Analytical solution to a time-varying LIP model for quadrupedal walking on a vertically oscillating surface[☆]

Amir Iqbal [a], Sushant Veer [b], Yan Gu [c],*

[a] Department of Mechanical Engineering, University of Massachusetts Lowell, One University Avenue, Lowell, 01854, MA, USA
[b] NVIDIA Research, 2788 San Tomas Expressway, Santa Clara, 95051, CA, USA
[c] School of Mechanical Engineering, Purdue University, 585 Purdue Mall, West Lafayette, 47907, IN, USA

ARTICLE INFO

ABSTRACT

This paper introduces an analytically tractable and computationally efficient model for legged robot dynamics during locomotion on a dynamic rigid surface (DRS), along with an approximate analytical solution and a real-time walking pattern generator synthesized based on the model and solution. By relaxing the static-surface assumption, we extend the classical, *time-invariant* linear inverted pendulum (LIP) model for legged locomotion on a static surface to dynamic-surface locomotion, resulting in a *time-varying* LIP model termed as "DRS-LIP". Sufficient and necessary stability conditions of the time-varying DRS-LIP model are obtained based on the Floquet theory. This model is also transformed into Mathieu's equation to derive an approximate analytical solution that provides reasonable accuracy with a relatively low computational cost. Using the extended model and its solution, a walking pattern generator is developed to efficiently plan physically feasible trajectories for quadrupedal walking on a vertically oscillating surface. Finally, simulations and hardware experiments from a Laikago quadrupedal robot walking on a pitching treadmill (with a maximum vertical acceleration of 1 m/s^2) confirm the accuracy and efficiency of the proposed analytical solution, as well as the efficiency, feasibility, and robustness of the pattern generator, under various surface motions and gait parameters.

1. Introduction

Legged robots have the potential to traverse various surfaces that are prohibitively challenging for tracked or wheeled robots, including stationary (uneven or discrete) surfaces [1–6] and dynamic rigid surfaces (i.e., rigid surfaces moving in the inertial frame) [7,8]. Legged robots capable of reliably traversing a dynamic rigid surface (DRS) can aid in various critical real-world applications such as firefighting, maintenance, inspection, surveillance, and disinfection on ships and public transit vehicles. The objective of this study is to model and analyze the essential dynamic behaviors of a legged robot that walks on a vertically oscillating DRS, and to exploit the analytical results for efficient and feasible pattern generation of quadrupedal walking. There has been ample work on reduced-order modeling and trajectory generation of legged locomotion on stationary surfaces, but not for DRS. This paper constitutes one of the first attempts to build a reduced-order model and leverage such a model in motion planning for DRS locomotion. However, reduced-order modeling and analysis of DRS locomotion are fundamentally complex due to the nonlinear robot dynamics [9–11] and the time-varying movement of surface-foot contact points [7,12].

1.1. Reduced-order models of legged locomotion on stationary or dynamic surfaces

A reduced-order dynamics model of legged locomotion captures the robot's essential dynamic behaviors [13]. One of the most widely studied reduced-order models for stationary surface walking is the linear inverted pendulum (LIP) model [14–17], which approximates a legged robot as a point mass atop a massless leg.

Due to its simplicity, the LIP model is analytically tractable and can provide physical insights into the essential robot dynamics. It also explicitly reveals the simplified relationship between the center of pressure (CoP), which can be used to infer the feasibility of ground contact forces (i.e., no foot rolling about any edge of the region of contact), and the center of mass (CoM). Although the LIP model is not as accurate as a full-order model [18] in capturing the complete full-body robot dynamics, many of today's walking robots can be relatively accurately modeled as an LIP for planning and control [3,19], since they typically have a heavy upper body and lightweight legs. Thus, the LIP can serve as a basis of walking pattern generation to ensure planning efficiency and feasibility, as reviewed later.

The classical LIP model [14] for static surfaces has been extended to various complex scenarios such as foot slippage [20], a varying CoM height [15], CoM motions on 3-D planes [21], nontrivial centroidal angular momentum [19,22], balancing on a seasaw [23], and hybrid robot dynamics [16,24]. Due to their static-surface assumption, they may not be suitable for DRSes with significant motions.

For locomotion on a DRS whose motions are affected by the robot (e.g., passive and relatively lightweight surfaces), several reduced-order robot dynamics models have been recently introduced, including an extended LIP [25], centroidal dynamics [26], and rimless-wheel models [27]. Still, it is unclear how to expand these models to DRSes whose motion cannot be affected by the robot (e.g., trains, vessels, and elevators). For such substantially heavy or rigidly actuated DRSes, the effects of the DRS motion on a spring-loaded inverted pendulum model have been numerically studied [28]. However, the stability conditions and analytical solution of the model remain unknown.

Beyond the scope of legged locomotion, the modeling and analysis of an inverted pendulum with a vertically oscillating support, i.e., the Kapitza pendulum [29], is a classical physics problem. The Kapitza pendulum has an intriguing property that under high-frequency support oscillations, the pendulum's upper equilibrium becomes stable whereas its lower one is unstable. Yet, it is an open question whether and when the Kapitza pendulum is a reasonable approximation of DRS locomotion. Furthermore, the motion frequencies of real-world DRSes (e.g., vessels [30]) are commonly too low to meet the conditions underlying the Kapitza pendulum.

1.2. Walking pattern generation based on pendulum models

Since the LIP model represents the low-dimensional CoM dynamics of robot walking, it has been utilized to efficiently plan physically feasible walking motions on a static surface. Given the user-specified footstep and CoP positions, the exact closed-form analytical solution of the classical LIP [17,31,32] has been used to enable real-time planning of feasible CoM trajectories for static-surface walking. This analytical solution has been augmented with the discrete-time jump of the CoM position (relative to the CoP) at a foot-landing event, which is then used to derive the desired footstep locations that provably stabilizes the hybrid LIP model [16,24]. Recently, the exact capturability conditions of a LIP model with a time-varying CoM height have been derived based on the closed-form solution of the model's time-varying damping function at a robot's desired final CoM state [15]. These conditions are then used to plan the desired CoM and CoP trajectories with provable capturability guarantees. As reviewed earlier, the underlying LIP models of these planners assume a stationary walking surface, and therefore the planner may not be directly used for DRS locomotion when the surface motion is significant.

1.3. Contributions

This study focuses on addressing the open questions in reduced-order dynamic modeling and walking pattern generation for locomotion on DRS, as discussed in Sections 1.1 and 1.2. Motivated by these research needs, this study aims to theoretically extend the classical LIP model [14] from stationary surfaces to substantially heavy or rigidly actuated DRSes (e.g., ships), introduce an analytical approximate solution to the extended LIP model (termed as "DRS-LIP"), and develop and experimentally validate a real-time walking pattern generator that uses the proposed solution to ensure planning efficiency and feasibility. A preliminary version of this work appeared in [33] where we derived the DRS-LIP model. The new, substantial contributions of this study are:

(a) Forming the analytical approximate solution of the DRS-LIP under a vertical, sinusoidal DRS motion.

(b) Providing the sufficient and necessary stability conditions of the DRS-LIP model based on the Floquet theory, and performing stability analysis under common ranges of surface motions and gait parameters.
(c) Designing a hierarchical walking pattern generator that utilizes the proposed analytical solution to efficiently plan feasible robot motions.
(d) Validating the accuracy and computational efficiency of the proposed analytical approximate solution through comparison with a highly accurate numerical solution.
(e) Demonstrating the planner efficiency, feasibility, and robustness through both realistic PyBullet simulations and hardware experiments under various surface movements, gait parameters, and uncertainties.

The paper is organized as follows. Section 2 introduces the derivation of the proposed reduced-order model of locomotion on DRS. Section 3 presents the analytical approximate solution of the proposed model under a vertical, sinusoidal surface motion. Section 4 explains the stability condition and analysis of the proposed model. Section 5 develops an efficient walking pattern generator based on the analytical results. Section 6 reports the validation outcomes of the proposed analytical solution and planner. Section 7 discusses the capabilities and limitations of the proposed methods. Finally, Section 8 provides the concluding remarks.

1.4. Abbreviations and notations

Abbreviation	Description
CoM	Center of mass.
CoP	Center of pressure.
DRS	Dynamic rigid surface.
LIP	Linear inverted pendulum.
$Re(\cdot)$	Real part of a number $(\cdot)$.

2. Reduced-order model of DRS locomotion

This section introduces an analytically tractable and computationally efficient reduced-order model that captures the essential robot dynamics associated with legged walking on a DRS. The model is derived by extending the classical LIP model [14] from a static surface to a DRS, and is called "DRS-LIP".

Since today's legged robots typically have a heavy upper body and lightweight legs, their CoM dynamics can be approximately described by a LIP, i.e., a point mass atop a massless leg [14], under the following assumption:

(A1) The robot's rate of whole-body angular momentum about the CoM is negligible.

Assumption (A1) is reasonable for real-world locomotion because the robot's trunk is typically controlled to maintain a steady orientation for housing sensors (e.g., cameras).

In this study, we use a 3-D LIP to capture the essential dynamics of a 3-D legged robot walking on a DRS, as shown in Fig. 1. The point mass and support point S in Fig. 1 correspond to the robot's CoM and CoP, respectively.

Let

$$\mathbf{r}_{wc} = [x_{wc}, \ y_{wc}, \ z_{wc}]^T \tag{1}$$

and

$$\mathbf{r}_{ws} = [x_{ws}, \ y_{ws}, \ z_{ws}]^T \tag{2}$$

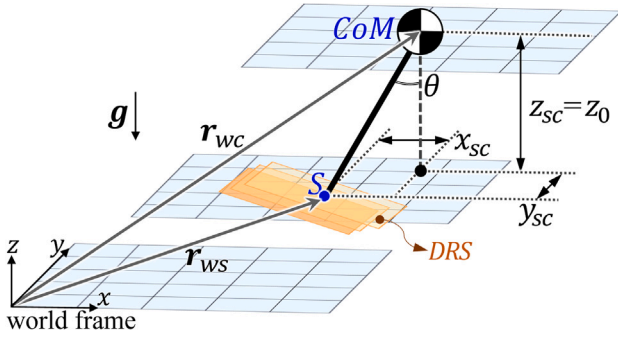respectively denote the positions of the CoM and point S in the world frame.

Fig. 1. Illustration of the proposed DRS-LIP model. All three grid planes are horizontal. The top and middle ones pass through the CoM and the leg's far end S, respectively. The bottom one is fixed to the world frame.

Then, the CoM position relative to point S, denoted as $\mathbf{r}_{sc}$, is defined as:

$$\mathbf{r}_{sc} = \mathbf{r}_{wc} - \mathbf{r}_{ws} =: [x_{sc}, \ y_{sc}, \ z_{sc}]^T. \tag{3}$$

The CoM dynamics during DRS locomotion can be obtained as:

$$\ddot{x}_{wc} = \frac{f_a x_{sc}}{mr} \sin\theta,$$
$$\ddot{y}_{wc} = \frac{f_a y_{sc}}{mr} \sin\theta, \quad \text{and} \tag{4}$$
$$\ddot{z}_{wc} = \frac{f_a}{m} \cos\theta - g,$$

where m is the robot's total mass, θ is the angle of the position vector $\mathbf{r}_{sc}$ relative to the vertical axis, g is the norm of the gravitational acceleration, r is the projected length of $\mathbf{r}_{sc}$ on the horizontal plane, and f_a is the norm of the ground contact force pointing from point S to the CoM.

2.1. DRS-LIP under a general vertical surface motion

We consider the following assumption on the vertical distance z_{sc} between the CoM and point S (see Fig. 1):

(A2) The CoM maintains a constant height z_0 above the support point S; that is,

$$z_{sc} = z_0.$$

This assumption is analogous to the simplifying assumption of the classical LIP model that the point-mass height over the stationary surface is constant [14].

Under assumption (A2), the relationships

$$\dot{z}_{wc} = \dot{z}_{ws} \quad \text{and} \quad \ddot{z}_{wc} = \ddot{z}_{ws}$$

hold, and thus the norm of the axial force, f_a, becomes

$$f_a = \frac{m(\ddot{z}_{ws} + g)}{\cos\theta}. \tag{5}$$

Accordingly, the horizontal components of the LIP dynamics in Eq. (4) become:

$$\ddot{x}_{wc} = (\ddot{z}_{ws} + g)\frac{x_{sc}}{z_0} \quad \text{and}$$
$$\ddot{y}_{wc} = (\ddot{z}_{ws} + g)\frac{y_{sc}}{z_0}. \tag{6}$$

From Eq. (3), we know $\ddot{x}_{wc} = \ddot{x}_{ws} + \ddot{x}_{sc}$ and $\ddot{y}_{wc} = \ddot{y}_{ws} + \ddot{y}_{sc}$. Substituting them into Eq. (6) yields the following horizontal LIP dynamics:

$$\ddot{x}_{sc} - \frac{\ddot{z}_{ws} + g}{z_0}x_{sc} = -\ddot{x}_{ws} \quad \text{and}$$
$$\ddot{y}_{sc} - \frac{\ddot{z}_{ws} + g}{z_0}y_{sc} = -\ddot{y}_{ws}. \tag{7}$$

Note that the DRS' acceleration at S equals the acceleration of point S, i.e., $(\ddot{x}_{ws}, \ \ddot{y}_{ws}, \ \ddot{z}_{ws})$ in Eq. (7), when there is no slippage between the support point S and the surface.

To account for the effects of the DRS motion on the LIP dynamics in Eq. (7), we choose to treat the DRS acceleration $(\ddot{x}_{ws}, \ddot{y}_{ws}, \ddot{z}_{ws})$ as explicit time functions, instead of building a separate dynamics model for the DRS [23,26,27]. The rationale of this design choice is twofold. First, for real-world rigidly actuated and/or heavy DRSes such as vessels, the physical interaction between a robot and such a DRS has a negligible effect on the dynamics of a DRS. Second, in real-world applications, the time profiles of the DRS motion are commonly sensed, estimated, and predicted by real-time monitoring systems [34].

With the surface motion modeled as time functions, the LIP dynamics in Eq. (7) are linear, nonhomogeneous, and time-varying. Since DRSes, such as cruising ships in regular sea waves, have relatively small horizontal acceleration compared with vertical acceleration [12,35,36], we assume:

(A3) The horizontal accelerations of point S (i.e., $\ddot{x}_{ws}$ and $\ddot{y}_{ws}$) are negligible.

Under assumption (A3), the forcing terms in Eq. (7) (i.e., $-\ddot{x}_{ws}$ and $-\ddot{y}_{ws}$) vanish, and the horizontal LIP dynamics in Eq. (7) become linear, time-varying, and homogeneous:

$$\ddot{x}_{sc} - \frac{\ddot{z}_{ws} + g}{z_0}x_{sc} = 0 \quad \text{and}$$
$$\ddot{y}_{sc} - \frac{\ddot{z}_{ws} + g}{z_0}y_{sc} = 0. \tag{8}$$

Recall that the vertical CoM position relative to the support point S, z_{sc}, is constant under assumption (A2), and given by:

$$z_{sc} = z_0. \tag{9}$$

Remark 1 (*DRS-LIP*). The LIP model in Eqs. (8) and (9) describe the simplified dynamics of DRS walking under assumptions (A1)-(A3), i.e., a DRS-LIP model for a vertically moving surface.

2.2. DRS-LIP under a vertical sinusoidal surface motion

A real-world DRS, such as a vessel in regular sea waves, typically exhibits a vertical, sinusoidal motion with a constant amplitude and frequency [36]. Thus, we focus on such motions for further analysis of the DRS-LIP model in Eqs. (8) and (9).

Under a vertical, sinusoidal surface motion, the vertical acceleration $\ddot{z}_{ws}$ of point S is sinusoidal, and Eq. (8) becomes the well-known Mathieu's equation [37], as explained next.

For generality, the vertical sinusoidal motion of the DRS at the surface-foot contact point can be expressed as:

$$z_{ws} = A\sin\omega t, \tag{10}$$

where the real scalar parameters A and ω are the amplitude and frequency of the vertical surface motion, respectively.

Then, the surface acceleration $\ddot{z}_{ws}$ at the support point is $\ddot{z}_{ws} := -A\omega^2 \sin\omega t$, with which the horizontal CoM dynamics in Eq. (8) becomes:

$$\ddot{x}_{sc} - \frac{g - A\omega^2 \sin\omega t}{z_0}x_{sc} = 0 \quad \text{and}$$
$$\ddot{y}_{sc} - \frac{g - A\omega^2 \sin\omega t}{z_0}y_{sc} = 0. \tag{11}$$

In Eq. (11), the two equations of motion in the x- and y-directions are decoupled and share the same structure. Therefore, their solutions have the same form. For brevity, we focus on the DRS-LIP model in the x-direction for solution derivation and stability analysis in the rest of this paper.

With a new time variable defined as

$$\tau := \frac{\pi + 2\omega t}{4}, \tag{12}$$

the DRS-LIP dynamics in Eq. (11) can be transformed into the standard Mathieu's equation as:

$$\frac{d^2 x_{sc}}{d\tau^2} + (c_0 - 2c_1 \cos 2\tau) x_{sc} = 0, \tag{13}$$

where the real scalar coefficients c_0 and c_1 are defined as

$$c_0 := -\frac{4g}{\omega^2 z_0} \quad \text{and} \quad c_1 := \frac{2A}{z_0}.$$

3. Approximate analytical solution

This section introduces a reasonably accurate and computationally efficient approximate analytical solution of the proposed DRS-LIP dynamics model under a vertical, sinusoidal surface motion.

3.1. Exact analytical solution

The DRS-LIP model in Eq. (13) generally does not have an exact, closed-form analytical solution. One straightforward approach to derive an approximate analytical solution is to utilize the fundamental solution matrix based on the Floquet theory [37]. Alternatively, we choose to exploit the existing analytical results of the well-studied Mathieu's equation to obtain a more computationally efficient solution.

There are various existing analytical approximate solutions of Mathieu's equation, including periodic solutions [38] and those expressed through power series [37]. In this study, we adopt the exact analytical solution from [39] because of its generality and accuracy:

Theorem 1 (*Exact Solution of Mathieu's Equation*). *The exact, general (periodic or non-periodic) analytical solution of Mathieu's equation in Eq. (13) is as follows:*

$$x_{sc}(\tau) = \alpha_1 e^{\mu\tau} \sum_{n=-\infty}^{\infty} C_{2n} e^{i2n\tau} + \alpha_2 e^{-\mu\tau} \sum_{n=-\infty}^{\infty} C_{2n} e^{-i2n\tau}. \tag{14}$$

Here, μ is the characteristic exponent of Eq. (13), α_1 and α_2 are real scalar coefficients, n is an integer, i is a unit imaginary number, and C_{2n}'s are complex scalar coefficients.

The proof of Theorem 1 is omitted for brevity, which can be readily obtained based on derivations in [39].

Despite its generality and exact accuracy, the analytical solution in Eq. (14) may demand an overly high computational load for real-time trajectory planning. Therefore, we use the exact solution to obtain an approximate solution that is reasonably accurate with a low computational cost.

To compute such an approximate solution, we need to determine the number of terms to keep in the approximate solution as well as the values of the parameters μ, α_1, α_2, and C_{2n}'s, which is explained in the rest of this section.

3.2. Recurrence relationship between characteristic exponent μ and solution parameters β_n's

To compute the characteristic exponent μ for obtaining the approximate solution, we first derive the recurrence relationship between μ and the solution parameters β_n's.

The solution of Mathieu's equation can be assumed as [40]:

$$x_{sc}(\tau) = e^{\mu\tau} \sum_{n=-\infty}^{\infty} C_{2n} e^{i2n\tau}. \tag{15}$$

Substituting Eq. (15) into Mathieu's equation in Eq. (13), we obtain:

$$\sum_{n=-\infty}^{\infty} [(\mu^2 + (i2n)^2 + 4i\mu n)C_{2n}$$
$$+ (c_0 - 2c_1 \cos 2\tau)C_{2n}] e^{(i2n+\mu)\tau} = 0.$$

With $\cos(2\tau) = \frac{e^{i2\tau} + e^{-i2\tau}}{2}$, this equation becomes:

$$\sum_{n=-\infty}^{\infty} \Big[\big(\mu^2 - (2n)^2 + 2(i\mu)(2n) + c_0\big)C_{2n}$$
$$- c_1(e^{i2\tau} + e^{-i2\tau})C_{2n}\Big] e^{(i2n+\mu)\tau} = 0, \tag{16}$$

which can be further rearranged as:

$$\sum_{n=-\infty}^{\infty} \Big[\big((i\mu)^2 + (2n)^2 - 2(i\mu)(2n) - c_0\big)C_{2n} e^{(i2n+\mu)\tau}$$
$$+ c_1 C_{2n} e^{(i2(n+1)+\mu)\tau} + c_1 C_{2n} e^{(i2(n-1)+\mu)\tau} \Big] = 0. \tag{17}$$

Since the sum in Eq. (17) is over $n \in (-\infty, \infty)$, we can relabel all n's in $c_1 C_{2n} e^{(i2(n+1)+\mu)\tau}$ and $C_{2n} e^{(i2(n-1)+\mu)\tau}$ as $n - 1$ and $n + 1$, respectively.

This relabeling transforms Eq. (17) into:

$$\sum_{n=-\infty}^{\infty} \Big[\big((2n - i\mu)^2 - c_0\big)C_{2n}$$
$$+ c_1 C_{2(n+1)} + c_1 C_{2(n-1)} \Big] e^{(i2n+\mu)\tau} = 0. \tag{18}$$

From Eq. (18), we obtain the recurrence relationship between the characteristic exponent μ and the solution parameters β_n's as follows:

$$\beta_n(\mu)C_{2(n+1)} + C_{2n} + \beta_n(\mu)C_{2(n-1)} = 0, \tag{19}$$

where the complex scalar function β_n is defined as

$$\beta_n(\mu) := \frac{c_1}{(2n - i\mu)^2 - c_0}. \tag{20}$$

3.3. Analytical expression of characteristic exponent μ

Eq. (19) for all $n \in \mathbb{Z}^+$ generates the following infinite set of linear homogeneous equations with the coefficients C_{2n}'s as the unknown variables:

$$\Delta(\mu) \big[\cdots, \ C_{-6}, \ C_{-4}, \ C_{-2} \ C_0, \ C_2, \ C_4, \ C_6, \ldots \big]^T = \mathbf{0}, \tag{21}$$

where $\mathbf{0}$ is a zero column vector with an infinite dimension and

$$\Delta(\mu) := \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \\ \cdots & 0 & \beta_{-1} & 1 & \beta_{-1} & 0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & \beta_0 & 1 & \beta_0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & \beta_1 & 1 & \beta_1 & 0 & \cdots \\ \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{22}$$

This set of linear equations have nontrivial solutions for the unknown coefficients C_{2n}'s if the determinant of $\Delta(\mu)$, denoted as $|\Delta(\mu)|$, equals zero. From [40], we know $|\Delta(\mu)| = 0$ can be compactly expressed as:

$$2|\Delta(0)| \sin^2 \Big(\frac{\pi\sqrt{c_0}}{2} \Big) = 1 - \cosh(\mu\pi). \tag{23}$$

Solving Eq. (23) provides the exact analytical expression of μ as:

$$\mu = \pm \frac{1}{\pi} \cosh^{-1} \Big(1 - 2|\Delta(0)| \sin^2 \Big(\frac{\pi\sqrt{c_0}}{2} \Big) \Big). \tag{24}$$

Remark 2 (*Offline Computation of Parameters μ and $\Delta(\mu)$*). Recall that $c_0 := -\frac{4g}{\omega^2 z_0}$ and $c_1 := \frac{2A}{z_0}$. Therefore, the values of c_0 and c_1 are known if the user-specified CoM height z_0 is known and if the surface motion frequency ω and magnitude A are measured, estimated, or predicted in real-time (e.g., by a surface motion monitoring system [34]). With known c_0 and c_1, the values of $\beta_n(0)$ (for all $n \in \mathbb{Z}^+$) and $|\Delta(0)|$ are known. Then, we can pre-compute μ using its analytical expression in Eq. (24), which could then be used to compute the analytical solution during online walking pattern generation.

3.4. Truncation of infinite series

Since the two infinite series in the exact solution in Eq. (14) are absolutely and uniformly convergent for any $0 < \tau < \infty$ [41], the exact solution is also convergent and can be approximated as a sum of finite terms.

We use $\hat{x}_{sc}(\tau)$ to denote the approximate solution. With N terms kept, the approximate solution is given by:

$$\hat{x}_{sc}(\tau) = \alpha_1 e^{\mu\tau} \sum_{n=-N}^{N} C_{2n} e^{i2n\tau} + \alpha_2 e^{-\mu\tau} \sum_{n=-N}^{N} C_{2n} e^{-i2n\tau}. \tag{25}$$

To simultaneously ensure sufficient accuracy and efficiency for the solution computation, we can determine the number of terms kept, N, offline for the considered range of DRS motion parameters and the user-specified solution tolerance. Specifically, we can numerically compute the minimum number of terms kept that results in a series truncation error less than the tolerance for the given DRS parameter range, which can then be used as the value of N.

3.5. Computation of coefficients C_{2n}'s

With the characteristic exponent, μ, and the number of terms kept, N, determined, we can obtain the value of the coefficient C_{2n} ($n \in \{0, 1, \ldots, N\}$) recursively based on Eq. (19), by setting $C_{2N} = 0$ and $C_0 = A$ [39].

The recurrence relationship in Eq. (19) indicates that the coefficient satisfies

$$|C_{2n}| \ll |C_{2(n-1)}| \tag{26}$$

for sufficiently large index n (e.g., $n > N$).

Hence, the coefficients with sufficiently large indices can be neglected; that is, $C_{2(N+1)} \approx 0$.

With $C_{2(N+1)} = 0$, solving the recurrence relation in Eq. (19) for all indices $n \in \{0, 1, \ldots, N\}$ gives:

for $n = N$:

$$\beta_N C_{2(N+1)} + C_{2N} + \beta_N C_{2(N-1)} = 0,$$
$$\Rightarrow C_{2N} = -\beta_N C_{2(N-1)}, \text{ since } C_{2(N+1)} = 0$$

for $n = N - 1$:

$$\beta_{N-1} C_{2N} + C_{2(N-1)} + \beta_{N-1} C_{2(N-2)} = 0,$$
$$\Rightarrow C_{2(N-1)} = \frac{-\beta_{N-1}}{1 - \beta_N \beta_{N-1}} C_{2(N-2)}$$

for $n = N - 2$:

$$\beta_{N-2} C_{2(N-1)} + C_{2(N-2)} + \beta_{N-2} C_{2(N-3)} = 0, \tag{27}$$
$$\Rightarrow C_{2(N-2)} = \frac{-\beta_{N-2}}{1 - \frac{\beta_{N-2}\beta_{N-1}}{1-\beta_{N-1}\beta_N}} C_{2(N-3)}$$

$$\ldots$$

Hence, C_{2n} ($n \in \{0, 1, \ldots, N\}$) can be expressed as:

$$C_{2n} = \frac{-\beta_n}{1 - \frac{\beta_n \beta_{n+1}}{1-\frac{\beta_{n+1}\beta_{n+2}}{1-\frac{\beta_{n+2}\beta_{n+3}}{1-\cdots}}}} C_{2(n-1)}. \tag{28}$$

By setting $C_0 = A$ in Eq. (28), all other coefficients are determine using Eq. (28). The relation in Eq. (28) is utilize to find the coefficients C_{-2n}, by replacing the index n with its additive inverse $-n$.

3.6. Computation of coefficients α_1 and α_2

The last set of parameters that need to be determined for computing the approximate analytical solution $\hat{x}_{sc}$ are α_1 and α_2, which can be obtained based on the given initial condition of $\hat{x}_{sc}$ as introduced next.

Recall that the coefficient β_n is defined in Eq. (20). This definition indicates that β_{-n} is the complex conjugate of β_n, and accordingly C_{-2n} is the complex conjugate of C_{2n}.

We can express the scalar, complex coefficient C_{2n} using the following generic form:

$$C_{2n} = r_{2n} e^{i\theta_{2n}}, \tag{29}$$

where r_{2n} and θ_{2n} are real constants.

Then, substituting $C_{2n} = r_{2n} e^{i\theta_{2n}}$ into the approximate analytical solution in Eq. (25) gives:

$$\begin{aligned}
\hat{x}_{sc}(\tau) &= \alpha_1 e^{\mu\tau} \sum_{n=-N}^{N} r_{2n} e^{i\theta_{2n}} e^{i2n\tau} + \alpha_2 e^{-\mu\tau} \sum_{n=-N}^{N} r_{2n} e^{i\theta_{2n}} e^{-i2n\tau} \\
&= \alpha_1 e^{\mu\tau} \sum_{n=1}^{N} \left[r_0 + r_{2n}(e^{i(2n\tau+\theta_{2n})} + e^{-i(2n\tau+\theta_{2n})}) \right] \\
&\quad + \alpha_2 e^{-\mu\tau} \sum_{n=1}^{N} \left[r_0 + r_{2n}(e^{-i(2n\tau-\theta_{2n})} + e^{i(2n\tau-\theta_{2n})}) \right] \\
&= \alpha_1 e^{\mu\tau} \sum_{n=1}^{N} \left[r_0 + 2r_{2n}\cos(2n\tau + \theta_{2n}) \right] \\
&\quad + \alpha_2 e^{-\mu\tau} \sum_{n=1}^{N} \left[r_0 + 2r_{2n}\cos(2n\tau - \theta_{2n}) \right].
\end{aligned} \tag{30}$$

Recall $\tau := \frac{\frac{\pi}{2}+\omega t}{2}$. Replacing τ with $\frac{\frac{\pi}{2}+\omega t}{2}$ in Eq. (30) yields:

$$\begin{aligned}
\hat{x}_{sc}(t) &= \alpha_1 e^{\mu \frac{\frac{\pi}{2}+\omega t}{2}} \sum_{n=1}^{N} \left[r_0 + 2r_{2n}\cos\left(\frac{n\pi}{2} + n\omega t + \theta_{2n} \right) \right] \\
&\quad + \alpha_2 e^{-\mu \frac{\frac{\pi}{2}+\omega t}{2}} \sum_{n=1}^{N} \left[r_0 + 2r_{2n}\cos\left(\frac{n\pi}{2} + n\omega t - \theta_{2n} \right) \right].
\end{aligned} \tag{31}$$

Given initial condition $(\hat{x}_{sc}(0), \dot{\hat{x}}_{sc}(0))$, we can compute the coefficients α_1 and α_2 based on the solution in Eq. (31).

Finally, with the values of the parameters μ, C_{2n}'s, α_1, and α_2 obtained as explained earlier, we can readily compute the approximate solution by substituting those parameter values in the solution expression in Eq. (25).

4. Stability analysis

This section presents the stability condition of the DRS-LIP model associated with locomotion on a vertically moving surface, along with the stability analysis based on the proposed approximate analytical solution.

4.1. Sufficient and necessary stability condition

Before introducing the stability condition, we first define the notion of stability for the DRS-LIP model.

Definition 1 (*Stability of the DRS-LIP Model*). By the Floquet theory [42], the DRS-LIP in Eq. (11) is called "stable" if all its solutions are bounded for any $t > 0$, and is "unstable" if an unbounded solution exists for $t > 0$.

The stability of the DRS-LIP, as defined in Definition 1, can be determined with the characteristic exponents μ. Since the DRS-LIP is a linear, second-order ordinary differential equation, it has two characteristic exponents, denoted as μ_1 and μ_2.

Theorem 2 (*Stability Condition*). *Let $Re(\mu_1)$ and $Re(\mu_2)$ respectively denote the real parts of μ_1 and μ_2. Suppose that $Re(\mu_1) \leq Re(\mu_2)$. Then, by the Floquet theory, the model is stable if and only if $Re(\mu_1), Re(\mu_2) < 0$.*
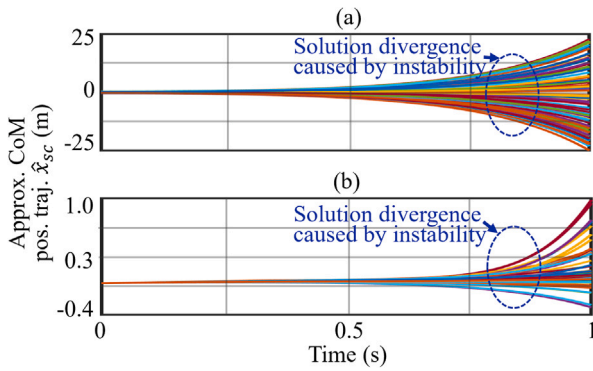
Fig. 2. Unbounded time evolution of solution $\hat{x}_{sc}(t)$ of the DRS-LIP model under: (a) the same model parameters ($A = 7$ cm, $\omega = \pi$ rad/s, and $z_0 = 42$ cm) but 100 different initial conditions satisfying $|x_{sc}(0)| < 0.4$ m and $|\dot{x}_{sc}(0)| < 0.4$ m/s and (b) different parameters ($0 < \omega \leq 2\pi$ rad/s, $0 < A \leq 100$ cm, and $30 \leq z_0 \leq 55$ cm) but the same initial condition ($x_{sc}(0) = 0.02$ m and $\dot{x}_{sc}(0) = 0.1$ m/s).

4.2. Numerical stability analysis

For typical ship motions in regular sea waves, the parameters (i.e., displacement magnitude A and frequency ω) of the DRS-LIP model in Eq. (13) take values within [30]:

$$A \leq 100 \text{ cm} \quad \text{and} \quad \omega \leq 2\pi \text{ rad/s}.$$

Also, the kinematically feasible CoM height z_0 of a typical quadrupedal robot (e.g., Unitree's Laikago) is within:

$$0.3 \text{ m} < z_0 < 0.55 \text{ m}.$$

Under these parameter ranges, we use Eq. (24) to numerically compute the characteristic exponents and obtain that

$$\text{Re}(\mu_2) > 0 \quad \text{and} \quad \text{Re}(\mu_1) < 0.$$

Therefore, by the Floquet theory [42], the DRS-LIP is unstable (i.e., an unbounded solution exists on $t > 0$) under the considered operating condition.

To illustrate this physical insight, Fig. 2 presents the corresponding approximate analytical solutions. Subplot (a) displays the approximate solutions under different initial conditions ($|x_{sc}(0)| < 0.4$ m and $|\dot{x}_{sc}(0)| < 0.4$ m/s) and DRS-LIP parameters ($\omega = \pi$ rad/s, $A = 7$ cm, and $z_0 = 42$ cm). Subplot (b) shows the solutions under the same initial condition ($x_{sc}(0) = 0.02$ m and $\dot{x}_{sc}(0) = 0.1$ m/s) but different model parameters ($0 < \omega \leq 2\pi$ rad/s, $0 < A \leq 100$ cm, and $30 \leq z_0 \leq 55$ cm). In all cases except for the trivial initial condition $x_{sc}(0)$, $\dot{x}_{sc}(0) = 0$, the solutions grow towards infinity as time t increases, confirming that the DRS-LIP model is unstable under the considered operating condition.

Remark 3 (*Effects of DRS-LIP Model Stability on Robot Walking Stability*). Despite the instability of the DRS-LIP model during continuous stance phases, the model is useful for the planning and control of a full-order robot to ensure robot walking stability. This is essentially because as long as the desired CoM motion is feasible during continuous phases, there exists a wide class of nonlinear control approaches (e.g., our prior input–output linearizing controller [7,43]) that can provably guarantee the walking stability for the overall hybrid full-order robot model. In this study, we implement such a controller to indirectly validate the feasibility of the proposed planner (see Section 6.5).

5. DRS-LIP based walking pattern generation

To demonstrate the practical uses of the DRS-LIP model and its analytical solution, this section presents a hierarchical walking pattern generator that exploits these theoretical results to enable efficient and
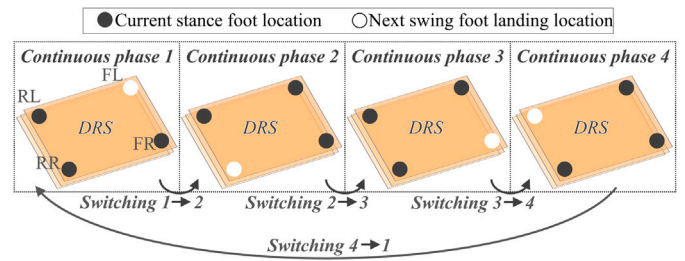

Fig. 3. A complete quadrupedal walking cycle, with the four feet marked as Front Left (FL), Front Right (FR), Rear Left (RL), and Rear Right (RR).

feasible planning for quadrupedal walking on a vertically oscillating DRS.

The planner is designed for quadrupedal walking [3,7] whose gait cycle comprises four continuous foot-swinging phases and four discrete foot-landing events (see Fig. 3).

The planner has two layers as illustrated in Fig. 4: (i) the higher layer generates kinematically and dynamically feasible CoM position trajectories, by using the proposed analytical solution of the DRS-LIP model and incorporating necessary feasibility constraints, and (ii) the lower layer utilizes trajectory interpolation to efficiently translate the CoM trajectories into the desired motion for all degrees of freedom of the full-order robot model.

Since the time profiles of the DRS motion are commonly sensed or estimated by real-time motion monitoring systems (e.g., shipboard sensors [34]), the planner design assumes that the nominal profile of the surface motion is known with bounded *inaccuracy*. Hardware experiment results that demonstrate the inherent robustness of the proposed framework under inaccurate surface motion knowledge is provided in Section 6.

5.1. Higher-layer CoM trajectory planner

The higher-layer planner uses the DRS-LIP as a basis to efficiently generate feasible reference trajectories of the CoM position $\mathbf{r}_{sc}(t)$ through nonlinear optimization.

As the planner is introduced primarily for highlighting the usefulness of the analytical results, we construct a simple higher-layer planner that is computationally efficient for real-world implementations, by reducing the number of optimization variables through the pre-specification of common gait parameters (which characterize the desired gait features). While a simple CoM trajectory planner is developed in this study based on the proposed model and analytical solution, the model and solution could be incorporated in a more complex planner (e.g., [44]) that demands fewer user-defined gait parameters and is more versatile for generating complex locomotion tasks.

5.1.1. User-defined gait parameters

Similar to [45], the pre-specified gait parameters in this study are chosen as: (i) average walking velocity (i.e., horizontal CoM velocity), (ii) foot contact sequence (see Fig. 3), (iii) stance foot positions, (iv) constant CoM height z_0 above the surface (for respecting assumption (A2)), and (v) gait period. The values of parameters (i)-(iv) are typically set to help ensure a kinematically feasible gait. The value of the parameter (v) is selected such that the quotient of the DRS' nominal motion period and the desired gait period is an integer (i.e., the desired CoM motion complies with the DRS motion).

5.1.2. Optimization variables

We choose the optimization variables α of the planner as the initial CoM position (x_{sc}, y_{sc}) and velocity ($\dot{x}_{sc}$, $\dot{y}_{sc}$) within each continuous
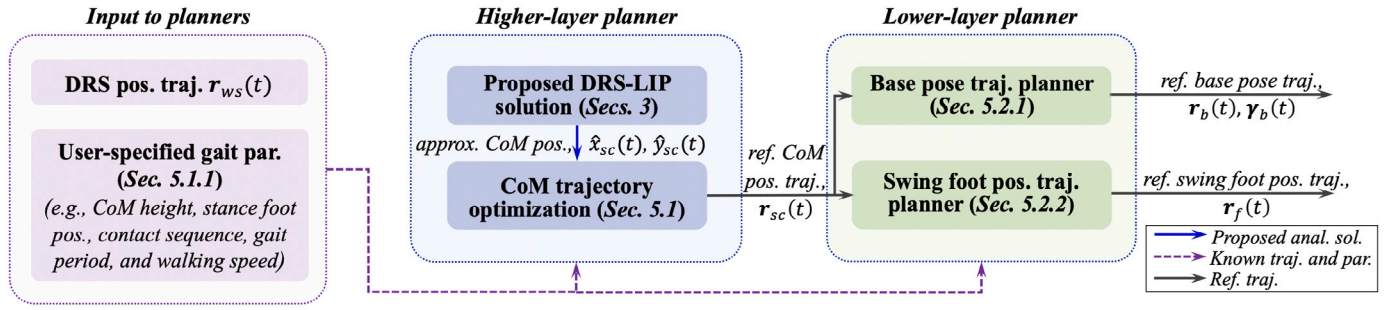
Fig. 4. Overview of the proposed hierarchical walking pattern generator. The higher layer exploits the proposed analytical solution of the DRS-LIP model to ensure efficient and physically feasible planning of the desired CoM position trajectories $\mathbf{r}_{sc}(t)$. The lower layer converts the reference CoM trajectories $\mathbf{r}_{sc}(t)$ into full-body reference motions ($\mathbf{r}_b(t)$, $\boldsymbol{\gamma}_b(t)$, and $\mathbf{r}_f(t)$) through trajectory interpolation.

phase. The rationale for this choice is that these variables, along with the DRS-LIP model parameters and the DRS motion profile, completely determine the horizontal CoM position trajectories. The vertical CoM position z_{sc} is not included as an optimization variable because it can be readily obtained from the user-defined CoM height z_0.

5.1.3. Constraints

We design the constraints to enforce gait feasibility and desired gait features. Note that these constraints are formed based on the proposed analytical approximate solution $\hat{x}_{sc}$. The equality constraints include: (i) continuity of the CoM trajectories at the foot-landing events and (ii) the desired walking velocity. The inequality constraints are: (i) friction cone constraint for avoiding foot slipping, (ii) confinement of CoM trajectories within the polygon of support for approximately respecting the CoP constraint, and (iii) upper and lower bounds on $\boldsymbol{\alpha}$.

To meet the constraints, $\boldsymbol{\alpha}$ is obtained by solving the following optimization problem:

$$\min_{\boldsymbol{\alpha}} \quad h(\boldsymbol{\alpha})$$
$$\text{subject to} \quad \mathbf{f}_{eq}(\boldsymbol{\alpha}) = \mathbf{0}, \ \mathbf{g}_{ineq}(\boldsymbol{\alpha}) \leq \mathbf{0}, \tag{32}$$

where $h(\boldsymbol{\alpha})$ is a scalar cost function (e.g., energy cost of transport), and the vector-valued functions $\mathbf{f}_{eq}$ and $\mathbf{g}_{ineq}$ are the sets of all aforementioned equality and inequality constraints, respectively. The expressions of $\mathbf{f}_{eq}$ and $\mathbf{g}_{ineq}$ are omitted for space consideration.

5.2. Lower-layer full-body trajectory generator

The lower-layer planner is essentially trajectory interpolation that translates the reference CoM trajectory $\mathbf{r}_{sc}(t)$ (supplied by the higher-layer planner) into the robot's full-body trajectories. To impose a steady trunk/base pose and to avoid swing foot scuffing on the surface, we choose these full-order trajectories to be the absolute base pose (position $\mathbf{r}_b$ and orientation $\boldsymbol{\gamma}_b$) and the swing foot position $\mathbf{r}_f$ relative to the base.

The input to the lower-layer planner (see Fig. 4) are: the nominal DRS motion that is vertical and sinusoidal, the CoM position trajectories provided by the higher-layer planner, and the user-defined parameters (e.g., CoM height, stance foot locations, and maximum swing foot height).

5.2.1. Base pose trajectories

The CoM of the robot is approximated as the base (i.e., the geometric center of the trunk) because a quadruped's trunk typically has a symmetric mass distribution and is substantially heavier than the legs. Thus, we set the desired base position trajectories $\mathbf{r}_b(t)$ same as the desired CoM position trajectories $\mathbf{r}_{sc}(t)$. As real-world locomotion tasks are typically encoded by a robot's absolute global/base position, we choose to transform these relative position trajectories into the absolute ones.

With the DRS position $\mathbf{r}_{ws}(t)$ at the support point S, we obtain the absolute base position trajectories $\mathbf{r}_b(t)$ as:

$$\mathbf{r}_b(t) = \mathbf{r}_{sc}(t) + \mathbf{r}_{ws}(t). \tag{33}$$

To avoid overly stretched leg joints for ensuring kinematic feasibility, the desired base orientation trajectories $\boldsymbol{\gamma}_b(t)$ are designed to comply with the DRS orientation.

5.2.2. Swing foot position trajectories

The desired swing foot trajectories $\mathbf{r}_f(t)$ (relative to the support point S) are designed to agree with the desired stance foot locations and to respect the kinematic limits of the robot's leg joints. Specifically, we plan the desired swing foot trajectory during a continuous phase by using Bézier polynomials [7] to connect the adjacent desired stance foot positions.

We use s to denote the scalar normalized phase variable that represents how far a walking step has progressed. Let $\mathbf{r}_{f,i}$ and $\mathbf{r}_{f,e}$ respectively denote the desired swing foot locations at the initial and end instants of a continuous phase. We assign the values of $\mathbf{r}_{f,i}$ and $\mathbf{r}_{f,e}$ to match the desired stance foot locations for the given continuous phase.

Then, we use the following Bézier curve to express the desired swing foot position $\mathbf{r}_f$ within the given phase:

$$\mathbf{r}_f(s) = \mathbf{r}_{f,i} + \mathbf{P}(s)(\mathbf{r}_{f,e} - \mathbf{r}_{f,i}), \tag{34}$$

where $\mathbf{P}(s)$ is a 3×3 diagonal matrix-valued function with each diagonal term an nth-order Bézier polynomial interpolation.

For walking along a straight line, we can design the lateral swing foot position as constant for simplicity. We design the forward and vertical trajectories to have a relatively fast initial velocity within Continuous Phases 1 and 3, as illustrated in Fig. 5(a). This relatively fast initial velocity allows the robot's full body to have sufficient momentum to leave the previous support polygon and enter the planned current polygon, thereby indirectly meeting the CoP constraints under the desired contact sequence.

Also, as inspired by previous quadrupedal robot planning [3], a brief four-leg-in-support phase is inserted upon a foot-landing event when the two consecutive polygons of support only share a common edge (i.e., "Switching $1 \rightarrow 2$" and "Switching $3 \rightarrow 4$" in Fig. 3), so as to ensure smooth and feasible transitions during these events. This transitional phase is highlighted with a gray background in Fig. 5. Thanks to this transitional phase, the initial forward and vertical swing foot velocities within Phases 2 and 4 do not need to be as fast as Phases 1 and 3 (see Fig. 5(b)).

Remark 4 (*Effects of LIP Model Accuracy on Trajectory Generation Feasibility*). The dynamic feasibility of the planned trajectories partly depends on the closeness between the DRS-LIP model and the actual robot dynamics. The DRS-LIP model is a relatively faithful representation of an actual DRS-robot system when the robot and DRS behaviors meet the assumptions (A1)-(A3) that underlie the proposed model and its
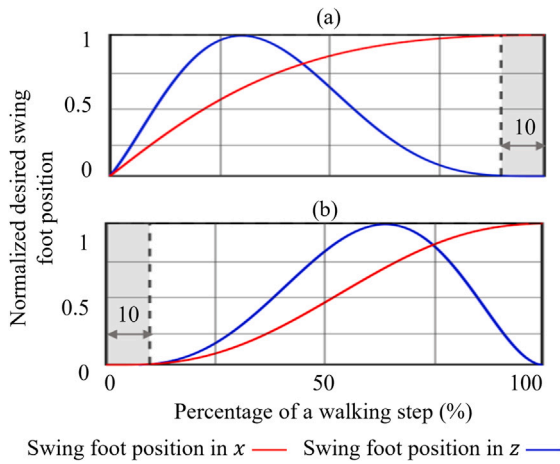
Fig. 5. Normalized swing foot position trajectories in x- and z-directions during (a) Continuous Phases 1 and 3 and (b) Continuous Phases 2 and 4. The gray background highlights the transitional four-leg-in-support phase.
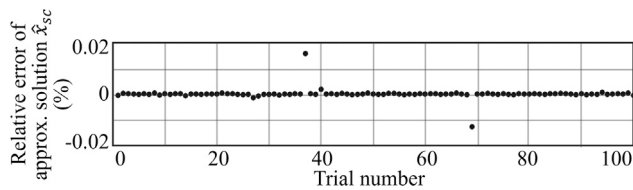

Fig. 6. Mean percentage error of the proposed analytical approximate solution compared with the high-accuracy numerical solution under model parameters $A = 7$ cm, $\omega = \pi$ rad/s, and $z_0 = 42$ cm for 100 random initial conditions satisfying $|x_{sc}(0)| < 0.2$ m and $|\dot{x}_{sc}(0)| < 0.2$ m/s.

Table 1
Average computation time of analytical and numerical solutions for 1000 trials in MATLAB (mean ± SD).

Solution method	Computation time (ms)
Numerical	2.61 ± 0.43
Analytical (proposed)	0.16 ± 0.02

solution. Indeed, assumption (A3) holds when the surface motion is vertical and sinusoidal, and the planner explicitly imposes assumption (A2). Moreover, as the planner enforces the desired base orientation to comply with the surface orientation for kinematic feasibility, the planned motion will reasonably respect assumption (A1) for surfaces that translate without rotary motions. Even for real-world DRSes that rotate (e.g., rocking vessels), the rate of the robot's centroidal angular momentum will be negligible under the typical angular movement range of those DRSes [30], thus still respecting assumption (A1).

6. Simulation and experiment validation

This section presents the simulation results that validate the proposed approximate analytical solution and walking pattern generator.

6.1. Validation of solution accuracy and efficiency

The accuracy and computational efficiency of the proposed analytical approximate solution in Eq. (25) is assessed through comparison with a highly accurate numerical solution. For fairness of comparison, both solutions are computed in MATLAB on $t \in [0, 0.5]$ s. The
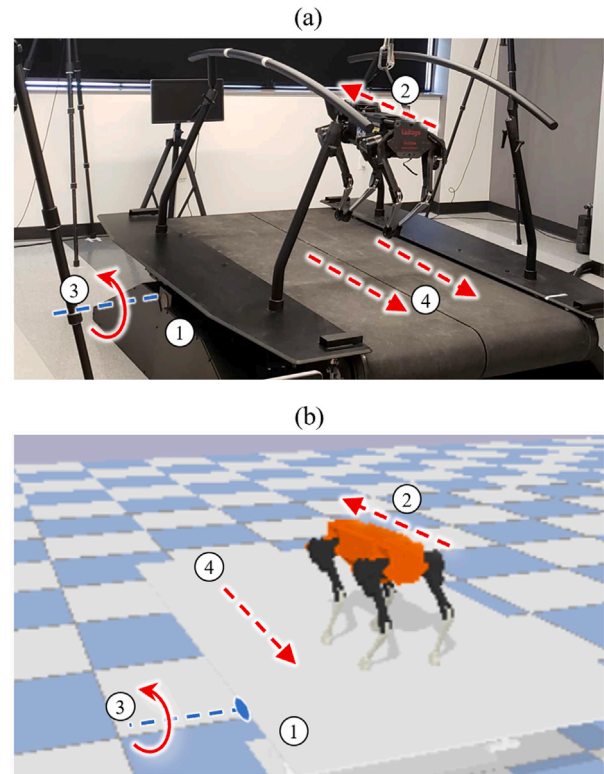
approximate solution has ten terms kept (i.e., $N = 10$) for a reasonable trade-off between accuracy and computational efficiency. The comparative numerical solution is computed using MATLAB's ODE45 solver with an error tolerance of 10^{-9} and at a time interval of 0.5 ms.

To validate the proposed solution under different initial conditions, 1000 sets of initial conditions are randomly chosen within a common movement range of quadrupedal walking [3]: $|x_{sc}(0)| < 0.2$ m and $|\dot{x}_{sc}(0)| < 0.2$ m/s. The DRS-LIP model parameters are chosen to be within realistic ranges of DRS motions [30,36] and quadrupedal robot dimensions [7]: $A = 7$ cm, $\omega = \pi$ rad/s, and $z_0 = 42$ cm.

Fig. 6 shows the accuracy of the approximate analytical solution (with ten terms kept) compared with the numerical solution for 100 out of the 1000 trials. Within those 100 trials, the maximum value of the absolute percentage error is lower than 0.02% in magnitude, indicating the reasonable accuracy of the proposed approximate solution. For all 1000 trials, the absolute percentage error, measured by mean ± one standard deviation (SD), is $(0.0012 \pm 0.005)\%$.

Table 1 displays the comparison of the average computational time cost (measured by mean±SD) for the aforementioned 1000 trials. The proposed approximate analytical solution is about 15 times faster to compute than the numerical one.

6.2. Simulation and experimental setup for planner validation

The setup of simulations and hardware experiments is shown in Fig. 7. The experiment video is provided as a supplementary file and is available at https://youtu.be/F9LH8mdhedg.
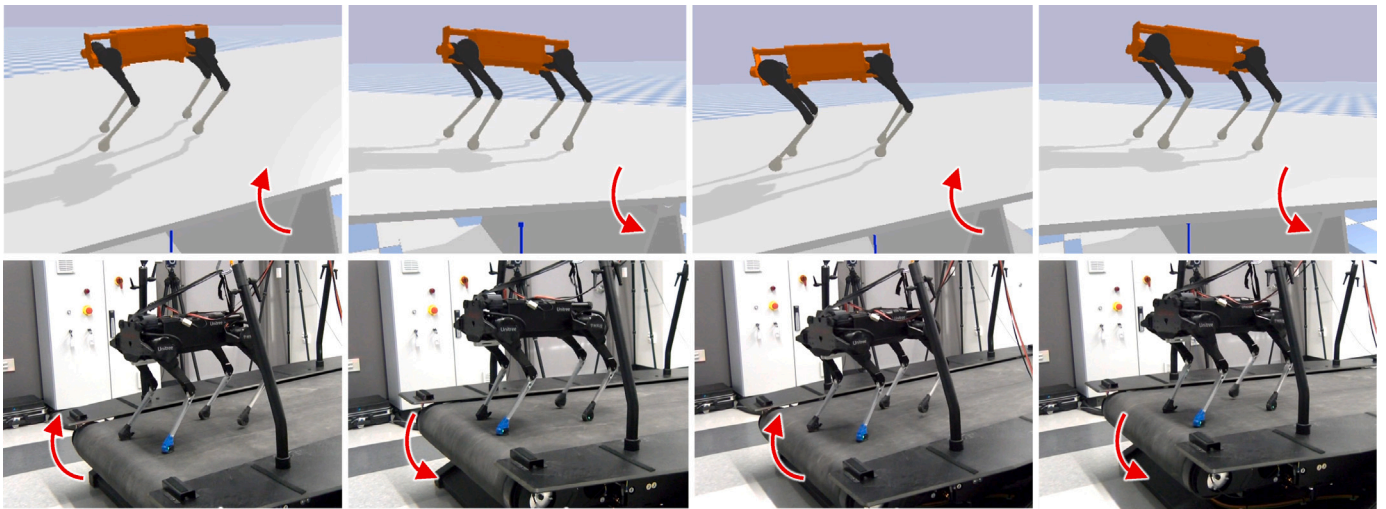

Fig. 7. Setup of (a) experiments and (b) PyBullet simulations for testing the planner effectiveness using a pitching Motek treadmill (①) and a Laikago quadruped (②). The treadmill has a split belt (④) that moves at a constant speed while the treadmill rocks about the horizontal axis (③).

Fig. 8. Image tiles of quadrupedal walking on a pitching treadmill under the proposed walking pattern generator synthesized based on the DRS-LIP model and its analytical solution. The top and bottom rows respectively show PyBullet simulations and hardware experiments.

Table 2
User-defined gait parameters in walking pattern generation.

Gait parameter	(G1)	(G2)
Friction coefficient	0.5	0.5
Robot's base height z_0 (cm)	42	42
Gait duration (s)	2	2
Average walking velocity (cm/s)	5	6
Step length (cm)	10	12
Max. step height (cm)	5	5

6.2.1. Robot

The validation of the planner utilizes a Laikago quadruped (see Fig. 7) developed by Unitree Robotics. The dimension of the robot is 55 cm × 35 cm × 60 cm. The robot's total mass is 25 kg and a power density of 0.8 kW/kg. The robot is powered by a 650 Wh lithium-ion battery weighing 4.4 kg. It has twelve independently actuated joints that can produce up to 18 kW of instantaneous power. Each leg weighs 2.9 kg and has three brushless DC motors located close to the trunk. The torque limits of the hip-roll, hip-pitch, and knee-pitch motors are 20 Nm, 55 Nm, and 55 Nm, respectively. The robot is equipped with an IMU at the trunk, 12 joint encoders, and a contact sensor at each foot.

6.2.2. Nominal and uncertain DRS motion

Three DRS motions are tested to assess the efficiency, feasibility, and robustness of the planner under different surface motions that emulate vessel movements in regular sea waves [36].

Due to our limited equipment access to programmable, actuated DRS that exclusively exhibits vertical motion, we focus on PyBullet simulations for the planner validation on a vertically moving surface with the following nominal profile without movement uncertainties:

(DRS1) The DRS motion is vertical and sinusoidal with $A = 10$ cm and $\omega = \pi$ rad/s.

Meanwhile, we use a Motek M-Gait treadmill that performs a pitching motion (see Fig. 7) to approximate a vertically moving DRS both in hardware experiments and PyBullet simulations. The Motek treadmill can be pre-programmed to perform user-defined pitching (but not vertical) motions and belt translation. The treadmill weighs 750 kg with a dimension of 2.3 m × 1.82 m × 0.5 m. A 4.5 kW servo motor powers each of the treadmill's two belts. During the hardware experiments, the robot is placed approximately 1 m away from the treadmill's axis of rotation, and the belt speed is set to be the same as the desired walking

speed. Fig. 8 shows images of the Laikago robot walking on the rocking treadmill in simulations and experiments.

The actual movement of the pitching treadmill at the foot-surface contact points is different from the nominal, vertical DRS motion used in planning and control, since the pitching treadmill naturally possesses non-negligible horizontal motion and the robot's actual location on the treadmill directly affects the actual surface motion at the support feet. Both the nominal and actual surface motions are listed below for the two surface profiles tested:

(DRS2) The nominal, vertical surface motion used in planning and control is sinusoidal with $A = 7$ cm and $\omega = \pi$ rad/s. The actual, pitching DRS motion is a sinusoidal function with an amplitude of 5° and frequency of 0.5 Hz.

(DRS3) The nominal, vertical surface motion used in planning and control is sinusoidal with $A = 11$ cm and $\omega = \pi$ rad/s. The actual, pitching DRS motion is a sinusoidal function with an amplitude of 7° and frequency of 0.5 Hz.

For all surface motions (DRS1)-(DRS3), the surface accelerations in the vertical direction are relatively significant for planner validation, with peak contact-point accelerations approximately at 100 cm/s^2, 70 cm/s^2, and 110 cm/s^2 in magnitude, respectively, when the robot stands about 1 m away from the treadmill's axis of pitching. The corresponding magnitude of contact-point displacements are 10 cm, 7 cm, and 11 cm, respectively.

For the uncertain surface motions (DRS2) and (DRS3), the upper bounds of the uncertainty in the absolute acceleration of the surface-robot contact points are approximately 20 cm/s^2 and 10 cm/s^2 in vertical and horizontal directions, respectively. Thus, the maximum absolute uncertainties are about 10%–20% of the peak vertical acceleration at the contact points.

6.2.3. Validation cases

We validate the efficiency, feasibility, and robustness of the proposed planner under four different combinations of surface motions and desired gait features, both through hardware experiments and simulations. The gait features are sampled from the two parameter sets (G1) and (G2) as specified in Table 2.

To assess the planning efficiency, computations in MATLAB and C++ are conducted under:

(Case 1) Combination of surface motion (DRS2) and gait parameters (G1).

Table 3
Average time cost of 1000 runs of higher-layer planning (mean±SD) under Case 1.

Solution method	MATLAB (fmincon)	C++ (Ipopt)
Numerical (ms)	1320.7 ± 13.8	72.3 ± 6.7
Analytical (ms)	269.1 ± 12.9	8.6 ± 2.2

To validate the planner feasibility under no surface motion uncertainty, we perform PyBullet simulations under:

(Case 2) Combinations of vertical surface motion (DRS1) and gait (G1).

Finally, both hardware experiments and Pybullet simulations are used to evaluate the planner robustness in the presence of uncertain surface motions:

(Case 3) Combination of uncertain surface motion (DRS2) and gait parameters (G1) and

(Case 4) Combination of uncertain surface motion (DRS3) and gait parameters (G2).

6.3. Planner efficiency validation

To demonstrate that utilizing the proposed analytical solution improves the planner efficiency compared with the numerical solution, the higher-layer CoM trajectory planning problem is solved based on both solutions under Case 1.

For simplicity, the cost function h in Eq. (32) is chosen as trivial. A 6th-order Bézier curve is used to design the desired swing foot trajectory for allowing adequate freedom in trajectory design. Also, we choose to lower the load of computing the proposed analytical solution by pre-computing its solution parameters μ and C_{2n}, as discussed in Remark 2.

To assess the planner efficiency under different common solvers, both MATLAB and C++ are used to solve the optimization-based planning problem in Eq. (32) for 1000 runs with the same initial guess of the optimization variable α. For fairness of comparison, the optimality and constraint tolerances are set as 10^{-6} in all runs. In MATLAB, fmincon is used with an interior-point solver. For the C++ optimization, the nonlinear optimization solver of the Ipopt package [46] is utilized. We solve the optimization problem on a Windows 10 PC with 32 GB DDR4 RAM and an Intel Xeon W-10855M processor running at a base speed of 2.8 GHz.

For those 1000 runs, Table 3 shows that the mean time costs of the analytical solution based higher-layer planning is approximately 4 and 7 times shorter than the numerical solution based one in MATLAB and C++, respectively.

Furthermore, Table 3 indicates that the higher-layer planner takes 8.6 ± 2.2 ms to generate the desired CoM trajectory when it is solved by C++ using the approximate analytical solution. The median time cost of those 1000 runs of computations is 8.4 ms. Also, solving the lower-layer planner is typically fast (e.g., MATLAB can solve it within 2 ms) since the planning is essentially trajectory interpolation. Therefore, the mean time cost for solving both higher and lower layers will be less than 11 ms. Since such a time cost is much smaller than the typical quadrupedal walking gait period (i.e., about 2 s [3]) and real-world DRS motion periods (e.g., 1-100 s for vessels [30]), the proposed planner would be adequately fast to timely regenerate the desired full-order trajectories in case of any significant changes in the DRS motion.

6.4. Comparing planner efficiency

To highlight the significant reduction in planning time of the proposed walking pattern generator thanks to the use of our proposed time-varying LIP model, we compare the time cost of the proposed generator for planning 2 s of walking motion with the existing planning
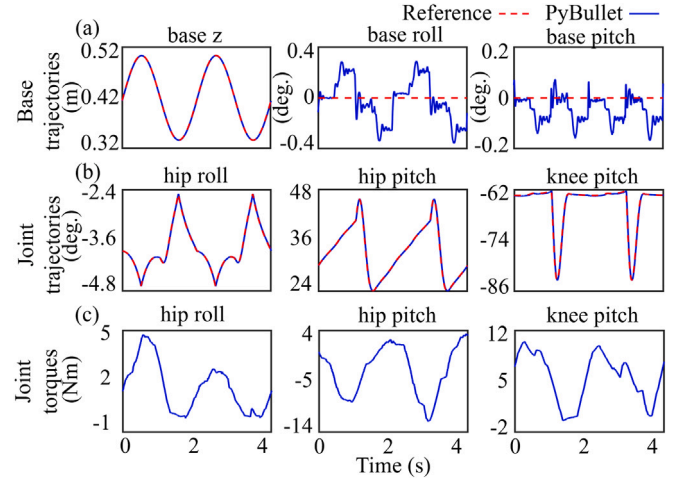

Fig. 9. PyBullet simulation results for the robot's base and front-right leg under Case 2.

method that utilizes a full-order robot model [7,47]. This comparative planner is formulated as an optimization problem similar to the proposed pattern generator.

The optimization variables of the full-order model based planner is the vector of Bézier coefficients β associated with the base pose and swing foot trajectories.

The equality constraints include: (i) continuity of the desired base pose trajectories across walking phases and (ii) the desired walking velocity. The inequality constraints are: (i) bounds on the joint configuration, (ii) bounds on the CoM position trajectories, and (iii) upper and lower bounds on β. The cost function is set as a scalar cost function with the same physical meaning as the proposed walking pattern generator.

Under the same settings for the optimization problem, computing unit, and user-defined parameters as in Section 6.3, the MATLAB planning time for 2 s of walking pattern based on the full-order robot model, among 10 trials, is 162.4 ± 13.7 s while the time cost is 0.27 ± 0.01 s for the proposed planner based on the time-varying LIP model. This comparative simulation demonstrates a substantial improvement in planning efficiency based on the proposed model simplification, enabling real-time walking pattern generation.

6.5. Planner feasibility validation

Besides efficiency, the proposed DRS-LIP and its solution can also be used to guarantee planning feasibility, which is validated in PyBullet simulations under Case 2.

To test the feasibility of the planned motion, we choose to implement our previous nonlinear feedback controller [7] that does not explicitly ensure the feasibility of ground contact forces during hardware experiments (Remark 3). If the controller turns out to be effective in sustaining stable walking on a DRS, we can infer that the planned trajectory is at least approximately physically feasible.

This controller is derived based on the hybrid full-order robot model and proportional derivative (PD) control. Given feasible desired trajectories, it provably guarantees the walking stability. To help ensure a reasonable tracking performance, the PD gains are tuned as 0.7 and 1.0 in simulations, and 5.5 and 0.15 on hardware.

As shown in Fig. 9, the robot sustains stable walking for the entire testing period of 50 gait cycles. The base and joint trajectories closely track their reference values, as shown in subplots (a) and (b). Also, subplot (c) indicates that the actual robot motion indeed respects the torque limits.
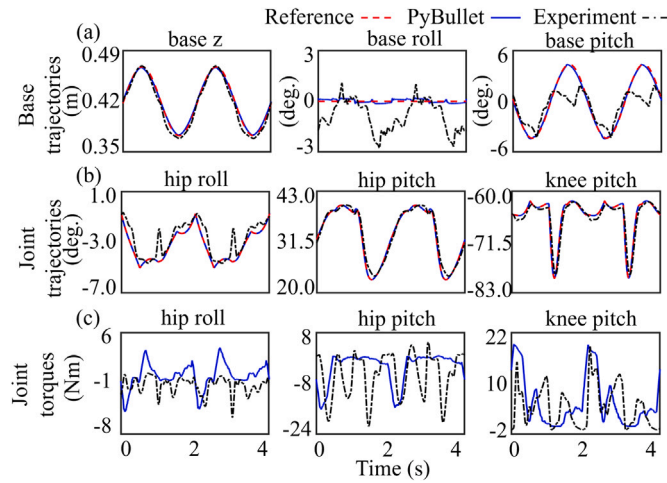
Fig. 10. Hardware experiment and PyBullet simulation results for the robot's base and front-right leg under Case 3.
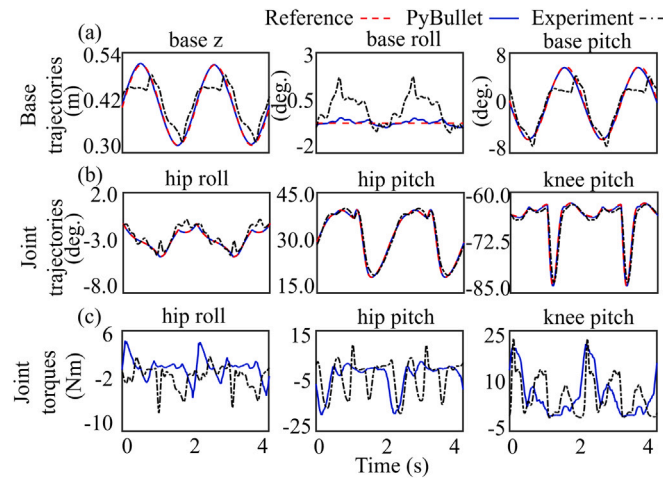

Fig. 11. Hardware experiment and PyBullet simulation results for the robot's base and front-right leg under Case 4.

6.6. Robustness validation under surface motion uncertainty

To evaluate the inherent robustness of the proposed planner in the presence of moderate levels of uncertain DRS motions as specified in Section 6.2.2, we test the planner under Cases 3 and 4, with results respectively presented in Figs. 10 and 11.

In both simulations and experiments, the robot walking is stable, as indicated by the trajectory tracking accuracy in subplots (a) and (b) of Figs. 10 and 11 as well as the experiment video. Moreover, subplots (c) of Figs. 10 and 11 confirm that the joint torque limits are met in both simulations and experiments.

Yet, the torque profiles of the front-right leg's three joints display notable discrepancies between PyBullet and experiment results, possibly due to the differences between the simulated and actual robot dynamics as well as the different inherent meanings of their effective PD gains.

Also, the experiment video shows that the robot experiences relatively notable rebounding and slipping at contact-switching events when a rear leg lands on the surface. This violation of the planned contact sequence is directly due to the temporary loss of contact force feasibility, and could be mitigated through improved controller design as discussed in Section 7.

7. Discussion

This paper has introduced a reduced-order dynamic model of a legged robot that walks on a DRS, by analytically extending the classical LIP model from stationary surfaces [14] to a DRS (e.g., a vessel). The resulting DRS-LIP model in Eq. (8) is a linear, second-order differential equation, similar to the classical LIP. However, the DRS-LIP is explicitly time-varying whereas the classical LIP is time-invariant. This fundamental difference is due to the time-varying movement of the surface at the surface-foot contact points. This study also investigates the stability of the DRS-LIP based on the Floquet theory. Similar to the classical LIP that describes stationary-surface locomotion [14], the DRS-LIP is unstable under the usual movement range of real-world DRSes such as vessels [30].

The DRS-LIP is valid under assumption (A1) that the actual robot's rate of whole-body angular momentum about the CoM is negligible. To relax this assumption, the point mass of the proposed DRS-LIP could be augmented with a flywheel [19,21] to account for the nonzero rate of angular momentum. Moreover, the DRS-LIP can be generalized from a constant CoM height (as enforced by assumption (A2)) to a varying height by integrating with the variable-height LIP for stationary surfaces [15].

This study also derives the approximate analytical solution of the DRS-LIP for vertical, sinusoidal surface motions. Its sufficient accuracy and improved computational efficiency compared with numerical solutions are confirmed through MATLAB simulations (Fig. 6 and Table 1). Although the proposed reduced-order model in Eq. (7) does not assume a specific form of surface motion, the proposed analytical solution is derived for vertical and sinusoidal surface motions, which are typical for real-world ship motions in regular sea waves [30,35]. To address surface motions that are vertical and nonperiodic with their time profiles pieced together by periods of different sinusoidal waves, which cover a wide range of DRS motions [30], the proposed analytical solution could be extended by: (i) forming the individual analytical solutions for those different periods based on the proposed solution derivation method and then (ii) piecing them together to form the needed overall solution. Moreover, if the vertical nonperiodic surface motion comprises periods of general periodic functions instead of sinusoidal waves, we could potentially use the Floquet theory [42] to derive the analytical solution by numerically precomputing the fundamental matrix of the reduced-order model and then constructing the analytical solution using the fundamental matrix. Our future work will also tackle the modeling and planning problem for legged locomotion under general surface motions [48] that contain horizontal movements [49,50].

To highlight the practical usefulness of the analytical results, the DRS-LIP model and its solution have been used as a basis to synthesize a two-layer walking pattern generator that efficiently produces desired, physically feasible motions for quadrupedal DRS walking. The feasibility of the planned motion is validated by using our previous tracking controller [7] to command a quadrupedal robot to follow the planned motion during DRS walking. As discussed in Section 6.1, simulation and experiment results indicate the reasonable feasibility and robustness of the proposed planner under different gait parameters and surface motions (Figs. 9–11). Furthermore, validations through hardware experiments demonstrated its robustness to uncertainties in DRS motions. To mitigate the temporary violation of the planned gait sequence observed in experiments, which is partly induced by the discrepancies between the DRS-LIP and the actual robot dynamics, the planned motion could be tracked by an optimization-based controller [51] that explicitly ensures physical feasibility.

With the peak absolute acceleration of the surface-foot contact points at around 1 m/s², the quadrupedal walking speed relative to the rocking treadmill is about 5 cm/s in the hardware experiments, which, to our best knowledge, is the fastest quadruped walking speed on a vigorously rotating surface for hardware experiments [7]. To achieve faster walking despite the inevitably higher inaccuracy of the

reduced-order model induced by the higher nonlinearity of actual robot dynamics, we can augment the controller described in Section 6.1, which does not reason about the feasibility of necessary constraints (e.g., ground contact forces), with an optimization-based controller [2, 3] that explicitly guarantee the feasibility for actual walking. Moreover, while we consider a constant average walking speed (relative to the dynamic surface) in the validation of the proposed walking pattern generator, the proposed planner can be readily extended from constant to variable speed walking, because the proposed reduced-order model and its analytical solution are valid in describing the robot dynamics within any walking cycle of constant or variable speed walking.

8. Conclusions

This paper has presented an analytically tractable and computationally efficient reduced-order robot dynamics model, the approximate analytical solution of the model, and a real-time motion generator for legged locomotion on a dynamic rigid surface (DRS). The proposed model was derived by theoretically extending the classical linear inverted pendulum (LIP) model from a stationary surface to a DRS, and describes the essential time-varying robot dynamics associated with DRS walking, which is fundamentally different from the classical time-invariant LIP model. The analytical solution of the extended LIP model was obtained based on the conversion of the model into the well-studied Mathieu's equation. Exploiting these analytical results, a real-time walking pattern generator was developed to efficiently plan feasible robot motions for quadrupedal walking on a vertically oscillating surface. Simulation results revealed the continuous-phase stability property of the proposed time-varying LIP model, and demonstrated the efficiency and accuracy of the analytical solution under a common range of real-world DRS movement. Finally, both 3-D realistic PyBullet simulations and hardware experiments on a physical Laikago quadrupedal robot confirmed the computational efficiency, physical feasibility, and inherent robustness of the proposed framework under various gait parameters and surface motions.

CRediT authorship contribution statement

Amir Iqbal: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft. **Sushant Veer:** Methodology, Investigation, Formal analysis, Writing – review & editing. **Yan Gu:** Conceptualization, Methodology, Formal analysis, Supervision, Project administration, Funding acquisition, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.mechatronics.2023.103073.

References

[1] Hutter M, Gehring C, Jud D, Lauber A, Bellicoso CD, Tsounis V, et al. ANYmal-a highly mobile and dynamic quadrupedal robot. In: Proc. IEEE/RSJ Int. Conf. Intel. Rob. Syst.. 2016, p. 38–44.

[2] Fawcett RT, Pandala A, Ames AD, Hamed KA. Robust stabilization of periodic gaits for quadrupedal locomotion via QP-based virtual constraint controllers. IEEE Contr. Syst. L. 2021;6:1736–41.

[3] Mastalli C, Havoutis I, Focchi M, Caldwell DG, Semini C. Motion planning for quadrupedal locomotion: Coupled planning, terrain mapping, and whole-body control. IEEE Trans. Rob. 2020;36(6):1635–48.

[4] Gu Y, Yao B, Lee CSG. Exponential stabilization of fully actuated planar bipedal robotic walking with global position tracking capabilities. ASME J. Dyn. Syst. Meas. Contr. 2018;140(5).

[5] Gao Y, Gu Y. Global-position tracking control of multi-domain planar bipedal robotic walking. In: Proc. ASME Dyn. Sys. Contr. Conf.. 2019, p. V001T03A009.

[6] Xu K, Wang S, Yue B, Wang J, Peng H, Liu D, et al. Adaptive impedance control with variable target stiffness for wheel-legged robot on complex unknown terrain. Mechatron. 2020;69:102388.

[7] Iqbal A, Gao Y, Gu Y. Provably stabilizing controllers for quadrupedal robot locomotion on dynamic rigid platforms. IEEE/ASME Trans. Mechatron. 2020;25(4):2035–44.

[8] Gao Y, Yuan C, Gu Y. Invariant extended Kalman filtering for hybrid models of bipedal robot walking. In: Proc. IFAC Mod. Est. Contr. Conf, vol. 54, no. 20. 2021, p. 290–7.

[9] Shih C-L, Grizzle JW, Chevallereau C. From stable walking to steering of a 3D bipedal robot with passive point feet. Rob. 2012;30(7):1119–30.

[10] Motahar MS, Veer S, Poulakakis I. Composing limit cycles for motion planning of 3D bipedal walkers. In: Proc. IEEE Conf. Dec. Contr.. 2016, p. 6368–74.

[11] Gao Y, Gu Y. Global-position tracking control of a fully actuated NAO bipedal walking robot. In: Proc. Amer. Contr. Conf.. 2019, p. 4596–601.

[12] Gao Y, Yuan C, Gu Y. Invariant filtering for legged humanoid locomotion on dynamic rigid surfaces. IEEE/ASME Trans. Mechatron. 2022;27(4):1900–9.

[13] Chen Y-M, Posa M. Optimal reduced-order modeling of bipedal locomotion. In: Proc. Int. Conf. Rob. Autom.. 2020, p. 8753–60.

[14] Kajita S, Kanehiro F, Kaneko K, Yokoi K, Hirukawa H. The 3D linear inverted pendulum mode: A simple modeling for a biped walking pattern generation. In: Proc. IEEE Int. Conf. Intel. Rob. Syst. vol. 1. 2001, p. 239–46.

[15] Caron S, Escande A, Lanari L, Mallein B. Capturability-based pattern generation for walking with variable height. IEEE Trans. Rob. 2019;36(2):517–36.

[16] Xiong X, Ames A. 3-D underactuated bipedal walking via H-LIP based gait synthesis and stepping stabilization. IEEE Trans. Rob. 2022;38(4):2405–25.

[17] Gong Y, Grizzle JW. Zero dynamics, pendulum models, and angular momentum in feedback control of bipedal locomotion. ASME J. Dyn. Syst. Meas. Contr. 2022;144(12):121006.

[18] Westervelt ER, Grizzle JW, Chevallereau C, Choi JH, Morris B. Feedback control of dynamic bipedal robot locomotion, vol. 28. CRC Press; 2007.

[19] Pratt J, Carff J, Drakunov S, Goswami A. Capture point: A step toward humanoid push recovery. In: Proc. IEEE Int. Conf. Humanoid Rob.. 2006, p. 200–7.

[20] Mihalec M, Zhao Y, Yi J. Recoverability estimation and control for an inverted pendulum walker model under foot slip. In: Proc. IEEE/ASME Int. Conf. Adv. Int. Mechatron.. 2020, p. 771–6.

[21] Zhao Y, Fernandez BR, Sentis L. Robust optimal planning and control of non-periodic bipedal locomotion with a centroidal momentum model. Int. J. Rob. Res. 2017;36(11):1211–42.

[22] Li C, Ding Y, Park H-W. Centroidal-momentum-based trajectory generation for legged locomotion. Mechatron. 2020;68:102364.

[23] Kimura K, Okada K, Inaba M. Extended balance stabilization control for humanoid robot on rotational slope based on seesaw-inverted-pendulum model. Adv. Rob. 2021;35(18):1116–30.

[24] Dai M, Xiong X, Ames AD. Data-driven step-to-step dynamics based adaptive control for robust and versatile underactuated bipedal robotic walking. 2022, arXiv preprint arXiv:2209.08458.

[25] Zheng Y, Yamane K. Ball walker: A case study of humanoid robot locomotion in non-stationary environments. In: Proc. IEEE Int. Conf. Rob. Autom.. 2011, p. 2021–8.

[26] Yang C, Zhang B, Zeng J, Agrawal A, Sreenath K. Dynamic legged manipulation of a ball through multi-contact optimization. In: Proc. IEEE Int. Conf. Intel. Rob. Syst.. 2020, p. 7513–20.

[27] Asano F. Modeling and control of stable limit cycle walking on floating island. In: Proc. IEEE Int. Conf. Mechatron.. 2021, p. 1–6.

[28] Iqbal A, Mao Z, Gu Y. Modeling, analysis, and control of SLIP running on dynamic platforms. ASME L. Dyn. Syst. Contr. 2021;1(2).

[29] Kapitza P. Dynamic stability of a pendulum with an oscillating point of support. Zh. Eksp. Teor. Fiz. 1951;21:588.

[30] Tannuri EA, Sparano JV, Simos AN, Da Cruz JJ. Estimating directional wave spectrum based on stationary ship motion measurements. App. Ocean Res. 2003;25(5):243–61.

[31] Kajita S, Kanehiro F, Kaneko K, Fujiwara K, Harada K, Yokoi K, et al. Biped walking pattern generation by using preview control of zero-moment point. In: Proc. IEEE Int. Conf. Rob. Autom. vol. 2. 2003, p. 1620–6.

[32] Paredes VC, Hereid A. Resolved motion control for 3d underactuated bipedal walking using linear inverted pendulum dynamics and neural adaptation. In: Proc. IEEE Int. Conf. Int. Rob. Sys.. 2022, p. 6761–7.

[33] Iqbal A, Gu Y. Extended capture point and optimization-based control for quadrupedal robot walking on dynamic rigid surfaces. In: Proc. IFAC Mod. Est. Contr. Conf. vol. 54, no. 20. 2021, p. 72–7.

[34] Yoon H-K, Lee G-J, Lee D-K. Development of the motion monitoring system of a ship. J. Navig. Port. Res. 2008;32(1):15–22.

[35] Gahlinger PM. Cabin location and the likelihood of motion sickness in cruise ship passengers. J. Trav. Med. 2000;7(3):120–4.

[36] Benjamin TB, Ursell FJ. The stability of the plane free surface of a liquid in vertical periodic motion. Proc. R. Soc. London 1954;225(1163):505–15.

[37] Farkas M. Periodic motions. Springer; 2013.

[38] Phelps III F, Hunter Jr J. An analytical solution of the inverted pendulum. Amer. J. Phys. 1965;33(4):285–95.

[39] Werth F, Gheorghe N, Major F, Gheorghe V, Werth G, Major S, Werth G. Charged particle traps: Physics and techniques of charged particle field confinement. In: Springer Ser. Atom., Opt., Plas. Phys., Springer; 2005.

[40] Bateman H. Higher transcendental functions [Volumes I-Iii], vol. 1. McGraw-Hill Book Company; 1953.

[41] Dougall J. The solution of Mathteu's differential equation. Proc. Edinb. Math. Soc. 1915;34:176–96.

[42] Floquet G. Sur les équations différentielles linéaires à coefficients périodiques. In: Annales scientifiques de l'École normale supérieure, vol. 12. 1883, p. 47–88.

[43] Gu Y, Gao Y, Yao B, Lee CG. Global-position tracking control for three-dimensional bipedal robots via virtual constraint design and multiple Lyapunov analysis. ASME J. Dyn. Syst. Meas. Contr. 2022;144(11):111001.

[44] Winkler AW, Bellicoso CD, Hutter M, Buchli J. Gait and trajectory optimization for legged systems through phase-based end-effector parameterization. IEEE Rob. Autom. L. 2018;3(3):1560–7.

[45] Xin S, Orsolino R, Tsagarakis N. Online relative footstep optimization for legged robots dynamic walking using discrete-time model predictive control. In: Proc. IEEE/RSJ Int. Conf. Intel. Rob. Syst.. 2019, p. 513–20.

[46] Wächter A, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math. Prog. 2006;106(1):25–57.

[47] Grizzle JW, Abba G, Plestan F. Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. IEEE Trans. Autom. Contr. 2001;46(1):51–64.

[48] Iqbal A, Veer S, Gu Y. Asymptotic stabilization of aperiodic trajectories of a hybrid-linear inverted pendulum walking on a vertically moving surface. In: Proc. Amer. Contr. Conf.. 2023, p. 3030–5.

[49] Gao Y, Paredes V, Hereid A, Gu Y. Exponential stabilization of periodic LIP walking on a horizontally moving surface. Proc. Dyn. Walk. Conf. 2022.

[50] Gao Y, Gong Y, Paredes V, Hereid A, Gu Y. Time-varying alip model and robust foot-placement control for underactuated bipedal robotic walking on a swaying rigid surface. In: Proc. Amer. Contr. Conf.. 2023, p. 3282–7.

[51] Bledt G, Powell MJ, Katz B, Di Carlo J, Wensing PM, Kim S. MIT cheetah 3: design and control of a robust, dynamic quadruped robot. In: Proc. IEEE/RSJ Int. Conf. Intel. Rob. Syst.. 2018, p. 2245–52.

Amir Iqbal is a Ph.D. candidate in Mechanical Engineering at the University of Massachusetts Lowell and a Research Intern at Purdue University. He received a B.Tech. degree in Aerospace Engineering from the Indian Institute of Space Science and Technology, Thiruvananthapuram, Kerala, India in 2012. He was a former Scientist/Engineer at the ISRO Satellite Center, Bengaluru, India. His research interests include legged locomotion, control theory, trajectory optimizations, and exoskeleton control.

Sushant Veer is a Senior Research Scientist at NVIDIA Research. In the past he was a Postdoctoral Research Associate in the Mechanical and Aerospace Engineering Department at Princeton University. He received his Ph.D. in Mechanical Engineering from the University of Delaware in 2018 and a B. Tech. in Mechanical Engineering from the Indian Institute of Technology Madras in 2013. His research interests lie at the intersection of control theory and machine learning with the goal of enabling safe decision making for robotic systems. He has received the Yeongchi Wu International Education Award (2013 International Society of Prosthetics and Orthotics World Congress), Singapore Technologies Scholarship (ST Engineering Pte Ltd), and Sri Chinmay Deodhar Prize (Indian Institute of Technology Madras).

Yan Gu received the B.S. degree in Mechanical Engineering from Zhejiang University, China, in June 2011 and the Ph.D. degree in Mechanical Engineering from Purdue University, West Lafayette, IN, USA, in August 2017. She joined the faculty of the School of Mechanical Engineering at Purdue University in July 2022. Prior to joining Purdue, she was an Assistant Professor with the Department of Mechanical Engineering at the University of Massachusetts Lowell. Her research interests include nonlinear control, hybrid systems, legged locomotion, and wearable robotics. She received the National Science Foundation CAREER Award in 2021 and the Office of Naval Research Young Investigator Program Award in 2023.