf{R}_t^\mathsf{T}\mathbf{d}_t \\ \mathbf{0}_{1,3} & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{X}_t^{-1}} \underbrace{\begin{bmatrix} \mathbf{0}_{3,1} \\ 0 \\ 1 \\ -1 \end{bmatrix}}_{\mathbf{b}} + \underbrace{\begin{bmatrix} \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{V}_t}$$

Therefore, the innovation depends solely on the invariant error and the update equations take the form (Barrau and Bonnabel, 2017: Section 3.1.2)

$$\begin{aligned} \bar{\mathbf{X}}_t^+ &= \exp\left(\mathbf{L}_t\left(\bar{\mathbf{X}}_t\mathbf{Y}_t - \mathbf{b}\right)\right)\bar{\mathbf{X}}_t \\ \boldsymbol{\eta}_t^{r+} &= \exp\left(\mathbf{L}_t\left(\boldsymbol{\eta}_t^r\mathbf{b} - \mathbf{b} + \bar{\mathbf{X}}_t\mathbf{V}_t\right)\right)\boldsymbol{\eta}_t^r \end{aligned} \tag{18}$$

where $\exp(\cdot)$ is the exponential map corresponding to the state matrix Lie group, $\mathcal{G}$, $\mathbf{L}_t$ is a gain matrix to be defined later, $\mathbf{b}^\mathsf{T} = \begin{bmatrix}\mathbf{0}_{1,3} & 0 & 1 & -1\end{bmatrix}$, and $\mathbf{Y}_t^\mathsf{T} = \begin{bmatrix}\boldsymbol{h}_p^\mathsf{T}(\tilde{\boldsymbol{\alpha}}_t) & 0 & 1 & -1\end{bmatrix}$. Because the last three rows of $\bar{\mathbf{X}}_t\mathbf{Y}_t - \mathbf{b}$ are identically zero, we can express the update equations using a reduced dimensional gain, $\mathbf{K}_t$, and an auxiliary selection matrix $\boldsymbol{\Pi} \triangleq \begin{bmatrix}\mathbf{I} & \mathbf{0}_{3,3}\end{bmatrix}$, so that $\mathbf{L}_t\left(\bar{\mathbf{X}}_t\mathbf{Y}_t - \mathbf{b}\right) = \mathbf{K}_t\boldsymbol{\Pi}\bar{\mathbf{X}}_t\mathbf{Y}_t$ as detailed in Barrau (2015).

Using the first-order approximation of the exponential map, $\boldsymbol{\eta}_t^r = \exp(\boldsymbol{\xi}_t) \approx \boldsymbol{I}_d + \boldsymbol{\xi}_t^\wedge$, and dropping higher-order terms, we can linearize the update equation (18),

$$\begin{aligned} \boldsymbol{\eta}_t^{r+} &\approx \boldsymbol{I}_d + \boldsymbol{\xi}_t^{+\wedge} \approx \boldsymbol{I}_d + \boldsymbol{\xi}_t^\wedge \\ &+ \left(\mathbf{K}_t\boldsymbol{\Pi}\left(\left(\boldsymbol{I}_d + \boldsymbol{\xi}_t^\wedge\right)\begin{bmatrix}\mathbf{0}_{3,1}\\0\\1\\-1\end{bmatrix} + \bar{\mathbf{X}}_t\begin{bmatrix}\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha\\0\\0\\0\end{bmatrix}\right)\right)^\wedge \end{aligned}$$

Therefore,

$$\begin{aligned} \boldsymbol{\xi}_t^{+\wedge} &= \boldsymbol{\xi}_t^\wedge + \\ &\left(\mathbf{K}_t\boldsymbol{\Pi}\left(\begin{bmatrix}\mathbf{I}+(\boldsymbol{\xi}_t^R)_\times & \boldsymbol{\xi}_t^v & \boldsymbol{\xi}_t^p & \boldsymbol{\xi}_t^d \\ \mathbf{0}_{3,1} & 1 & 0 & 0 \\ \mathbf{0}_{3,1} & 0 & 1 & 0 \\ \mathbf{0}_{3,1} & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}\mathbf{0}_{3,1}\\0\\1\\-1\end{bmatrix} + \bar{\mathbf{X}}_t\begin{bmatrix}\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha\\0\\0\\0\end{bmatrix}\right)\right)^\wedge \\ &= \boldsymbol{\xi}_t^\wedge + \left(\mathbf{K}_t\boldsymbol{\Pi}\left(\begin{bmatrix}\boldsymbol{\xi}_t^p - \boldsymbol{\xi}_t^d\\0\\1\\-1\end{bmatrix} + \bar{\mathbf{X}}_t\begin{bmatrix}\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha\\0\\0\\0\end{bmatrix}\right)\right)^\wedge \end{aligned}$$

Taking $(\cdot)^\vee$ of both sides yields the linear update equation,

$$\begin{aligned} \boldsymbol{\xi}_t^+ &= \boldsymbol{\xi}_t - \mathbf{K}_t\left(\begin{bmatrix}\mathbf{0}_{3,3} & \mathbf{0}_{3,3} & -\mathbf{I} & \mathbf{I}\end{bmatrix}\boldsymbol{\xi}_t - \bar{\mathbf{R}}_t\left(\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha\right)\right) \\ &\triangleq \boldsymbol{\xi}_t - \mathbf{K}_t\left(\mathbf{H}_t\boldsymbol{\xi}_t - \bar{\mathbf{R}}_t\left(\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha\right)\right) \end{aligned}$$

Finally, we can write down the full state and covariance update equations of the RIEKF using the derived linear update equation and the theory of Kalman filtering (Anderson and Moore, 1979; Bar-Shalom et al., 2001; Maybeck, 1982) as

$$\begin{aligned} \bar{\mathbf{X}}_t^+ &= \exp\left(\mathbf{K}_t\boldsymbol{\Pi}\bar{\mathbf{X}}_t\mathbf{Y}_t\right)\bar{\mathbf{X}}_t \\ \mathbf{P}_t^+ &= (\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)\,\mathbf{P}_t(\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)^\mathsf{T} + \mathbf{K}_t\bar{\mathbf{N}}_t\mathbf{K}_t^\mathsf{T} \end{aligned} \tag{19}$$

where the gain $\mathbf{K}_t$ is computed using

$$\mathbf{S}_t = \mathbf{H}_t\mathbf{P}_t\mathbf{H}_t^\mathsf{T} + \bar{\mathbf{N}}_t, \quad \mathbf{K}_t = \mathbf{P}_t\mathbf{H}_t^\mathsf{T}\mathbf{S}_t^{-1}$$

and from (19), the matrices $\mathbf{H}_t$ and $\bar{\mathbf{N}}_t$ are given by

$$\begin{aligned} \mathbf{H}_t &= \begin{bmatrix}\mathbf{0}_{3,3} & \mathbf{0}_{3,3} & -\mathbf{I} & \mathbf{I}\end{bmatrix}, \\ \bar{\mathbf{N}}_t &= \bar{\mathbf{R}}_t\,\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathrm{Cov}(\mathbf{w}_t^\alpha)\,\mathbf{J}_p^\mathsf{T}(\tilde{\boldsymbol{\alpha}}_t)\,\bar{\mathbf{R}}_t^\mathsf{T} \end{aligned} \tag{20}$$

5.4. Observability analysis

Because the error dynamics are log-linear (cf. Theorem 2), we can determine the unobservable states of the filter without having to perform a nonlinear observability analysis (Barrau, 2015). Noting that the linear error dynamics matrix in our case is time-invariant and nilpotent (with a degree of three), the discrete-time state transition matrix is a polynomial in $\mathbf{A}_t$,

$$\boldsymbol{\Phi} = \exp_m(\mathbf{A}_t\Delta t) = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ (\mathbf{g})_\times\,\Delta t & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \frac{1}{2}(\mathbf{g})_\times\,\Delta t^2 & \mathbf{I}\Delta t & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

It follows that the discrete-time observability matrix is

$$\mathcal{O} = \begin{bmatrix}\mathbf{H}\\ \mathbf{H}\boldsymbol{\Phi}\\ \mathbf{H}\boldsymbol{\Phi}^2\\ \vdots\end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} \\ -\frac{1}{2}(\mathbf{g})_\times\,\Delta t^2 & -\mathbf{I}\Delta t & -\mathbf{I} & \mathbf{I} \\ -2(\mathbf{g})_\times\,\Delta t^2 & -2\mathbf{I}\Delta t^2 & -\mathbf{I} & \mathbf{I} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

The last six columns (i.e., two matrix columns) of the observability matrix are clearly linearly dependent, which indicates the absolute position of the robot is unobservable. In addition, because the gravity vector only has a z component, the third column of $\mathcal{O}$ is all zeros. Therefore, a rotation about the gravity vector (yaw) is also unobservable. This linear observability analysis agrees with the nonlinear observability results of Bloesch et al. (2012), albeit with much less computation. Furthermore, as the error dynamics do not depend on the estimated state, there is no chance of the linearization spuriously increasing the numerical rank of the observability matrix (Barrau, 2015). This latter effect was previously known and studied by Bloesch et al. (2012), and in order to resolve this problem, an observability-constrained EKF (Huang et al., 2010) was developed. In our

proposed framework, by default, the discrete RIEKF has the same unobservable states as the underlying nonlinear system; hence, the developed discrete RIEKF intrinsically solves this problem.

6. Simulation results

To investigate potential benefits or drawbacks of the proposed filter, we compare it against a state-of-the-art quaternion-based extended Kalman filter (QEKF), similar to those described by Bloesch et al. (2012); Rotella et al. (2014). For implementation, the filter equations were discretized; see Appendix A for more details.

6.1. Quaternion-based filter equations

The choice of error variables is the main difference between the InEKF and the QEKF. Instead of the right-invariant error (1), a QEKF typically uses decoupled error states

$$\begin{aligned} \exp(\delta\boldsymbol{\theta}_t) &\triangleq \mathbf{R}_t^{\mathsf{T}}\bar{\mathbf{R}}_t \\ \delta\mathbf{v}_t &\triangleq \mathbf{v}_t - \bar{\mathbf{v}}_t \\ \delta\mathbf{p}_t &\triangleq \mathbf{p}_t - \bar{\mathbf{p}}_t \\ \delta\mathbf{d}_t &\triangleq \mathbf{d}_t - \bar{\mathbf{d}}_t \end{aligned} \tag{21}$$

Using this definition of error, the QEKF deterministic error dynamics can be approximated as

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\delta\boldsymbol{\theta}_t\\ \delta\mathbf{v}_t\\ \delta\mathbf{p}_t\\ \delta\mathbf{d}_t\end{bmatrix} = \begin{bmatrix}-(\tilde{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ -\bar{\mathbf{R}}_t(\tilde{\mathbf{a}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\delta\boldsymbol{\theta}_t\\ \delta\mathbf{v}_t\\ \delta\mathbf{p}_t\\ \delta\mathbf{d}_t\end{bmatrix}$$

while the linearized observation matrix becomes

$$\mathbf{H}_t = \begin{bmatrix}(\bar{\mathbf{R}}_t^{\mathsf{T}}(\bar{\mathbf{d}}_t - \bar{\mathbf{p}}_t))_\times & \mathbf{0} & -\bar{\mathbf{R}}_t^{\mathsf{T}} & \bar{\mathbf{R}}_t^{\mathsf{T}}\end{bmatrix}$$

The above linearizations are clearly dependent on the state estimate. Therefore, when the estimated state deviates from the true state, the linearizations are potentially wrong, reducing accuracy and consistency in the QEKF. In contrast, the deterministic right-invariant error dynamics are exactly log-linear (14). In addition, the linearized observation matrix for our InEKF (20) is also independent of the state estimate.

6.2. Convergence comparison

A dynamic *simulation* of a Cassie-series bipedal robot (described in Section 9) was performed in which the robot slowly walked forward after a small drop, accelerating from 0.0 to 0.3 m/s. The discrete, simulated measurements were corrupted by additive white Gaussian noise, which are specified in Table 1 along with the initial state covariance values. The same values were used in both simulation and experimental convergence evaluations of the filters. The IMU bias estimation was turned off for these simulations. The simulation was performed with MATLAB and Simulink (Simscape Multibody™) where the simulation environment models ground contact forces with a linear force law (having a stiffness and damping term) and a Coulomb friction model. A typical walking gait is shown in Figure 2.

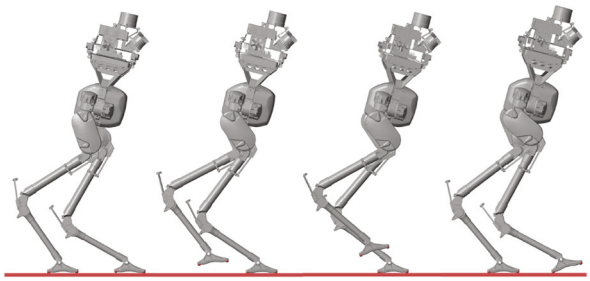

Fig. 2. A typical walking gait that is used for filter comparisons. The Cassie bipedal robot is simulated using Simscape Multibody™.

To compare the convergence properties of the two filters, 100 simulations of each filter were performed using identical measurements, noise statistics, initial covariance, and various random initial orientations and velocities. The initial Euler angle estimates were sampled uniformly from -30° to 30°. The initial velocity estimates were sampled uniformly from -1.0 m/s to 1.0 m/s. The pitch and roll estimates as well as the (body frame) velocity estimates for both filters are shown in Figure 3. Although both filters converge for this set of initial conditions, the proposed RIEKF converges considerably faster than the standard quaternion-based EKF.

6.3. Accuracy of linearized dynamics

The superior performance of the InEKF over the QEKF comes from the improved accuracy of the linearized error dynamics. As indicated by Theorem 2, the deterministic error dynamics of the InEKF are actually exact, whereas the QEKF version is only an approximation. To demonstrate this, a simulation was performed where propagation of the true error is compared with the propagation of the linearized error dynamics.

We first analyzed the deterministic dynamics. Given an initial error vector, $\boldsymbol{\xi}_0^{\text{true}}$, the initial state estimate for the InEKF was computed using the definition of right-invariant error (1), and the initial state estimate for the QEKF was computed using Equation (21). The true state was initialized to the identity element. The true and estimated states for both filters were then propagated for 1 second (1,000 time steps), using randomly sampled IMU measurements. The error states for both the InEKF and the QEKF were also propagated using their respective linearized error dynamics. The resulting error, $\boldsymbol{\xi}_1^{\text{true}}$, between the final estimated and true states were computed and compared with the propagated error states, $\boldsymbol{\xi}_1^{\text{prop}}$ to yield a measure of linearization accuracy, $\|\boldsymbol{\xi}_1^{\text{true}} - \boldsymbol{\xi}_1^{\text{prop}}\|$. This test was performed multiple

Table 1. Experiment discrete noise statistics and initial covariance.

Measurement type	Noise SD	State element	Initial SD
Linear acceleration	0.04 m/s^2	Orientation of IMU	30.0°
Angular velocity	0.002 rad/s	Velocity of IMU	1.0 m/s
Accelerometer bias	0.001 m/s^3	Position of IMU	0.1 m
Gyroscope bias	0.001 rad/s^2	Position of right foot	0.1 m
Contact linear velocity	0.05 m/s	Position of left foot	0.1 m
Joint encoders	1.0°	Gyroscope bias	0.005 rad/s
		Accelerometer bias	0.05 m/s^2

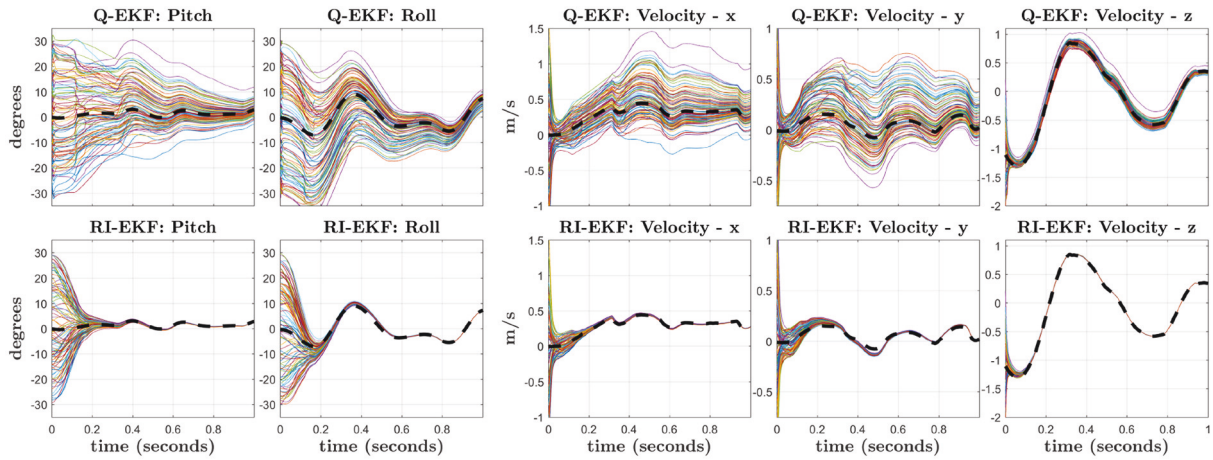

Fig. 3. A quaternion-based extended Kalman filter (QEKF) and the proposed right-invariant extended Kalman filter (RIEKF) were run 100 times using the same measurements, noise statistics, and initial covariance, but with random initial orientations and velocities. The noisy measurements came from a dynamic simulation of a Cassie-series biped robot where the robot walks forwards after a small drop, accelerating from 0.0 to 0.3 m/s. The above plots show the state estimate for the initial second of data, where the dashed black line represents the true state. The RIEKF (bottom row) converges considerably faster than the QEKF (top row) for all observable states. The estimated yaw angle (not shown) does not converge for either filter because it is unobservable. Therefore, to compare convergence, the velocities shown are represented in the estimated IMU (body) frame.

times while linearly scaling the initial orientation error from vec(0, 0, 0) to vec($\pi/2, \pi/2, \pi/2$). The results are shown in Figure 4.

As expected, when the initial error is zero, the difference between the true and propagated error is zero. This indicates that the linearized error dynamics for both the InEKF and the QEKF are correct. As the initial error increases, the difference between the true and propagated error states for the QEKF grows due to the decreased accuracy of the linearization. In contrast, the difference between the true and propagated error states for the InEKF are always exactly zero regardless of the initial error. In other words, assuming the initial error is known, the true propagated state can be exactly recovered from solving the linearized error dynamics system; see Theorem 2.

In the non-deterministic case (with sensor noise), Theorem 2 no longer holds. This can be seen in Figure 5, where the same test was performed, but with sensor noise corrupting the propagated state estimate. The difference between the true and propagated error is no longer exactly zero for

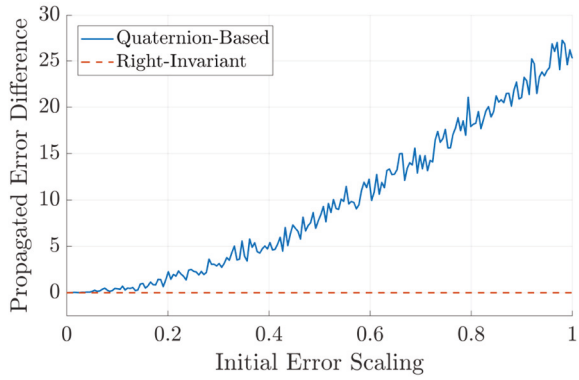

Fig. 4. Analyzing accuracy of the deterministic error dynamics. The difference between the true error and the propagated error is shown as the initial true error increases. The state and errors were propagated for 1 second using randomly sampled IMU measurements.

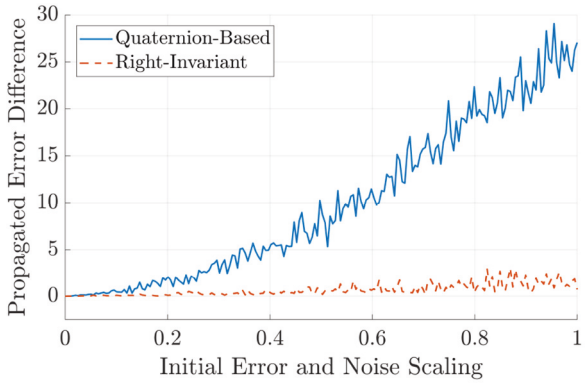

Fig. 5. Difference between the true and propagated errors when measurements contain noise. The log-linear error dynamics of the InEKF are no longer exact.

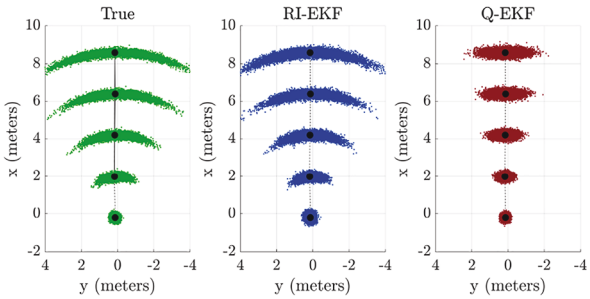

Fig. 6. A set of 10,000 samples taken from the estimated filter covariances for a simulation where Cassie walked forward with an average speed of 1 m/s. The position distributions at times 0, 2, 4, 6, and 8 s are shown.

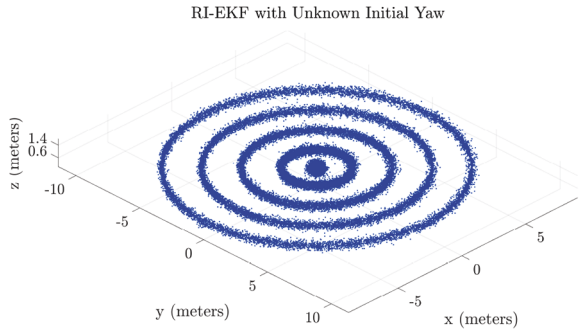

Fig. 7. Samples taken from the InEKF's estimated position distribution for a walking simulation with a completely uncertain initial yaw angle. The robot moved forward at an average speed of 1 m/s. Each ring represents the position uncertainty at times 0, 2, 4, 6, and 8 s.

the InEKF. However, the InEKF linearization remains more accurate due to the reduced sensitivity to initial state errors. This helps to further explain the improved convergence properties shown in Figure 3.

6.4. Covariance ellipse comparison

The error states in both the QEKF and the InEKF are assumed to be zero-mean Gaussian random vectors. However, owing to the differing choice of error variables, the state uncertainty will differ. In the QEKF, all states and errors are decoupled (21). For example, the true position only depends on the position estimate and the position error, $\mathbf{p} = \bar{\mathbf{p}} + \delta\mathbf{p}$. Therefore, the position estimate is a Gaussian centered at $\bar{\mathbf{p}}$. In contrast, when using the InEKF, the position and orientation are actually coupled together, $\mathbf{X} = \exp(\boldsymbol{\xi})\bar{\mathbf{X}}$. Although $\boldsymbol{\xi}$ is a Gaussian random vector, after applying the group's exponential map and matrix multiplication, the state estimate's uncertainty distribution is no longer Gaussian. This distribution is known as a *concentrated Gaussian on a Lie group* (Wang and Chirikjian, 2006, 2008). This type of distribution can often capture the underlying system uncertainty better than a standard Gaussian defined in Euclidean space (Barfoot and Furgale, 2014; Long et al., 2013).

To demonstrate the difference, a simple simulation was performed where the Cassie robot walked forward for 8 s at an average speed of 1 m/s. The standard deviation for the initial position uncertainty was set to 0.1 m about each axis, whereas the standard deviation of the initial yaw uncertainty was set to 10°. A set of 10,000 particles sampled from this distribution were propagated forward to represent the robot's true uncertainty distribution. After running both the InEKF and the QEKF, particles were sampled from the resulting filter covariances to provide a picture of the estimated position uncertainties. This result is shown in Figure 6.

The curved position distribution comes from the initial yaw uncertainty that continually grows due to its unobservability (Section 5.4). The InEKF is able to closely match this distribution since the samples are taken in the Lie algebra and are mapped to the group through the exponential map. This couples the orientation and position errors leading to a curved position distribution. In contrast, the QEKF position uncertainty can only have the shape of the standard Gaussian ellipse, which may not represent the true uncertainty well.

The InEKF can even accurately model the case of complete yaw uncertainty. To demonstrate this, the initial yaw standard deviation was set to 360°, and the same 8 s simulation was performed. Each ring in Figure 7 shows the sample position distribution spaced 2 s apart. This type of uncertainty cannot be captured with a standard Gaussian covariance ellipse. These examples illustrate that even if the means are identical, the covariance estimate of the InEKF can provide a more accurate representation of the state's uncertainty than the standard QEKF.

In these examples, the initial covariance of all states (except yaw) was small. It is interesting to note what happens if we remove the dynamics noise and set the covariance

for some states to exactly zero (rank-deficient covariance matrix). In this case, the initial covariance is supported by a subgroup and the InEKF will keep the state estimate within a time-dependent subset of the Lie group at all times. This theoretical result was proved by Chauchat et al. (2017) and Barrau and Bonnabel (2019) to create EKFs with state equality constraints.

7. IMU bias augmentation

Implementation of an IMU-based state estimator on hardware typically requires modeling additional states, such as gyroscope and accelerometer biases. Unfortunately, as noted in Barrau (2015), there is no Lie group that includes the bias terms while also having the dynamics satisfy the group affine property (2). Even though many of the theoretical properties of the RIEKF will no longer hold, it is possible to design an "imperfect InEKF" that still outperforms the standard EKF (Barrau, 2015).

7.1. State representation

The IMU biases are slowly varying signals that corrupt the measurements in an additive manner:

$$\begin{aligned}\tilde{\boldsymbol{\omega}}_t &= \boldsymbol{\omega}_t + \mathbf{b}^g_t + \mathbf{w}^g_t, \qquad \mathbf{w}^g_t \sim \mathcal{GP}\left(\mathbf{0}_{3,1}, \boldsymbol{\Sigma}^g\, \delta(t-t')\right)\\ \tilde{\mathbf{a}}_t &= \mathbf{a}_t + \mathbf{b}^a_t + \mathbf{w}^a_t, \qquad \mathbf{w}^a_t \sim \mathcal{GP}\left(\mathbf{0}_{3,1}, \boldsymbol{\Sigma}^a\, \delta(t-t')\right)\end{aligned}$$

These biases form a parameter vector that needs to be estimated as part of the RIEKF state,

$$\boldsymbol{\theta}_t \triangleq \begin{bmatrix}\mathbf{b}^g(t)\\ \mathbf{b}^a(t)\end{bmatrix} \triangleq \begin{bmatrix}\mathbf{b}^g_t\\ \mathbf{b}^a_t\end{bmatrix} \in \mathbb{R}^6 \tag{22}$$

The model's state now becomes a tuple of our original matrix Lie group and the parameter vector, $(\mathbf{X}_t, \boldsymbol{\theta}_t) \in \mathcal{G} \times \mathbb{R}^6$. The augmented right-invariant error is now defined as

$$\mathbf{e}^r_t \triangleq (\bar{\mathbf{X}}_t \mathbf{X}_t^{-1}, \bar{\boldsymbol{\theta}}_t - \boldsymbol{\theta}_t) \triangleq (\boldsymbol{\eta}^r_t, \boldsymbol{\zeta}_t) \tag{23}$$

Written explicitly, the right-invariant error is

$$\boldsymbol{\eta}^r_t = \begin{bmatrix}\bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T} & \bar{\mathbf{v}}_t - \bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T}\mathbf{v}_t & \bar{\mathbf{p}}_t - \bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T}\mathbf{p}_t & \bar{\mathbf{d}}_t - \bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T}\mathbf{d}_t\\ \mathbf{0}_{1,3} & 1 & 0 & 0\\ \mathbf{0}_{1,3} & 0 & 1 & 0\\ \mathbf{0}_{1,3} & 0 & 0 & 1\end{bmatrix}$$

whereas the parameter vector error is given by

$$\boldsymbol{\zeta}_t = \begin{bmatrix}\bar{\mathbf{b}}^g_t - \mathbf{b}^g_t\\ \bar{\mathbf{b}}^a_t - \mathbf{b}^a_t\end{bmatrix} \triangleq \begin{bmatrix}\boldsymbol{\zeta}^g_t\\ \boldsymbol{\zeta}^a_t\end{bmatrix}$$

7.2. System dynamics

With IMU biases included, the system dynamics are now expressed as

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t &= \mathbf{R}_t(\tilde{\boldsymbol{\omega}}_t - \mathbf{b}^g_t - \mathbf{w}^g_t)_\times\\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}_t &= \mathbf{R}_t(\tilde{\mathbf{a}}_t - \mathbf{b}^a_t - \mathbf{w}^a_t) + \mathbf{g}\\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{p}_t &= \mathbf{v}_t\\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{d}_t &= \mathbf{R}_t\, \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)(-\mathbf{w}^v_t)\end{aligned} \tag{24}$$

The IMU bias dynamics are modeled using the typical "Brownian motion" model, i.e., the derivatives are white Gaussian noise, to capture the slowly time-varying nature of these parameters,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{b}^g_t &= \mathbf{w}^{bg}_t, \qquad \mathbf{w}^{bg}_t \sim \mathcal{GP}(\mathbf{0}_{3,1}, \boldsymbol{\Sigma}^{bg}\,\delta(t-t'))\\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{b}^a_t &= \mathbf{w}^{ba}_t, \qquad \mathbf{w}^{ba}_t \sim \mathcal{GP}(\mathbf{0}_{3,1}, \boldsymbol{\Sigma}^{ba}\,\delta(t-t'))\end{aligned} \tag{25}$$

The deterministic system dynamics now depend on both the inputs, $\mathbf{u}_t$, and the parameters, $\boldsymbol{\theta}_t$

$$f_{\mathbf{u}_t}(\bar{\mathbf{X}}_t, \bar{\boldsymbol{\theta}}_t) = \begin{bmatrix}\bar{\mathbf{R}}_t(\bar{\boldsymbol{\omega}}_t)_\times & \bar{\mathbf{R}}_t\bar{\mathbf{a}}_t + \mathbf{g} & \bar{\mathbf{v}}_t & \mathbf{0}_{3,1}\\ \mathbf{0}_{1,3} & 0 & 0 & 0\\ \mathbf{0}_{1,3} & 0 & 0 & 0\\ \mathbf{0}_{1,3} & 0 & 0 & 0\end{bmatrix}$$

where $\bar{\boldsymbol{\omega}}_t \triangleq \tilde{\boldsymbol{\omega}}_t - \bar{\mathbf{b}}^g_t$ and $\bar{\mathbf{a}}_t \triangleq \tilde{\mathbf{a}}_t - \bar{\mathbf{b}}^a_t$ are the "bias-corrected" inputs. To compute the linearized error dynamics, the augmented right-invariant error (23) is first differentiated with respect to time,

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{e}^r_t = \left(\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\eta}^r_t, \begin{bmatrix}\mathbf{w}^{bg}_t\\ \mathbf{w}^{ba}_t\end{bmatrix}\right) \tag{26}$$

After carrying out the chain rule and making the first-order approximation, $\boldsymbol{\eta}^r_t = \exp(\boldsymbol{\xi}_t) \approx \boldsymbol{I}_d + \boldsymbol{\xi}^\wedge_t$, the individual terms of the invariant error dynamics become

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\left(\bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T}\right) &\approx (\bar{\mathbf{R}}_t\left(\mathbf{w}^g_t - \boldsymbol{\zeta}^g_t\right))_\times\\ \frac{\mathrm{d}}{\mathrm{d}t}\left(\bar{\mathbf{v}}_t - \bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T}\mathbf{v}_t\right) &\approx (\mathbf{g})_\times\,\boldsymbol{\xi}^R_t + (\bar{\mathbf{v}}_t)_\times\,\bar{\mathbf{R}}_t(\mathbf{w}^g_t - \boldsymbol{\zeta}^g_t)\\ &\quad + \bar{\mathbf{R}}_t(\mathbf{w}^a_t - \boldsymbol{\zeta}^a_t)\\ \frac{\mathrm{d}}{\mathrm{d}t}\left(\bar{\mathbf{p}}_t - \bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T}\mathbf{p}_t\right) &\approx \boldsymbol{\xi}^v_t + (\bar{\mathbf{p}}_t)_\times\,\bar{\mathbf{R}}_t(\mathbf{w}^g_t - \boldsymbol{\zeta}^g_t)\\ \frac{\mathrm{d}}{\mathrm{d}t}\left(\bar{\mathbf{d}}_t - \bar{\mathbf{R}}_t\mathbf{R}_t^\mathsf{T}\mathbf{d}_t\right) &\approx (\bar{\mathbf{d}}_t)_\times\,\bar{\mathbf{R}}_t(\mathbf{w}^g_t - \boldsymbol{\zeta}^g_t)\\ &\quad + \bar{\mathbf{R}}_t\,\boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}^v_t\end{aligned} \tag{27}$$

Importantly, the augmented invariant error dynamics only depends on the estimated trajectory though the noise and bias errors, $\boldsymbol{\zeta}_t$ (this is expected because when there are no

bias errors, there is no dependence on the estimated trajectory). A linear system can now be constructed from (27) to yield

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\boldsymbol{\xi}_t\\ \boldsymbol{\zeta}_t\end{bmatrix} = \mathbf{A}_t\begin{bmatrix}\boldsymbol{\xi}_t\\ \boldsymbol{\zeta}_t\end{bmatrix} + \begin{bmatrix}\mathrm{Ad}_{\bar{\mathbf{X}}_t} & \mathbf{0}_{12,6}\\ \mathbf{0}_{6,12} & \mathbf{I}_6\end{bmatrix}\mathbf{w}_t$$

where the noise vector is augmented to include the bias terms,

$$\mathbf{w}_t \triangleq \mathrm{vec}(\mathbf{w}_t^g,\ \mathbf{w}_t^a,\ \mathbf{0}_{3,1},\ \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^v, \mathbf{w}_t^{bg}, \mathbf{w}_t^{ba})$$

7.3. Forward kinematic measurements

The forward kinematics position measurement (17) does not depend on the IMU biases. Therefore, the $\mathbf{H}_t$ matrix can simply be appended with zeros to account for the augmented variables. The linear update equation becomes

$$\begin{bmatrix}\boldsymbol{\xi}_t^+\\ \boldsymbol{\zeta}_t^+\end{bmatrix} = \begin{bmatrix}\boldsymbol{\xi}_t\\ \boldsymbol{\zeta}_t\end{bmatrix} - \begin{bmatrix}\mathbf{K}_t^{\xi}\\ \mathbf{K}_t^{\zeta}\end{bmatrix}\left(\mathbf{H}_t\begin{bmatrix}\boldsymbol{\xi}_t\\ \boldsymbol{\zeta}_t\end{bmatrix} - \bar{\mathbf{R}}_t\left(\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^{\alpha}\right)\right)$$

7.4. Final continuous RIEKF equations

The final "imperfect" RIEKF equations that include IMU biases can now be written down. The estimated state tuple is predicted using the following set of differential equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\bar{\mathbf{X}}_t, \bar{\boldsymbol{\theta}}_t\right) = \left(f_{\mathbf{u}_t}(\bar{\mathbf{X}}_t, \bar{\boldsymbol{\theta}}_t), \mathbf{0}_{6,1}\right)$$

The covariance of the augmented right invariant error dynamics is computed by solving the Riccati equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{P}_t = \mathbf{A}_t\mathbf{P}_t + \mathbf{P}_t\mathbf{A}_t^{\mathsf{T}} + \bar{\mathbf{Q}}_t$$

where the matrices $\mathbf{A}_t$ and $\bar{\mathbf{Q}}_t$ are now defined using (27),

$$\mathbf{A}_t = \begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\bar{\mathbf{R}}_t & \mathbf{0}\\ (\mathbf{g})_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\mathbf{v}}_t)_\times\bar{\mathbf{R}}_t & -\bar{\mathbf{R}}_t\\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & -(\bar{\mathbf{p}}_t)_\times\bar{\mathbf{R}}_t & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\mathbf{d}}_t)_\times\bar{\mathbf{R}}_t & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix} \tag{28}$$

$$\bar{\mathbf{Q}}_t = \begin{bmatrix}\mathrm{Ad}_{\bar{\mathbf{X}}_t} & \mathbf{0}_{12,6}\\ \mathbf{0}_{6,12} & \mathbf{I}_6\end{bmatrix}\mathrm{Cov}(\mathbf{w}_t)\begin{bmatrix}\mathrm{Ad}_{\bar{\mathbf{X}}_t} & \mathbf{0}_{12,6}\\ \mathbf{0}_{6,12} & \mathbf{I}_6\end{bmatrix}^{\mathsf{T}}$$

The estimated state tuple and its covariance are corrected though the update equations

$$\begin{aligned}\left(\bar{\mathbf{X}}_t^+, \boldsymbol{\theta}_t^+\right) &= \left(\exp\left(\mathbf{K}_t^{\xi}\,\boldsymbol{\Pi}\bar{\mathbf{X}}_t\mathbf{Y}_t\right)\bar{\mathbf{X}}_t,\ \ \bar{\boldsymbol{\theta}}_t + \mathbf{K}_t^{\zeta}\,\boldsymbol{\Pi}\bar{\mathbf{X}}_t\mathbf{Y}_t\right)\\ \mathbf{P}_t^+ &= (\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)\,\mathbf{P}_t(\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)^{\mathsf{T}} + \mathbf{K}_t\bar{\mathbf{N}}_t\mathbf{K}_t^{\mathsf{T}}\end{aligned} \tag{29}$$

where the gains $\mathbf{K}_t^{\xi}$ and $\mathbf{K}_t^{\zeta}$ are computed from

$$\mathbf{S}_t = \mathbf{H}_t\mathbf{P}_t\mathbf{H}_t^{\mathsf{T}} + \bar{\mathbf{N}}_t \qquad \mathbf{K}_t = \begin{bmatrix}\mathbf{K}_t^{\xi}\\ \mathbf{K}_t^{\zeta}\end{bmatrix} = \mathbf{P}_t\mathbf{H}_t^{\mathsf{T}}\mathbf{S}_t^{-1}$$

with the following measurement, output, and noise matrices,

$$\begin{aligned}\mathbf{Y}_t^{\mathsf{T}} &= \begin{bmatrix}\boldsymbol{h}_p^{\mathsf{T}}(\tilde{\boldsymbol{\alpha}}_t) & 0 & 1 & -1\end{bmatrix},\\ \mathbf{H}_t &= \begin{bmatrix}\mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0}\end{bmatrix},\\ \bar{\mathbf{N}}_t &= \bar{\mathbf{R}}_t\,\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\ \mathrm{Cov}(\mathbf{w}_t^{\alpha})\ \mathbf{J}_p^{\mathsf{T}}(\tilde{\boldsymbol{\alpha}}_t)\ \bar{\mathbf{R}}_t^{\mathsf{T}}\end{aligned}$$

Remark 3. The upper-right block of the new linearized dynamics matrix (28) is related to the adjoint of the current state estimate (7). Intuitively, this maps the bias error (measured in the body frame) to the world frame.

8. Addition and removal of contact points

Sections 5 and 7 derived the equations for the RIEKF under the assumption that the contact point is unchanging with time. However, for legged robots, contacts are discrete events that are created and broken as a robot navigates through the environment. Therefore, it is important to be able to conveniently add and remove contact points states to and from the observer's state.

8.1. Removing contact points

To remove a previous contact point from the state, we marginalize the corresponding state variable by simply removing the corresponding column and row from the matrix Lie group. The corresponding elements of the covariance matrix are also eliminated. This can be done through a simple linear transformation. For example, if the robot is going from one contact to zero contacts, then the newly reduced covariance would be computed by

$$\begin{bmatrix}\boldsymbol{\xi}_t^R\\ \boldsymbol{\xi}_t^v\\ \boldsymbol{\xi}_t^p\end{bmatrix} = \begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}_t^R\\ \boldsymbol{\xi}_t^v\\ \boldsymbol{\xi}_t^p\\ \boldsymbol{\xi}_t^d\end{bmatrix} \tag{30}$$

$$\begin{aligned}\boldsymbol{\xi}_t^{\mathrm{new}} &\triangleq \mathbf{M}\,\boldsymbol{\xi}_t\\ \implies \mathbf{P}_t^{\mathrm{new}} &= \mathbf{M}\,\mathbf{P}_t\,\mathbf{M}^{\mathsf{T}}\end{aligned}$$

Remark 4. This marginalization matrix, $\mathbf{M}$, does not depend on the choice of right- or left-invariant error.

8.2. Adding contact points

When the robot makes a new contact with the environment, the state and covariance matrices need to be augmented. Special attention needs to be given to initialize the mean and covariance for the new estimated contact point. For example, if the robot is going from zero contacts to one contact, the initial mean is obtained though the forward kinematics relation

$$\bar{\mathbf{d}}_t = \bar{\mathbf{p}}_t + \bar{\mathbf{R}}_t\,\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) \tag{31}$$

In order to compute the new covariance, we need to look at the right-invariant error,

$$\begin{aligned}\boldsymbol{\eta}_t^d &= \bar{\mathbf{d}}_t - \bar{\mathbf{R}}_t \mathbf{R}_t^\mathsf{T} \mathbf{d}_t \\ &= \bar{\mathbf{p}}_t + \bar{\mathbf{R}}_t\, \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) - \bar{\mathbf{R}}_t \mathbf{R}_t^\mathsf{T} \mathbf{d}_t \\ &= \bar{\mathbf{p}}_t + \bar{\mathbf{R}}_t\, \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) - \bar{\mathbf{R}}_t \mathbf{R}_t^\mathsf{T} \left(\mathbf{p}_t + \mathbf{R}_t\, \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t - \mathbf{w}_t^\alpha)\right) \\ &\approx \boldsymbol{\eta}_t^p + \bar{\mathbf{R}}_t \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\, \mathbf{w}_t^\alpha \\ \implies \boldsymbol{\xi}_t^d &\approx \boldsymbol{\xi}_t^p + \bar{\mathbf{R}}_t \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\, \mathbf{w}_t^\alpha\end{aligned}$$

Therefore, covariance augmentation can be done using the following linear map,

$$\begin{aligned}\begin{bmatrix}\boldsymbol{\xi}_t^R \\ \boldsymbol{\xi}_t^v \\ \boldsymbol{\xi}_t^p \\ \boldsymbol{\xi}_t^d\end{bmatrix} &= \begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}\end{bmatrix} \begin{bmatrix}\boldsymbol{\xi}_t^R \\ \boldsymbol{\xi}_t^v \\ \boldsymbol{\xi}_t^p\end{bmatrix} + \begin{bmatrix}\mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \bar{\mathbf{R}}_t \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\end{bmatrix} \mathbf{w}_t^\alpha \\ \boldsymbol{\xi}_t^{\text{new}} &\triangleq \mathbf{F}_t\, \boldsymbol{\xi}_t + \mathbf{G}_t \mathbf{w}_t^\alpha \\ \implies \mathbf{P}_t^{\text{new}} &= \mathbf{F}_t\, \mathbf{P}_t\, \mathbf{F}_t^\mathsf{T} + \mathbf{G}_t\, \text{Cov}(\mathbf{w}_t^\alpha)\, \mathbf{G}_t^\mathsf{T}\end{aligned} \tag{32}$$

Remark 5. The error augmentation matrix, $\mathbf{F}_t$, and the noise matrix, $\mathbf{G}_t$, will depend on the choice of error variable. Here they are derived for the right-invariant error case. The matrices will differ in the left-invariant error formulation, as detailed in Section 10.

9. Experimental results on a Cassie robot

We now present an experimental evaluation of the proposed contact-aided RIEKF observer using a 3D bipedal robot. The Cassie-series robot, shown in Figure 1, developed by Agility Robotics, has 20 degrees of freedom coming from the body pose, 10 actuators, and 4 springs. The robot is equipped with an IMU along with 14 joint encoders that can measure all actuator and spring angles. The proposed and baseline algorithms (along with the robot's feedback controller) are implemented in MATLAB (Simulink Real-Time). The IMU (VectorNav-100) is located in the robot's torso and provides angular velocity and linear acceleration measurements at 800 Hz. The encoders provide joint angle measurements at 2,000 Hz. The robot has two springs on each leg that are compressed when the robot is standing on the ground. The spring deflections are measured by encoders and serve as a binary contact sensor. The controller used for these experiments was developed by Gong et al. (2019).

9.1. Convergence comparison

An experiment was performed where the robot walked forwards at approximately 0.3 m/s. The QEKF and the proposed RIEKF were run (off-line) 100 times using the same logged measurements, noise statistics, and initial covariance with random initial orientations and velocities. The noise statistics and initial covariance estimates are provided in Table 1. As with the simulation comparison presented in Section 6, the initial mean estimate for the Euler angles were uniformly sampled from -30° to 30° and the initial mean estimate for velocities were sampled uniformly from -1.0 m/s to 1.0 m/s. Bias estimation was turned on and the initial bias estimate was obtained from processing the IMU data when the robot was static. The pitch and roll estimates as well as the (body frame) velocity estimates for both filters are shown in Figure 8. The "ground truth" trajectory estimates (black lines) were computed by initializing each filter with a good state estimate at a time before the beginning of the plot (to allow for convergence). The initial orientation was obtained from the VectorNav-100's onboard EKF and the initial velocity was obtained through kinematics alone.

The experimental results for comparing filter convergence matches those of the simulation. The proposed RIEKF converges faster and more reliably in all 100 runs than the QEKF; therefore, owing to the convenience of initialization and reliability for tracking the developed RIEKF is the preferred observer.

When the state estimate is initialized close to the true value, the RIEKF and QEKF have similar performance (black lines), because the linearization of the error dynamics accurately reflects the underlying nonlinear dynamics. However, when the state estimate is far from the true value, the simulation and experimental results show that RIEKF consistently converges faster than the QEKF. The relatively poor performance of the QEKF is due to the error dynamics being linearized around the wrong operating point, in which case the linear system does not accurately reflect the nonlinear dynamics. In addition, when bias estimation is turned off, the invariant error dynamics of the RIEKF do not depend on the current state estimate. As a result, the linear error dynamics can be accurately used even when the current state estimate is far from its true value, leading to better performance over the QEKF. Although this theoretical advantage is lost when bias estimation is turned on, the experimental results (shown in Figure 8) indicate that the RIEKF still is the preferred observer owing to less sensitivity to initialization.

9.2. Motion capture experiment

In order to verify the accuracy of the InEKF state estimate, we performed a motion capture experiment in the University of Michigan's M-Air facility. This outdoor space is equipped with 18 Qualisys cameras that allows for position tracking. We had the Cassie robot walk untethered for 60 sec along an approximately 15 m path. A top-down view of the estimated trajectory is shown in Figure 9.

Although there is noticeable drift due to the unobservability of the position and yaw, the final position error accounts for less than 5% of the distance traveled. This drift error is due to a combination of sensor noise and imperfect modeling of the robot's kinematics which may introduce biases to the forward kinematic measurements.

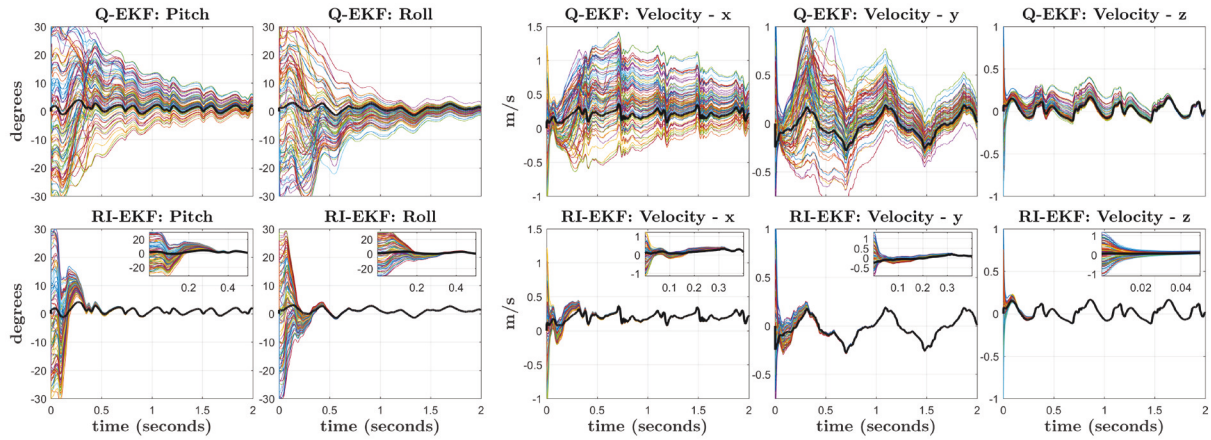

Fig. 8. An experiment was performed where an actual Cassie-series robot slowly walked forward at approximately 0.3 m/s. The noisy measurements came from the on-board IMU (VN-100) and the robot's joint encoders. The quaternion-based extended Kalman filter (QEKF) and the proposed right-invariant extended Kalman filter (RIEKF) were run (off-line) 100 times using the same measurements, noise statistics, and initial covariance, but with random initial orientations and velocities. The black line represents the filter state estimates when initialized with a good estimate. The RIEKF (bottom row) converges considerably faster than the QEKF (top row) for all observable states. Zoomed-in plots of the RIEKF performance are provided in the top-right corner.

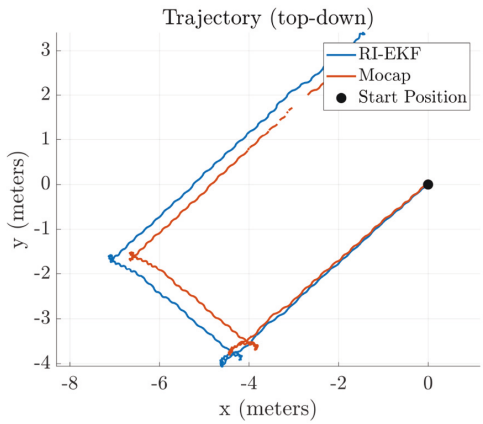

Fig. 9. Top-down view of the InEKF's estimated trajectory for a motion capture experiment. The position drift is unobservable, however, the final drift is less than 5% of the distance traveled.

The orientation, velocity, and position estimates along with their 3σ covariance hulls are shown in Figure 10. Owing to the unobservability of the yaw angle, the velocity estimate is given in the body frame instead of the world frame. The orientation is plotted using exponential coordinates, $\exp(\boldsymbol{\phi}) = \mathbf{R}$. Owing to an inaccurate orientation estimate from the motion capture system, the "ground truth" for the orientation is given by the VectorNav-100, which runs a state-of-the-art QEKF that fuses angular velocity, linear acceleration, and magnetometer measurements to estimate orientation only.

As expected, the error for all observable states remains small. The absolute position and the orientation about the gravity vector are unobservable, so some drift will occur for these states. While the drift is largely imperceptible for $\boldsymbol{\phi}_z$, it is interesting to note that the covariance slowly grows over time owing to this unobservability. More significant drift occurs on the absolute position states. This drift can be attributed to a combination of sensor noise, foot slip, and kinematic modeling errors.

In order to plot the 3σ covariance hull, the right-invariant error covariance needed to be converted to a covariance where the error is defined by Euclidean distance. Up to a first-order approximation, this mapping is done using

$$\begin{bmatrix} \delta\boldsymbol{\phi}_t \\ \delta\mathbf{v}_t \\ \delta\mathbf{p}_t \end{bmatrix} = \begin{bmatrix} -\boldsymbol{\Gamma}_1^{-1}(\bar{\boldsymbol{\phi}}_t) & \mathbf{0} & \mathbf{0} \\ (\bar{\mathbf{v}}_t)_\times & -\mathbf{I} & \mathbf{0} \\ (\bar{\mathbf{p}}_t)_\times & \mathbf{0} & -\mathbf{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{\xi}_t^R \\ \boldsymbol{\xi}_t^v \\ \boldsymbol{\xi}_t^p \end{bmatrix} \tag{33}$$

where the "Euclidean orientation error" is defined as $\delta\boldsymbol{\phi}_t \triangleq \boldsymbol{\phi}_t - \bar{\boldsymbol{\phi}}_t$, and the velocity and position errors match the QEKF error states (21). The matrix $\boldsymbol{\Gamma}_1(\bar{\boldsymbol{\phi}}_t)$ is known as the left Jacobian of SO(3) and has an analytical form. Further explanation and the derivation of the above equation is given in Appendix C.

9.3. Long odometry experiment

In addition to providing accurate estimates of states vital for legged robot control (orientation and velocity), this InEKF can also provide reliable odometry for a higher-level mapping or SLAM system. To demonstrate the accuracy of long-term odometry, we had Cassie walk about 200 m along a sidewalk around the University of Michigan's Wave Field. In total, the walk took 7 minutes and 45 seconds. The estimated path from the InEKF overlaid onto Google Earth imagery is shown in Figure 11. A video of this experiment can be found at https://youtu.be/jRUltB_dMlo.

Even though the absolute position is unobservable, the odometry estimate from the InEKF contains low enough drift to keep the estimate on the sidewalk for the duration of

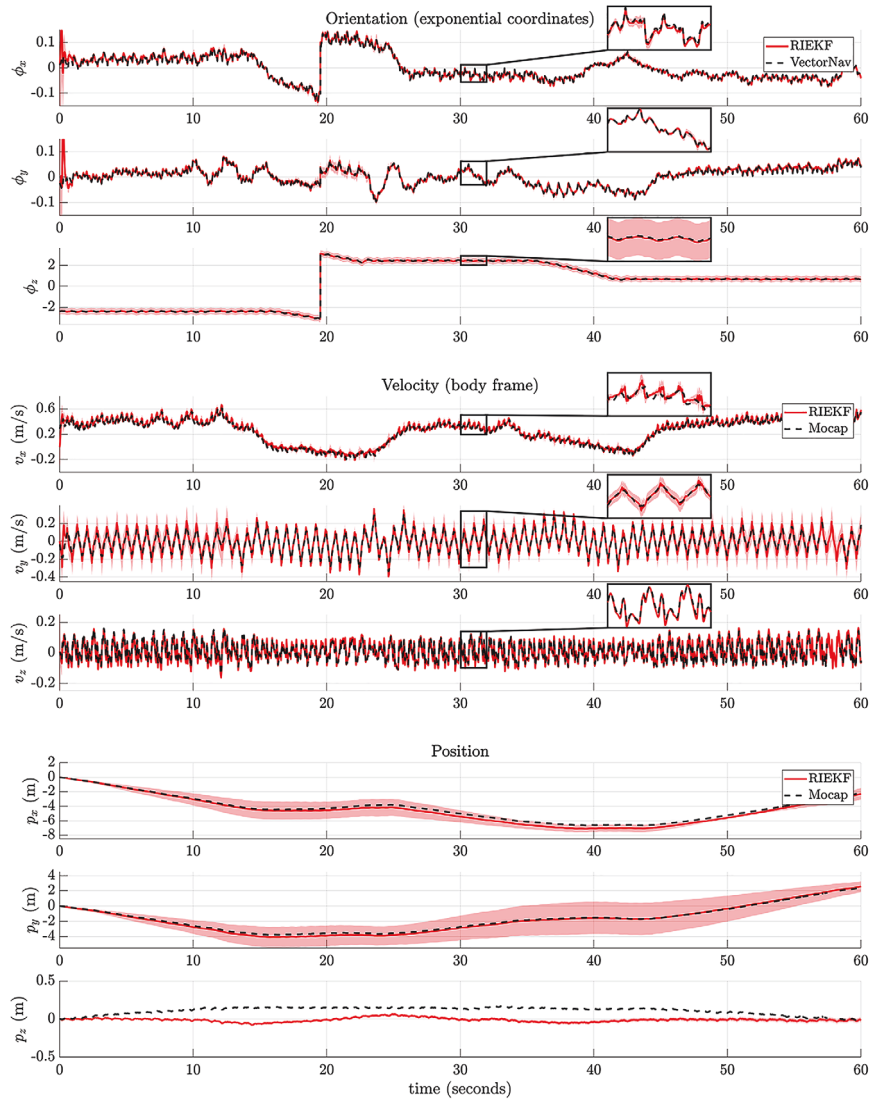

Fig. 10. Motion capture experiment conducted in the University of Michigan's M-Air facility. The dashed black line represents ground truth, the solid red line is the right-invariant EKF estimate, and the red shaded area represents the 3σ covariance hull. The ground truth for position and velocity were obtained using 18 Qualisys cameras. Owing to poor orientation estimates from the motion capture system, the "ground truth" for orientation was obtained from the VectorNav-100, which runs a highly accurate on-board QEKF. The orientation data is plotted using the exponential coordinates.

the experiment. The final position estimate is within a few meters of the true position, and the yaw drift is imperceptible. This odometry estimate is readily available, as it only depends on inertial, contact, and kinematic data, which barring sensor failure, always exist. It does not require the use of any vision systems that may be susceptible to changes in environment or lighting conditions.

9.4. LiDAR Mapping Application

One application for the InEKF odometry is the building of local maps of the environment. We equipped the Cassie-series robot with a new torso that houses a Velodyne VLP-32C LiDAR. With the filter running, we can project each received packet of point cloud data into the world frame based on the current state estimate. This point cloud data can then be accumulated to create a map of the environment. Figure 12 shows a few still frames from several LiDAR mapping experiments. Videos of these results can be viewed at https://youtu.be/pNyXsZ5zVZk and https://youtu.be/nbQTQw0gJ-k.

10. Alternative left-invariant formulation

For the derivations in Sections 5–8, we were assuming the use of the right-invariant error. This choice was due to the forward kinematic measurement having the right-invariant observation form. However, it is possible to derive a left-invariant form of this filter, which may be more appropriate to use when dealing with left-invariant observations.

Fig. 11. Long outdoor odometry experiment where Cassie walked roughly 200 m along a sidewalk over 7 minutes and 45 seconds. A video of this experiment can be found at https://youtu.be/jRUltB_dMlo

For example, GPS measurements are left-invariant observations for the world-centric observer; see Section 12. Written explicitly, the left-invariant error is

$$\boldsymbol{\eta}_t^l \triangleq \mathbf{X}_t^{-1}\bar{\mathbf{X}}_t = \begin{bmatrix} \mathbf{R}_t^\mathsf{T}\bar{\mathbf{R}}_t & \mathbf{R}_t^\mathsf{T}(\bar{\mathbf{v}}_t - \mathbf{v}_t) & \mathbf{R}_t^\mathsf{T}(\bar{\mathbf{p}}_t - \mathbf{p}_t) & \mathbf{R}_t^\mathsf{T}(\bar{\mathbf{d}}_t - \mathbf{d}_t) \\ \mathbf{0}_{1,3} & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 \end{bmatrix}$$

After carrying out the chain rule and making the first-order approximation, $\boldsymbol{\eta}_t^l = \exp(\boldsymbol{\xi}_t) \approx \boldsymbol{I}_d + \boldsymbol{\xi}_t^\wedge$, the individual terms of the left-invariant error dynamics become

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t^\mathsf{T}\bar{\mathbf{R}}_t &\approx (-(\tilde{\boldsymbol{\omega}}_t - \boldsymbol{\zeta}_t^g)_\times \boldsymbol{\xi}_t^R - \boldsymbol{\zeta}_t^g + \mathbf{w}_t^g)_\times \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t^\mathsf{T}(\bar{\mathbf{v}}_t - \mathbf{v}_t) &\approx -(\tilde{\mathbf{a}}_t - \bar{\mathbf{b}}_t^a)_\times \boldsymbol{\xi}_t^R \\ &\quad -(\tilde{\boldsymbol{\omega}}_t - \bar{\mathbf{b}}_t^g)_\times \boldsymbol{\xi}_t^v - \boldsymbol{\zeta}_t^a + \mathbf{w}_t^a \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t^\mathsf{T}(\bar{\mathbf{p}}_t - \mathbf{p}_t) &\approx \boldsymbol{\xi}_t^v - (\tilde{\boldsymbol{\omega}}_t - \bar{\mathbf{b}}_t^g)_\times \boldsymbol{\xi}_t^p \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t^\mathsf{T}(\bar{\mathbf{d}}_t - \mathbf{d}_t) &\approx -(\tilde{\boldsymbol{\omega}}_t - \bar{\mathbf{b}}_t^g)_\times \boldsymbol{\xi}_t^d + \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\mathbf{w}_t^v \end{aligned}$$

Using these results, the log-linear left-invariant dynamics can be expressed using the following linear system

$$\begin{aligned} &\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\xi}_t = \mathbf{A}_t\boldsymbol{\xi}_t + \mathbf{w}_t \\ \Longrightarrow\ &\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{P}_t = \mathbf{A}_t\mathbf{P}_t + \mathbf{P}_t\mathbf{A}_t^\mathsf{T} + \bar{\mathbf{Q}}_t \end{aligned}$$

where the dynamics and noise matrices are

$$\mathbf{A}_t^l = \begin{bmatrix} -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} \\ -(\bar{\mathbf{a}}_t)_\times & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{I} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\bar{\mathbf{Q}}_t = \mathrm{Cov}(\mathbf{w}_t) \tag{34}$$

Similar to the right-invariant case, the dynamics only depend on the state through the IMU bias. When a left-invariant observation comes in, the state estimate is corrected using

$$\left(\bar{\mathbf{X}}_t^+, \boldsymbol{\theta}_t^+\right) = \left(\bar{\mathbf{X}}_t \exp\left(\mathbf{K}_t^\xi \boldsymbol{\Pi}\bar{\mathbf{X}}_t^{-1}\mathbf{Y}_t\right),\ \bar{\boldsymbol{\theta}}_t + \mathbf{K}_t^\zeta \boldsymbol{\Pi}\bar{\mathbf{X}}_t^{-1}\mathbf{Y}_t\right) \tag{35}$$

where the exponential map is now multiplied on the right-hand side (Barrau and Bonnabel, 2017).

10.1. Switching between left and right-invariant errors

Because forward kinematic measurements have the right-invariant observation form, the innovation equations are only autonomous when the right-invariant error is used. Fortunately, it is possible to switch between the left and right error forms through the use of the adjoint map:

$$\begin{aligned} &\boldsymbol{\eta}_t^r = \bar{\mathbf{X}}_t\mathbf{X}_t^{-1} = \bar{\mathbf{X}}_t\boldsymbol{\eta}_t^l\bar{\mathbf{X}}_t^{-1} \\ \Longrightarrow\ &\exp(\boldsymbol{\xi}_t^r) = \bar{\mathbf{X}}_t\exp(\boldsymbol{\xi}_t^l)\bar{\mathbf{X}}_t^{-1} = \exp(\mathrm{Ad}_{\bar{\mathbf{X}}_t}\boldsymbol{\xi}_t^l) \\ \Longrightarrow\ &\boldsymbol{\xi}_t^r = \mathrm{Ad}_{\bar{\mathbf{X}}_t}\boldsymbol{\xi}_t^l \end{aligned}$$

This transformation is exact, which means that we can easily switch between the covariance of the left- and right-invariant errors using

$$\mathbf{P}_t^r = \mathrm{Ad}_{\bar{\mathbf{X}}_t}\mathbf{P}_t^l\mathrm{Ad}_{\bar{\mathbf{X}}_t}^\mathsf{T} \tag{36}$$

Therefore, when handling a right-invariant observation, we can map the propagated left-invariant covariance to the right-invariant covariance temporarily, apply the right-invariant update equations (29), then map the corrected covariance back to the left-invariant form.

It is also possible to compute the log-linear left-invariant dynamics starting from the right-invariant form. Substituting (36) into the (right-invariant) covariance propagation equation (13) and solving for the left-invariant covariance yields

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{P}_t^l = &\left(\mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}}\mathbf{A}_t^r\mathrm{Ad}_{\bar{\mathbf{X}}_t} - \mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}}\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathrm{Ad}_{\bar{\mathbf{X}}_t}\right)\right)\mathbf{P}_t^l \\ &+ \mathbf{P}_t^l\left(\mathrm{Ad}_{\bar{\mathbf{X}}_t}^\mathsf{T}\mathbf{A}_t^{r\mathsf{T}}\mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}}^\mathsf{T} - \frac{\mathrm{d}}{\mathrm{d}t}\left(\mathrm{Ad}_{\bar{\mathbf{X}}_t}^\mathsf{T}\right)\mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}}^\mathsf{T}\right) \\ &+ \mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}}\bar{\mathbf{Q}}_t^r\mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}}^\mathsf{T} \end{aligned}$$

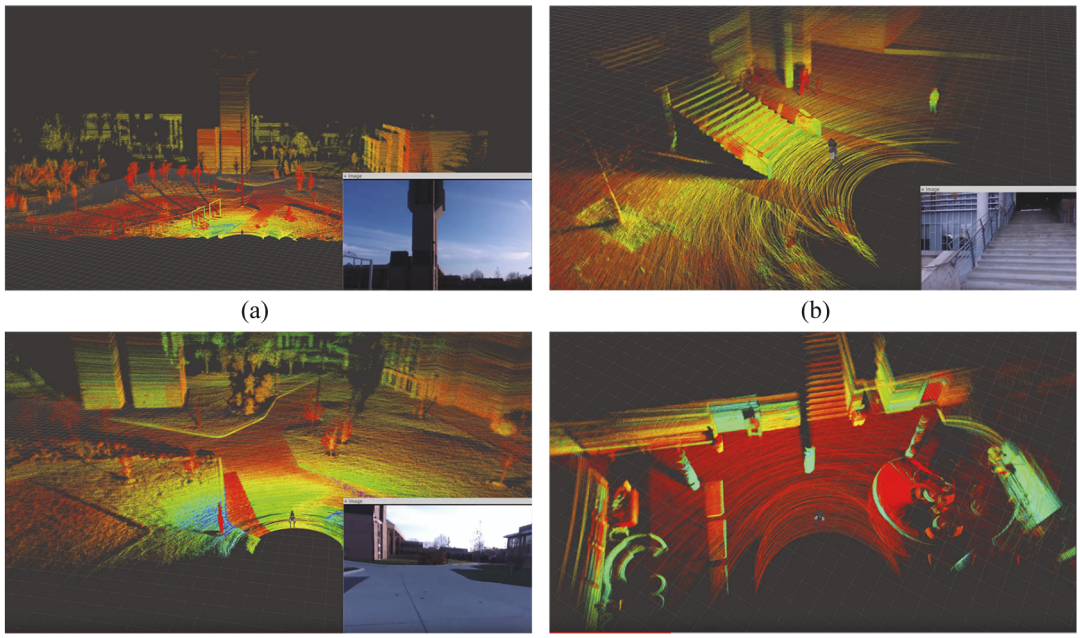

Fig. 12. LiDAR maps created by transforming 10 seconds of point cloud data onto the pose trajectory estimated by the InEKF: (a) University of Michigan's North Campus with the bell tower; (b) looking towards a staircase; (c) walking along a sidewalk; (d) inside the Bob and Betty Beyster Building. The high-frequency odometry estimate allows for motion compensation within a single scan of the LiDAR (10Hz). Videos of this experiment can be found at https://youtu.be/pNyXsZ5zVZk and https://youtu.be/nbQTQw0gJ-k.

Therefore, we learn that the right and left dynamics and noise matrices are related by the following expressions

$$\begin{aligned}\mathbf{A}_t^l &\triangleq \mathrm{Ad}_{\tilde{\mathbf{X}}_t^{-1}}\mathbf{A}_t^r\mathrm{Ad}_{\tilde{\mathbf{X}}_t} - \mathrm{Ad}_{\tilde{\mathbf{X}}_t^{-1}}\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathrm{Ad}_{\tilde{\mathbf{X}}_t}\right)\\ \bar{\mathbf{Q}}_t^l &\triangleq \mathrm{Ad}_{\tilde{\mathbf{X}}_t^{-1}}\bar{\mathbf{Q}}_t^r\mathrm{Ad}_{\tilde{\mathbf{X}}_t^{-1}}^{\mathsf{T}}\end{aligned} \tag{37}$$

Remark 6. Intuitively, the left-invariant error represents an error measured in the body frame of the robot, whereas the right-invariant error represents an error measured in the world or spatial frame. The frame of measurement dictates whether the exponential map appears on the right or left in the update equations. The error can be moved between these two frames using the adjoint map of the Lie group.

10.2. Adding new contact points

The process for removing a contact point from the state (marginalization) is identical to the right-invariant error case, described in Section 8.1. Likewise, when a new contact is detected, the state can be augmented using the same kinematics relation (31) as before. However, owing to the change in error variable, the process for augmenting the covariance will be different.

In order to compute the new covariance, we need to look at the left-invariant error,

$$\begin{aligned}\boldsymbol{\eta}_t^d &= \mathbf{R}_t^{\mathsf{T}}(\bar{\mathbf{d}}_t - \mathbf{d}_t)\\ &= \mathbf{R}_t^{\mathsf{T}}\left(\bar{\mathbf{p}}_t + \bar{\mathbf{R}}_t\,\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t)\right)\\ &\quad - \mathbf{R}_t^{\mathsf{T}}\left(\mathbf{p}_t + \mathbf{R}_t\,\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t - \mathbf{w}_t^{\alpha})\right)\\ &\approx \boldsymbol{\eta}_t^p + \boldsymbol{\eta}_t^R\,\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) - \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) + \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^{\alpha}\\ \implies \boldsymbol{\xi}_t^d &\approx \boldsymbol{\xi}_t^p - (\,\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t))_{\times}\,\boldsymbol{\xi}_t^R + \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^{\alpha}\end{aligned}$$

Therefore, covariance augmentation can be done using the following linear map,

$$\begin{aligned}\begin{bmatrix}\boldsymbol{\xi}_t^R\\ \boldsymbol{\xi}_t^v\\ \boldsymbol{\xi}_t^p\\ \boldsymbol{\xi}_t^d\end{bmatrix} &= \begin{bmatrix}\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{0} & \mathbf{I}\\ (-\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t))_{\times} & \mathbf{0} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}_t^R\\ \boldsymbol{\xi}_t^v\\ \boldsymbol{\xi}_t^p\end{bmatrix} + \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \mathbf{0}\\ \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\end{bmatrix}\mathbf{w}_t^{\alpha}\\ \boldsymbol{\xi}_t^{\text{new}} &\triangleq \mathbf{F}_t\,\boldsymbol{\xi}_t + \mathbf{G}_t\mathbf{w}_t^{\alpha}\\ \implies \mathbf{P}_t^{\text{new}} &= \mathbf{F}_t\,\mathbf{P}_t\,\mathbf{F}_t^{\mathsf{T}} + \mathbf{G}_t\,\mathrm{Cov}(\mathbf{w}_t^{\alpha})\,\mathbf{G}_t^{\mathsf{T}}\end{aligned} \tag{38}$$

11. Robo-centric estimator

In this section, we derive a "robot-centric" version of the contact-aided InEKF where the estimated state is measured in the robot's base (IMU) frame. When switching to a robot-centric model, the forward kinematics measurements take the left-invariant observation form. In addition, the left-/right-invariant error dynamics equations are identical to the world-centric form, albeit swapped. The right-invariant error dynamics for the world-centric estimator are equivalent to the left-invariant error dynamics for the robo-centric estimator.

The properties of this filter are identical to the RIEKF derived in Section 5. However, in some cases this filter may be preferred as it directly estimates states that are useful for controlling a legged robot (namely the velocity measured in the body frame).

11.1. State and dynamics

We are interested in estimating the same states as before, though measured in the robot's body frame. Again, the state variables can form a matrix Lie group, $\mathcal{G}$. Specifically, for N contact points, $\mathbf{X}_t \in \mathrm{SE}_{N+2}(3)$ can be represented by the following matrix (which is simply the inverse of the world-centric state)[5]:

$$\overset{\text{redefine}}{\mathbf{X}_t \triangleq} \begin{bmatrix} \mathbf{R}_{\mathrm{BW}}(t) & -{}_{\mathrm{B}}\mathbf{v}_{\mathrm{B}}(t) & {}_{\mathrm{B}}\mathbf{p}_{\mathrm{BW}}(t) & {}_{\mathrm{B}}\mathbf{p}_{\mathrm{C}_1\mathrm{W}}(t) & \cdots & {}_{\mathrm{B}}\mathbf{p}_{\mathrm{C}_N\mathrm{W}}(t) \\ \mathbf{0}_{1\times 3} & 1 & 0 & 0 & \cdots & 0 \\ \mathbf{0}_{1\times 3} & 0 & 1 & 0 & \cdots & 0 \\ \mathbf{0}_{1\times 3} & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{1\times 3} & 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

Without loss of generality, assume a single contact point. Furthermore, for the sake of readability, we redefine our shorthand notation to be body-centric states,

$$\mathbf{X}_t \overset{\text{redefine}}{\triangleq} \begin{bmatrix} \mathbf{R}_t & \mathbf{v}_t & \mathbf{p}_t & \mathbf{d}_t \\ \mathbf{0}_{1\times 3} & 1 & 0 & 0 \\ \mathbf{0}_{1\times 3} & 0 & 1 & 0 \\ \mathbf{0}_{1\times 3} & 0 & 0 & 1 \end{bmatrix} \tag{39}$$

Using the bias corrected IMU measurements, the individual terms of the new robot-centric system dynamics can be derived as (Bloesch et al., 2013)

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{R}_t &= -(\bar{\boldsymbol{\omega}}_t - \mathbf{w}_t^g)_\times \mathbf{R}_t \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}_t &= -(\bar{\mathbf{a}}_t - \mathbf{w}_t^a) - \mathbf{R}_t\,\mathbf{g} - (\bar{\boldsymbol{\omega}}_t - \mathbf{w}_t^g)_\times \mathbf{v}_t \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{p}_t &= \mathbf{v}_t - (\bar{\boldsymbol{\omega}}_t - \mathbf{w}_t^g)_\times \mathbf{p}_t \\ \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{d}_t &= -(\bar{\boldsymbol{\omega}}_t - \mathbf{w}_t^g)_\times \mathbf{d}_t - \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^v \end{aligned} \tag{40}$$

Written in matrix form, this becomes

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{X}_t &= \begin{bmatrix} -(\bar{\boldsymbol{\omega}}_t)_\times \mathbf{R}_t & -\bar{\mathbf{a}}_t - \mathbf{R}_t\mathbf{g} - (\bar{\boldsymbol{\omega}}_t)_\times \mathbf{v}_t & \mathbf{v}_t - (\bar{\boldsymbol{\omega}}_t)_\times \mathbf{p}_t & -(\bar{\boldsymbol{\omega}}_t)_\times \mathbf{d}_t \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix} \\ &\quad - \begin{bmatrix} (\mathbf{w}_t^g)_\times & \mathbf{w}_t^a & \mathbf{0}_{3,1} & \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^v \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}_t & \mathbf{v}_t & \mathbf{p}_t & \mathbf{d}_t \\ \mathbf{0}_{1,3} & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 \end{bmatrix} \\ &\triangleq f_u(\mathbf{X}_t, \boldsymbol{\theta}_t) - (\mathbf{w}_t)^\wedge \mathbf{X}_t \end{aligned} \tag{41}$$

with $\mathbf{w}_t \triangleq \mathrm{vec}[\mathbf{w}_t^g,\ \mathbf{w}_t^a,\ \mathbf{0}_{3,1},\ \boldsymbol{h}_R(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^v]$. The deterministic dynamics function $f_u(\cdot)$ can be shown to satisfy the group affine property (2). Therefore, the left- (and right-) invariant error dynamics depends solely on the invariant error.

We can derive the log-linear dynamics matrices for the body-centric estimator following a similar derivation process to the world-centric version in Section 5. In fact, without IMU bias, the linearization of the left-invariant dynamics for the body-centric estimator is the same as the right-invariant dynamics for the world-centric estimator:

$$\begin{aligned} \mathbf{A}_t^l \quad &\textit{(body-centric)} = \mathbf{A}_t^r \quad \textit{(world-centric)} \\ \mathbf{A}_t^r \quad &\textit{(body-centric)} = \mathbf{A}_t^l \quad \textit{(world-centric)} \end{aligned} \tag{42}$$

When IMU bias is included, the above relation still holds, though with the bias terms negated.[6] The noise covariance matrices for the body-centric left-/right-invariant propagation models are given by

$$\begin{aligned} \bar{\mathbf{Q}}_t^l &= \begin{bmatrix} \mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}} & \mathbf{0}_{15,6} \\ \mathbf{0}_{6,15} & \mathbf{I}_6 \end{bmatrix} \mathrm{Cov}(\mathbf{w}_t) \begin{bmatrix} \mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}} & \mathbf{0}_{15,6} \\ \mathbf{0}_{6,15} & \mathbf{I}_6 \end{bmatrix}^\mathsf{T} \\ \bar{\mathbf{Q}}_t^r &= \mathrm{Cov}(\mathbf{w}_t) \end{aligned} \tag{43}$$

which are also swapped versions of the world-centric noise matrices, after accounting for the redefinition of $\bar{\mathbf{X}}_t$ as its inverse. A comparison of the world-centric and body-centric equations are given in Tables 2 and 3.

11.2. Left-invariant forward kinematic measurement model

We use forward-kinematics to measure the relative position of the contact point with respect to the body, $\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t)$. Using the new robot-centric state variables, the measurement model (16) becomes

$$\boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) = \mathbf{p}_t - \mathbf{d}_t + \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t \tag{44}$$

Re-written in matrix form, this measurement will have now have the left-invariant observation form (4),

$$\underbrace{\begin{bmatrix} \boldsymbol{h}_p(\tilde{\boldsymbol{\alpha}}_t) \\ 0 \\ 1 \\ -1 \end{bmatrix}}_{\mathbf{Y}_t} = \underbrace{\begin{bmatrix} \mathbf{R}_t & \mathbf{v}_t & \mathbf{p}_t & \mathbf{d}_t \\ \mathbf{0}_{1,3} & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{X}_t} \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}}_{\mathbf{b}} + \underbrace{\begin{bmatrix} \mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\,\mathbf{w}_t^\alpha \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{V}_t} \tag{45}$$

The state update equation will take the left-invariant form (35), where the matrices $\mathbf{H}_t$ and $\bar{\mathbf{N}}_t$ can be derived to be

$$\begin{aligned} \mathbf{H}_t &= \begin{bmatrix} \mathbf{0}_{3,3} & \mathbf{0}_{3,3} & \mathbf{I} & -\mathbf{I} \end{bmatrix} \\ \bar{\mathbf{N}}_t &= \bar{\mathbf{R}}_t^\mathsf{T}\,\mathbf{J}_p(\tilde{\boldsymbol{\alpha}}_t)\ \mathrm{Cov}(\mathbf{w}_t^\alpha)\ \mathbf{J}_p^\mathsf{T}(\tilde{\boldsymbol{\alpha}}_t)\ \bar{\mathbf{R}}_t \end{aligned} \tag{46}$$

It is important to note that the forward kinematics measurement is a right-invariant observation for the world-centric estimator, whereas the same measurement becomes a left-invariant observation for the body-centric version. In addition, the above linearization (46) is similar to the world-centric, right-invariant one (20). Namely, $\mathbf{H}_t$ is simply negated, whereas $\bar{\mathbf{N}}_t$ is identical after accounting for the redefinition of the state as its inverse.

Table 2. Summary of world-centric state estimator.

State definition

$$\mathbf{X}_t \triangleq \begin{bmatrix} \mathbf{R}_{\mathrm{WB}} & {}^{\mathrm{W}}\mathbf{v}_{\mathrm{B}} & {}^{\mathrm{W}}\mathbf{p}_{\mathrm{WB}} & {}^{\mathrm{W}}\mathbf{p}_{\mathrm{WC}} & {}^{\mathrm{W}}\mathbf{p}_{\mathrm{WL}} \\ \mathbf{0}_{1,3} & 1 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 1 \end{bmatrix}$$

Deterministic nonlinear dynamics

$$f_{\mathbf{u}_t}(\bar{\mathbf{X}}_t, \bar{\boldsymbol{\theta}}_t) = \begin{bmatrix} \bar{\mathbf{R}}_t(\bar{\boldsymbol{\omega}}_t)_\times & \bar{\mathbf{R}}_t\bar{\mathbf{a}}_t + \mathbf{g} & \bar{\mathbf{v}}_t & \mathbf{0}_{3,1} & \mathbf{0}_{3,1} \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \end{bmatrix}$$

Log-linear right-invariant dynamics

$$\mathbf{A}_t^r = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\bar{\mathbf{R}}_t & \mathbf{0} \\ (\mathbf{g})_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\mathbf{v}}_t)_\times \bar{\mathbf{R}}_t & -\bar{\mathbf{R}}_t \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\mathbf{p}}_t)_\times \bar{\mathbf{R}}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\mathbf{d}}_t)_\times \bar{\mathbf{R}}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\mathbf{l}}_t)_\times \bar{\mathbf{R}}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\bar{\mathbf{Q}}_t^r = \begin{bmatrix} \mathrm{Ad}_{\bar{\mathbf{X}}_t} & \mathbf{0}_{15,6} \\ \mathbf{0}_{6,15} & \mathbf{I}_6 \end{bmatrix} \mathrm{Cov}(\mathbf{w}_t) \begin{bmatrix} \mathrm{Ad}_{\bar{\mathbf{X}}_t} & \mathbf{0}_{15,6} \\ \mathbf{0}_{6,15} & \mathbf{I}_6 \end{bmatrix}^\mathsf{T}$$

Log-linear left-invariant dynamics

$$\mathbf{A}_t^l = \begin{bmatrix} -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} \\ -(\bar{\mathbf{a}}_t)_\times & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{I} \\ \mathbf{0} & \mathbf{I} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\bar{\mathbf{Q}}_t^l = \mathrm{Cov}(\mathbf{w}_t)$$

Measurement	Observation matrix, $\mathbf{H}$	Observation type
Forward kinematic	$\begin{bmatrix}\mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Right-invariant
Relative landmark position	$\begin{bmatrix}\mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Right-invariant
Absolute landmark position	$\begin{bmatrix}(\mathbf{l})_\times & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Right-invariant
Magnetometer	$\begin{bmatrix}(\mathbf{m})_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Right-invariant
GPS position	$\begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Left-Invariant

Table 3. Summary of robo-centric state estimator.

State definition

$$\mathbf{X}_t \triangleq \begin{bmatrix} \mathbf{R}_{\mathrm{BW}} & -{}^{\mathrm{B}}\mathbf{v}_{\mathrm{B}} & {}^{\mathrm{B}}\mathbf{p}_{\mathrm{BW}} & {}^{\mathrm{B}}\mathbf{p}_{\mathrm{CW}} & {}^{\mathrm{B}}\mathbf{p}_{\mathrm{LW}} \\ \mathbf{0}_{1,3} & 1 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 1 \end{bmatrix}$$

Deterministic nonlinear dynamics

$$f_{\mathbf{u}_t}(\bar{\mathbf{X}}_t, \bar{\boldsymbol{\theta}}_t) = \begin{bmatrix} -(\bar{\boldsymbol{\omega}}_t)_\times \mathbf{R}_t & -\bar{\mathbf{a}}_t - \mathbf{R}_t\mathbf{g} - (\bar{\boldsymbol{\omega}}_t)_\times \mathbf{v}_t & \mathbf{v}_t - (\bar{\boldsymbol{\omega}}_t)_\times \mathbf{p}_t & -(\bar{\boldsymbol{\omega}}_t)_\times \mathbf{d}_t & -(\bar{\boldsymbol{\omega}}_t)_\times \mathbf{l}_t \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & 0 \end{bmatrix}$$

Log-linear right-invariant dynamics

$$\mathbf{A}_t^r = \begin{bmatrix} -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ -(\bar{\mathbf{a}}_t)_\times & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{I} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -(\bar{\boldsymbol{\omega}}_t)_\times & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\bar{\mathbf{Q}}_t^r = \mathrm{Cov}(\mathbf{w}_t)$$

Log-linear left-invariant dynamics

$$\mathbf{A}_t^l = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \bar{\mathbf{R}}_t & \mathbf{0} \\ (\mathbf{g})_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & (\bar{\mathbf{v}}_t)_\times \bar{\mathbf{R}}_t & \bar{\mathbf{R}}_t \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & (\bar{\mathbf{p}}_t)_\times \bar{\mathbf{R}}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & (\bar{\mathbf{d}}_t)_\times \bar{\mathbf{R}}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & (\bar{\mathbf{l}}_t)_\times \bar{\mathbf{R}}_t & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\bar{\mathbf{Q}}_t^l = \begin{bmatrix} \mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}} & \mathbf{0}_{15,6} \\ \mathbf{0}_{6,15} & \mathbf{I}_6 \end{bmatrix} \mathrm{Cov}(\mathbf{w}_t) \begin{bmatrix} \mathrm{Ad}_{\bar{\mathbf{X}}_t^{-1}} & \mathbf{0}_{15,6} \\ \mathbf{0}_{6,15} & \mathbf{I}_6 \end{bmatrix}^\mathsf{T}$$

Measurement	Observation matrix, $\mathbf{H}$	Observation type
Forward kinematic	$\begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{I} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Left-invariant
Relative landmark position	$\begin{bmatrix}\mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Left-invariant
Absolute landmark position	$\begin{bmatrix}-(\mathbf{l})_\times & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Left-invariant
Magnetometer	$\begin{bmatrix}-(\mathbf{m})_\times & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Left-invariant
GPS position	$\begin{bmatrix}\mathbf{0} & \mathbf{0} & -\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0}\end{bmatrix}$	Right-invariant

12. Additional sensor observations

This article extends upon our conference paper results (Hartley et al., 2018b) where we originally presented the contact-aided InEKF. As described in earlier sections, this filter uses an inertial-contact dynamics model with corrections coming from forward kinematics. In addition to forward kinematics, other groups have discovered that a number of measurements common to robotics can also fit the invariant observation model (4). Bonnabel (2007) developed an invariant observer that uses magnetometer and acceleration measurements to solve the attitude estimation problem. Barczyk and Lynch (2011) and Barrau (2015) described methods for invariant observer design for GPS and magnetometer-aided navigation. Wu et al. (2017) developed an InEKF to solve visual–inertial navigation. It has also been shown that an InEKF can be used for SLAM (Barrau and Bonnabel, 2015; Bonnabel, 2012; Zhang et al., 2017).

In particular, it is interesting to note the similarities between our contact-aided InEKF and landmark-based SLAM. In the simplest case, this SLAM problem involves jointly estimating the robot's state along with the position of static landmarks in the environment. The robot is often assumed to have a sensor capable of measuring the position of the landmark relative to the robot. This formulation is identical to our developed InEKF with the contact positions acting as landmarks and forward kinematics measuring the relative translation between the base and contact frames. This similarity was also mentioned by Bloesch et al. (2012). However, there are a few notable differences. The contact frame velocity is assumed to be white noise to allow for foot slip, whereas landmarks are usually treated as static. Forward kinematics measurements often come at high frequencies (2,000 Hz on Cassie). In contrast, landmarks measurements are often at a much lower frequency. Finally, with landmark observations, a data association problem often has to be solved which associates the measurement with a particular landmark state. This problem does not exist with forward kinematic measurements.

Owing to the similarities between contacts and landmarks, in our state matrix, the contact states could be easily replaced with landmark states, ${}^{\mathrm{W}}\mathbf{p}_{\mathrm{WL}_i}(t)$,

$$\mathbf{X}_t \triangleq \begin{bmatrix} \mathbf{R}_{\mathrm{WB}}(t) & {}^{\mathrm{W}}\mathbf{v}_{\mathrm{B}}(t) & {}^{\mathrm{W}}\mathbf{p}_{\mathrm{WB}}(t) & {}^{\mathrm{W}}\mathbf{p}_{\mathrm{WL}_1}(t) & \cdots & {}^{\mathrm{W}}\mathbf{p}_{\mathrm{WL}_N}(t) \\ \mathbf{0}_{1,3} & 1 & 0 & 0 & \cdots & 0 \\ \mathbf{0}_{1,3} & 0 & 1 & 0 & \cdots & 0 \\ \mathbf{0}_{1,3} & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{1,3} & 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

or a combination of contact and landmark positions. In this way, it is possible to develop an observer that contains no unobservable states. Of course, like EKF-SLAM, the filter can become too computationally expensive to run in real-time if the number of landmarks grows too large.

Tables 2 and 3 give a summary of the left/right world-centric and robo-centric InEKF equations assuming a single contact and landmark position. The linearized observation matrix and observation type for several different sensors are also provided. In these tables, $\mathbf{l}_t$ is shorthand for the true landmark position, and $\mathbf{m}$ denotes the true magnetic field vector. Using these tables, it is clear to see the relation between the left/right invariant error dynamics and the world/robo-centric formulations. All of these dynamics and observations are supported in an open-source C++ Library released alongside this article, available at: https://github.com/RossHartley/invariant-ekf.

13. Conclusion and future Work

Using recent results on a group-invariant form of the EKF, this article has derived an observer for a contact-aided INS for a 3D legged robot. Contact and IMU sensors are available on all modern bipedal robots; therefore, the developed system has the potential to become an essential part of such platforms because an observer with a large basin of attraction can improve the reliability of perception and control algorithms. We also included IMU biases in the state estimator and showed that, while some of the theoretical guarantees are lost, in real experiments, the proposed system has better convergence performance than that of a commonly used quaternion-based EKF. Although the latter is a discrete EKF on a Lie group, it does not exploit symmetries present in the system dynamics and observation models, namely, invariance of the estimation error under a group action. In addition to the original right-invariant form of the filter, the left-invariant dynamics have been discussed as well as a robo-centric version of the filter.

A series of experiments have been conducted while running the filter on a Cassie-series biped robot. The accuracy of the invariant-EKF has been demonstrated through a motion capture comparison, whereas long-term odometry was compared by overlaying the trajectory on Google Earth imagery. In addition to improving signals used for feedback control, the pose estimate from this filter can be used alongside a vision sensor to build maps of the environment. This was demonstrated on Cassie by building LiDAR-based point cloud maps while walking.

Future work includes developing an invariant smoother based on this filter. This would utilize the framework developed by Chauchat et al. (2018) to potentially improve IMU preintegration (Eckenhoff et al., 2018; Forster et al., 2017; Lupton and Sukkarieh, 2012) as well as contact preintegration (Hartley et al., 2018a) to perform SLAM. One interesting extension would be online estimation of kinematic parameters, which may help remove biases in the forward kinematic measurements. Another possibility for improvement of legged robot odometry is the detection of mode changes such as standing, flat-ground walking, and turning. During these modes, additional constraints may be inferred, which could improve state estimation (Brossard

et al., 2019). Finally, additional experiments need to be conducted to explore the potential for visual–inertial–contact odometry using an InEKF as well as incorporating prior terrain information into the filter.

Acknowledgments

The authors would like to thank Yukai Gong for the development of the feedback controller utilized in the Cassie experiments as well as Bruce Huang, Ray Zhang, Lu Gan, Zhenyu Gan, Omar Harib, Eva Mungai, and Grant Gibson for their help in collecting experimental data.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: Funding for RH and MG was given by the Toyota Research Institute (TRI), partly under award number N021515. However, this article solely reflects the opinions and conclusions of its authors and not TRI or any other Toyota entity. Funding for JG was in part provided by TRI and in part by the NSF (Award No. 1808051).

Notes

1. We use the InEKF acronym to distinguish from an iterated-EKF (IEKF).
2. The quaternion-based formulation of EKFs is also sometimes called "multiplicative filtering" (MEKF) due to the orientation correction being done through quaternion multiplication (Markley, 2004).
3. The set of quaternions, along with quaternion multiplication, actually forms a Lie group.
4. With input noise, Theorem 2 no longer holds, and the linearization is only approximate.
5. The negative sign on body velocity appears when inverting the world-centric state: $-{}_{\mathrm{B}}\mathbf{v}_{\mathrm{B}} = -\mathbf{R}_{\mathrm{WB}}^{\mathsf{T}}\,{}_{\mathrm{W}}\mathbf{v}_{\mathrm{B}}$. This sign is removed on the position vectors by swapping the start and end points, ${}_{\mathrm{B}}\mathbf{p}_{\mathrm{BW}} = -\mathbf{R}_{\mathrm{WB}}^{\mathsf{T}}\,{}_{\mathrm{B}}\mathbf{p}_{\mathrm{WB}}$.
6. If definition of bias error is negated, even these terms would remain the same.

ORCID iDs

Ross Hartley https://orcid.org/0000-0002-4605-4333

Maani Ghaffari https://orcid.org/0000-0002-4734-4295

References

Aghannan N and Rouchon P (2002) On invariant asymptotic observers. In: *IEEE Conference on Decision and Control*, Vol. 2. IEEE, pp. 1479–1484.

Anderson BD and Moore JB (1979) *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall.

Bar-Shalom Y, Li XR and Kirubarajan T (2001) *Estimation With Applications to Tracking and Navigation: Theory Algorithms and Software*. New York: John Wiley & Sons.

Barczyk M and Lynch AF (2011) Invariant extended Kalman filter design for a magnetometer-plus-GPS aided inertial navigation system. In: *IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*. IEEE, pp. 5389–5394.

Barczyk M and Lynch AF (2013) Invariant observer design for a helicopter UAV aided inertial navigation system. *IEEE Transactions on Control Systems Technology* 21(3): 791–806.

Barfoot TD (2017) *State Estimation for Robotics*. Cambridge: Cambridge University Press.

Barfoot TD and Furgale PT (2014) Associating uncertainty with three-dimensional poses for use in estimation problems. *IEEE Transactions on Robotics* 30(3): 679–693.

Barrau A (2015) *Non-linear state error based extended Kalman filters with applications to navigation*. PhD Thesis, Mines Paristech.

Barrau A and Bonnabel S (2015) An EKF-SLAM algorithm with consistency properties. *arXiv preprint arXiv:1510.06263*.

Barrau A and Bonnabel S (2017) The invariant extended Kalman filter as a stable observer. *IEEE Transactions on Automatic Control* 62(4): 1797–1812.

Barrau A and Bonnabel S (2018) Invariant Kalman filtering. *Annual Review of Control, Robotics, and Autonomous Systems* 1: 237–257.

Barrau A and Bonnabel S (2019) Extended Kalman filtering with nonlinear equality constraints: A geometric approach. *IEEE Transactions on Automatic Control*, in press.

Barshan B and Durrant-Whyte HF (1995) Inertial navigation systems for mobile robots. *IEEE Transactions on Robotics and Automation* 11(3): 328–342.

Benallegue M and Lamiraux F (2015) Estimation and stabilization of humanoid flexibility deformation using only inertial measurement units and contact information. *International Journal of Humanoid Robotics* 12(3): 1550025.

Bloesch M, Gehring C, Fankhauser P, Hutter M, Hoepflinger MA and Siegwart R (2013) State estimation for legged robots on unstable and slippery terrain. In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 6058–6064.

Bloesch M, Hutter M, Hoepflinger MA, et al. (2012) State estimation for legged robots: Consistent fusion of leg kinematics and IMU. In: *Robotics: Science and Systems Conference (RSS 2012)*.

Bloesch MA (2017) *State Estimation for Legged Robots—Kinematics, Inertial Sensing, and Computer Vision*. PhD Thesis, ETH Zürich. Available at: https://www.research-collection.ethz.ch/handle/20.500.11850/129873

Bonnabel S (2007) Left-invariant extended Kalman filter and attitude estimation. In: *IEEE Conference on Decision and Control*. IEEE, pp. 1027–1032.

Bonnabel S (2012) Symmetries in observer design: Review of some recent results and applications to EKF-based SLAM. In: *Robot Motion and Control 2011*. Springer, pp. 3–15.

Bonnabel S, Martin P and Rouchon P (2009) Non-linear symmetry-preserving observers on Lie groups. *IEEE Transactions on Automatic Control* 54(7): 1709–1713.

Bonnable S, Martin P and Salaün E (2009) 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, 2009 held jointly with the 2009 28th Chinese Control Conference (CDC/CCC 2009)*. IEEE, pp. 1297–1304.

Bourmaud G, Mégret R, Arnaudon M and Giremus A (2015) Continuous-discrete extended Kalman filter on matrix Lie groups using concentrated Gaussian distributions. *Journal of Mathematical Imaging and Vision* 51(1): 209–228.

Bourmaud G, Mégret R, Giremus A and Berthoumieu Y (2013) Discrete extended Kalman filter on Lie groups. In: *2013 Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*. IEEE, pp. 1–5.

Brossard M, Barrau A and Bonnabel S (2019) RINS-W: Robust inertial navigation system on wheels. *arXiv preprint arXiv:1903.02210*.

Chauchat P, Barrau A and Bonnabel S (2017) Kalman filtering with a class of geometric state equality constraints. In: *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, pp. 2581–2586.

Chauchat P, Barrau A and Bonnabel S (2018) Invariant smoothing on lie groups. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 1703–1710.

Chirikjian GS (2011) *Stochastic Models, Information Theory, and Lie Groups, Volume 2: Analytic Methods and Modern Applications*. Springer Science & Business Media.

Chitta S, Vemaza P, Geykhman R and Lee DD (2007) Proprioceptive localization for a quadrupedal robot on known terrain. In: *2007 IEEE International Conference on Robotics and Automation*. IEEE, pp. 4582–4587.

Clifford C (2018) This start-up is building a humanoid robot that could soon be delivering packages to your door. Available at: https://www.cnbc.com/2018/05/01/agility-robotics-building-two-legged-cassie-robot-and-delivery-robots.html.

Cobano JA, Estremera J and De Santos PG (2008) Location of legged robots in outdoor environments. *Robotics and Autonomous Systems* 56(9): 751–761.

Da X, Harib O, Hartley R, Griffin B and Grizzle JW (2016) From 2D design of underactuated bipedal gaits to 3D implementation: Walking with speed tracking. *IEEE Access* 4: 3469–3478.

Da X, Hartley R and Grizzle JW (2017) Supervised learning for stabilizing underactuated bipedal robot locomotion, with outdoor experiments on the wave field. In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, pp. 3476–3483.

Diebel J (2006) Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Matrix* 58(15–16): 1–35.

Eckenhoff K, Geneva P and Huang G (2018) Closed-form preintegration methods for graph-based visual–inertial navigation. *The International Journal of Robotics Research* 38(5): 563–586.

Eljaik J, Kuppuswamy N and Nori F (2015) Multimodal sensor fusion for foot state estimation in bipedal robots using the extended Kalman filter. In: *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 2698–2704.

Fallon MF, Antone M, Roy N and Teller S (2014) Drift-free humanoid state estimation fusing kinematic, inertial and lidar sensing. In: *IEEE-RAS International Conference on Humanoid Robots*. IEEE, pp. 112–119.

Fankhauser P, Bloesch M, Gehring C, Hutter M and Siegwart R (2014) Robot-centric elevation mapping with uncertainty estimates. In: *International Conference on Climbing and Walking Robots*, pp. 433–440.

Forster C, Carlone L, Dellaert F and Scaramuzza D (2017) On-manifold preintegration for real-time visual–inertial odometry. *IEEE Transactions on Robotics* 33(1): 1–21.

Gebre-Egziabher D, Hayward RC and Powell JD (2004) Design of multi-sensor attitude determination systems. *IEEE Transactions on aerospace and electronic systems* 40(2): 627–649.

Gong Y, Hartley R, Da X, et al. (2019) Feedback control of a cassie bipedal robot: Walking, standing, and riding a segway. In: *2019 American Control Conference (ACC)*. IEEE, pp. 4559–4566.

Grewal MS (2011) Kalman filtering. In: *International Encyclopedia of Statistical Science*. Berlin: Springer, pp. 705–708.

Grizzle J and Song Y (1995) The extended Kalman filter as a local asymptotic observer for nonlinear discrete-time systems. *Journal of Mathematical Systems, Estimation and Control* 5(1): 59–78.

Hall B (2015) *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction* (*Graduate Texts in Mathematics*, Vol. 222). New York: Springer.

Hartley R, Da X and Grizzle JW (2017) Stabilization of 3D underactuated biped robots: Using posture adjustment and gait libraries to reject velocity disturbances. In: *2017 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, pp. 1262–1269.

Hartley R, Jadidi MG, Gan L, Huang JK, Grizzle JW and Eustice RM (2018a) Hybrid contact preintegration for visual-inertial-contact state estimation using factor graphs. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 3783–3790.

Hartley R, Jadidi MG, Grizzle JW and Eustice RM (2018b) Contact-aided invariant extended Kalman filtering for legged robot state estimation. In: *Proceedings of Robotics: Science and Systems*.

Hodgins JK (1989) *Legged robots on rough terrain: Experiments in adjusting step length*. PhD Thesis, Carnegie Mellon University, Pittsburgh, PA.

Huai Z and Huang G (2018) Robocentric visual–inertial odometry. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 6319–6326.

Huang GP, Mourikis AI and Roumeliotis SI (2010) Observability-based rules for designing consistent EKF SLAM estimators. *The International Journal of Robotics Research* 29(5): 502–528.

Hwangbo J, Bellicoso CD, Fankhauser P and Huttery M (2016) Probabilistic foot contact estimation by fusing information from dynamics and differential/forward kinematics. In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 3872–3878.

Kalman RE (1960) A new approach to linear filtering and prediction problems. *Journal of Basic Engineering* 82(1): 35–45.

Krener AJ (2003) The convergence of the extended Kalman filter. In: *Directions in Mathematical Systems Theory and Optimization*. Berlin: Springer, pp. 173–182.

Kuindersma S, Deits R, Fallon M, et al. (2016) Optimization-based locomotion planning, estimation, and control design for the Atlas humanoid robot. *Autonomous Robots* 40(3): 429–455.

Lefferts EJ, Markley FL and Shuster MD (1982) Kalman filtering for spacecraft attitude estimation. *Journal of Guidance, Control, and Dynamics* 5(5): 417–429.

Lin PC, Komsuoglu H and Koditschek DE (2005) A leg configuration measurement system for full-body pose estimates in a hexapod robot. *IEEE Transactions on robotics* 21(3): 411–422.

Lin PC, Komsuoglu H and Koditschek DE (2006) Sensor data fusion for body state estimation in a hexapod robot with dynamical gaits. *IEEE Transactions on Robotics* 22(5): 932–943.

Long AW, Wolfe KC, Mashner MJ and Chirikjian GS (2013) The banana distribution is Gaussian: A localization study with exponential coordinates. *Robotics: Science and Systems VIII* 265.

Lupton T and Sukkarieh S (2012) Visual–inertial-aided navigation for high-dynamic motion in built environments without initial conditions. *IEEE Transactions on Robotics* 28(1): 61–76.

Madyastha V, Ravindra V, Mallikarjunan S and Goyal A (2011) Extended Kalman filter vs. error state Kalman filter for aircraft attitude estimation. In: *AIAA Guidance, Navigation, and Control Conference*, p. 6615.

Markley FL (2004) Multiplicative vs. additive filtering for spacecraft attitude determination. In: *Proceedings of the 6th Cranfield Conference on Dynamics and Control of Systems and Structures in Space*, Riomaggiore, Italy, 18—22 July 2004, pp. 467-–474.

Maybeck PS (1982) *Stochastic Models, Estimation, and Control*, Vol. 3. New York: Academic Press.

Merhav SJ and Koifman M (1991) Autonomously aided strapdown attitude reference system. *Journal of Guidance, Control, and Dynamics* 14(6): 1164–1172.

Murrell J (1978) Precision attitude determination for multimission spacecraft. In: *Guidance and Control Conference*, p. 1248.

Nobili S, Camurri M, Barasuol V, et al. (2017) Heterogeneous sensor fusion for accurate state estimation of dynamic legged robots. In: *Robotics: Science and Systems*.

Qin T, Li P and Shen S (2018) VINS-MONO: A robust and versatile monocular visual–inertial state estimator. *IEEE Transactions on Robotics* 34(4): 1004–1020.

Roston GP and Krotkov EP (1991) Dead reckoning navigation for walking robots. Technical report, Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA.

Rotella N, Bloesch M, Righetti L and Schaal S (2014) State estimation for a humanoid robot. In: *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 952–958.

Roumeliotis SI, Sukhatme GS and Bekey GA (1999) Circumventing dynamic modeling: Evaluation of the error-state Kalman filter applied to mobile robot localization. In: *Proceedings 1999 IEEE International Conference on Robotics and Automation, 1999*, Vol. 2. IEEE, pp. 1656–1663.

Sebesta KD and Boizot N (2014) A real-time adaptive high-gain EKF, applied to a quadcopter inertial navigation system. *IEEE Transactions on Industrial Electronics* 61(1): 495–503.

Setoodeh P, Khayatian A and Frajah E (2004) Attitude estimation by separate-bias Kalman filter-based data fusion. *The Journal of Navigation* 57(2): 261–273.

Singh SP, Csonka PJ and Waldron KJ (2006) Optical flow aided motion estimation for legged locomotion. In: *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 1738–1743.

Smith D and Singh S (2006) Approaches to multisensor data fusion in target tracking: A survey. *IEEE Transactions on Knowledge and Data Engineering* 18(12): 1696–1710.

Smith R, Self M and Cheeseman P (1990) Estimating uncertain spatial relationships in robotics. In: *Autonomous Robot Vehicles*. New York: Springer, pp. 167–193.

Sola J (2017) Quaternion kinematics for the error-state Kalman filter. *arXiv preprint arXiv:1711.02508*.

Song Y and Grizzle JW (1992) The extended Kalman filter as a local asymptotic observer for nonlinear discrete-time systems. In: *American Control Conference*. IEEE, pp. 3365–3369.

Thrun S, Burgard W and Fox D (2005) *Probabilistic Robotics*. Cambridge, MA: MIT Press.

Titterton D, Weston JL and Weston J (2004) *Strapdown Inertial Navigation Technology*, Vol. 17. IET.

Trawny N and Roumeliotis SI (2005) Indirect Kalman filter for 3D attitude estimation. Technical Report 2005-002, Department of Computer Science and Engineering University of Minnesota, Minneapolis, MN. Available at: http://mars.cs.umn.edu/tr/reports/Trawny05b.pdf.

Wang Y and Chirikjian GS (2006) Error propagation on the Euclidean group with applications to manipulator kinematics. *IEEE Transactions on Robotics* 22(4): 591–602.

Wang Y and Chirikjian GS (2008) Nonparametric second-order theory of error propagation on motion groups. *The International Journal of Robotics Research* 27(11–12): 1258–1273.

Woodman OJ (2007) An introduction to inertial navigation. Technical Report UCAM-CL-TR-696, University of Cambridge Computer Laboratory. Available at: https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-696.pdf.

Wu K, Zhang T, Su D, Huang S and Dissanayake G (2017) An invariant-EKF VINS algorithm for improving consistency. In: *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems.*

Xinjilefu X, Feng S, Huang W and Atkeson CG (2014) Decoupled state estimation for humanoids using full-body dynamics. *Proceedings IEEE International Conference on Robotics and Automation*.

Yang Z and Shen S (2017) Monocular visual–inertial state estimation with online initialization and camera–IMU extrinsic calibration. *IEEE Transactions on Automation Science and Engineering* 14(1): 39–51.

Zhang T, Wu K, Song J, Huang S and Dissanayake G (2017) Convergence and consistency analysis for a 3-D invariant-EKF SLAM. *IEEE Robotics and Automation Letters* 2(2): 733–740.

Appendix A. Discretization of filter equations

In the preceding sections, the filters' equations were in continuous time. However, in order to implement these filters using software and physical sensors, these equations need to be discretized. For our implementation, we assumed a zero-order hold on the inertial measurements, and performed analytical integration (Eckenhoff et al., 2018; Huai and Huang, 2018). In particular, analytical integration was important for the resulting error dynamics to satisfy Theorem 2.

A.1. World-centric dynamics

This appendix demonstrates how to derive the deterministic, discrete-time (world-centric) dynamics through analytical integration of the continuous state (24) and bias (25) dynamics. The contact, landmark, and bias dynamics are simply Gaussian noise. Therefore, the discrete, deterministic dynamics are simply

$$\bar{\mathbf{d}}_{t_{k+1}} = \bar{\mathbf{d}}_{t_k}, \quad \bar{\mathbf{b}}^g_{t_{k+1}} = \bar{\mathbf{b}}^g_{t_k}, \quad \bar{\mathbf{b}}^a_{t_{k+1}} = \bar{\mathbf{b}}^a_{t_k} \tag{47}$$

Assuming a zero-order hold on the incoming IMU measurements between times t_k and t_{k+1}, the orientation can be updated using the exponential map of SO(3):

$$\begin{aligned}\bar{\mathbf{R}}_{t_{k+1}} &= \int_{t_k}^{t_{k+1}} \bar{\mathbf{R}}_{t_k} (\bar{\boldsymbol{\omega}}_t)_\times \, dt \\ &= \bar{\mathbf{R}}_{t_k} \exp\left(\bar{\boldsymbol{\omega}}_{t_k} \Delta t\right)\end{aligned}$$

where $\Delta t \triangleq t_{k+1} - t_k$. Integrating the velocity dynamics yields an equation that involves the integral of the exponential map:

$$\begin{aligned}\bar{\mathbf{v}}_{t_{k+1}} &= \bar{\mathbf{v}}_{t_k} + \int_{t_k}^{t_{k+1}} \bar{\mathbf{R}}_t \bar{\mathbf{a}}_t + \mathbf{g} \, dt \\ &= \bar{\mathbf{v}}_{t_k} + \mathbf{g}\Delta t + \bar{\mathbf{R}}_{t_k} \left(\int_{t_k}^{t_{k+1}} \exp\left(\bar{\boldsymbol{\omega}}_{t_k} t\right) \, dt \right) \bar{\mathbf{a}}_{t_k}\end{aligned}$$

Likewise, analytically solving for the discrete position dynamics involves computing the double integral:

$$\begin{aligned}\bar{\mathbf{p}}_{t_{k+1}} =& \bar{\mathbf{p}}_{t_k} + \bar{\mathbf{v}}_{t_k} \Delta t + \frac{1}{2}\mathbf{g}\Delta t^2 \\ &+ \bar{\mathbf{R}}_{t_k} \left(\int_{t_k}^{t_{k+1}} \int_{t_k}^{\tau} \exp\left(\bar{\boldsymbol{\omega}}_{t_k} t\right) \, dt \, d\tau \right) \bar{\mathbf{a}}_{t_k}\end{aligned}$$

To solve these integrals, it is useful to define an auxiliary function (Bloesch et al., 2012):

$$\boldsymbol{\Gamma}_m(\boldsymbol{\phi}) \triangleq \left(\sum_{n=0}^{\infty} \frac{1}{(n+m)!} (\boldsymbol{\phi})_\times^n \right) \tag{48}$$

which allows integrals to be easily expressed and computed using the Taylor series form of the SO(3) exponential map.

$$\begin{aligned}\int_{t_k}^{t_{k+1}} \exp\left(\boldsymbol{\omega} t\right) \, dt &= \int_{t_k}^{t_{k+1}} \boldsymbol{\Gamma}_0(\boldsymbol{\omega} t) \, dt \\ &= \left(\sum_{n=0}^{\infty} \frac{1}{(n+1)!} (\boldsymbol{\omega}\Delta t)_\times^n \right) \Delta t = \boldsymbol{\Gamma}_1(\boldsymbol{\omega}\Delta t)\, \Delta t \\ \int_{t_k}^{t_{k+1}} \int_{t_k}^{\tau} \exp\left(\boldsymbol{\omega} t\right) \, dt \, d\tau &= \int_{t_k}^{t_{k+1}} \boldsymbol{\Gamma}_1(\boldsymbol{\omega} t) \, t \, dt \\ &= \left(\sum_{n=0}^{\infty} \frac{1}{(n+2)!} (\boldsymbol{\omega}\Delta t)_\times^n \right) \Delta t^2 = \boldsymbol{\Gamma}_2(\boldsymbol{\omega}\Delta t)\, \Delta t^2\end{aligned}$$

Closed-form expressions also exist, allowing fast and easy computation of these quantities (Sola, 2017):

$$\begin{aligned}\boldsymbol{\Gamma}_0(\boldsymbol{\phi}) &= \mathbf{I}_3 + \frac{\sin(||\boldsymbol{\phi}||)}{||\boldsymbol{\phi}||} (\boldsymbol{\phi})_\times + \frac{1 - \cos(||\boldsymbol{\phi}||)}{||\boldsymbol{\phi}||^2} (\boldsymbol{\phi})_\times^2 \\ \boldsymbol{\Gamma}_1(\boldsymbol{\phi}) &= \mathbf{I}_3 + \frac{1 - \cos(||\boldsymbol{\phi}||)}{||\boldsymbol{\phi}||^2} (\boldsymbol{\phi})_\times + \frac{||\boldsymbol{\phi}|| - \sin(||\boldsymbol{\phi}||)}{||\boldsymbol{\phi}||^3} (\boldsymbol{\phi})_\times^2 \\ \boldsymbol{\Gamma}_2(\boldsymbol{\phi}) &= \frac{1}{2}\mathbf{I}_3 + \frac{||\boldsymbol{\phi}|| - \sin(||\boldsymbol{\phi}||)}{||\boldsymbol{\phi}||^3} (\boldsymbol{\phi})_\times \\ &\quad + \frac{||\boldsymbol{\phi}||^2 + 2\cos(||\boldsymbol{\phi}||) - 2}{2||\boldsymbol{\phi}||^4} (\boldsymbol{\phi})_\times^2\end{aligned} \tag{49}$$

Remark 7. The map $\boldsymbol{\Gamma}_0(\boldsymbol{\phi})$ is simply the exponential map of SO(3), whereas $\boldsymbol{\Gamma}_1(\boldsymbol{\phi})$ is also known as the left Jacobian of SO(3) (Barfoot, 2017; Chirikjian, 2011).

Using these expressions, we can write the discrete dynamics for the rotation, velocity, and positions states as

$$\begin{aligned}\bar{\mathbf{R}}_{k+1} &= \bar{\mathbf{R}}_k \, \boldsymbol{\Gamma}_0(\bar{\boldsymbol{\omega}}_k \Delta t) \\ \bar{\mathbf{v}}_{k+1} &= \bar{\mathbf{v}}_k + \bar{\mathbf{R}}_k \boldsymbol{\Gamma}_1(\bar{\boldsymbol{\omega}}_k \Delta t)\, \bar{\mathbf{a}}_k \Delta t + \mathbf{g}\Delta t \\ \bar{\mathbf{p}}_{k+1} &= \bar{\mathbf{p}}_k + \bar{\mathbf{v}}_k \Delta t + \bar{\mathbf{R}}_k \boldsymbol{\Gamma}_2(\bar{\boldsymbol{\omega}}_k \Delta t)\, \bar{\mathbf{a}}_k \Delta t^2 + \frac{1}{2}\mathbf{g}\Delta t^2\end{aligned} \tag{50}$$

where the t is dropped from the subscript for readability. These discrete dynamics are an exact integration of the continuous-time system under the assumption that the IMU measurements are constant over Δt.

A.2. Covariance propagation

In order to propagate the covariance, a continuous-time Riccati equation needs to be solved:

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{P}_t = \mathbf{A}_t \mathbf{P}_t + \mathbf{P}_t \mathbf{A}_t^\mathsf{T} + \bar{\mathbf{Q}}_t$$

The analytical solution to the differential equation above is given by (Maybeck, 1982):

$$\mathbf{P}_{t_{k+1}} = \boldsymbol{\Phi}(t_{k+1}, t_k)\, \mathbf{P}_{t_k} \boldsymbol{\Phi}(t_{k+1}, t_k)^\mathsf{T} + \bar{\mathbf{Q}}_d \tag{51}$$

where the discrete noise covariance matrix is computed by

$$\bar{\mathbf{Q}}_d = \int_{t_k}^{t_{k+1}} \boldsymbol{\Phi}(t_{k+1}, t)\, \bar{\mathbf{Q}}_t \boldsymbol{\Phi}(t_{k+1}, t)^\mathsf{T} \, dt \tag{52}$$

and the state transition matrix, $\boldsymbol{\Phi}(t_{k+1}, t_k)$, satisfies

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{\Phi}(t, t_k) = \mathbf{A}_t \boldsymbol{\Phi}(t, t_k) \quad \text{with } \boldsymbol{\Phi}(t_k, t_k) = \mathbf{I} \tag{53}$$

The (world-centric) left-invariant error dynamics matrix only depends on the IMU inputs and the estimated bias terms, see Table 2. As both are assumed to be constant between times t_k and t_{k+1}, the state transition matrix can be simply computed from the matrix exponential:

$$\boldsymbol{\Phi}^l(t_{k+1}, t_k) = \exp_m(\mathbf{A}_t^l \Delta t) \tag{54}$$

This state transition matrix also has an analytical solution of the form:

$$\boldsymbol{\Phi}^l(t_{k+1}, t_k) = \begin{bmatrix} \boldsymbol{\Phi}_{11}^l & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{\Phi}_{15}^l & \mathbf{0} \\ \boldsymbol{\Phi}_{21}^l & \boldsymbol{\Phi}_{22}^l & \mathbf{0} & \mathbf{0} & \boldsymbol{\Phi}_{25}^l & \boldsymbol{\Phi}_{26}^l \\ \boldsymbol{\Phi}_{31}^l & \boldsymbol{\Phi}_{32}^l & \boldsymbol{\Phi}_{33}^l & \mathbf{0} & \boldsymbol{\Phi}_{35}^l & \boldsymbol{\Phi}_{36}^l \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{\Phi}_{44}^l & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{55}$$

where the individual terms are

$$
\begin{aligned}
\mathbf{\Phi}^l_{11} &= \mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\\
\mathbf{\Phi}^l_{21} &= -\mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,(\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\bar{\mathbf{a}}_k)_\times\,\Delta t\\
\mathbf{\Phi}^l_{31} &= -\mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,(\mathbf{\Gamma}_2(\bar{\boldsymbol{\omega}}_k\Delta t)\,\bar{\mathbf{a}}_k)_\times\,\Delta t^2\\
\mathbf{\Phi}^l_{22} &= \mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\\
\mathbf{\Phi}^l_{32} &= \mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t\\
\mathbf{\Phi}^l_{33} &= \mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\\
\mathbf{\Phi}^l_{44} &= \mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\\
\mathbf{\Phi}^l_{15} &= -\mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t\\
\mathbf{\Phi}^l_{25} &= \mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,\mathbf{\Psi}_1\\
\mathbf{\Phi}^l_{35} &= \mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,\mathbf{\Psi}_2\\
\mathbf{\Phi}^l_{26} &= -\mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t\\
\mathbf{\Phi}^l_{36} &= -\mathbf{\Gamma}_0^\mathsf{T}(\bar{\boldsymbol{\omega}}_k\Delta t)\,\mathbf{\Gamma}_2(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t^2
\end{aligned}
$$

The matrices $\mathbf{\Psi}_1$ and $\mathbf{\Psi}_2$ involve computing the solution to a more complicated integral. However, these integrals still have analytical solutions which can be expressed easier after defining $\phi \triangleq \|\bar{\boldsymbol{\omega}}_k\|$ and $\theta \triangleq \phi\Delta t$:

$$
\begin{aligned}
\mathbf{\Psi}_1 &\triangleq \int_{t_k}^{t_{k+1}} (\mathbf{\Gamma}_0(\bar{\boldsymbol{\omega}}_k t)\,\bar{\mathbf{a}}_k)_\times\,\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k t)\; t\; dt\\
&= (\bar{\mathbf{a}}_k)_\times\,\mathbf{\Gamma}_2(-\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t^2\\
&\quad\Big(\frac{\sin(\theta)-\theta\cos(\theta)}{\phi^3}(\bar{\boldsymbol{\omega}}_k)_\times(\bar{\mathbf{a}}_k)_\times\\
&\quad+\frac{\cos(2\theta)-4\cos(\theta)+3}{4\phi^4}(\bar{\boldsymbol{\omega}}_k)_\times(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times\\
&\quad+\frac{4\sin(\theta)+\sin(2\theta)-4\theta\cos(\theta)-2\theta}{4\phi^5}(\bar{\boldsymbol{\omega}}_k)_\times(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times^2\\
&\quad+\frac{\theta^2-2\theta\sin(\theta)-2\cos(\theta)+2}{2\phi^4}(\bar{\boldsymbol{\omega}}_k)_\times^2(\bar{\mathbf{a}}_k)_\times\\
&\quad+\frac{6\theta-8\sin(\theta)+\sin(2\theta)}{4\phi^5}(\bar{\boldsymbol{\omega}}_k)_\times^2(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times\\
&\quad+\frac{2\theta^2-4\theta\sin(\theta)-\cos(2\theta)+1}{4\phi^6}(\bar{\boldsymbol{\omega}}_k)_\times^2(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times^2\Big)
\end{aligned}
\tag{56}
$$

$$
\begin{aligned}
\mathbf{\Psi}_2 &\triangleq \int_{t_k}^{t_{k+1}} \mathbf{\Gamma}_0(\bar{\boldsymbol{\omega}}_k t)\,\mathbf{\Phi}^l_{25}(t,t_k)\; dt\\
&= (\bar{\mathbf{a}}_k)_\times\,\mathbf{\Gamma}_3(-\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t^3\\
&\Big(\frac{\theta\sin(\theta)+2\cos(\theta)-2}{\phi^4}(\bar{\boldsymbol{\omega}}_k)_\times(\bar{\mathbf{a}}_k)_\times\\
&+\frac{6\theta-8\sin(\theta)+sin(2\theta)}{8\phi^5}(\bar{\boldsymbol{\omega}}_k)_\times(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times\\
&+\frac{2\theta^2+8\theta\sin(\theta)+16\cos(\theta)+\cos(2\theta)-17}{8\phi^6}(\bar{\boldsymbol{\omega}}_k)_\times(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times^2\\
&+\frac{\theta^3+6\theta-12\sin(\theta)+6\theta\cos(\theta)}{6\phi^5}(\bar{\boldsymbol{\omega}}_k)_\times^2(\bar{\mathbf{a}}_k)_\times\\
&+\frac{6\theta^2+16\cos(\theta)-\cos(2\theta)-15}{8\phi^6}(\bar{\boldsymbol{\omega}}_k)_\times^2(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times\\
&+\frac{4\theta^3+6\theta-24\sin(\theta)-3\sin(2\theta)+24\theta\cos(\theta)}{24\phi^7}(\bar{\boldsymbol{\omega}}_k)_\times^2(\bar{\mathbf{a}}_k)_\times(\bar{\boldsymbol{\omega}}_k)_\times^2\Big)
\end{aligned}
\tag{57}
$$

The (world-centric) right-invariant error dynamics matrix depends on the state estimates, $\bar{\mathbf{R}}_t$, $\bar{\mathbf{v}}_t$, and $\bar{\mathbf{p}}_t$, which will change between times t_k and t_{k+1}, see Table 2. Therefore, the state transition matrix will not simply be the matrix exponential, as in the left-invariant case. Solving (53) yields a state transition matrix of the form:

$$
\mathbf{\Phi}^r(t_{k+1},t_k) = \begin{bmatrix}
\mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{\Phi}^r_{15} & \mathbf{0}\\
\mathbf{\Phi}^r_{21} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{\Phi}^r_{25} & \mathbf{\Phi}^r_{26}\\
\mathbf{\Phi}^r_{31} & \mathbf{\Phi}^r_{32} & \mathbf{I} & \mathbf{0} & \mathbf{\Phi}^r_{35} & \mathbf{\Phi}^r_{36}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{\Phi}^r_{45} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0}\\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}
\end{bmatrix}
\tag{58}
$$

where the individual terms can be analytically computed as

$$
\begin{aligned}
\mathbf{\Phi}^r_{21} &= (\mathbf{g})_\times\,\Delta t\\
\mathbf{\Phi}^r_{31} &= \frac{1}{2}(\mathbf{g})_\times\,\Delta t^2\\
\mathbf{\Phi}^r_{32} &= \mathbf{I}\Delta t\\
\mathbf{\Phi}^r_{15} &= -\bar{\mathbf{R}}_k\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t\\
\mathbf{\Phi}^r_{25} &= -(\bar{\mathbf{v}}_{k+1})_\times\,\bar{\mathbf{R}}_k\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t + \bar{\mathbf{R}}_k\mathbf{\Psi}_1\\
\mathbf{\Phi}^r_{35} &= -(\bar{\mathbf{p}}_{k+1})_\times\,\bar{\mathbf{R}}_k\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t + \bar{\mathbf{R}}_k\mathbf{\Psi}_2\\
\mathbf{\Phi}^r_{45} &= -(\bar{\mathbf{d}}_{k+1})_\times\,\bar{\mathbf{R}}_k\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t\\
\mathbf{\Phi}^r_{26} &= -\bar{\mathbf{R}}_k\mathbf{\Gamma}_1(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t\\
\mathbf{\Phi}^r_{36} &= -\bar{\mathbf{R}}_k\mathbf{\Gamma}_2(\bar{\boldsymbol{\omega}}_k\Delta t)\,\Delta t^2
\end{aligned}
$$

As the left-/right-invariant errors are related through the adjoint, the two state transition and discrete noise matrices also satisfy the following relations (Barrau, 2015).

$$
\begin{aligned}
\mathbf{\Phi}^r &= \mathrm{Ad}_{\mathbf{X}_{k+1}}\mathbf{\Phi}^l\mathrm{Ad}_{\bar{\mathbf{X}}_k^{-1}}\\
\bar{\mathbf{Q}}^r_d &= \mathrm{Ad}_{\mathbf{X}_{k+1}}\bar{\mathbf{Q}}^l_d\mathrm{Ad}^\mathsf{T}_{\bar{\mathbf{X}}_{k+1}}
\end{aligned}
\tag{59}
$$

Therefore, the right-invariant state transition matrix can alternatively be computed using

$$
\mathbf{\Phi}^r(t_{k+1},t_k) = \mathrm{Ad}_{\bar{\mathbf{X}}_{k+1}}\,\exp_m(\mathbf{A}^l_t\Delta t)\,\mathrm{Ad}_{\bar{\mathbf{X}}_k^{-1}}
\tag{60}
$$

which can simplify implementation because many software libraries already contain efficient methods to compute the matrix exponential.

Similar to the state transition matrices, the discrete noise covariance matrix (52) also has an analytical solution. In practice, this matrix is often approximated as

$$
\bar{\mathbf{Q}}_d \approx \mathbf{\Phi}\bar{\mathbf{Q}}_k\mathbf{\Phi}^\mathsf{T}\Delta t
\tag{61}
$$

This approximated discrete noise matrix was used for all results in this article.

Appendix B. Useful Lie group expressions

The matrix Lie group $\mathrm{SE}_K(3)$ is known as the group of K direct isometries (Barrau, 2015). This group is composed of a rotation matrix, $\mathbf{R} \in \mathrm{SO}(3)$, and K vectors in $\mathbb{R}^3$,

$\mathbf{p}_1, \ldots, \mathbf{p}_K$. Let $\mathbf{X}$ be an element of $\mathrm{SE}_K(3)$, which can be written as a $(3+K)\times(3+K)$ matrix:

$$\mathbf{X} \triangleq \begin{bmatrix} \mathbf{R} & \mathbf{p}_1 & \cdots & \mathbf{p}_K \\ \mathbf{0}_{3,3} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{3,3} & 0 & \cdots & 1 \end{bmatrix} \tag{62}$$

The group action is matrix multiplication. The adjoint is a linear map that can be used to move vectors between the tangent spaces of two group elements. The matrix representation of the adjoint of $\mathrm{SE}_K(3)$ is given by

$$\mathrm{Ad}_{\mathbf{X}} = \begin{bmatrix} \mathbf{R} & \mathbf{0} & \cdots & \mathbf{0} \\ (\mathbf{p}_1)_\times \mathbf{R} & \mathbf{R} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ (\mathbf{p}_K)_\times \mathbf{R} & \mathbf{0} & \cdots & \mathbf{R} \end{bmatrix} \tag{63}$$

The matrix representation of a vector, $\boldsymbol{\xi} \triangleq \mathrm{vec}(\boldsymbol{\phi}, \boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_K) \in \mathbb{R}^{3+3K}$, in the Lie algebra can obtained using the "hat" operator:

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} (\boldsymbol{\phi})_\times & \boldsymbol{\xi}_1 & \cdots & \boldsymbol{\xi}_K \\ \mathbf{0}_{3,3} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{3,3} & 0 & \cdots & 0 \end{bmatrix} \in \mathfrak{se}_K(3) \tag{64}$$

where $(\boldsymbol{\phi})_\times$ denotes the skew-symmetric matrix of a vector $\boldsymbol{\phi} = \mathrm{vec}(\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \boldsymbol{\phi}_3) \in \mathbb{R}^3$:

$$(\boldsymbol{\phi})_\times \triangleq \begin{bmatrix} 0 & -\boldsymbol{\phi}_3 & \boldsymbol{\phi}_2 \\ \boldsymbol{\phi}_3 & 0 & -\boldsymbol{\phi}_1 \\ -\boldsymbol{\phi}_2 & \boldsymbol{\phi}_1 & 0 \end{bmatrix} \in \mathfrak{so}(3) \tag{65}$$

The same vector can be moved to the Lie group through the exponential map:

$$\exp(\boldsymbol{\xi}) = \begin{bmatrix} \boldsymbol{\Gamma}_0(\boldsymbol{\phi}) & \boldsymbol{\Gamma}_1(\boldsymbol{\phi})\boldsymbol{\xi}_1 & \cdots & \boldsymbol{\Gamma}_1(\boldsymbol{\phi})\boldsymbol{\xi}_K \\ \mathbf{0}_{3,3} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{3,3} & 0 & \cdots & 1 \end{bmatrix} \tag{66}$$

where $\boldsymbol{\Gamma}_0(\boldsymbol{\phi})$ is the exponential map of $\mathrm{SO}(3)$, and $\boldsymbol{\Gamma}_1(\boldsymbol{\phi})$ is the left Jacobian of $\mathrm{SO}(3)$.

Appendix C. Error-state conversions

Throughout this article, several versions of error states are used. These include the left-/right-invariant error (1), the QEKF error states (21), and the "Euclidean orientation error" used for plotting the covariance hull (33). It is often necessary to convert between these error states for plotting or when initializing the filters to provide fair comparisons. For example, if the QEKF and the InEKF are initialized with identical covariance matrices, the underlying distribution that they represent may be substantially different. This makes it impossible to show an accuracy comparison between the InEKF and QEKF with identical initial uncertainty. This appendix provides details on how to convert between these error states up to a first-order approximation.

When designing a QEKF, the orientation error can be defined in either the local or global frame (Sola, 2017). These errors are equivalent to the left- and right-invariant errors for $\mathrm{SO}(3)$. In this document, the orientation error in the QEKF was chosen to be the error defined in the local frame (left-invariant error). As the invariant errors are related through the group's adjoint, the exact relation between right-invariant and QEKF orientation errors is

$$\exp(\boldsymbol{\xi}_t^R) = \exp(\bar{\mathbf{R}}_t \delta\boldsymbol{\theta}_t) \tag{67}$$

When plotting the individual axes of the orientation error covariance hull, a "Euclidean orientation error" should be used. Let $\delta\boldsymbol{\phi}_t \triangleq \boldsymbol{\phi}_t - \bar{\boldsymbol{\phi}}_t$ be this Euclidean error where $\exp(\boldsymbol{\phi}) \triangleq \mathbf{R}$ is the exponential coordinate representation of a particular orientation. When the errors are small, a first-order approximation can be used to find a mapping between the right-invariant orientation error and this "Euclidean orientation error":

$$\begin{aligned} \exp(\boldsymbol{\xi}_t^R) = \bar{\mathbf{R}}_t \mathbf{R}_t^\mathsf{T} &= \exp(\bar{\boldsymbol{\phi}}_t)\exp(-\bar{\boldsymbol{\phi}}_t - \delta\boldsymbol{\phi}) \\ &\approx \exp(-\boldsymbol{\Gamma}_1(\bar{\boldsymbol{\phi}}_t)\,\delta\boldsymbol{\phi}) \end{aligned} \tag{68}$$

A similar first-order approximation can be used to find the relation between the right-invariant error and the QEKF velocity errors:

$$\begin{aligned} \boldsymbol{\eta}_t^v &= \bar{\mathbf{v}}_t - \bar{\mathbf{R}}_t \mathbf{R}_t^\mathsf{T} \mathbf{v}_t = \bar{\mathbf{v}}_t - \exp(\boldsymbol{\xi}_t^R)\,\mathbf{v}_t \\ &\approx \bar{\mathbf{v}}_t - \mathbf{v}_t - (\boldsymbol{\xi}_t^R)_\times \mathbf{v}_t = -\delta\mathbf{v}_t + (\mathbf{v}_t)_\times \boldsymbol{\xi}_t^R \\ \implies \boldsymbol{\xi}_t^v &\approx -\delta\mathbf{v}_t + (\bar{\mathbf{v}}_t)_\times \boldsymbol{\xi}_t^R \end{aligned} \tag{69}$$

The same process can be repeated for the position states:

$$\begin{aligned} \boldsymbol{\xi}_t^p &\approx -\delta\mathbf{p}_t + (\bar{\mathbf{p}}_t)_\times \boldsymbol{\xi}_t^R \\ \boldsymbol{\xi}_t^d &\approx -\delta\mathbf{d}_t + (\bar{\mathbf{d}}_t)_\times \boldsymbol{\xi}_t^R \end{aligned} \tag{70}$$