

Two-Stage Proprioceptive State Estimation with Stability Guarantee for Legged Robots

Jun Li, Yuhui Wan, Weihua Li, Jianfeng Wang and Chengxu Zhou

Abstract—To achieve dynamic legged locomotion, real-time acquisition of the robot floating base state is critical. However, current full-coupled state estimation solutions may suffer from convergence issues. To address this, a two-stage state estimator that uses proprioceptive sensors is proposed to estimate the floating base’s velocity and pose. The approach decomposes the floating-base state estimation problem into base orientation estimation and base velocity and position estimation, resulting in a two-stage multi-sensor fusion algorithm with stability guarantees. Additionally, a covariance inflation method is introduced to consider the influences of contact switching by adjusting noise covariances. The proposed state estimator has been successfully implemented and verified on a real quadrupedal robot, supporting dynamic motions such as multi-gaited locomotion and push recovery.

I. INTRODUCTION

Legged robots excel in highly unstructured environments thanks to their intermittent contact, which complicates locomotion control. A key challenge for dynamic legged locomotion is accurately estimating the robot’s floating base velocity and pose in real-time. Current state estimation algorithms for legged robots fall into two categories: filtering methods [1]–[9] and smoothing methods [10]–[16]. While many of these methods are popular, their stability—particularly during contact switching—receives limited attention.

Filtering methods predict the robot’s state with previous measurements and correct it with new ones. The first filtering method [1] for legged robots is the Extended Kalman Filter (EKF) based state estimator fusing Inertial Measurement Units (IMU) readings and encoder measurements, eliminating the need for prior knowledge of terrain and robot dynamics, but potentially suffering from observability and divergence issues. To overcome these, an invariant EKF [2] based on invariant observer theory [17] is proposed, and its stability conditions are provided [5]. However, the invariant property is forfeited when the IMU biases are considered. In short, the filtering method has no stability guarantee, and

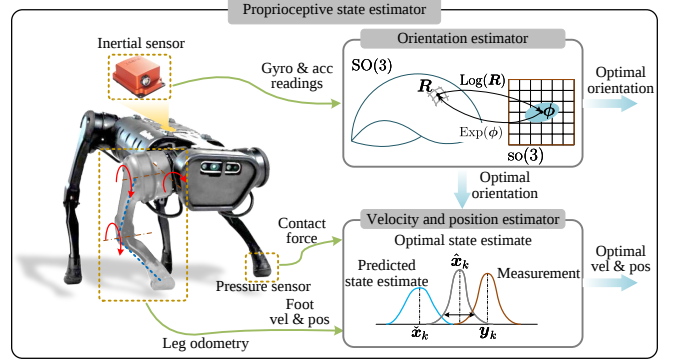


Fig. 1. Overview of the two-stage proprioceptive state estimator.

the impact of contact switching on stability has not been discussed.

Smoothing methods estimate the robot’s state at a specific time using all previous measurements and some future ones. The work in [10] is the first one to apply smoothing to legged robot state estimation, presenting a fusion of inertial, kinematic and contact information into a factor graph framework. Initially, focusing on a single contact point, the authors later extended it in [11] to accommodate multiple footsteps. Several enhancements have improved the reliability of the smoothing method under challenging conditions [12], [13], ultimately developing a legged navigation system tailored for all-terrain legged robots [14]. While the smoothing method is gaining popularity, none of these studies provide a stability analysis that considers the impact of contact switching.

To address stability issues in filtering and smoothing methods for legged robot state estimation, we propose a two-stage state estimator for legged robots shown in Fig. 1, validated using the A1 robot. The state estimation is decomposed into two stages: a complementary filter-based base orientation estimation and a linear Kalman filter-based base velocity and position estimation, importantly, both stages undergo rigorous stability analysis. The key contribution of this research is that a two-stage state estimator with stability guarantee for dynamic legged locomotion is proposed that only need proprioceptive sensors and are validated through real-world experiments.

This paper is structured as follows: Section II introduces the system and sensor models and outlines the estimator’s structure. Section III and Section IV present the estimator’s submodules and stability analysis. Experiments are detailed in Section V, and Section VI concludes the article.

This work was supported by the Natural Science Foundation of Shandong Province [grant number ZR2024QE256 and ZR2024YQ035], the National Natural Science Foundation of China [grant number 52175007], the Scientific Research Foundation of Harbin Institute of Technology, and the Royal Society [grant number RG\R2\232409]. (Corresponding authors: Weihua Li and Chengxu Zhou)

Jun Li, Weihua Li, and Jianfeng Wang are with the Special Vehicle Institute, Harbin Institute of Technology, Weihai, 264209, China. (e-mail: liweihua@hit.edu.cn)

Yuhui Wan is with the School of Mechanical Engineering, University of Leeds, Leeds, LS2 9JT, United Kingdom.

Chengxu Zhou is with the Department of Computer Science, University College London, London, WC1E 6EA, United Kingdom. (e-mail: chengxu.zhou@ucl.ac.uk)

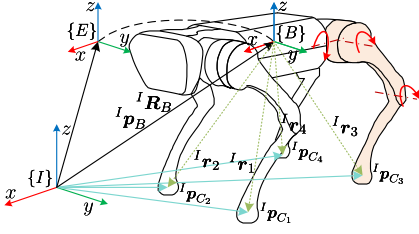


Fig. 2. Illustration of the required coordinate systems including the inertial frame Σ_I , base frame Σ_B and estimator frame Σ_E .

II. PROBLEM FORMULATION

This section presents the system and sensor models for a legged robot, along with the structure of the proprioceptive state estimator. Fig. 2 depicts the required coordinate systems comprising the inertial frame Σ_I and the base frame Σ_B . The state estimator computes the base frame velocity and pose relative to the inertial frame with sensor measurements.

A. System Model

The origin of Σ_B is located at the center of mass of the robot base with the x-axis pointing to the robot's front, the y-axis pointing to the left, and the z-axis pointing upward. The system model can be represented as follows:

$${}^I\dot{\mathbf{p}}_B = {}^I\mathbf{v}_{I,B}, \quad {}^I\dot{\mathbf{v}}_{I,B} = {}^I\mathbf{a}_{I,B}, \quad (1)$$

$${}^I\dot{\mathbf{R}}_B = {}^I\mathbf{R}_B {}^B\boldsymbol{\omega}_{I,B}^\wedge, \quad (2)$$

where ${}^I\mathbf{p}_B$, ${}^I\mathbf{v}_{I,B}$, and ${}^I\mathbf{a}_{I,B}$ are the base's absolute position, velocity, and acceleration, respectively. ${}^I\mathbf{R}_B$ and ${}^B\boldsymbol{\omega}_{I,B}$ are the base's orientation and angular velocity, and ${}^B\boldsymbol{\omega}_{I,B}$ is expressed in Σ_B . Operators $(\cdot)^\wedge$ and $(\cdot)^\vee$ convert 3D vectors to skew-symmetric matrices and vice versa.

B. Sensor Models

The state estimator relies on proprioceptive sensors and their stochastic measurement models, assuming measurements are corrupted by additive white Gaussian noise.

1) *Inertial Sensors*: IMU measures base angular velocity ${}^B\tilde{\boldsymbol{\omega}}_B$ and acceleration ${}^B\tilde{\mathbf{c}}_B$, respectively, in Σ_B . These measurements are affected by noise [1], [16]

$${}^B\tilde{\boldsymbol{\omega}}_{I,B} = {}^B\boldsymbol{\omega}_{I,B} + \mathbf{b}_g + \mathbf{w}_g, \quad \mathbf{w}_g \sim \mathcal{N}(\mathbf{0}_{3,1}, \sigma_g), \quad (3)$$

$${}^B\tilde{\mathbf{c}}_{I,B} = {}^B\mathbf{c}_{I,B} + \mathbf{b}_a + \mathbf{w}_a, \quad \mathbf{w}_a \sim \mathcal{N}(\mathbf{0}_{3,1}, \sigma_a), \quad (4)$$

where ${}^B\boldsymbol{\omega}_{I,B}$ and ${}^B\mathbf{c}_{I,B}$ denote the true angular velocity and proper acceleration of Σ_B relative to Σ_I , respectively. $\mathbf{b}_g$ and $\mathbf{b}_a$ are the gyroscope and acceleration biases.

2) *Joint Encoders*: Encoders provide angles and velocities of all n_j joints. The encoder measurements $\tilde{\boldsymbol{\theta}}$ of joint angles $\boldsymbol{\theta}$ and $\tilde{\dot{\boldsymbol{\theta}}}$ of velocities $\dot{\boldsymbol{\theta}}$ are corrupted by noises [1], [16]:

$$\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} + \mathbf{w}_\theta, \quad \mathbf{w}_\theta \sim \mathcal{N}(\mathbf{0}_{n_j,1}, \sigma_\theta), \quad (5)$$

$$\tilde{\dot{\boldsymbol{\theta}}} = \dot{\boldsymbol{\theta}} + \mathbf{w}_{\dot{\theta}}, \quad \mathbf{w}_{\dot{\theta}} \sim \mathcal{N}(\mathbf{0}_{n_j,1}, \sigma_{\dot{\theta}}). \quad (6)$$

3) *Contact Force Sensors*: The pneumatic pressure sensor is often placed at each foot of a legged robot. The measurement $\tilde{f}_i$ of the i th foot's contact force f_i is affected by noise [18]:

$$\tilde{f}_i = f_i + w_{f_i}, \quad w_{f_i} \sim \mathcal{N}(0, \sigma_{f_i}). \quad (7)$$

C. Structure of State Estimator

The two-stage proprioceptive state estimator fuses measurements from inertial sensors, joint encoders, and force sensors, and contains a base orientation estimator and a base velocity and position estimator submodule. Its overall procedure is succinctly summarized in Algorithm 1. Further elaboration on these submodules can be found in the subsequent sections.

Algorithm 1 Two-Stage Multi-Sensor Fusion Algorithm

Input: ${}^B\tilde{\boldsymbol{\omega}}_{I,B}$, ${}^B\tilde{\mathbf{c}}_{I,B}$, $\tilde{\boldsymbol{\theta}}$, $\tilde{\dot{\boldsymbol{\theta}}}$, $\tilde{f}_i$

Output: ${}^I\hat{\mathbf{R}}_B$, ${}^I\hat{\mathbf{p}}_B$, ${}^I\hat{\mathbf{v}}_{I,B}$, ${}^I\hat{\mathbf{p}}_{C_i}$

- 1: **procedure** EST(${}^B\tilde{\boldsymbol{\omega}}_{I,B}$, ${}^B\tilde{\mathbf{c}}_{I,B}$: ${}^I\hat{\mathbf{R}}_B$)
 - 2: Compute $\boldsymbol{\omega}_{\text{cor}}$ and ${}^I\hat{\mathbf{R}}_B$ with Eq. (15).
 - 3: **return** ${}^I\hat{\mathbf{R}}_B$
 - 4: **end procedure**
 - 5: **procedure**
EST(${}^I\hat{\mathbf{R}}_B$, ${}^B\tilde{\mathbf{c}}_{I,B}$, $\tilde{\boldsymbol{\theta}}$, $\tilde{\dot{\boldsymbol{\theta}}}$, $\tilde{f}_i$: ${}^I\hat{\mathbf{p}}_B$, ${}^I\hat{\mathbf{v}}_{I,B}$, ${}^I\hat{\mathbf{p}}_{C_i}$)
 - 6: Compute $\mathbf{u}$ with Eq. (28).
 - 7: Compute $\mathbf{z}$ with Eqs. (21) and (23).
 - 8: Inflate $\mathbf{W}_{I_{\mathbf{p}_{C_i}}}$, $\mathbf{V}_{I_{\mathbf{r}_{C_i}}}$, $\mathbf{V}_{I_{\dot{\mathbf{r}}_{C_i}}}$, $\mathbf{V}_{h_i}$ with Eq. (34).
 - 9: Compute predicted $\hat{\mathbf{x}}$ and $\hat{\mathbf{P}}$ with Eq. (32).
 - 10: Compute corrected $\hat{\mathbf{x}}$ and $\hat{\mathbf{P}}$ with Eq. (33).
 - 11: Slice ${}^I\hat{\mathbf{p}}_B$, ${}^I\hat{\mathbf{v}}_{I,B}$ and ${}^I\hat{\mathbf{p}}_{C_i}$ from $\hat{\mathbf{x}}$.
 - 12: **return** ${}^I\hat{\mathbf{p}}_B$, ${}^I\hat{\mathbf{v}}_{I,B}$, ${}^I\hat{\mathbf{p}}_{C_i}$
 - 13: **end procedure**
-

III. BASE ORIENTATION ESTIMATION

A. Complementary Filtering on SO(3)

Classical complementary filtering on Euclidean space has been extended to the SO(3) manifold, leveraging its natural Lie group structure [19]. Given the orientation ${}^I\mathbf{R}_B$ and angular velocity ${}^B\boldsymbol{\omega}_{I,B}$ of a rigid body, dynamics of the true system are

$${}^I\dot{\mathbf{R}}_B = {}^I\mathbf{R}_B {}^B\boldsymbol{\omega}_{I,B}^\wedge = ({}^I\mathbf{R}_B {}^B\boldsymbol{\omega}_{I,B})^\wedge {}^I\mathbf{R}_B, \quad (8)$$

where ${}^I\mathbf{R}_B {}^B\boldsymbol{\omega}_{I,B}$ transforms ${}^B\boldsymbol{\omega}_{I,B}$ to ${}^I\boldsymbol{\omega}_{I,B}$. Since the true orientation ${}^I\mathbf{R}_B$ evolves on SO(3), its estimate ${}^I\hat{\mathbf{R}}_B$ is required to evolve on SO(3) and has a similar dynamics form. Since the true value ${}^I\mathbf{R}_B$ is not available, an alternative needs to be used. If the estimate ${}^I\hat{\mathbf{R}}_B$ is used, the filter is called a passive complementary filter.

The dynamics of passive complementary filters [19] is

$$\begin{aligned} {}^I\dot{\hat{\mathbf{R}}}_B &= \left({}^I\hat{\mathbf{R}}_B {}^B\boldsymbol{\omega}_{I,B} + \kappa_{\text{cor}} {}^I\hat{\mathbf{R}}_B \boldsymbol{\omega}_{\text{cor}} \right)^\wedge {}^I\hat{\mathbf{R}}_B \\ &= {}^I\hat{\mathbf{R}}_B ({}^B\boldsymbol{\omega}_{I,B} + \kappa_{\text{cor}} \boldsymbol{\omega}_{\text{cor}})^\wedge, \end{aligned} \quad (9)$$

where $\kappa_{\text{cor}} > 0$ is a gain, $\boldsymbol{\omega}_{\text{cor}}$ is correction term, and ${}^I\hat{\mathbf{R}}_B(0) = \hat{\mathbf{R}}_0$ is initial value. The correction term $\boldsymbol{\omega}_{\text{cor}}$ expressed in Σ_E is a nonlinear approximation of the orientation error between the true rotation ${}^I\mathbf{R}_B$ and its estimate ${}^I\hat{\mathbf{R}}_B$.

B. IMU-based Orientation Estimation

The base orientation estimator, as the first stage of the proposed state estimator, fuses IMU gyroscope and accelerometer readings. The gyroscope provides high-frequency gravity direction measurements, while the accelerometer provides low-frequency linear acceleration measurements, enabling complementary filters for orientation estimation.

The accelerometer measures the non-gravitational linear accelerations, given by ${}^B\mathbf{c}_{I,B} = {}^I\mathbf{R}_B^T ({}^I\mathbf{a}_{I,B} - \mathbf{a}_g)$, where $\mathbf{a}_g = -g\mathbf{e}_3$ is the gravitational field, g is the gravitational constant, and $\mathbf{e}_3 = [0, 0, 1]^T$ is the unit vector expressed in Σ_I . Therefore, for the direction of gravity in frame Σ_B ,

$$\mathbf{v}_1 := -{}^I\mathbf{R}_B^T \mathbf{e}_3 = {}^I\mathbf{R}_B^T (-\mathbf{e}_3) = {}^I\mathbf{R}_B^T \mathbf{v}_g, \quad (10)$$

where $\mathbf{v}_g$ is the direction of gravity in frame Σ_I . There is a reasonable low-frequency estimate calculated from the IMU acceleration measurement, given by

$$\hat{\mathbf{v}}_1 := \frac{{}^B\tilde{\mathbf{c}}_{I,B}}{\|{}^B\tilde{\mathbf{c}}_{I,B}\|} = {}^I\hat{\mathbf{R}}_B^T \left(\frac{{}^I\mathbf{a}_{I,B} - \mathbf{a}_g}{\|{}^B\tilde{\mathbf{c}}_{I,B}\|} \right) = {}^I\hat{\mathbf{R}}_B^T \mathbf{v}_g, \quad (11)$$

where ${}^B\tilde{\mathbf{c}}_{I,B}$ is the acceleration measurement, and ${}^I\hat{\mathbf{R}}_B$ is the estimated orientation. The error between the true direction of gravity $\mathbf{v}_1$ and its estimate $\hat{\mathbf{v}}_1$ is

$$E_v = 1 - \langle \hat{\mathbf{v}}_1, \mathbf{v}_1 \rangle = 1 - \text{tr}(\hat{\mathbf{v}}_1 \mathbf{v}_1^T) = 1 - \text{tr}(\boldsymbol{\eta} \mathbf{M}_g) \quad (12)$$

where $\boldsymbol{\eta} = {}^I\hat{\mathbf{R}}_B^T {}^I\mathbf{R}_B$ is the estimation error, $\mathbf{M}_g = {}^I\mathbf{R}_B^T \mathbf{v}_g \mathbf{v}_g^T {}^I\mathbf{R}_B$ is positive semi-definite.

With two available measurements, Eq. (10) and Eq. (11), the passive complementary filter can be designed as follows:

$${}^I\dot{\mathbf{R}}_B = {}^I\hat{\mathbf{R}}_B ({}^B\boldsymbol{\omega}_{I,B} + \kappa_{\text{cor}} \boldsymbol{\omega}_{\text{cor}})^\wedge, \quad (13)$$

$$\boldsymbol{\omega}_{\text{cor}} = -(\hat{\mathbf{v}}_1 \times \mathbf{v}_1). \quad (14)$$

In practice, the above filter can be made more robust and easier to implement with a few tricks, such as replacing the true rotation ${}^I\mathbf{R}_B$ in Eq. (10) with its estimate ${}^I\hat{\mathbf{R}}_B$. With the zero-order hold assumption, the above continuous dynamics can be discretized as

$$\begin{aligned} {}^I\hat{\mathbf{R}}_{B,k} &= {}^I\hat{\mathbf{R}}_{B,k-1} \text{Exp} \left[({}^B\tilde{\boldsymbol{\omega}}_{I,B,k} + \kappa_{\text{cor}} \boldsymbol{\omega}_{\text{cor},k}) \Delta t \right] \\ \boldsymbol{\omega}_{\text{cor},k} &= \frac{{}^B\tilde{\mathbf{c}}_{I,B,k}}{\|{}^B\tilde{\mathbf{c}}_{I,B,k}\|} \times \left({}^I\hat{\mathbf{R}}_{B,k-1}^T \mathbf{e}_3 \right) \end{aligned} \quad (15)$$

where $\Delta t = t_k - t_{k-1}$, $\text{Exp}(\cdot)$ is the exponential map, $\kappa_{\text{cor}} = 0.1$ is a constant, and $\max$ and $\min$ are functions that return the maximum or minimum of two inputs.

C. Stability Analysis

To analyze the stability of the filter from equations (13) and (14), we introduce a candidate Lyapunov function

$$V(\boldsymbol{\eta}) = E_v = 1 - \text{tr}(\boldsymbol{\eta} \mathbf{M}_g). \quad (16)$$

The correction term $\boldsymbol{\omega}_{\text{cor}}$ can be expressed as

$$\boldsymbol{\omega}_{\text{cor}}^\wedge = -(\hat{\mathbf{v}}_1 \times \mathbf{v}_1)^\wedge = \hat{\mathbf{v}}_1^\wedge \mathbf{v}_1^\wedge - \mathbf{v}_1^\wedge \hat{\mathbf{v}}_1^\wedge = 2\pi_a(\boldsymbol{\eta} \mathbf{M}_g) \quad (17)$$

where $\pi_a(\mathbf{S}) = \frac{1}{2}(\mathbf{S} - \mathbf{S}^T)$ and $\pi_s(\mathbf{S}) = \frac{1}{2}(\mathbf{S} + \mathbf{S}^T)$ are the antisymmetric and symmetric projection operators.

The derivative of the estimation error is given by

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= {}^I\dot{\mathbf{R}}_B^T {}^I\mathbf{R}_B + {}^I\hat{\mathbf{R}}_B^T {}^I\dot{\mathbf{R}}_B \\ &= {}^I\hat{\mathbf{R}}_B^T {}^I\mathbf{R}_B {}^B\boldsymbol{\omega}_{I,B}^\wedge - {}^B\boldsymbol{\omega}_{I,B}^\wedge {}^I\hat{\mathbf{R}}_B^T {}^I\mathbf{R}_B \\ &\quad + \kappa_{\text{cor}} (\boldsymbol{\omega}_{\text{cor}}^\wedge)^T {}^I\hat{\mathbf{R}}_B^T {}^I\mathbf{R}_B \\ &= \boldsymbol{\eta} {}^B\boldsymbol{\omega}_{I,B}^\wedge - {}^B\boldsymbol{\omega}_{I,B}^\wedge \boldsymbol{\eta} + \kappa_{\text{cor}} (\boldsymbol{\omega}_{\text{cor}}^\wedge)^T \boldsymbol{\eta}. \end{aligned} \quad (18)$$

Using the fact that the trace of a commutator is zero, $\text{tr}([\boldsymbol{\eta}, {}^B\boldsymbol{\omega}_{I,B}^\wedge]) = 0$, and that for any symmetric matrix π_s and any antisymmetric matrix π_a , $\text{tr}(\pi_s \pi_a) = 0$, we can obtain the time derivative of the Lyapunov function as

$$\begin{aligned} \dot{V}(\boldsymbol{\eta}) &= -\text{tr}(\dot{\boldsymbol{\eta}} \mathbf{M}_g + \boldsymbol{\eta} \dot{\mathbf{M}}_g) \\ &= -\text{tr}([\boldsymbol{\eta} \mathbf{M}_g, {}^B\boldsymbol{\omega}_{I,B}^\wedge] - \kappa_{\text{cor}} \boldsymbol{\omega}_{\text{cor}}^\wedge \boldsymbol{\eta} \mathbf{M}_g) \\ &= \kappa_{\text{cor}} \text{tr}[\boldsymbol{\omega}_{\text{cor}}^\wedge (\pi_a(\boldsymbol{\eta} \mathbf{M}_g) + \pi_s(\boldsymbol{\eta} \mathbf{M}_g))] \\ &= -2\kappa_{\text{cor}} \|\pi_a(\boldsymbol{\eta} \mathbf{M}_g)\|^2, \end{aligned} \quad (19)$$

which is negative semi-definite. Therefore, Lyapunov's direct method completes the proof, and we can conclude that the filter is asymptotically stable.

IV. BASE VELOCITY AND POSITION ESTIMATION

A. Forward Differential Kinematics

The generalized coordinates for a robot is denoted by $\mathbf{q}$, and the generalized velocities by $\mathbf{v}$. Let Σ_{C_i} be the i th foot-end frame, then the homogeneous transformation between the foot-end frame Σ_{C_i} and the inertial frame Σ_I is

$${}^I\mathbf{H}_{C_i}(\mathbf{q}) = \begin{bmatrix} {}^I\mathbf{R}_{C_i}(\mathbf{q}) & {}^I\mathbf{p}_{C_i}(\mathbf{q}) \\ \mathbf{0} & 1 \end{bmatrix} \quad (20)$$

where ${}^I\mathbf{p}_{C_i}(\mathbf{q})$ and ${}^I\mathbf{R}_{C_i}(\mathbf{q})$ are the relative position and orientation. The relative position of foot i is

$${}^I\mathbf{r}_{C_i} := {}^I\mathbf{p}_{C_i} - {}^I\mathbf{p}_B. \quad (21)$$

Since the body velocity of frame Σ_{C_i} can be computed with

$${}^{C_i}\mathbf{V}_{I,C_i} = \begin{bmatrix} {}^{C_i}\boldsymbol{\omega}_{I,C_i} \\ {}^{C_i}\mathbf{v}_{I,C_i} \end{bmatrix} = \left({}^I\mathbf{H}_{C_i}^{-1}(\mathbf{q})^T \dot{{}^I\mathbf{H}}_{C_i}(\mathbf{q}) \right)^\vee, \quad (22)$$

the relative velocity of foot i is given by

$${}^I\dot{\mathbf{r}}_{C_i} := {}^I\mathbf{R}_{C_i} {}^{C_i}\mathbf{v}_{I,C_i} - {}^I\mathbf{R}_B {}^B\mathbf{v}_{I,B}. \quad (23)$$

B. Kinematics-Based Velocity and Position Estimation

The second stage of the proposed estimator estimates the base velocity ${}^I\mathbf{v}_{I,B}$ and position ${}^I\mathbf{p}_B$ using the orientation estimate ${}^I\hat{\mathbf{R}}_B$ and leg kinematics measurements. With the orientation estimated, the second stage can be formulated as a linear Kalman filter. To maintain consistent prediction and correction equations, the state variable of the filter includes the absolute position of the base center ${}^I\mathbf{p}_B$, its velocity ${}^I\mathbf{v}_{I,B}$, and the absolute foot positions, ${}^I\mathbf{p}_{C_i}$, to account for leg kinematics. The state variables are represented as

$$\mathbf{x} = [{}^I\mathbf{p}_B^T, {}^I\mathbf{v}_{I,B}^T, {}^I\mathbf{p}_{C_1}^T, \dots, {}^I\mathbf{p}_{C_n}^T]^T, \quad (24)$$

where n is the number of feet.

The prediction equations are modeled as

$${}^I\dot{\tilde{\mathbf{p}}}_B = {}^I\mathbf{v}_{I,B} + \mathbf{w}_p, \quad (25)$$

$${}^I\dot{\tilde{\mathbf{v}}}_{I,B} = {}^I\hat{\mathbf{R}}_B^B \tilde{\mathbf{c}}_{I,B} + \mathbf{a}_g + \mathbf{w}_v, \quad (26)$$

$${}^I\dot{\tilde{\mathbf{p}}}_{C_i} = \mathbf{w}_d \quad \forall i = \{1, \dots, n\}, \quad (27)$$

where $(\tilde{\cdot})$ denotes a priori quantity. $\mathbf{w}_p$, $\mathbf{w}_v$ and $\mathbf{w}_d$ are the white noise terms characterizing the process noise in the base position, velocity, and feet positions variability, respectively. The term

$$\mathbf{u} = {}^I\hat{\mathbf{R}}_B^B \tilde{\mathbf{c}}_{I,B} + \mathbf{a}_g \quad (28)$$

can be viewed as an input to the system because it can be computed ahead of time.

The correction equations are modeled as

$${}^I\mathbf{r}_{C_i} = ({}^I\tilde{\mathbf{p}}_{C_i} - {}^I\tilde{\mathbf{p}}_B) + \mathbf{v}_r, \quad (29)$$

$${}^I\dot{\mathbf{r}}_{C_i} = (-{}^I\tilde{\mathbf{v}}_{I,B}) + \mathbf{v}_{\dot{r}}, \quad (30)$$

$$h_i = \mathbf{e}_3^T {}^I\tilde{\mathbf{p}}_{C_i} + v_h, \quad (31)$$

where ${}^I\mathbf{r}_{C_i}$, ${}^I\dot{\mathbf{r}}_{C_i}$ and h_i represent the relative position, relative velocity, and foot height of foot i respectively, and are obtained from leg forward kinematics using Eq. (21) and Eq. (23). The base position and velocity, ${}^I\tilde{\mathbf{p}}_B$, ${}^I\tilde{\mathbf{v}}_{I,B}$, and the position of foot i , ${}^I\tilde{\mathbf{p}}_{C_i}$ are predicted in the previous prediction process from Eq.(25) to (27). The terms $\mathbf{v}_r$, $\mathbf{v}_{\dot{r}}$, and v_h represent the white noise terms characterizing the measurement noise in the relative foot position and velocity, and absolute foot height. The above three correction equations are applicable for every foot.

With the zero-order hold assumption, the filter for a legged robot can be reformulated as a discrete-time linear Kalman filter, which involves a prediction step:

$$\begin{aligned} \tilde{\mathbf{x}}_k &= \mathbf{F}_k \hat{\mathbf{x}}_{k-1} + \mathbf{B}_k \mathbf{u}_k \\ \tilde{\mathbf{P}}_k &= \mathbf{F}_k \hat{\mathbf{P}}_{k-1} \mathbf{F}_k^T + \mathbf{W}_k \end{aligned} \quad (32)$$

where $\mathbf{P}_k$ is the estimate covariance matrix, $(\hat{\cdot})$ denotes a posterior quantity, and

$$\mathbf{F}_k = \begin{bmatrix} \mathbf{I} & \Delta t \mathbf{I} & 0 \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix}, \mathbf{B}_k = \begin{bmatrix} \frac{1}{2} \Delta t^2 \mathbf{I} \\ \Delta t \mathbf{I} \\ 0 \end{bmatrix},$$

$$\mathbf{W}_k = \text{diag}(\mathbf{W}_{I\mathbf{p}_B,k}, \mathbf{W}_{I\mathbf{v}_{I,B},k}, \mathbf{W}_{I\mathbf{p}_{C_1},k}, \dots, \mathbf{W}_{I\mathbf{p}_{C_n},k})$$

are the state transition matrix, the control-input matrix and the covariance of process noise. The correction step involves the following equations:

$$\begin{aligned} \tilde{\mathbf{y}}_k &= \mathbf{z}_k - \mathbf{H}_k \tilde{\mathbf{x}}_k \\ \mathbf{S}_k &= \mathbf{H}_k \tilde{\mathbf{P}}_k \mathbf{H}_k^T + \mathbf{V}_k \\ \mathbf{K}_k &= \tilde{\mathbf{P}}_k \mathbf{H}_k^T \mathbf{S}_k^{-1} \\ \hat{\mathbf{x}}_k &= \tilde{\mathbf{x}}_k + \mathbf{K}_k \tilde{\mathbf{y}}_k \\ \hat{\mathbf{P}}_k &= (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \tilde{\mathbf{P}}_k \end{aligned} \quad (33)$$

$$\text{where } \mathbf{z}_k = \begin{bmatrix} {}^I\mathbf{r}_{C_1,k} \\ \vdots \\ {}^I\mathbf{r}_{C_n,k} \\ {}^I\dot{\mathbf{r}}_{1,k} \\ \vdots \\ {}^I\dot{\mathbf{r}}_{n,k} \\ h_1 \\ \vdots \\ h_n \end{bmatrix}, \mathbf{H}_k = \begin{bmatrix} -\mathbf{I} & 0 & \mathbf{I} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\mathbf{I} & 0 & 0 & \cdots & \mathbf{I} \\ 0 & -\mathbf{I} & & & \\ \vdots & \vdots & & 0 & \\ 0 & -\mathbf{I} & & & \\ & & \mathbf{e}_3^T & \cdots & 0 \\ & & \vdots & \ddots & \vdots \\ & & 0 & \cdots & \mathbf{e}_3^T \end{bmatrix},$$

$$\mathbf{V}_k = \text{diag}(\mathbf{V}_{I\mathbf{r}_{C_1,k}}, \dots, \mathbf{V}_{I\mathbf{r}_{C_n,k}}, \mathbf{V}_{I\dot{\mathbf{r}}_{C_1,k}}, \dots, \mathbf{V}_{I\dot{\mathbf{r}}_{C_n,k}}, \mathbf{V}_{h_1,k}, \dots, \mathbf{V}_{h_n,k}),$$

are the measurement, the observation matrix, and the covariance of the observation noise.

The covariance inflation method adjusts the covariance matrices $\mathbf{W}_k$ and $\mathbf{V}_k$ by inflating certain state covariances to large values to reduce their effect. The equation:

$$\mathbf{M}_b = (1 + (1 - \kappa_{i,k})C) \mathbf{M}_b^0, \quad (34)$$

where $\kappa_{i,k} = \min(\max(\tilde{f}_{i,k}/f_{\text{thre}}, 0), 1)$, $f_{\text{thre}} = mg/n$ represents the threshold value determined by the gravity of the entire robot (mg) and the number of legs (n). $\tilde{f}_{i,k}$ denotes the measured normal force of foot i at timestep t_k . C is a large constant, usually 1000. $\mathbf{M}_b$ stands for matrix blocks $\mathbf{W}_{I\mathbf{p}_{C_i},k}$, $\mathbf{V}_{I\mathbf{r}_{C_i},k}$, $\mathbf{V}_{I\dot{\mathbf{r}}_{C_i},k}$ and $\mathbf{V}_{h_i,k}$, and $\mathbf{M}_b^0$ stands for constant covariance $\mathbf{W}_{I\mathbf{p}_{C_i}}^0$, $\mathbf{V}_{I\mathbf{r}_{C_i}}^0$, $\mathbf{V}_{I\dot{\mathbf{r}}_{C_i}}^0$ and $\mathbf{V}_h^0$ for the foot in stance. Each covariance is represented as a diagonal matrix with uniform diagonal entries, as noted in [1].

C. Stability Analysis

The Kalman filter is typically used in linear Gaussian systems, and its stability and robustness are independent of stochastic issues [20]. These properties are related only to the deterministic continuous-time or discrete-time Riccati equation. Thus, we perform stability analysis using deterministic form of the filter given by Eqs. (32) and (33), shown as

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{F}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u} \\ \mathbf{m} = \mathbf{H}(t)\mathbf{x} \end{cases} \quad (35)$$

where $\mathbf{F}(t)$, $\mathbf{B}(t)$ and $\mathbf{H}(t)$ are well-defined continuous and bounded matrix-valued functions.

A Riccati observer is a linear time-varying system of the above form, given by

$$\dot{\hat{\mathbf{x}}} = \mathbf{F}(t)\hat{\mathbf{x}} + \mathbf{B}(t)\mathbf{u} + \mathbf{K}(t)(\mathbf{m} - \mathbf{H}(t)\hat{\mathbf{x}}), \quad (36)$$

where the observer gain is calculated using

$$\mathbf{K}(t) = \kappa_r(t)\mathbf{P}_r(t)\mathbf{H}^T\mathbf{Q}_r(t), \quad \kappa_r \geq 0.5. \quad (37)$$

Here, $\mathbf{P}_r(t)$ is the solution to the continuous Riccati equation

$$\dot{\mathbf{P}}_r = \mathbf{F}\mathbf{P}_r + \mathbf{P}_r\mathbf{F}^T - \mathbf{P}_r\mathbf{H}^T\mathbf{Q}_r\mathbf{H}\mathbf{P}_r + \mathbf{R}_r \quad (38)$$

with $\mathbf{P}_r(0)$ being any positive definite matrix, and $\mathbf{Q}_r$, and $\mathbf{R}_r$ being positive semi-definite matrices that need to be specified. The Kalman filter is a special case of Riccati observer where $\kappa_r = 1$ and the matrices $\mathbf{R}_r$ and $\mathbf{Q}_r^{-1}$ are covariance matrices.

With estimation error $\tilde{\mathbf{x}} := \mathbf{x} - \hat{\mathbf{x}}$, the error equation is

$$\dot{\tilde{\mathbf{x}}} = (\mathbf{F}(t) - \mathbf{K}(t)\mathbf{H}(t)) \tilde{\mathbf{x}}. \quad (39)$$

A candidate Lyapunov function can then be defined as

$$V(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^T \mathbf{P}_r^{-1} \tilde{\mathbf{x}}. \quad (40)$$

Using the fact that the time-derivative of $\mathbf{P}_r^{-1}$ is given by

$$\dot{\mathbf{P}}_r^{-1} = -\mathbf{P}_r^{-1} \mathbf{F} - \mathbf{F}^T \mathbf{P}_r^{-1} + \mathbf{H}^T \mathbf{Q}_r \mathbf{H} - \mathbf{P}_r^{-1} \mathbf{R}_r \mathbf{P}_r^{-1}, \quad (41)$$

it is easy to verify that the time-derivative of $V(\tilde{\mathbf{x}})$ is

$$\begin{aligned} \dot{V}(\tilde{\mathbf{x}}) &= -\tilde{\mathbf{x}}^T \left((2\kappa_r - 1) \mathbf{H}^T \mathbf{Q}_r \mathbf{H} + \mathbf{P}_r^{-1} \mathbf{R}_r \mathbf{P}_r^{-1} \right) \tilde{\mathbf{x}} \\ &\leq -\tilde{\mathbf{x}}^T (\mathbf{P}_r^{-1} \mathbf{R}_r \mathbf{P}_r^{-1}) \tilde{\mathbf{x}} \leq -\frac{p_{\inf}}{p_{\sup}} r_{\inf} V(\tilde{\mathbf{x}}) \leq 0, \end{aligned} \quad (42)$$

where $p_{\inf}$, $p_{\sup}$ are the infimum and supremum over time of the smallest and largest eigenvalues of the function $\mathbf{P}_r$. $r_{\inf}$ is the infimum over time of the smallest of $\mathbf{R}_r$. Covariance inflation shown in Eq. (34) adds a bounded positive definite matrix, keeping $\mathbf{W}_k$ and $\mathbf{V}_k$ bounded and positive definite. Thus, $\mathbf{P}_r$ is always positive definite and satisfies

$$p_{\inf} \mathbf{I} \leq \mathbf{P}_r \leq p_{\sup} \mathbf{I}, \quad \text{with } p_{\sup} \geq p_{\inf} \geq 0. \quad (43)$$

Lyapunov's direct method completes the proof, and the filter is asymptotically stable.

V. EXPERIMENTAL RESULTS

To evaluate the proposed estimator, a planning and control framework [21] was used to generate and stabilize motions on the A1 robot. Laboratory experiments were conducted in a cage with an OptiTrack motion capture system, while field experiments took place in a real-world environment.

A. Laboratory Experiments

In the first experiment, the robot twisted floating-base with its feet fixed on the ground and an unchanged base position to verify the orientation estimator's effectiveness. In the second experiment, it performed an S-turn in a trotting gait to validate the velocity and position estimator. Both experiments, shown in Fig. 3, used ground truth from the OptiTrack system.

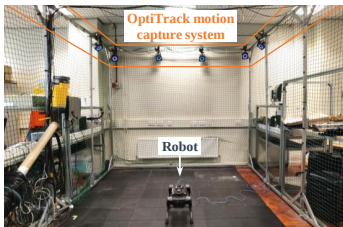


Fig. 3. OptiTrack system and A1 robot used in the experiment.

Fig. 4 presents snapshots of the robot performing the body twisting motion, rotating in the x -, y -, and z -directions. Fig. 5 compares the estimated orientation $\mathbf{R}_e$ from Eq. (15) with the OptiTrack ground truth $\mathbf{R}_m$. The estimator provides precise orientation estimates, with maximum absolute errors of 0.025 rad and 0.015 rad in the z -, x -, and y -directions.

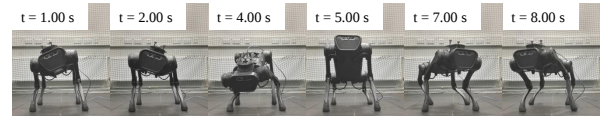


Fig. 4. Snapshots of the robot performing the body-twisting motion.

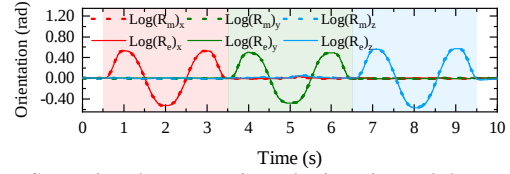


Fig. 5. Comparison between estimated orientation and the ground truth.

Fig. 6 displays the S-turn motion, which includes forward movement, S-turn, and straight motion. Fig. 7 compares the estimated position with the ground truth. In the top and middle sub-pictures, there is a slight drift in the estimates in the x - and y -directions, which is inevitable since proprioceptive sensors cannot provide position measurements to correct predictions. However, no drift occurs in the z -direction, with a maximum error of 0.01 m, owing to the pseudo measurements from Eq. (31) that correct the base height. Position estimates in z -direction fluctuate due to intermittent ground contact.

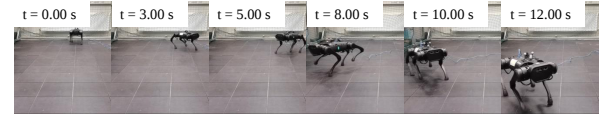


Fig. 6. Snapshots of the robot performing the S-turn motion.

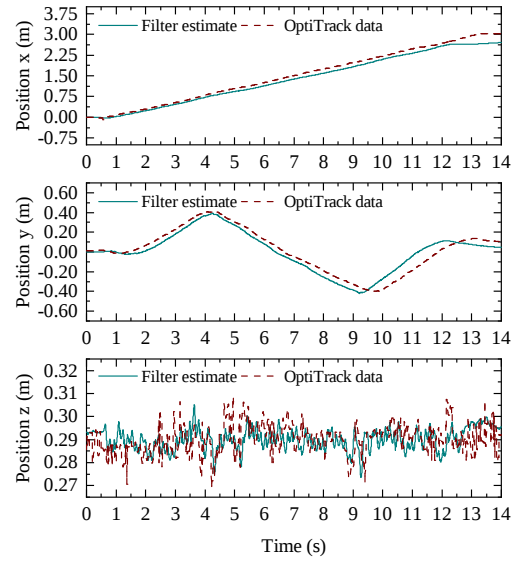


Fig. 7. Comparison between the estimated position and the ground truth.

B. Field Experiments

Two field experiments were conducted to assess the estimator's performance in real-world environments. The multi-gaited locomotion experiment involved the robot running at varying speeds, adopting static-walk, walking-trot, trot, flying-trot, trot, walking-trot, and static-walk gaits in sequence. In the push recovery experiment, the robot was kicked and recovered twice while trotting in place.

Fig. 8 shows the robot's multi-gait locomotion, and Fig. 9 shows base position variations across gaits. The z -direction estimates exhibit larger amplitudes in trot (0.015 m) and flying-trot (0.020 m), but smaller ones in static-walk (0.005 m) and walking-trot (0.010 m), reflecting shorter contact times and fewer grounded feet in dynamic gaits. This experiment showed the estimator could handle arbitrary gaits, independent of the number of grounded feet.

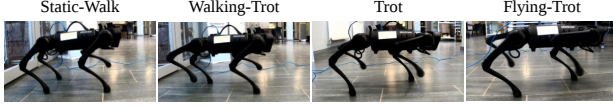


Fig. 8. Snapshots of the robot locomoting with multiple gaits.

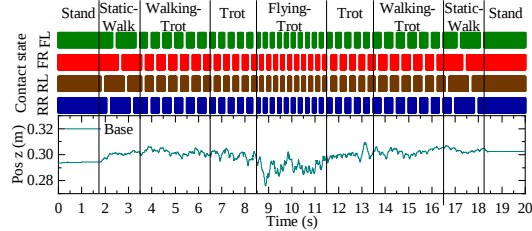


Fig. 9. Graph showing the gait type, contact state, base velocity in z -direction, and base position in z -direction.

Fig. 10 presents the push recovery motion, and Fig. 11 displays base and foot position variations. Within the two gray oval boxes, the black line crossing red and blue lines indicates the robot base's center exceeding the area enclosed by the four feet during sidekicks. Despite this, the estimator accurately estimated the robot's state, enabling for the motion controller to restore balance. The sudden change in the base's position from 0.00 m to -0.30 m increases the proportion of the base position control task in motion controller tasks, leading the robot to walk back to its start position after strong sidekicks. This experiment showed the estimator's capability to estimate the base state even under strong disturbances.

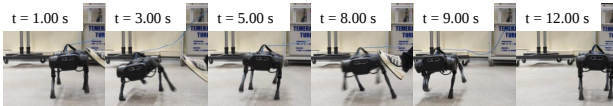


Fig. 10. Snapshots of the push recovery experiment.

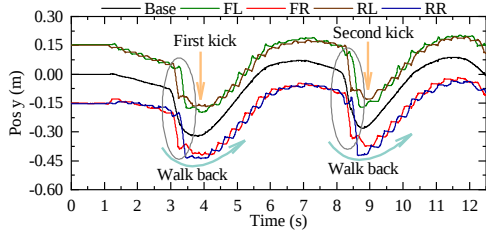


Fig. 11. Estimated base and foot positions in y -direction.

VI. CONCLUSION

This paper presents a two-stage proprioceptive state estimator for legged robots, enabling dynamic locomotion and improved resistance to external disturbances. The estimator decomposes the challenging floating base state estimation problem into two components: base orientation estimation and base velocity/position estimation. These two components are implemented with complementary filter and linear

Kalman filter, respectively. By leveraging this decomposition, the proposed estimator provides a stability guarantee, a crucial feature lacking in existing filtering and smoothing methods. Validated on a quadrupedal robot performing dynamic motions like multi-gaited locomotion and push recovery, the estimator shows promise for challenging environments. Future work includes integration with perception systems and comparative experiments with existing methods.

REFERENCES

- [1] M. Bloesch, M. Hutter, M. A. Hoepfner, et al., "State Estimation for Legged Robots - Consistent Fusion of Leg Kinematics and IMU," in *Proc. Robot.: Sci. Syst.*, Sydney, Australia, Jul. 2012.
- [2] R. Hartley, M. G. Jadidi, J. Grizzle, et al., "Contact-Aided Invariant Extended Kalman Filtering for Legged Robot State Estimation," in *Proc. Robot.: Sci. Syst.*, Pittsburgh, Pennsylvania, Jun. 2018.
- [3] M. Camurri, et al., "Pronto: A multi-sensor state estimator for legged robots in real-world scenarios," *Front. Robot. AI*, vol. 7, no. 68, 2020.
- [4] G. Fink and C. Semini, "Proprioceptive Sensor Fusion for Quadruped Robot State Estimation," in *Proc. IEEE Int. Conf. Robot. Autom.*, Las Vegas, NV, USA, 2020, pp. 10914-10920.
- [5] Y. Gao, et al., "Invariant Extended Kalman Filtering for Hybrid Models of Bipedal Robot Walking," *IFAC-Pap.*, vol. 54, 2021, pp. 290-297.
- [6] P. Ramadoss, et al., "DILIGENT-KIO: A Proprioceptive Base Estimator for Humanoid Robots using Extended Kalman Filtering on Matrix Lie Groups," in *Proc. IEEE Int. Conf. Robot. Autom.*, Xi'an, China, 2021.
- [7] S. Teng, et al., "Legged Robot State Estimation in Slippery Environments Using Invariant Extended Kalman Filter with Velocity Update," in *Proc. IEEE Int. Conf. Robot. Autom.*, Xi'an, 2021, pp. 3104-3110.
- [8] S. Yang, et al., "Multi-IMU Proprioceptive Odometry for Legged Robots," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, Detroit, MI, USA, 2023, pp. 774-779.
- [9] Q. Zhao et al., "LIKO: LiDAR, Inertial, and Kinematic Odometry for Bipedal Robots," *Proc. IEEE Int. Conf. Robot. Autom.*, Yokohama, Japan, 2024, pp. 1180-1185.
- [10] R. Hartley, J. Mangelson, L. Gan, et al., "Legged Robot State Estimation Through Combined Forward Kinematic and Preintegrated Contact Factors," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 4422-4429.
- [11] R. Hartley, et al., "Hybrid Contact Preintegration for Visual-Inertial-Contact State Estimation Using Factor Graphs," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, Madrid, Spain, 2018, pp. 3783-3790.
- [12] D. Wisth, M. Camurri, and M. Fallon, "Robust Legged Robot State Estimation Using Factor Graph Optimization," *IEEE Robot. Autom. Lett.*, vol. 4, no. 4, pp. 4507-4514, 2019.
- [13] D. Wisth, et al., "Preintegrated Velocity Bias Estimation to Overcome Contact Nonlinearities in Legged Robot Odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, Paris, France, 2020, pp. 392-398.
- [14] D. Wisth, M. Camurri and M. Fallon, "VILENS: Visual, Inertial, Lidar, and Leg Odometry for All-Terrain Legged Robots," *IEEE Trans. Robot.*, vol. 39, no. 1, pp. 309-326, 2023.
- [15] Y. Kim, B. Yu, E. M. Lee, et al., "STEP: State Estimator for Legged Robots Using a Preintegrated Foot Velocity Factor," *IEEE Trans. Automat. Contr.*, vol. 7, no. 2, pp. 4456-4463, 2022.
- [16] Z. Yoon, J. -H. Kim and H. -W. Park, "Invariant Smoother for Legged Robot State Estimation With Dynamic Contact Event Information," *IEEE Trans. Robot.*, vol. 40, pp. 193-212, 2024.
- [17] A. Barrau, et al., "The Invariant Extended Kalman Filter as a Stable Observer," *IEEE Trans. Automat. Contr.*, vol. 62, pp. 1797-1812, 2017.
- [18] G. Bledt, M. J. Powell, B. Katz, et al., "MIT Cheetah 3: Design and Control of a Robust, Dynamic Quadrupedal Robot," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, 2018, pp. 2245-2252.
- [19] R. Mahony, T. Hamel and J. -M. Pfimlin, "Nonlinear Complementary Filters on the Special Orthogonal Group," *IEEE Trans. Automat. Contr.*, vol. 53, no. 5, pp. 1203-1218, June 2008.
- [20] T. Hamel and C. Samson, "Position estimation from direction or range measurements," *Automatica*, vol. 82, pp. 137-144, 2017.
- [21] J. Li, H. Gao, Y. Wan, H. Yu, and C. Zhou, "A Real-Time Planning and Control Framework for Robust and Dynamic Quadrupedal Locomotion," *J. Bionic. Eng.*, Feb. 2023.