

Invariant Filtering for Legged Humanoid Locomotion on a Dynamic Rigid Surface

Yuan Gao , Student Member, IEEE, Chengzhi Yuan , Member, IEEE, and Yan Gu , Member, IEEE

Abstract—State estimation for legged locomotion over a dynamic rigid surface (DRS), which is a rigid surface moving in the world frame (e.g., ships, aircraft, and trains), remained an underexplored problem. This article introduces an invariant extended Kalman filter that estimates the robot's pose and velocity during DRS locomotion by using common sensors of legged robots [e.g., inertial measurement units (IMUs), joint encoders, and RDB-D camera]. A key feature of the filter lies in that it explicitly addresses the nonstationary surface–foot contact point and the hybrid robot behaviors. Another key feature is that in the absence of IMU biases, the filter satisfies the attractive group affine and invariant observation conditions, and is thus provably convergent for the deterministic continuous phases. The observability analysis is performed to reveal the effects of DRS movement on the state observability, and the convergence property of the hybrid deterministic filter system is examined for the observable state variables. Experiments of a Digit humanoid robot walking on a pitching treadmill validate the effectiveness of the proposed filter under large estimation errors and moderate DRS movement. The video of the experiments can be found at: <https://youtu.be/ScQIBFUSKzo>.

Index Terms—Dynamic environments, legged locomotion, state estimation.

I. INTRODUCTION

STATE estimation is essential to providing the estimates of a robot's movement state (e.g., pose and velocity) needed for planning and control. While state estimation for locomotion on

static [1] or relatively unstable surfaces [2] has been extensively studied, state estimation for dynamic rigid surfaces (DRSs), such as ships and aircraft [3], has not been fully investigated. This article aims to solve the estimation problem for DRS locomotion. Yet, solving this problem is challenging due to the nonstationary surface–foot contact point [3], [4] and the hybrid robot dynamics involving both continuous behaviors and discrete foot-landing events [5]–[7].

Extended Kalman filtering (EKF) [8], [9] has been used to achieve real-time state estimation of legged locomotion on static surfaces by fusing the data returned by common on-board sensors, such as encoders (which measure the joint angles), as well as inertial measurement units (IMUs) attached to the robot's base (which measure the base's linear acceleration and angular velocity in the base frame). Recently, EKF-based estimators have been created to estimate a robot's base pose and velocity [1], [10]. These methods can be applied to general legged locomotion because their formulation is independent of robot dynamics and gait types. Specifically, the process model includes the IMU motion and bias dynamics, and the measurement model is based on the the leg odometry formed via the forward kinematics between the base and the ground contact point. Yet, they may not be effective for DRS locomotion because they assume the foot–ground contact point is static in the world frame. Also, similar to standard EKF, they may not handle large estimation errors well, because the underlying system linearization depends on the true state, but is evaluated at the state estimate [11].

To ensure provable, rapid convergence under large estimation error, the previous EKF-based design [1] has been transformed into an invariant extended Kalman filter (InEKF) for legged locomotion on static surfaces [12], [13]. By the theory of InEKF [14], when the InEKF system meets the group affine and invariant observation conditions, the system linearization is independent of the true state, and thus is valid even under relatively large errors. Still, the effectiveness of these methods in handling DRS locomotion is unclear, especially under a relatively significant surface motion (e.g., ship motion under sea waves [15]), because of the underlying assumption of stationary surface–foot contact.

To address hybrid robot behaviors, state estimators for hybrid models of legged locomotion [16], [17] have been derived. However, the convergence property of invariant filters for hybrid locomotion models has not been examined.

This article introduces an InEKF method to produce real-time accurate state estimation for bipedal humanoid walking on a

Manuscript received 30 December 2021; revised 26 March 2022; accepted 29 April 2022. Date of publication 2 August 2022; date of current version 16 August 2022. Recommended by Technical Editor Z. Chen and Senior Editor X. Chen. This work was supported by National Science Foundation under Grant CMMI-1934280 and Grant CMMI-2046562. (Corresponding author: Yan Gu.)

Yuan Gao is with the Department of Mechanical Engineering, University of Massachusetts Lowell, Lowell, MA 01854 USA (e-mail: yuan_gao@student.uml.edu).

Yan Gu is with the University of Massachusetts Lowell, Lowell, MA 01854 USA, and also with the School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: yangu@purdue.edu).

Chengzhi Yuan is with the Department of Mechanical, Industrial and Systems Engineering, University of Rhode Island, Kingston, RI 02881 USA (e-mail: cyuan@uri.edu).

This article has supplementary material provided by the authors and color versions of one or more figures available at <https://doi.org/10.1109/TMECH.2022.3176015>.

Digital Object Identifier 10.1109/TMECH.2022.3176015

DRS even under relatively large estimation errors. The main contributions of this article are as follows:

- 1) Deriving an InEKF that considers the DRS motion and hybrid robot behaviors and meets the group affine and invariant observation properties without IMU biases.
- 2) Building a right-invariant measurement model based on the rotational constraint at the surface–foot contact area, enhancing the convergence rate, and rendering the base yaw angle observable under general DRS movement.
- 3) Performing observability analysis that reveals how the DRS pose affects state observability, and proving the stability for the hybrid deterministic error dynamics.
- 4) Demonstrating the computational efficiency, accuracy, and robustness of the proposed filter through experiments of humanoid walking on a rocking treadmill.

Some results in this article have been reported [18]. The new substantial contributions of this article are the last three items in the aforementioned list.

The rest of this article is organized as follows. Section II provides a brief background on matrix Lie groups. Section III presents the problem formulation. Section IV introduces the proposed InEKF for the hybrid model of DRS locomotion. Section V provides observability and convergence analysis. Section VI reports experiment results. Section VII discusses the capabilities and limitations of the filter. Finally, Section VIII concludes this article.

II. PRELIMINARIES

The matrix Lie group, denoted as $\mathcal{G}$, is a subset of $n \times n$ invertible square matrices. The associated Lie algebra $\mathfrak{g}$ with a dimension of $\dim \mathfrak{g}$ is a set of $n \times n$ square matrices. The linear operator $(\cdot)^\wedge$ maps any vector $\xi \in \mathbb{R}^{\dim \mathfrak{g}}$ onto $\mathfrak{g}$. The exponential map $\exp : \mathbb{R}^{\dim \mathfrak{g}} \rightarrow \mathcal{G}$ is defined as $\exp(\xi) \triangleq \text{expm}(\xi^\wedge)$, where expm is the usual exponential of $n \times n$ matrices. The inverse operator of $(\cdot)^\wedge$ is denoted as $(\cdot)^\vee : \mathfrak{g} \rightarrow \mathbb{R}^{\dim \mathfrak{g}}$. The adjoint matrix Ad_X at X for any vector $\xi \in \mathbb{R}^{\dim \mathfrak{g}}$ is defined as $\text{Ad}_X \xi = (X \xi^\wedge X^{-1})^\vee$. More detailed introductions to matrix Lie groups can be found in [19]. A nomenclature table is given in supplementary material.

III. PROBLEM FORMULATION

In the proposed filter design, the state is chosen as variables that are often needed in locomotion planning and control, including the linear velocity $\mathbf{v} \in \mathbb{R}^3$ and orientation $\mathbf{R} \in SO(3)$ of the robot's base (e.g., chest) expressed in the world frame (see Fig. 1). The state also includes the base position $\mathbf{p} \in \mathbb{R}^3$ and contact-point position $\mathbf{p}^c \in \mathbb{R}^3$ (see Fig. 1), so as to exploit the forward kinematics between the base and contact/foot frames in the filter design [1], [12].

The DRS of interest possesses two common characteristics of real-world DRSs, such as aircraft and vessels. First, when traveling on such surfaces, a robot can only see landmarks attached to the surface instead of the world frame, e.g., due to the concealed environment on the surface. Second, the DRS orientation $\mathbf{R}^{\text{DRS}} \in SO(3)$ and linear and angular velocities

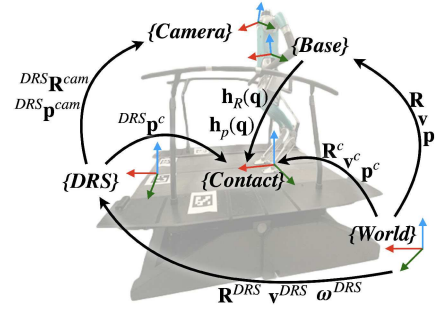


Fig. 1. Illustration of coordinate frames and key variables. The treadmill is a DRS that rotates in the world frame.

$\mathbf{v}^{\text{DRS}}, \omega^{\text{DRS}} \in \mathbb{R}^3$, respectively, (see Fig. 1) are relatively accurately known. This is a reasonable assumption because these real-world DRSs are typically equipped with high-accuracy motion monitoring systems [20]. Also, the proposed filter design explicitly treats the inaccurate knowledge of surface pose and motion, as explained in Section IV.

The sensors considered in this study are common on-board sensors, which are an IMU attached to the robot's base, joint encoders, a red green blue-depth (RGB-D) camera, and a contact indicator. The RGB-D camera tracks the landmarks attached to the DRS, which is used to obtain the camera pose in the DRS frame. The contact indicator detects foot landing events.

The encoders measure the joint angles $\mathbf{q} \in \mathbb{R}^m$ with m number of joints. Corrupted by the white zero-mean Gaussian noise $\mathbf{w}^q$, the raw encoder data $\tilde{\mathbf{q}}$ are expressed as $\tilde{\mathbf{q}} = \mathbf{q} + \mathbf{w}^q$.

The IMU includes a gyroscope and an accelerator that measure the angular velocity $\omega \in \mathbb{R}^3$ and the linear acceleration $\mathbf{a} \in \mathbb{R}^3$ of the IMU, respectively, in the base frame. Corrupted by the white Gaussian zero-mean noise $\mathbf{w}^a, \mathbf{w}^\omega \in \mathbb{R}^3$, as well as biases $\mathbf{b}^a, \mathbf{b}^\omega \in \mathbb{R}^3$, the IMU readings $\tilde{\mathbf{a}}$ and $\tilde{\omega}$ are expressed as $\tilde{\mathbf{a}} = \mathbf{a} + \mathbf{b}^a + \mathbf{w}^a$ and $\tilde{\omega} = \omega + \mathbf{b}^\omega + \mathbf{w}^\omega$, respectively.

A. Continuous-Phase IMU Motion and Bias Dynamics

To form the process model, we choose to adopt the IMU motion dynamics due to its accuracy and simplicity [1]. At time t , the IMU motion dynamics is given by

$$\begin{aligned} \dot{\mathbf{R}}_t &= \mathbf{R}_t(\tilde{\omega}_t - \mathbf{b}_t^\omega - \mathbf{w}_t^\omega)_\times \\ \dot{\mathbf{v}}_t &= \mathbf{R}_t(\tilde{\mathbf{a}}_t - \mathbf{b}_t^a - \mathbf{w}_t^a) + \mathbf{g}, \text{ and } \dot{\mathbf{p}}_t = \mathbf{v}_t \end{aligned} \quad (1)$$

where $(\cdot)_\times$ is a skew-symmetric matrix and $\mathbf{g}$ is the gravitational acceleration vector. The IMU bias dynamics is modeled as the Brownian motion [12]

$$\dot{\mathbf{b}}_t^a = \mathbf{w}_t^{ba} \text{ and } \dot{\mathbf{b}}_t^\omega = \mathbf{w}_t^{b\omega} \quad (2)$$

where $\mathbf{w}_t^{ba}$ and $\mathbf{w}_t^{b\omega}$ are the white zero-mean Gaussian noises.

B. Continuous-Phase Contact-Point Motion Dynamics

During DRS locomotion, the foot moves in the world frame due to the surface movement. Thus, the deterministic motion model of the contact point is not $\dot{\mathbf{p}}_t^c = \mathbf{0}$ as in previous work [1], [12] on static surfaces. Instead, we explicitly consider the contact

point velocity $\mathbf{v}_t^c$ in the model

$$\dot{\mathbf{p}}_t^c = \mathbf{v}_t^c. \quad (3)$$

In this study, to inform the model in (3), we choose to directly measure the contact point velocity based on the known surface pose and motion and the measured contact position in the DRS frame through the following kinematics:

$$\mathbf{v}_t^c = \mathbf{v}_t^{\text{DRS}} + \boldsymbol{\omega}_t^{\text{DRS}} \times (\mathbf{R}_t^{\text{DRSDRS}} \mathbf{p}_t^c). \quad (4)$$

where $\mathbf{p}_t^c$ is the contact point position relative to the DRS frame, expressed in the DRS frame (see Fig. 1). Note that the computation of the velocity $\mathbf{v}_t^c$ depends on the robot's camera data and the joint angle data returned by encoders. Also, recall that the surface orientation $\mathbf{R}_t^{\text{DRS}}$ and the motions $\boldsymbol{\omega}_t^{\text{DRS}}$ and $\mathbf{v}_t^{\text{DRS}}$ are assumed to be known as explained earlier. An example of computing $\mathbf{v}_t^c$ is given in Section VI.

The velocity computation inaccuracy is considered as

$$\tilde{\mathbf{v}}_t^c = \mathbf{v}_t^c + \mathbf{R}_t \mathbf{w}_t^c \quad (5)$$

where $\tilde{\mathbf{v}}_t^c \in \mathbb{R}^3$ is the measured contact point velocity and the inaccuracy $\mathbf{w}_t^c$ is modeled as the white Gaussian zero-mean noise expressed in the base frame.

C. Discrete Jump Dynamics at a Foot Landing

At a foot landing, the swing and support legs switch roles, causing a discrete jump in the contact point position $\mathbf{p}_t^c$. To appropriately propagate the estimate and covariance at foot landings, we choose to explicitly consider the jump.

The jump map of the contact point position $\mathbf{p}_t^c$ is

$$\mathbf{p}_{t^+}^c = \mathbf{p}_t^c + \mathbf{R}_t \mathbf{h}_c(\mathbf{q}_t) \quad (6)$$

where the subscript t^+ denotes the timing just after the foot landing at t and the function $\mathbf{h}_c$ is the forward kinematics from the previous support-foot position to the new one, expressed in the base frame. Except for $\mathbf{p}_t^c$, all other state variables remain continuous at foot switching.

With the first-order Taylor expansion, the nonlinear term in the jump dynamics (6) can be approximated as $\mathbf{R}_t \mathbf{h}_c(\mathbf{q}_t) \approx \mathbf{R}_t \mathbf{h}_c(\tilde{\mathbf{q}}_t) - \mathbf{R}_t \frac{\partial \mathbf{h}_c}{\partial \mathbf{q}}(\tilde{\mathbf{q}}_t) \mathbf{w}_t^q$.

D. Position-Based Forward Kinematics Measurement

To connect the contact and the base frames, we adopt the leg odometry measurement in [1], [12] [see Fig. 2(b)]

$$\mathbf{R}_t^T (\mathbf{p}_t^c - \mathbf{p}_t) = \mathbf{h}_p(\mathbf{q}_t) \quad (7)$$

where the forward kinematics function $\mathbf{h}_p$ is the support foot position relative to the base expressed in the base frame. Given the inaccuracy of the encoder reading $\tilde{\mathbf{q}}_t = \mathbf{q}_t + \mathbf{w}_t^q$ and with the first-order Taylor expansion, the model in (7) can be rewritten as $\mathbf{h}_p(\mathbf{q}_t) \approx \mathbf{h}_p(\tilde{\mathbf{q}}_t) - \frac{\partial \mathbf{h}_p}{\partial \mathbf{q}}(\tilde{\mathbf{q}}_t) \mathbf{w}_t^q$.

E. Contact Orientation-Based Measurement

When the support foot and the surface have a full area contact, their normal vectors are parallel, whether the surface is stationary or moving [see Fig. 2(a)]. In this study, we utilize this

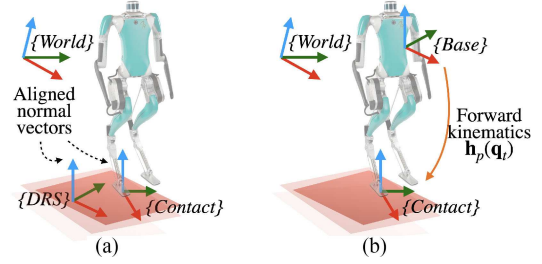


Fig. 2. Illustrations of the observations. (a) Normal vector alignment of the contact and DRS frames. (b) Contact point position in the base frame.

rotational kinematic relationship to form a measurement model. Suppose that the z -axis of the contact and surface frames are aligned and normal to the DRS. Then, the following hold:

$$\mathbf{R}_t^{\text{DRS}} [0 \ 0 \ 1]^T = \mathbf{R}_t^c [0 \ 0 \ 1]^T = \mathbf{R}_t \mathbf{h}_R(\mathbf{q}_t) [0 \ 0 \ 1]^T \quad (8)$$

where $\mathbf{R}_t^c \in SO(3)$ is the contact frame orientation and the forward kinematics matrix function $\mathbf{h}_R$ is the support foot orientation with respect to the base frame (see Fig. 1).

To address the inaccuracy of the known surface orientation $\tilde{\mathbf{R}}_t^{\text{DRS}}$, we assume that the true orientation is corrupted by the white Gaussian zero-mean uncertainty $\mathbf{w}_t^{\text{DRS}}$ as $\mathbf{R}_t^{\text{DRS}} = \exp(-\mathbf{w}_t^{\text{DRS}}) \tilde{\mathbf{R}}_t^{\text{DRS}} \approx (\mathbf{I}_3 - (\mathbf{w}_t^{\text{DRS}})_\times) \tilde{\mathbf{R}}_t^{\text{DRS}}$, where $\mathbf{I}_n$ is an $n \times n$ identity matrix.

To handle the inaccuracy of the encoder reading $\tilde{\mathbf{q}}_t$, the support foot orientation $\mathbf{R}_t \mathbf{h}_R(\mathbf{q}_t)$ is approximated as $\mathbf{R}_t \mathbf{h}_R(\mathbf{q}_t) \approx \mathbf{R}_t \mathbf{h}_R(\tilde{\mathbf{q}}_t) - \mathbf{R}_t \mathbf{J}_{h_R}(\tilde{\mathbf{q}}_t, \mathbf{w}_t^q)$, where the matrix $\mathbf{J}_{h_R}(\tilde{\mathbf{q}}_t, \mathbf{w}_t^q)$ is obtained based on the Jacobian of each column of $\mathbf{h}_R \triangleq [\mathbf{h}_{R,1}, \mathbf{h}_{R,2}, \mathbf{h}_{R,3}]$ as $\mathbf{J}_{h_R} \triangleq [\frac{\partial \mathbf{h}_{R,1}}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_t) \mathbf{w}_t^q, \frac{\partial \mathbf{h}_{R,2}}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_t) \mathbf{w}_t^q, \frac{\partial \mathbf{h}_{R,3}}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_t) \mathbf{w}_t^q]$.

Combining these equations yields

$$\begin{aligned} & \mathbf{R}_t^T \tilde{\mathbf{R}}_t^{\text{DRS}} [0 \ 0 \ 1]^T + \mathbf{R}_t^T (-\mathbf{w}_t^{\text{DRS}})_\times \tilde{\mathbf{R}}_t^{\text{DRS}} [0 \ 0 \ 1]^T \\ & \approx \mathbf{h}_R(\tilde{\mathbf{q}}_t) [0 \ 0 \ 1]^T - \frac{\partial \mathbf{h}_{R,3}}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_t) \mathbf{w}_t^q. \end{aligned} \quad (9)$$

IV. FILTER DESIGN

This section introduces the proposed InEKF design based on the models formulated in Section III.

The proposed filter derivation begins with proper state representation. We adopt the representation in [12] since our filters estimate the same state. First, the state variables $\mathbf{R}_t$, $\mathbf{v}_t$, $\mathbf{p}_t$, and $\mathbf{p}_t^c$ are expressed on the matrix Lie group $\mathcal{G}$ as

$$\mathbf{X}_t \triangleq \begin{bmatrix} \mathbf{R}_t & [\mathbf{v}_t, \mathbf{p}_t, \mathbf{p}_t^c] \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 \end{bmatrix} \in \mathcal{G} \quad (10)$$

where $\mathbf{0}_{m \times n}$ is an $m \times n$ zero matrix. The Lie group $\mathcal{G}$ is $\text{SE}_3(3)$, an extension of the special Euclidean group $\text{SE}(3)$.

To explicitly handle IMU biases, they are also chosen as state variables. These biases are typically expressed on the vector space instead of $\mathcal{G}$ [14], that is, $\boldsymbol{\theta}_t \triangleq [(\mathbf{b}_t^\omega)^T, (\mathbf{b}_t^a)^T]^T$.

Let $(\cdot)$ denote the estimate of the variable $(\cdot)$. Based on the InEKF framework [11], we use the right-invariant error $\boldsymbol{\eta}_t$ to

represent the estimation error of $\mathbf{X}_t$ on $\mathcal{G}$ as

$$\boldsymbol{\eta}_t \triangleq \bar{\mathbf{X}}_t \mathbf{X}_t^{-1} \in \mathcal{G}. \quad (11)$$

The log of the invariant error, denoted as $\boldsymbol{\xi}_t$, is a vector on $\mathbb{R}^{\dim \mathfrak{g}}$ defined via $\boldsymbol{\eta}_t \triangleq \exp(\boldsymbol{\xi}_t) = \expm(\boldsymbol{\xi}_t^\wedge)$. The expressions of $\boldsymbol{\eta}_t$, $\boldsymbol{\xi}_t$, and $\boldsymbol{\xi}_t^\wedge$ are in Section 2 of supplementary document. The IMU bias error ζ_t is defined as $\zeta_t \triangleq \bar{\boldsymbol{\theta}}_t - \boldsymbol{\theta}$.

A. Continuous-Phase Process Model and Propagation Step

This section introduces the process model and propagation step of the proposed filter for the continuous phases.

1) *Process Model*: Based on the IMU motion and bias dynamics and the contact point motion in (1)–(3), the process model is expressed as

$$\dot{\mathbf{X}}_t = \begin{bmatrix} \mathbf{R}_t(\tilde{\boldsymbol{\omega}}_t - \mathbf{b}_t^\omega) \times [\mathbf{R}_t(\tilde{\mathbf{a}}_t - \mathbf{b}_t^a) + \mathbf{g}, \mathbf{v}_t, \tilde{\mathbf{v}}_t^c] \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix} - \mathbf{X}_t(\mathbf{w}_t^X)^\wedge \triangleq \mathbf{f}_{ut}(\mathbf{X}_t, \boldsymbol{\theta}_t) - \mathbf{X}_t(\mathbf{w}_t^X)^\wedge \quad (12)$$

with the noise vector $\mathbf{w}_t^X \triangleq [(\mathbf{w}_t^\omega)^T, (\mathbf{w}_t^a)^T, \mathbf{0}_{1 \times 3}, (\mathbf{w}_t^c)^T]^T$. Here, we define the input $\mathbf{u}_t$ to consist of the IMU and encoder readings and the measured contact point velocity, i.e., $\mathbf{u}_t = [\tilde{\boldsymbol{\omega}}_t^T, \tilde{\mathbf{a}}_t^T, (\tilde{\mathbf{v}}_t^c)^T, \tilde{\mathbf{q}}_t^T]^T$. Note that the encoder reading $\tilde{\mathbf{q}}_t$ is not an input to the continuous-phase process model in (12), but is used later in the jump process model.

2) *Linearized Error Model*: By using the first-order Taylor expansion $\boldsymbol{\eta}_t \approx \mathbf{I} + \boldsymbol{\xi}_t^\wedge$ and by applying the chain rule to express $\dot{\boldsymbol{\eta}}_t$, we obtain the linearized error equation

$$\begin{bmatrix} \dot{\boldsymbol{\xi}}_t \\ \dot{\zeta}_t \end{bmatrix} = \mathbf{A}_t \begin{bmatrix} \boldsymbol{\xi}_t \\ \zeta_t \end{bmatrix} + \begin{bmatrix} \mathbf{Ad}_{\bar{\mathbf{X}}_t} \mathbf{0}_{12 \times 6} \\ \mathbf{0}_{6 \times 12} \quad \mathbf{I}_6 \end{bmatrix} \mathbf{w}_t \quad (13)$$

where the noise term $\mathbf{w}_t$ is $\mathbf{w}_t \triangleq [(\mathbf{w}_t^X)^T, (\mathbf{w}_t^{b\omega})^T, (\mathbf{w}_t^{ba})^T]^T$, the adjoint matrix $\mathbf{Ad}_{\bar{\mathbf{X}}_t}$ is given in Section 3 of supplementary material, and the matrix $\mathbf{A}_t$ is

$$\mathbf{A}_t = \begin{bmatrix} \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\bar{\mathbf{R}}_t & \mathbf{0}_{3 \times 3} \\ (\mathbf{g}) \times & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -(\tilde{\mathbf{v}}_t) \times \bar{\mathbf{R}}_t & -\bar{\mathbf{R}}_t \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -(\tilde{\mathbf{p}}_t) \times \bar{\mathbf{R}}_t & \mathbf{0}_{3 \times 3} \\ (\tilde{\mathbf{v}}_t^c) \times & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -(\tilde{\mathbf{p}}_t^c) \times \bar{\mathbf{R}}_t & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} \end{bmatrix}. \quad (14)$$

Note that $\mathbf{A}_t$ contains the contact point velocity $\tilde{\mathbf{v}}_t^c$ because the process model explicitly considers it. Derivation of (13) and (14) is in Section 3 of supplementary material.

3) *Propagation*: Let t_n ($n \in \{1, 2, \dots\}$) denote the time when sensors return data for estimation error correction. Then, during the propagation step on $t \in [t_{n-1}, t_n)$, the estimates $\bar{\mathbf{X}}_t$ and $\bar{\boldsymbol{\theta}}_t$ are obtained via $\dot{\bar{\mathbf{X}}}_t = \mathbf{f}_{ut}(\bar{\mathbf{X}}_t, \bar{\boldsymbol{\theta}}_t)$ and $\dot{\bar{\boldsymbol{\theta}}}_t = \mathbf{0}$ based on the process models in (2) and (12).

By the InEKF methodology, the covariance matrix $\mathbf{P}_t$ is propagated via the Riccati equation associated with the linearized error model in (13) as $\dot{\mathbf{P}}_t = \mathbf{A}_t \mathbf{P}_t + \mathbf{P}_t \mathbf{A}_t^T + \bar{\mathbf{Q}}_t$, where

$$\bar{\mathbf{Q}}_t \triangleq \begin{bmatrix} \mathbf{Ad}_{\bar{\mathbf{X}}_t} \mathbf{0}_{12 \times 6} \\ \mathbf{0}_{6 \times 12} \quad \mathbf{I}_6 \end{bmatrix} \text{Cov}(\mathbf{w}_t) \begin{bmatrix} \mathbf{Ad}_{\bar{\mathbf{X}}_t} \mathbf{0}_{12 \times 6} \\ \mathbf{0}_{6 \times 12} \quad \mathbf{I}_6 \end{bmatrix}^T.$$

Remark 1 (Group Affine Property): Without IMU biases, the continuous process model in (12) is group affine, as defined

in [11]. Thus, without biases and in the deterministic case, the linear error dynamics in (13) is exact and independent of the true state, and the covariance propagation is exact. Such features are different from the standard EKF whose linearization accuracy relies on estimation error.

B. Continuous-Phase Measurement Models and Update Step

This section formulates the two measurements in (7) and (9) into the right-invariant observation form defined in [11] and introduces the update step of the proposed InEKF at time t_n . These treatments result in an error update equation, which is independent of the true state.

1) *Right-Invariant Orientation-Based Measurement*: The orientation-based measurement in (9) can be rewritten into the following right-invariant observation form:

$$\underbrace{\begin{bmatrix} \mathbf{h}_R(\tilde{\mathbf{q}})_{t_n} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ \mathbf{0}_{3 \times 1} \end{bmatrix}}_{\mathbf{Y}_{1,t_n}} = \mathbf{X}_{t_n}^{-1} \underbrace{\begin{bmatrix} \tilde{\mathbf{R}}_{t_n}^{\text{DRS}} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ \mathbf{0}_{3 \times 1} \end{bmatrix}}_{\mathbf{d}_{1,t_n}} + \begin{bmatrix} \mathbf{V}_{1,t_n} \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \quad (15)$$

with $\mathbf{V}_{1,t_n} = \mathbf{R}_{t_n}^T (\tilde{\mathbf{R}}_{t_n}^{\text{DRS}} [001]^T) \times \mathbf{w}_{t_n}^{\text{DRS}} + \frac{\partial \mathbf{h}_{R,3}}{\partial \tilde{\mathbf{q}}}(\tilde{\mathbf{q}}_{t_n}) \mathbf{w}_{t_n}^q$.

2) *Right-Invariant Position Measurement*: The position measurement in (7) can be expressed as [12]

$$\underbrace{\begin{bmatrix} \mathbf{h}_p(\tilde{\mathbf{q}}_{t_n}) \\ 0 \\ 1 \\ -1 \end{bmatrix}}_{\mathbf{Y}_{2,t_n}} = \mathbf{X}_{t_n}^{-1} \underbrace{\begin{bmatrix} \mathbf{0}_{3 \times 1} \\ 0 \\ 1 \\ -1 \end{bmatrix}}_{\mathbf{d}_{2,t_n}} + \begin{bmatrix} \frac{\partial \mathbf{h}_p}{\partial \tilde{\mathbf{q}}}(\tilde{\mathbf{q}}_{t_n}) \mathbf{w}_{t_n}^q \\ \mathbf{0}_{3 \times 1} \end{bmatrix}. \quad (16)$$

3) *Update*: At time t_n , the updated estimates and covariance, denoted as $(\bar{\mathbf{X}}_{t_n}^\dagger, \bar{\boldsymbol{\theta}}_{t_n}^\dagger)$ and $\mathbf{P}_{t_n}^\dagger$, respectively, are given by [11]

$$\begin{aligned} \bar{\mathbf{X}}_{t_n}^\dagger &= \exp(\mathbf{L}_{t_n}^\xi \mathbf{z}_{t_n}) \bar{\mathbf{X}}_{t_n}, \bar{\boldsymbol{\theta}}_{t_n}^\dagger = \bar{\boldsymbol{\theta}}_{t_n} \\ &+ \mathbf{L}_{t_n}^\zeta \mathbf{z}_{t_n}, \mathbf{P}_{t_n}^\dagger = (\mathbf{I} - \mathbf{L}_{t_n} \mathbf{H}_{t_n}) \mathbf{P}_{t_n} \end{aligned} \quad (17)$$

where $\mathbf{L}_{t_n} \triangleq [(\mathbf{L}_{t_n}^\xi)^T, (\mathbf{L}_{t_n}^\zeta)^T]^T$ is the filter gain, $\mathbf{H}_{t_n}$ is the observation matrix, and $\mathbf{z}_{t_n} \triangleq [(\bar{\mathbf{X}}_{t_n} \mathbf{Y}_{1,t_n} - \mathbf{d}_{1,t_n})^T, (\bar{\mathbf{X}}_{t_n} \mathbf{Y}_{2,t_n} - \mathbf{d}_{2,t_n})^T]^T$.

To derive the observation matrix $\mathbf{H}_{t_n}$, we first decompose it into $\mathbf{H}_{t_n} = [\mathbf{H}_{1,t_n}^T, \mathbf{H}_{2,t_n}^T]^T$, where $\mathbf{H}_{1,t_n} \in \mathbb{R}^{6 \times 12}$ and $\mathbf{H}_{2,t_n} \in \mathbb{R}^{6 \times 12}$ are, respectively, associated with the measurement models in (15) and (16). Since the measurement models are not explicitly dependent on biases, the matrix $\mathbf{H}_{i,t_n}$ ($i = 1, 2$) can be further decomposed as $\mathbf{H}_{i,t_n} \triangleq [\tilde{\mathbf{H}}_{i,t_n}, \mathbf{0}_{3 \times 6}; \mathbf{0}_{3 \times 12}, \mathbf{0}_{3 \times 6}]$, where the element $\mathbf{0}_{3 \times 6}$ correspond to the bias terms and the element $\mathbf{0}_{3 \times 12}$ could be removed if a reduced-dimensional filter gain is instead used, as in [12]. Based on the right-InEKF methodology [11], we obtain the submatrix $\tilde{\mathbf{H}}_{i,t_n}$ via $\tilde{\mathbf{H}}_{i,t_n} \boldsymbol{\xi}_{t_n} = -(\boldsymbol{\xi}_{t_n})^\wedge \mathbf{d}_{i,t_n}$ as $\tilde{\mathbf{H}}_{1,t_n} \triangleq [(\mathbf{R}_{t_n}^{\text{DRS}} [0, 0, 1]^T) \times, \mathbf{0}_{3 \times 9}]$ and $\tilde{\mathbf{H}}_{2,t_n} \triangleq [\mathbf{0}_{3 \times 6}, -\mathbf{I}_3, \mathbf{I}_3]$.

To compute $\mathbf{L}_{t_n}$, the linearized error update equation is obtained based on the update (17) as

$$\begin{bmatrix} \xi_{t_n}^+ \\ \zeta_{t_n}^+ \end{bmatrix} = (\mathbf{I} - \mathbf{L}_{t_n} \mathbf{H}_{t_n}) \begin{bmatrix} \xi_{t_n} \\ \zeta_{t_n} \end{bmatrix} + \mathbf{L}_{t_n} \begin{bmatrix} \bar{\mathbf{R}}_{t_n} \frac{\partial \mathbf{h}_{R,3}}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_{t_n}) \\ \mathbf{0}_{3 \times 1} \\ \bar{\mathbf{R}}_{t_n} \frac{\partial \mathbf{h}_p}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_{t_n}) \\ \mathbf{0}_{3 \times 1} \end{bmatrix} \mathbf{w}_{t_n}^q \quad (18)$$

with derivation given in Section 4 of supplementary material.

Then, applying the standard Kalman filtering methodology to this linear error update equation, we obtain the filter gain $\mathbf{L}_{t_n} = \mathbf{P}_{t_n} \mathbf{H}_{t_n}^T \mathbf{S}_{t_n}^{-1}$, where $\mathbf{S}_{t_n} = \mathbf{H}_{t_n} \mathbf{P}_{t_n} \mathbf{H}_{t_n}^T + \bar{\mathbf{N}}_{t_n}$, $\bar{\mathbf{N}}_{t_n} \triangleq \text{diag}(\bar{\mathbf{N}}_{1,t_n}, \bar{\mathbf{N}}_{2,t_n})$, $\bar{\mathbf{N}}_{1,t_n} \triangleq \bar{\mathbf{R}}_{t_n} \frac{\partial \mathbf{h}_{R,3}}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_{t_n}) \text{Cov}(\mathbf{w}_{t_n}^q) (\frac{\partial \mathbf{h}_{R,3}}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_{t_n}))^T \bar{\mathbf{R}}_{t_n}^T$, and $\bar{\mathbf{N}}_{2,t_n} \triangleq \bar{\mathbf{R}}_{t_n} \frac{\partial \mathbf{h}_p}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_{t_n}) \text{Cov}(\mathbf{w}_{t_n}^q) (\frac{\partial \mathbf{h}_p}{\partial \mathbf{q}_t}(\tilde{\mathbf{q}}_{t_n}))^T \bar{\mathbf{R}}_{t_n}^T$.

Remark 2 (Independence of True State): The linearized error update (18) is independent of the true states $\mathbf{X}_t$ and θ_t in the deterministic case. This is because both measurement models satisfy the right-invariant observation form with respect to $\mathbf{X}_t$ and are independent of θ_t , and because the update equation of $\bar{\mathbf{X}}_t$ is in the exponential form as prescribed by the InEKF methodology [11].

C. Discrete Process Model and Propagation Step

Without loss of generality and for simplicity, suppose that the foot-landing events and the updates do not coincide. Thus, the proposed filtering for the state jump focuses on estimate and covariance propagation without update. Except for the true contact point position $\mathbf{p}_t^c$, the rest of the true state is continuous across a foot landing, as explained in Section III.

1) Process Model: From the proposed jump dynamics in Section III, the stochastic jump dynamics of $\mathbf{X}_t$ can be approximately expressed as

$$\mathbf{X}_{t+} = \mathbf{X}_t \begin{bmatrix} \mathbf{I}_3 & [\mathbf{0}_{3 \times 1}, \mathbf{0}_{3 \times 1}, \mathbf{h}_c(\tilde{\mathbf{q}}_t)] \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 \end{bmatrix} - \mathbf{X}_t \begin{bmatrix} \frac{\partial \mathbf{h}_c}{\partial \mathbf{q}}(\tilde{\mathbf{q}}_t) \mathbf{w}_t^q \\ \mathbf{0}_{3 \times 1} \end{bmatrix}^\Delta \triangleq \Delta_{u_t}(\mathbf{X}_t) - \mathbf{X}_t(\mathbf{w}_t^\Delta)^\Delta \quad (19)$$

where the function $\Delta_{u_t}(\mathbf{X}_t)$ is the deterministic jump dynamics and the encoder data $\tilde{\mathbf{q}}_t$ serves as the input. As the biases are continuous under a jump event, $\theta_{t+} = \theta_t$ holds.

Remark 3 (Group Affine Property): The jump map Δ_{u_t} of the state $\mathbf{X}_t$ possesses the discrete-time group affine property defined in [21], and is independent of IMU biases θ_t . Thus, the jump dynamics of the error ξ_t is independent of the true state and is exactly linear. Moreover, from the expression of Δ_{u_t} in (20), we can see that Δ_{u_t} is a group action on $\text{SE}_3(3)$, under which the error ξ_t naturally does not change.

2) Error Equation: From (20), we obtain the dynamics of the logarithmic error ξ_t as $\xi_{t+} = \xi_t - \text{Ad}_{\bar{\mathbf{X}}} \mathbf{w}_t^\Delta$. Indeed, as analyzed in Remark 3, the error does not jump under Δ_{u_t} . Also, $\zeta_{t+} = \zeta_t$ holds since the IMU biases are continuous.

3) Propagation: Based on the deterministic portion of the jump model in (20), the propagations of the state estimate at a jump event are $\bar{\mathbf{X}}_{t+} = \Delta_{u_t}(\bar{\mathbf{X}}_t)$ and $\bar{\theta}_{t+} = \bar{\theta}_t$.

With the linear error equation of ξ and ζ across a jump, the propagation of the covariance matrix is expressed as $\mathbf{P}_{t+} = \mathbf{P}_t + \bar{\mathbf{Q}}_t^\Delta$, where $\bar{\mathbf{Q}}_t^\Delta = \begin{bmatrix} \text{Ad}_{\bar{\mathbf{X}}} \text{Cov}(\mathbf{w}_t^\Delta) \text{Ad}_{\bar{\mathbf{X}}}^T & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \end{bmatrix}$.

The complete algorithm of the proposed right-InEKF is summarized as Algorithm 1 in supplementary material.

Remark 4 (Imperfect InEKF): In the presence of IMU biases, the proposed filter is no longer a “perfect” InEKF in the sense that the group affine and invariant form properties no longer hold for continuous phases. Although the linear equation in (13) is no longer independent of the true state, it depends on the true state only through the bias terms, while the remaining part of the Jacobian matrix $\mathbf{A}_t$ is still independent of the true state. Also, the measurement models are still independent of the true states $\mathbf{X}_t$ and θ_t as highlighted in Remark 2. For these reasons, the linearization inaccuracy induced by the biases has a limited impact on the continuous-phase propagation and update. Thus, the “imperfect InEKF” with biases considered can still ensure rapid and accurate convergence under large errors, which is experimentally confirmed on DRS locomotion, as reported in Section VI.

V. OBSERVABILITY AND CONVERGENCE ANALYSIS

A. Observability Analysis for Continuous Phases

As measurement update is performed during continuous phases, we only analyze the continuous-phase observability.

Recall that the deterministic continuous-phase dynamics in (12) is group affine in the absence of IMU biases θ_t (see Remark 1). Also, recall that the measurement models in (15) and (16) are in the right-invariant observation form with respect to $\mathbf{X}_t$, regardless of the presence of biases (see Remark 2). Then, by [14, Theorem 20], the observability of $\mathbf{X}_t$ for the complete continuous-phase system, which has both $\mathbf{X}_t$ and θ_t as its states, is the same as that of the simplified continuous-phase system without IMU biases. Thus, by [11, Theorem 5], the local observability of $\mathbf{X}_t$ for the complete system can be determined by the couple $(\mathbf{A}, \mathbf{H})$ with $\mathbf{A}$ and $\mathbf{H}$ updated with bias-related terms removed (see Section 6 of supplementary material).

With Δt the duration of one propagation step, the discrete state transition matrix Φ is given by $\Phi = \text{expm}(\mathbf{A}_t \Delta t)$ [11] (see Section 6 of supplementary material for the expression of Φ). Then, from $\mathfrak{D} = [(\mathbf{H})^T, (\mathbf{H}\Phi)^T, (\mathbf{H}\Phi^2)^T, \dots]^T$, we have

$$\mathfrak{D} = \begin{bmatrix} (\mathbf{R}^{\text{DRS}}[0, 0, 1]^T)_{\times} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\mathbf{I}_3 & \mathbf{I}_3 \\ (\mathbf{R}^{\text{DRS}}[0, 0, 1]^T)_{\times} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ -\frac{1}{2}(\mathbf{g})_{\times} \Delta t^2 & -\mathbf{I}_3 \Delta t & -\mathbf{I}_3 & \mathbf{I}_3 \\ \dots & \dots & \dots & \dots \end{bmatrix}. \quad (20)$$

As the first two columns of $(\mathbf{g})_{\times}$ are linearly independent, the base roll and pitch angles are observable. Because all columns in the second column block of $\mathfrak{D}$ are linearly independent, the base velocity $\mathbf{v}_t$ is observable. Yet, as the last two column blocks are linearly dependant, the base position $\mathbf{p}_t$ and contact point position $\mathbf{p}_t^c$ are unobservable.

The third column of $(\mathbf{g})_{\times}$ is always zero, because only its z -component is nonzero. Then, if the surface is nonhorizontal,

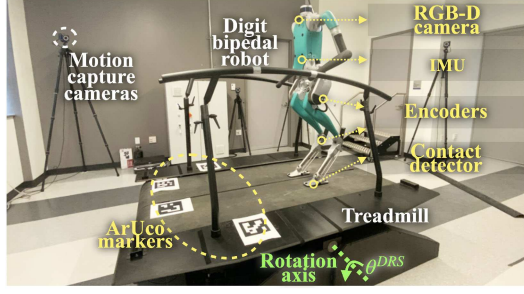


Fig. 3. Experimental setup that includes a Digit bipedal humanoid robot and a pitching treadmill (i.e., DRS).

(i.e., the third column of $(\mathbf{R}^{\text{DRS}}[0, 0, 1]^T)_{\times}$ is not all zero), the yaw will be observable; otherwise, it is unobservable.

From the expression of $\mathfrak{D}$, we also know that the contact velocity $\mathbf{v}_t^c$ does not affect observability, either measurement model ensures observable base roll and pitch, the proposed measurement in (15) renders base yaw observable and the previous measurement in (16) makes base velocity observable.

B. Convergence Property for the Hybrid Error System

The proposed convergence analysis for the hybrid error system is built upon previous analysis of the InEKF as a deterministic observer for systems without state-triggered jumps [11]. Different from the previous work, this section analyzes the effects of the jumps on the error convergence for the overall hybrid error system.

We first analyze the error evolution across the deterministic discrete jump of the system. Analyzing the state evolution across discrete, state-triggered jumps (e.g., foot-landing impacts) are typically complex [3]. Yet, since the jump map Δ_{u_t} is a group action, the error ξ_t does not jump under Δ_{u_t} despite the jump of the true state $\mathbf{X}_t$. Also, the bias error ζ_t is continuous across a jump event. Thus, the hybrid deterministic error dynamics is essentially continuous for all time, and its error convergence is equivalent to that of the deterministic continuous phases.

For continuous phases, the proposed filter meets the group affine condition and invariant observation form without biases, as discussed in Section IV. Thus, by the theory of InEKF [11], the proposed filter is locally asymptotically convergent for the observable variables of the deterministic continuous phases without biases. Accordingly, the local asymptotic convergence of the hybrid deterministic filter system is guaranteed in the absence of biases.

This analysis also supports the local asymptotic convergence of the existing InEKF [12] designed for static surface locomotion, because the jump model in [12] is a group action and its continuous-phase design also satisfies the group affine and invariant observation conditions without biases.

VI. EXPERIMENTS

A. Experimental Setup

The setup for experimental data collection (see Fig. 3) is as follows.

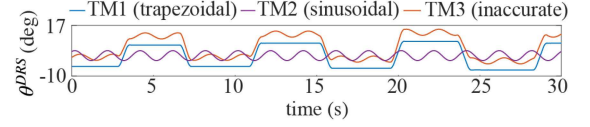


Fig. 4. Profiles (TM1)–(TM3) of the treadmill pitch angle $\theta^{\text{DRS}}(t)$.

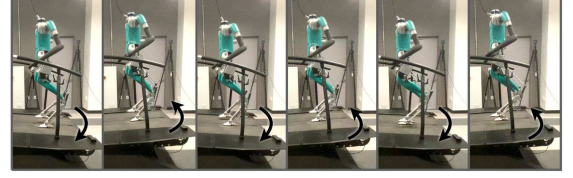


Fig. 5. Time-lapse figures of Digit walking on a rocking treadmill. The black arrow indicates the treadmill's direction of rotation.

1) **Treadmill (i.e., Tested DRS):** A split-belt Motek M-gait treadmill is used as a DRS. Its dimension is $2.3 \times 1.82 \times 0.5$ m. To emulate a rocking ship in sea waves, it performs a whole-body pitching motion without belt translation.

2) **Robot:** The Digit robot is 1.6-m tall, and each leg's kinematic chain used by the filter has 12 joints. Different robot movements are tested as (RM1) stepping and (RM2) standing. The robot is about 0.8-m away from the treadmill center.

3) **Treadmill motion Profiles:** To test filter performance under different DRS motions, two different profiles of the treadmill's pitch angle θ^{DRS} (see Fig. 3) are tested: a) (TM1), a nonperiodic trapezoidal wave $f_{\text{trap}}(t)$; and b) (TM2), a sine wave $2.5^\circ \sin(\pi t)$. Under (TM1) and (TM2), the maximum contact point speeds $\|\mathbf{v}_t^c\|$ are 0.41 and 0.11 m/s, respectively. To test the filter's robustness under surface motion inaccuracy, a fictitious profile is considered as (TM3) $\theta^{\text{DRS}}(t) = f_{\text{trap}}(t) + 5.1^\circ + 1.7^\circ \sin(\pi t)$ with the actual profile (TM1) used in experiments. Fig. 4 shows all profiles.

4) **Onboard Sensors Used:** Digit's onboard sensors used (see Fig. 3) are IMU, joint encoders, a RealSense RGB-D camera, and the robot's proprietary contact detector. The camera returns data at 15 Hz, and the remaining sensors stream data at the same rate within 60–90 Hz. Cortex motion capture cameras provide the ground truth. ArUco markers are attached to the treadmill, emulating the real-world scenario where legged robots that navigate within a DRS (e.g., a vessel at sea) can only see landmarks attached to the DRS, but not any landmarks on the Earth's ground. The markers are sensed by the camera to obtain the camera pose in the treadmill frame, which is then used to compute contact point velocity, as explained in the following section.

5) **Data Collection Cases:** Fig. 5 shows screenshots of experiments. The filter is simulated in MATLAB using four experimentally collected datasets under different robot and treadmill motions as follows.

- a) *Case A:* Combination of (RM1) and (TM1).
- b) *Case B:* Combination of (RM1) and (TM2).
- c) *Case C:* Combination of (RM2) and (TM1).

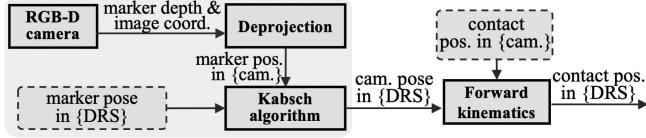


Fig. 6. Procedure of obtaining the 3-D contact point position in the DRS frame using the ArUco markers and the robot's onboard RGB-D camera.

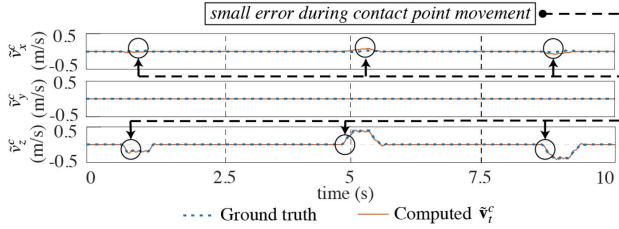


Fig. 7. Validation results of the proposed method for obtaining the contact point velocity $\tilde{\mathbf{v}}^c \triangleq [\tilde{v}_x^c, \tilde{v}_y^c, \tilde{v}_z^c]^T$ under case C. The velocity along the y -direction $\tilde{v}_y^c$ is zero because the treadmill does not move in that direction.

- d) *Case D*: Combination of (RM1) and (TM3), where the actual profile is (TM1), but the filter uses the inaccurate data (TM3).

The experiment video is online available at: <https://youtu.be/ScQIBFUSKzo>.

B. Filter Setting

1) *Filters Compared*: The proposed filter (denoted as “InEKF-DRS”) is compared with an InEKF, designed for locomotion on a static rigid surface [12] (denoted as “InEKF-SRS”). The InEKF-SRS models the deterministic contact point motion as $\dot{\mathbf{p}}^c = \mathbf{0}$, and uses the position measurement in (16) alone. It renders the base orientation (except for yaw) and velocity observable. It has realized substantially faster convergence under large errors during stationary surface locomotion, as compared with the EKF-based method [1]. Also, the proposed InEKF-DRS is compared with an EKF-based filter, which we formulate by augmenting the existing EKF designed for static surfaces [1] to explicitly handle nonstationary surfaces. Details of the augmentation and comparison results are in Section 8.3 of supplementary material. The rest of this section focuses on comparing InEKF-DRS and InEKF-SRS.

2) *Contact Point Velocity Computation*: The contact point velocity $\tilde{\mathbf{v}}^c$ serves as an input to the continuous-phase process model of the proposed InEKF-DRS. To obtain the contact point velocity $\tilde{\mathbf{v}}^c$ (see Fig. 6), we first obtain the camera pose in the DRS frame by processing the features of the ArUco markers in the camera images, which we then use to compute the 3-D contact point position in the DRS frame (${}^{\text{DRS}}\mathbf{p}^c$) through forward kinematics. Next, we estimate the contact point velocity $\tilde{\mathbf{v}}^c$ based on (4) using the known treadmill motion data. Details of this procedure are in Section 7 of supplementary material. Results in Fig. 7 validate the accuracy of the proposed contact point velocity sensing.

TABLE I
NOISE SD FOR InEKF-SRS AND InEKF-DRS

Measurement type	InEKF-SRS	InEKF-DRS
Linear acc. (m/s ²)	0.4	0.4
Angular vel. (rad/s)	0.01	0.01
Acc. bias (m/s ³)	0.001	0.001
Gyroscope bias (rad/s ²)	0.0001	0.0001
Contact vel. (m/s)	0.01	0.01
Encoder (°)	1	1
DRS orientation (°)	N/A	1

3) *Covariance Settings*: Table I gives the noise standard deviation (SD) of both filters. The SD for the accelerometer, gyroscope, and their corresponding biases are obtained from the manufacturer's manual with a slight adjustment for better performance. The SD for the encoder readings is adopted from the previous filter [12] designed for a similar robot. The SD for the contact-point velocity and orientation-based measurement are tuned for a reasonable performance. The initial value of the covariance $\mathbf{P}$ is set as an identity matrix.

4) *Initial Estimation Errors*: For a fair comparison, the two filters are simulated under the same large range of initial estimation errors. The initial velocity and orientation errors in each direction are uniformly distributed within $[-1.5, 1.5]$ m/s and $[-1, 1]$ rad, respectively.

C. Computation Time Comparison

In MATLAB, both filters take less than 1 ms to compute one estimation cycle (i.e., one propagation and one update step), confirming their validity for real-time estimation.

D. Convergence Rate and Yaw Observability Comparison

Fig. 8(a) shows the estimation results of InEKF-DRS (proposed) and InEKF-SRS under case A where the treadmill stays at a pitch angle of -8° for approximately 2.8 s, and then begins to pitch until reaching $+8^\circ$ in 0.5 s.

Both filters drive the error of base roll, pitch, and velocity closer to zero, indicating their observability, as predicted in Section V and previous work [1], [12]. In terms of the convergence rates for these variables, Fig. 8(a) shows that the proposed InEKF-DRS is faster than InEKF-SRS, driving the error close to zero within 1 s. This is because InEKF-DRS considers the surface motion and has an additional measurement (15) that corrects estimates.

Under InEKF-DRS, the yaw estimate converges close to the ground truth in approximately 3 s, which supports the observability analysis in Section V that the yaw angle is observable if the DRS/treadmill is not horizontal. Yet, the yaw convergence is slower than pitch and roll, possibly because both observations in (15) and (16) help correct the roll and pitch estimates, whereas only the former corrects the yaw estimate. Finally, as

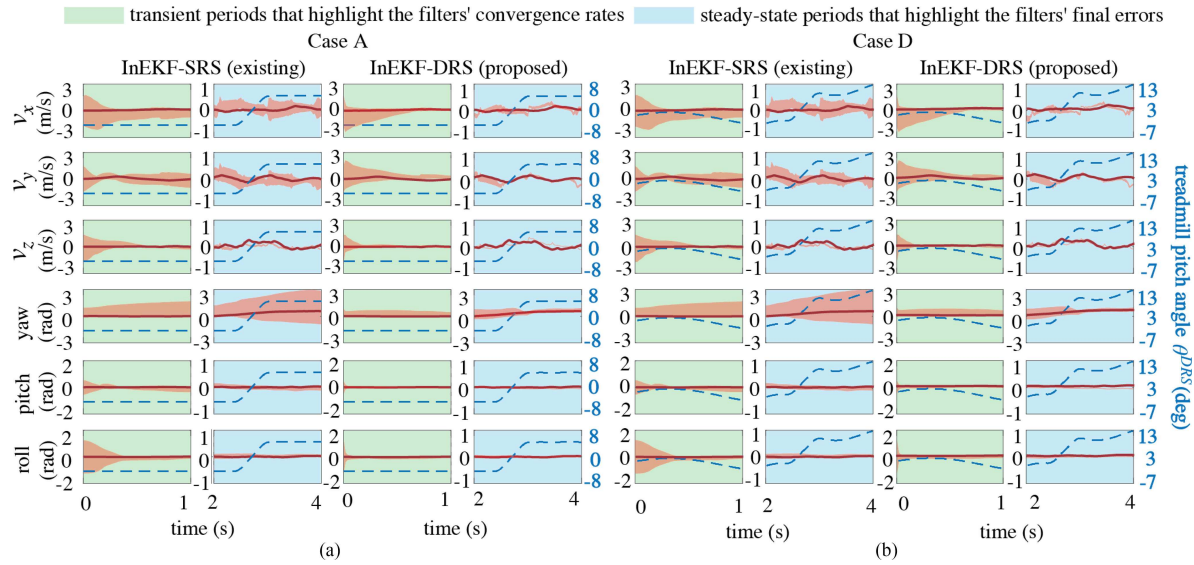


Fig. 8. Base velocity and orientation estimation results of the filters InEKF-DRS and InEKF-SRS for cases A and D. The red-shaded area indicates the range of the state estimates for 10 runs. The red solid line is the ground truth. The blue-dashed line is the treadmill angle. (a) Case A. (b) Case D.

TABLE II
RMS ERROR COMPARISON UNDER CASE A

State variables	InEKF-SRS	InEKF-DRS
v_x (m/s)	0.3320	0.2051
v_y (m/s)	0.2488	0.1955
v_z (m/s)	0.1438	0.1025
yaw (rad)	0.9294	0.2516
pitch (rad)	0.0897	0.0413
roll (rad)	0.1365	0.0318

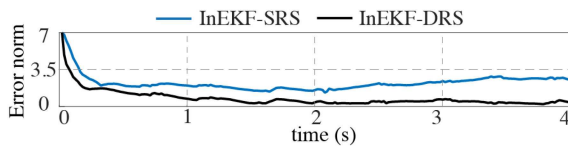


Fig. 9. Accuracy comparison of InEKF-SRS and InEKF-DRS (proposed) for the estimation of base velocity and roll and pitch angles under case A.

previously revealed [1], the yaw error divergence under InEKF-SRS confirms that the base yaw is indeed nonobservable with InEKF-SRS.

E. Accuracy Comparison

Table II gives the comparison of the root mean square (rms) estimation errors for base orientation (including yaw) and velocity under case A. Fig. 9 shows the corresponding time evolution of the errors for base roll, pitch, and velocity under case A. The table and the figure show that the proposed InEKF-DRS is more accurate in velocity and orientation estimation compared with InEKF-SRS.

F. Performance Under Different DRS and Robot Movements

Fig. 1(a) and (b) in supplementary material show the estimation results of the two filters under case B (where the treadmill motion is different from case A) and case C (where the robot stands on the treadmill instead of walking as in case A), respectively. The plots show that the performance comparison of the two filters under cases B and C are similar to case A [i.e., Fig. 8(a)], in terms of convergence rate and accuracy, indicating that the InEKF-DRS can effectively handle different DRS and robot movements.

Comparing the yaw estimate under the InEKF-DRS in cases A–C, we notice that the yaw estimate in case C converges faster than cases A and B. In case C, the treadmill remains horizontal for the first 10 s, during which the yaw estimate does not converge. Yet, once the treadmill begins to rock at $t = 10$ s, the yaw estimate converges close to the ground truth within 1 s, whereas it takes about 3 s for the yaw estimate to enter a similar neighborhood under cases A and B. This might be due to the fact that in case C, by the time the treadmill begins to pitch, the estimates of the rest observable state are already sufficiently accurate, making the yaw error correction faster than cases A and B.

G. Robustness Assessment

Results from cases A [see Fig. 8(a)] and D [see Fig. 8(b)] confirm the robustness of the proposed InEKF-DRS under inaccurate surface pose knowledge. Case D emulates the scenario where the DRSs motion monitoring system fails to provide accurate DRS pose. Fig. 8(a) and (b) show that the filter performance (e.g., convergence rate, accuracy, and yaw observability) under case D is similar to that under case A. Specifically, the velocity estimate under InEKF-DRS converges to the ground truth in all

directions within 1 s. Also, the orientation convergence rates are similar, the roll and pitch estimates converge close to the ground truth within 0.3 s, the yaw angle converges within 3 s. Longer periods (10–30 s) of estimation results for cases A–D are shown and discussed in supplementary material.

VII. DISCUSSION

This article has designed an InEKF that estimates the orientation and velocity of a bipedal robot that walks on a DRS with a known relatively significant motion, by fusing the known surface pose and the leg, visual, and inertial odometries. Similar to the InEKF [12] and EKF [1] designed for stationary surfaces, the filter uses the IMU motion dynamics as the process model and the 3-D contact point position and leg kinematics to form a measurement model. Different from the previous work, the proposed contact-point process model does not assume that the contact point is static, but instead explicitly considers its movement in the world. Also, we have introduced a right-invariant measurement model based on the rotational kinematic relationship between the surface and support foot. Thanks to these features, the filter ensures accurate estimation under relatively large surface motion and estimation errors, as given by the rms errors in Table II and the state trajectories in Figs. 8 and 9.

The proposed filter is suitable for a DRS with a relatively accurately known surface pose profile, but may not be effective under overly inaccurate or unknown profiles. One potential solution is to extend this filter to estimate the surface pose, by constructing a matrix Lie group that includes the surface pose in the state and formulating an InEKF with fundamental benefits.

This article also assumes the robot's feet do not persistently and significantly slip on the surface. When the surface is slippery [22], the support foot may move relative to the surface, causing discrepancy between the actual robot movement and the models. Yet, the proposed method could be extended to address slippage during DRS locomotion by incorporating existing techniques [2], [23], such as using an RGB-D sensor to measure the base velocity [23].

VIII. CONCLUSION

This article introduced a right-InEKF for bipedal humanoid walking on a moving surface. The filter design explicitly considered the known surface movement and hybrid robot behaviors while enjoying the fundamental benefits of satisfying the attractive group-affine condition and invariant observation form in the absence of IMU biases. Observability analysis for the continuous locomotion phases showed that the robot's base velocity and roll and pitch angles are observable, and the base yaw angle becomes observable when the DRS is not horizontal. Stability analysis proved the asymptotic error convergence of these observable states for the hybrid deterministic system. Experimental results of humanoid walking on a pitching treadmill validated the enhanced accuracy and convergence rate of the proposed filter over existing work, in the presence of large estimation errors and moderate DRS movement.

ACKNOWLEDGMENT

The authors would like to thank M. Ghaffari and A. Saccon for constructive feedback on theoretical derivation.

REFERENCES

- [1] M. Bloesch *et al.*, "State estimation for legged robots-consistent fusion of leg kinematics and IMU," *Robotics*, vol. 17, pp. 17–24, 2013.
- [2] J.-H. Kim *et al.*, "Legged robot state estimation with dynamic contact event information," *IEEE Trans. Robot. Autom.*, vol. 6, no. 4, pp. 6733–6740, Oct. 2021.
- [3] A. Iqbal, Y. Gao, and Y. Gu, "Provably stabilizing controllers for quadrupedal robot locomotion on dynamic rigid platforms," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 4, pp. 2035–2044, Aug. 2020.
- [4] A. Iqbal, Z. Mao, and Y. Gu, "Modeling, analysis, and control of slip running on dynamic platforms," *Lett. Dyn. Syst. Control*, vol. 1, no. 2, 2021, Art. no. 021007.
- [5] Y. Gao and Y. Gu, "Global-position tracking control of multi-domain planar bipedal robotic walking," in *Proc. ASME Dyn. Syst. Control Conf.*, 2019, p. V001T03A009, vol. 59148.
- [6] Y. Gu, B. Yao, and C. S. G. Lee, "Exponential stabilization of fully actuated planar bipedal robotic walking with global position tracking capabilities," *J. Dyn. Syst. Meas. Control*, vol. 140, no. 5, 2018, Art. no. 051008.
- [7] Y. Gao and Y. Gu, "Global-position tracking control of a fully actuated NAO bipedal walking robot," in *Proc. Amer. Control Conf.*, 2019, pp. 4596–4601.
- [8] P.-C. Lin, H. Komsuoglu, and D. E. Koditschek, "Sensor data fusion for body state estimation in a hexapod robot with dynamical gaits," *IEEE Trans. Robot.*, vol. 22, no. 5, pp. 932–943, Oct. 2006.
- [9] M. F. Fallon, M. Antone, N. Roy, and S. Teller, "Drift-free humanoid state estimation fusing kinematic, inertial and LIDAR sensing," in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 112–119.
- [10] P. Ramadoss, G. Romualdi, S. Dafarra, F. J. A. Chavez, S. Traversaro, and D. Pucci, "DILIGENT-KIO: A proprioceptive base estimator for humanoid robots using extended Kalman filtering on matrix Lie groups," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2021, pp. 2904–2910.
- [11] A. Barrau and S. Bonnabel, "The invariant extended Kalman filter as a stable observer," *IEEE Trans. Autom. Control*, vol. 62, no. 4, pp. 1797–1812, Apr. 2017.
- [12] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended Kalman filtering for robot state estimation," *Int. J. Robot. Res.*, vol. 39, no. 4, pp. 402–430, 2020.
- [13] T.-Y. Lin, R. Zhang, J. Yu, and M. Ghaffari, "Deep multi-modal contact estimation for invariant observer design on quadruped robots," 2021, *arXiv:2106.15713*.
- [14] A. Barrau, "Non-linear state error based extended Kalman filters with applications to navigation," Ph.D. dissertation, Mines ParisTech, Paris, France, 2015.
- [15] S. Küchler, J. K. Eberharter, K. Langer, K. Schneider, and O. Sawodny, "Heave motion estimation of a vessel using acceleration measurements," *IFAC Proc. World Congr.*, vol. 44, no. 1, pp. 14742–14747, 2011.
- [16] K. A. Hamed, A. D. Ames, and R. D. Gregg, "Observer-based feedback controllers for exponential stabilization of hybrid periodic orbits: Application to underactuated bipedal walking," in *Proc. Amer. Control Conf.*, 2018, pp. 1438–1445.
- [17] N. J. Kong, J. J. Payne, G. Council, and A. M. Johnson, "The salted Kalman filter: Kalman filtering on hybrid dynamical systems," *Automatica*, vol. 131, 2021, Art. no. 109752.
- [18] Y. Gao, C. Yuan, and Y. Gu, "Invariant extended Kalman filtering for hybrid models of bipedal robot walking," *IFAC PapersOnLine*, vol. 54, pp. 290–297, 2021.
- [19] J. Sola, J. Deray, and D. Atchuthan, "A micro Lie theory for state estimation in robotics," 2018, *arXiv:1812.01537*.
- [20] *VectorNav*, Accessed: Aug. 8, 2021. [Online]. Available: <https://www.vectornav.com/>
- [21] A. Barrau and S. Bonnabel, "Invariant Kalman filtering," *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 1, pp. 237–257, 2018.
- [22] M. Trkov, K. Chen, J. Yi, and T. Liu, "Inertial sensor-based slip detection in human walking," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 3, pp. 1399–1411, Jul. 2019.
- [23] S. Teng, M. W. Mueller, and K. Sreenath, "Legged robot state estimation in slippery environments using invariant extended Kalman filter with velocity update," in *Proc. IEEE Int. Conf. Rob. Autom.*, 2021, pp. 3104–3110.



Yuan Gao (Student Member, IEEE) received the B.S. degree from China Jiliang University, Hangzhou, China, in 2014, and the M.S. degree from Arizona State University, Tempe, AZ, USA, in 2016, all in mechanical engineering. He is currently working toward the Ph.D. degree in controls and robotics with the Department of Mechanical Engineering, University of Massachusetts Lowell, Lowell, MA, USA.



Yan Gu (Member, IEEE) received the B.S. degree from Zhejiang University, Hangzhou, China, in 2011, and the Ph.D. degree from Purdue University, West Lafayette, IN, USA, in 2017, all in mechanical engineering.

In 2022, she joined the Faculty of the School of Mechanical Engineering, Purdue University. Prior to this, she was with the Department of Mechanical Engineering, University of Massachusetts Lowell. Her research interests include nonlinear control, hybrid systems, legged locomotion, and wearable robots.

Dr. Gu was the recipient of the NSF CAREER Award in 2021.



Chengzhi Yuan (Member, IEEE) received the B.S. and M.S. degrees in control science and engineering from the South China University of Technology, Guangzhou, China, in 2009 and 2012, respectively, and the Ph.D. degree in mechanical engineering from North Carolina State University, Raleigh, NC, USA, in 2016.

He is currently an Assistant Professor with Mechanical, Industrial, and Systems Engineering Department, and the Director of the Intelligent Control and Robotics Laboratory, University of Rhode Island, Kingston, RI, USA. His research interests include adaptive learning and control, hybrid systems, and multirobot-distributed control.

His research interests include adaptive learning and control, hybrid systems, and multirobot-distributed control.