RESEARCH ARTICLE

Haitao YU, Haibo GAO, Liang DING, Zongquan DENG

Sagittal SLIP-anchored task space control for a monopode robot traversing irregular terrain

Abstract As a well-explored template that captures the essential dynamical behaviors of legged locomotion on sagittal plane, the spring-loaded inverted pendulum (SLIP) model has been extensively employed in both biomechanical study and robotics research. Aiming at fully leveraging the merits of the SLIP model to generate the adaptive trajectories of the center of mass (CoM) with maneuverability, this study presents a novel two-layered sagittal SLIP-anchored (SSA) task space control for a monopode robot to deal with terrain irregularity. This work begins with an analytical investigation of sagittal SLIP dynamics by deriving an approximate solution with satisfactory apex prediction accuracy, and a two-layered SSA task space controller is subsequently developed for the monopode robot. The higher layer employs an analytical approximate representation of the sagittal SLIP model to form a deadbeat controller, which generates an adaptive reference trajectory for the CoM. The lower layer enforces the monopode robot to reproduce a generated CoM movement by using a task space controller to transfer the reference CoM commands into joint torques of the multi-degree of freedom monopode robot. Consequently, an adaptive hopping behavior is exhibited by the robot when traversing irregular terrain. Simulation results have demonstrated the effectiveness of the proposed method.

Keywords legged robots, spring-loaded inverted pendulum, task space control, apex return map, deadbeat control, irregular terrain negotiation

1 Introduction

Diverse in terms of morphology and gait patterns [1],

Haitao YU (✉), Haibo GAO, Liang DING, Zongquan DENG
State Key Laboratory of Robotics and Systems, Harbin Institute of Technology, Harbin 150001, China
E-mail: yht@hit.edu.cn

cursorial mammals with legs used to propel their bodies and traverse complicated terrains display astonishing performance that outclasses that of any man-made walking device. Such dexterity and superiority when interacting with the surrounding environment have widely attracted the interest of both bio-mechanical and robotics researchers. Recently, numerous legged prototypes [2–5] have been developed extensively with the aim of reproducing these behaviors. However, a huge gap still exists in that artificial apparatuses have yet to attain comparable performance as those of animals.

Hopping, as the most ordinary movement observed in kangaroo and galago, is the fundamental gait pattern from which other complex gaits, such as bipedal running, quadrupedal trotting, bounding, and galloping, can be further evolved [6]. The spring-loaded inverted pendulum (SLIP) is regarded a versatile template in capturing the essential characteristics of hopping with satisfactory trajectory prediction accuracy of the center of mass (CoM); SLIP is also regarded effective in fitting biology data since it was first established in Ref. [7]. The sagittal SLIP dynamics, particular in stance phase, is intrinsically nonlinear. The exact analytical closed-form solution of sagittal SLIP dynamics is unavailable due to the non-integrable coupled terms contained in the stance formulation. Several studies in this field provide analytical approximation as an alternative. An analytical approximation is proposed in Ref. [8]. Picard's iteration with mean-value theorem is employed to derive a closed-form expression of sagittal SLIP dynamics in stance phase. However, the accuracy of the derived approximation relies on the iteration steps, and this reliance restricts the practicability of the solution when online computational cost is crucial for the motion planning and gait control of a legged robot. Ghigliazza et al. [9] presented another approximate solution for an ideal SLIP template based on the negligible gravity assumption, and stable hopping gait is also fulfilled with the fixed-leg reposition policy. Geyer et al. [10] developed an approximation by assuming the angular momentum of the SLIP model conserved in the

stance phase and offered a straightforward solution in simple form that works effectively with the symmetric movement of CoM. Arslan et al. [11] further improved this approximation by proposing a gravity correction scheme to compensate the effect of gravity on angular momentum for highly non-symmetric trajectories. Consequently, a two-step iteration with high accuracy is provided for predicting the apex state of the SLIP system. Shahbazi et al. [12] extended the approach from a single-leg configuration to a bipedal case. An analytical approximate representation for double-stance walking is derived and then used to construct an apex return map (ARM) without relying on numerical integration. Our previous work [13] presented a perturbation-based analytical approximation for the sagittal SLIP dynamics in stance phase and is valid for both the symmetric and asymmetric trajectories of CoM, and it can preserve mathematical tractability and high apex prediction accuracy.

The merits of simple formulation in mathematics and self-stability in stride-to-stride movement renders the SLIP model easy to implement for the locomotion control for legged robots. A SLIP template-based controller operating with an impact collision compensation scheme is presented in Ref. [14]. The spring–mass dynamics is numerically solved to generate the reference CoM trajectory for a segmented robotic leg. The SLIP model with leg actuation is reported in Ref. [15]. Active SLIP dynamics is analytically resolved to reduce online computational cost, and a corresponding two-part control strategy is developed to add/remove system energy from a series of strides of a single-leg robot. Reliable adaptive hopping performance is achieved in the presence of terrain perturbations. The swing-leg retraction policy was first proposed in Ref. [16] and further adapted in Ref. [17] to improve the hopping stability and robustness of the SLIP model. In Ref. [18], a comparison of energetic efficiency between the Raibert-style controller with embedded SLIP dynamics and the optimal trajectory-based controller for a three-link mono-pode model is conducted using cost of transport as a criterion. A nonlinear predictive control scheme is constructed in Ref. [19] to steer the SLIP trajectory over rough terrain footholds. Garofalo and Albu-Schäffer [20] developed a controller based on the dual-SLIP model to stabilize a five-link fully actuated bipedal walking robot, in which the reference CoM trajectory is generated by numerically computing the SLIP dynamics. Aside from the sagittal SLIP model used to control legged robots, the 3D-SLIP model can be applied to generate the reference CoM trajectory for humanoids [21], in which high-speed stabilized running as fast as 6.5 m/s is achieved. Subsequently, turning gait control with 3D-SLIP model is realized in Ref. [22] by modifying the control framework of straightforward running, which is established in Ref. [23].

Aiming at fully leveraging the aforementioned benefits of the sagittal SLIP model, this study presents a SLIP-anchored task space control for a monopode robot to deal with terrain perturbations. The main contributions of this study can be summarized as follows:

1) An analytical approximation-based deadbeat controller for the sagittal SLIP model is developed to regulate apex height and velocity. In stance phase, a leg adjustment policy with piecewise-constant stiffness is proposed to match the energy variation between the current apex and the desired apex. In flight phase, a falling time-dominant touchdown (TD) policy for the swing leg is proposed to adapt to terrain irregularities without priori ground truth knowledge.

2) A sagittal SLIP-anchored double-layered task space formulation for a monopode robot is presented. The high layer employs the SLIP model to generate an adaptive reference CoM trajectory for the monopode robot. The bottom layer employs the task space controller to enforce the robot to behave with SLIP dynamics when dealing with terrain perturbations.

3) The simulation results demonstrate the effectiveness of the proposed controller in steering the monopode robot. The robot not only can achieve stable hopping with desired apex height and velocity, but it also has the capability of traversing irregular terrains.

The remainder of this paper is structured as follows. Section 2 briefly reviews the general control framework by elaborating the control objective and the sagittal SLIP-anchored double-layered control architecture. The sagittal SLIP model with the corresponding analytical representation is presented in Section 3, followed by the task space controller design in Section 4. The simulation results are given in Section 5. Section 6 presents the discussions on the superiority of the proposed SLIP-anchored task space controller relative to the traditional SLIP controller. This paper ends with conclusions and perspectives of future work in Section 7.

2　General control framework

2.1　Control objective

Self-stability and ease of maneuverability are the main merits of the sagittal SLIP model when applied to the motion planning and gait control of legged robots. The operation of the former property may be valid in a wide range of model parameter combinations, as reported in Refs. [10,23]. The latter property simplifies the control of the SLIP model by tuning the swing-leg TD angle during flight and the leg stiffness during stance, respectively. The main purpose of this work is to endow the fully actuated monopode robot with the aforementioned merits of the sagittal SLIP model. In this regard, the control objective is to reproduce the sagittal SLIP model behavior on the

monopode robot as the target dynamics, in which the expected resulted is adaptive and robustness hopping performance in the presence of terrain perturbations.

2.2   Sagittal SLIP-anchored double-layered control architecture

The sagittal SLIP-anchored task space control architecture is illustrated in Fig. 1. The high layer consists of a sagittal SLIP model that employs a deadbeat controller to generate the reference CoM trajectory for the monopode robot. The derived analytical approximation can be regarded a representation of SLIP dynamics, and it provides apex prediction to resolve the shooting problem of the deadbeat controller, which will be detailed in Section 3.3. The TD angle, together with the leg stiffness, is regarded the tunable parameter of the SLIP model, in which adaptive movements can be produced in the presence of terrain perturbations. Scheduled by the finite state machine (FSM) that switches the flight/stance phase according to the contact detection triggered by monopode-ground interaction, the bottom layer transfers the target CoM trajectory into individual joint commands via the task space controller, thus enforcing the whole body of the actuated robot to bounce in accordance with the SLIP dynamics generated by the high layer.
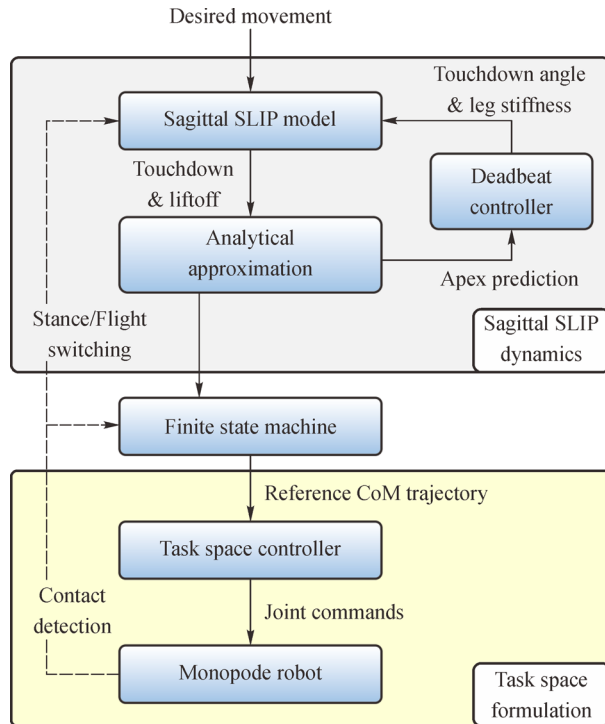

Fig. 1 Schematic of the sagittal SLIP-anchored task space control architecture. SLIP: Spring-loaded inverted pendulum.

3   Sagittal SLIP model and analytical representations

3.1   Sagittal SLIP model and the analytical approximate solution

The sagittal SLIP model with coordinates and relevant parameters (Fig. 2(a)) is represented by point mass m_s that connects a massless telescopic leg to the hip at rest length r_0. The entire gait cycle, defined as a complete mapping from the current apex to the next apex (Fig. 2(b)), entails a flight phase (when the leg swings in aerial mode) and a stance phase (when the leg touches the ground). Two switching events will be defined as TD triggered from flight to stance and lift-off (LO) from stance to flight. The stance leg undergoes compression and decompression with tunable stiffness k_s. The swing leg is assumed to be freely pre-positioned in its orientation at TD with the angle of attack (AoT) α_{TD}. The toe is assumed to be a fixed pivot, and no slipping is expected to occur during stance.

Given the CoM position vector of the SLIP model in a polar coordinate and expressed as $q_s = (-r\sin\theta, r\cos\theta)^T$, the dynamics for the stance phase is given by

$$\ddot{q}_s = -\frac{F_{spr}}{m_s} + g_s, \tag{1}$$

where the leg force F_{spr} resulting from the linear spring satisfies $F_{spr} = k_s(q_s - q_{s0}) \in R^2$, where q_{s0} is the leg length vector at TD given by $q_{s0} = (-r_{TD}\sin\theta_{TD}, r_{TD}\cos\theta_{TD})^T$, and g_s denotes the gravitational force vector given by $g_s = (0, -g)^T$. A system in flight phase is solely governed by gravity and exhibits a ballistic trajectory, and its dynamics is formulated as

$$\ddot{q}_f = g_s, \tag{2}$$

where q_f denotes the CoM position vector with the Cartesian coordinate form $q_f = (x, y)^T$.

Incidentally, the exact solution of the sagittal SLIP dynamics in stance phase is unknown due to the coupled non-integrable terms in Eq. (1) [10]. Alternatively, we employ in this study an analytical approximation, in which the proven high apex prediction accuracy has been derived in our previous work [13], instead of using the nonlinear Eq. (1) to formulate the entire SLIP dynamics in stance phase. The two switching event mappings for TD and LO can be further defined as

$$\begin{cases} \Delta_{S \to F} = \{(y, r) | y - r_0\sin\alpha_{TD} = 0, \dot{r} < 0\}, & \text{for TD,} \\ \Delta_{F \to S} = \{(r, \dot{r}) | r - r_0 = 0, \dot{r} > 0\}, & \text{for LO.} \end{cases} \tag{3}$$

3.2   Apex return map

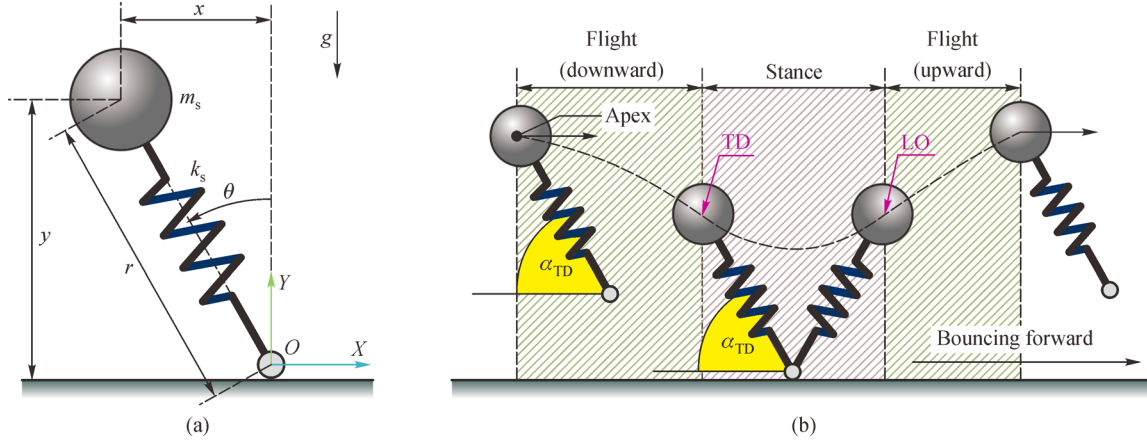ARM, as a reduced-order discrete version of the Poincaré

Fig. 2 Illustration of the sagittal SLIP model. (a) Coordinates and variable definitions; (b) entire gait cycle, including flight and stance sub-phase.

map used to analyze the periodic behaviors of hybrid systems, is utilized in the present study to characterize the stride-to-stride hopping behavior of the SLIP system. We take the so-called Poincaré section at the apex state, which is defined by the following vector at the ith step:

$$\boldsymbol{S}_i = (y_a(i), \dot{x}_a(i))^{\mathrm{T}}, \tag{4}$$

where $y_a(i)$ and $\dot{x}_a(i)$ denote apex height and velocity, respectively. As shown in Fig. 3, the ARM generally consists of three sub-maps, namely, the map $\boldsymbol{P}_{\mathrm{fd}}$ of the downward flight from the current apex to TD, the map $\boldsymbol{P}_{\mathrm{st}}$ of the stance from TD to LO, and the map $\boldsymbol{P}_{\mathrm{fu}}$ of the upward flight from LO to the succeeding apex.
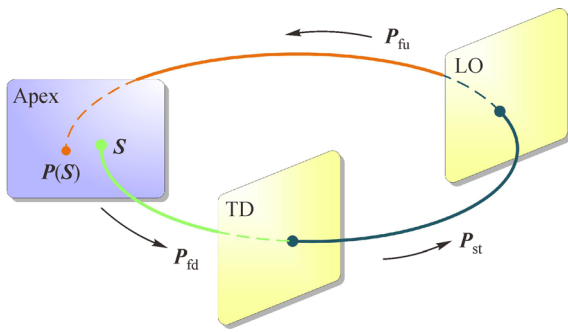
Fig. 3 Composition of ARM, including the three sub-maps of $\boldsymbol{P}_{\mathrm{fd}}$, $\boldsymbol{P}_{\mathrm{st}}$, and $\boldsymbol{P}_{\mathrm{fu}}$.

Aiming at regulating the apex vector during hopping, two variables are selected in this study as the control inputs for the SLIP model. The AoT α_{TD} of the swing leg with a pre-positioning policy in flight phase is chosen as one of the control inputs. Leg stiffness k_s in stance phase with a piecewise constant property is chosen as another control input (Fig. 4). This stance phase is further divided into a

compression sub-phase (with constant stiffness k_c) and a decompression sub-phase (with constant stiffness k_d). The instant stiffness variation at the instant stiffness variation at the bottom (BM) is commonly used in Refs. [13,22]. Let the control input vector be $\boldsymbol{u}(n) = \left(\alpha_{\mathrm{TD}}(n), k_c(n), k_d(n)\right)^{\mathrm{T}}$. ARM can thus be written as

$$\boldsymbol{S}_{n+1} = \boldsymbol{P}(\boldsymbol{S}_n, \boldsymbol{u}_n) = (\boldsymbol{P}_{\mathrm{fd}} \circ \boldsymbol{P}_{\mathrm{st}} \circ \boldsymbol{P}_{\mathrm{fu}})(\boldsymbol{S}_n, \boldsymbol{u}_n). \tag{5}$$

The above ARM formulation can be obtained by numerical integration only (i.e., fourth-order Runge–Kutta approach), considering that the stance map $\boldsymbol{P}_{\mathrm{st}}$ cannot be analytically solved. We therefore build an approximate apex return map (A²RM) that employs the previous analytical approximation presented in Ref. [13] to calculate $\boldsymbol{P}_{\mathrm{st}}$ and fully avoid the numerical process.

$$\tilde{\boldsymbol{S}}_{n+1} = \tilde{\boldsymbol{P}}(\boldsymbol{S}_n, \boldsymbol{u}_n) = \left(\boldsymbol{P}_{\mathrm{fd}} \circ \tilde{\boldsymbol{P}}_{\mathrm{st}} \circ \boldsymbol{P}_{\mathrm{fu}}\right)(\boldsymbol{S}_n, \boldsymbol{u}_n), \tag{6}$$

where the superscript "~" denotes the maps or the variable derived by using the analytical approximation in Ref. [13].

3.3 Deadbeat controller

Given the current apex vector $\boldsymbol{S}_0$ and the target apex vector $\boldsymbol{S}_d$, the deadbeat control applied to achieve the target apex height and velocity is formulated as an optimization problem as follows:

$$\begin{aligned} \min_{\alpha_{\mathrm{TD}}, k_c, k_d} & \left\| \boldsymbol{S}_d - \tilde{\boldsymbol{P}}(\boldsymbol{S}_0, \boldsymbol{u}) \right\|, \\ \text{with} \quad & \boldsymbol{S}_0 = (y_{a0}, \dot{x}_{a0})^{\mathrm{T}}, \\ & \boldsymbol{S}_d = (y_d, \dot{x}_d)^{\mathrm{T}}, \\ & \boldsymbol{u} = (\alpha_{\mathrm{TD}}, k_c, k_d)^{\mathrm{T}}. \end{aligned} \tag{7}$$
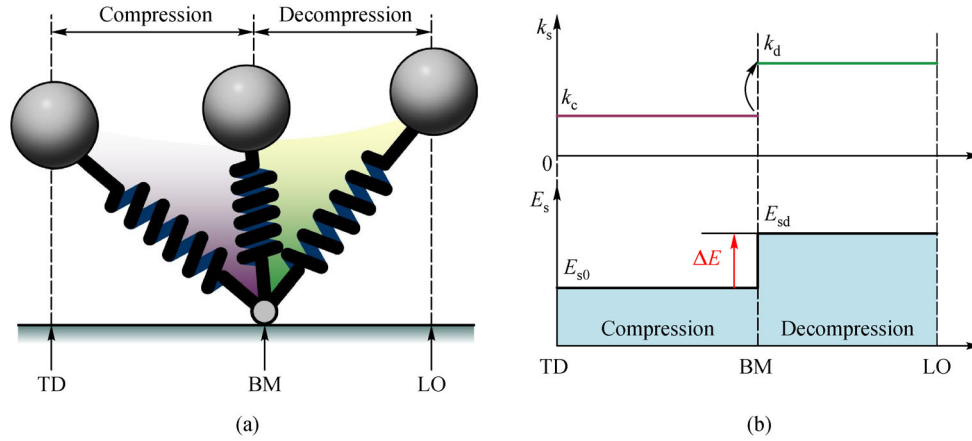
Fig. 4 Schematic of the variable stiffness policy with piecewise constant profile for the compression and decompression phases. (a) Division of the stance phase, with the instant stiffness variation at the bottom defined as the maximum leg compression; (b) variable stiffness spring with piecewise-constant leg stiffness.

The relationship between leg stiffness during compression and leg stiffness during decompression can be further determined by using the system energy matching of the current apex S_0 and the target apex S_d.

$$k_d = k_c + \frac{2\Delta E}{(r_0 - \tilde{r}_{BM})^2}, \tag{8}$$

where $\tilde{r}_{BM}$ is the approximate prediction of the leg length at BM, and ΔE is the energy variation between S_0 and S_d to obtain

$$\Delta E = \frac{1}{2}m_s(\dot{x}_d^2 - \dot{x}_0^2) + m_s g(y_d - y_0). \tag{9}$$

In this manner, the leg stiffness collection $\{k_c, k_d\}$ of the SLIP model can be transformed into AoT α_{TD} variables if the current apex vector S_0 and the target apex vector S_d are given. Consequently, the AoT α_{TD} of the swing leg can be determined by solving a 1D shooting problem as follows:

$$\alpha_{TD} = \underset{\frac{\pi}{4} < \alpha_{TD} < \frac{\pi}{2}}{\arg\min} \left\| S_d - \tilde{P}\left(S_0, (\alpha_{TD}, k_c, k_d)^T\right) \right\|. \tag{10}$$

As shown below, an algorithm that determines the TD angle α_{TD} together with the leg stiffness k_c, k_d is applied to solve the optimization problem Eq. (7):

Algorithm: Determination of the touchdown angle and the leg stiffness in solving problem Eq. (7)

Input:

The initial apex state S_0

The target apex state S_d

The initial leg stiffness k_s

Output:

The touchdown angle α_{TD}

The leg stiffness k_c and k_d

---

1. Initialize the current leg stiffness $k_c \leftarrow k_s$

2. Compute the energy variation ΔE by using Eq. (9)

3. **for** $\alpha_{TD} = \pi/4$ to $\pi/2$ **do**

4. Compute the approximation of the leg length at BM $\tilde{r}_{BM}$

5. $k_d \leftarrow k_c + 2\Delta E/(r - \tilde{r}_{BM})^2$

6. Compute sub-maps P_{fd}, $\tilde{P}_{st}$, P_{fu}

7. Compute the A²RM $\tilde{P} \leftarrow P_{fd} \circ \tilde{P}_{st} \circ P_{fu}$

8. Compute the predicted apex state $\tilde{S}_{n+1} \leftarrow \tilde{P}\left(S_0, (\alpha_{TD}, k_c, k_d)^T\right)$

9. $\alpha_{TD} = \arg\min \left\| S_d - \tilde{P}\left(S_0, (\alpha_{TD}, k_c, k_d)^T\right) \right\|$

10. **end for**

11. **redo** Steps 3 and 4

12. **return** α_{TD}, k_c, and k_d

13 Update the leg stiffness for the coming compression sub-phase with $k_s \leftarrow k_d$

14. **end algorithm**

The leg stiffness of the SLIP model during the compression sub-phase of the first gait cycle requires a customized initial value k_s to maintain the operation of the solving procedure. Furthermore, leg stiffness shall be updated by using Eq. (8) to match the energy variation, as illustrated in Fig. 4(b). Once the stiffness k_d in the decompression sub-phase is updated, the SLIP model will retain this value as the leg stiffness k_c in the forthcoming compression sub-phase. Thus far, the deadbeat controller with control input (α_{TD}, k_c, k_d) has been completely constructed for hopping on a flat terrain, in which the target apex state is approached in one stride. However, the devised deadbeat controller is twofold; it has a swing-leg pre-positioned controller with a preset AoT α_{TD} and a stance-leg controller with a piecewise-constant leg stiffness adjustment. The twofold scheme suggests that AoT

α_{TD} is the sole tunable control input for the swing leg during flight, while the leg stiffness collection $\{k_c, k_d\}$ is the remaining control input for the stance leg during stance.

3.4   Extension to an irregular terrain case

Traversing irregular terrains is an essential requirement for legged robots interacting with complex environments. In this section, we extend the developed deadbeat controller in Section 3.3 from one capable of flat surface hopping to one suitable for irregular terrain cases to achieve constant absolute altitude in the presence of terrain perturbations, as shown in Fig. 5.
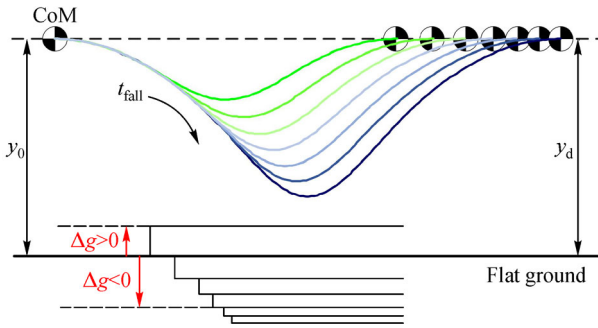

Fig. 5 Hopping with constant absolute altitude when traversing irregular terrains in the SLIP model. Colored curves represent CoM trajectories of different terrain irregularities. y_0 and y_d represents the initial and target hopping height, respectively.

We first introduce a perturbation indicator Δg to characterize terrain irregularities, in which $\Delta g > 0$ represents a convex profile and vice versa. Thus, the control problem of the SLIP model with terrain perturbation Δg can be transferred into the following shooting problem:

$$\min_{\alpha_{TD}, k_c, k_d} \left\| y_d - \tilde{P}_y(y_0, \boldsymbol{u}) \right\| \quad \text{with} \quad \boldsymbol{u} = (\alpha_{TD}, k_c, k_d)^T, \quad (11)$$

where $\tilde{P}_y$ is the corresponding A²RM based on the initial apex height y_0 and the control input $\boldsymbol{u}$. A constant absolute altitude requirement implies that $\Delta E = 0$, considering that the system energy of the sagittal SLIP model is conservative. According to Eq. (8), we have

$$k_d = k_c. \quad (12)$$

Then, the deadbeat controller is reduced to a 1D shooting problem as follows:

$$\alpha_{TD} = \underset{\frac{\pi}{4} < \alpha_{TD} < \frac{\pi}{2}}{\operatorname{argmin}} \left| y_d - \Delta g - \tilde{P}_y\left(y_0 - \Delta g, (\alpha_{TD}, k_c)^T\right) \right|, \quad (13)$$

which can be regarded an extension of Eq. (10) with the perturbation indicator Δg. On the purpose of devising a robust controller without prior knowledge of ground truth,

we introduce a time-scale variable t_{fall} measured from the apex state, as reported in Ref. [23] to record the time of falling. In this manner, the pre-positioning policy for the swing leg with AoT α_{TD} during downward flight can be transformed into a time-dependent policy by using

$$y_d - \Delta g = r_0 \sin\alpha_{TD} + \frac{1}{2}g t_{fall}^2. \quad (14)$$

Substituting Eq. (14) into Eq. (13) yields the falling-time relevant swing-leg policy as follows:

$$\alpha_{TD}(t_{fall}) = \underset{\frac{\pi}{4} < \alpha_{TD} < \frac{\pi}{2}}{\operatorname{argmin}} \left| r_0 \sin\alpha_{TD} + \frac{1}{2}g t_{fall}^2 \right.$$

$$\left. - \tilde{P}_y\left(r_0 \sin\alpha_{TD} + \frac{1}{2}g t_{fall}^2, (\alpha_{TD}, k_c)^T\right) \right|. \quad (15)$$

Thus far, we have completed the deadbeat controller design for the hopping of the sagittal SLIP model in irregular terrains. The resulting trajectory will be utilized as the reference CoM trajectory for the monopode robot and then reproduced by the task space controller, as presented in the following section.

4   Task space controller design

4.1   Dynamics of the monopode robot

The rigid body model of the monopode robot in this study (Fig. 6) has a two-segmented leg mounted at the hip of the robot's upper body, with rotary actuation located at the hip and the knee. The CoM of the upper body is assumed to coincide at the hip, and the toe of the foot is considered a massless point, as shown in Fig. 6(a). By considering the upper body position, the CoM coordinates of the two segments can be further given by

$$\begin{cases} x_1 = x_b - l_2 \sin q_2 - (l_1 - l_{C1}) \sin(q_2 - q_1), \\ y_1 = y_b - l_2 \cos q_2 - (l_1 - l_{C1}) \cos(q_2 - q_1), \end{cases} \quad (16)$$

$$\begin{cases} x_2 = x_b - (l_2 - l_{C2}) \sin q_2, \\ y_2 = y_b - (l_2 - l_{C2}) \cos q_2, \end{cases} \quad (17)$$

where (x_1, y_1) and (x_2, y_2) are the CoM coordinates of the thigh and the shank, respectively, (x_b, y_b) is the hip joint position on the upper body, l_1 and l_{C1} are the segment length and the CoM bias length of the thigh, respectively, l_2 and l_{C2} are the segment length and the CoM bias length of the shank, respectively, q_1 is the joint angle of the shank (anticlockwise measured) actuated by the torque u_1, while q_2 is the joint angle of the thigh (clockwise measured) actuated by the torque u_2. The CoM of the monopode robot are thus given by
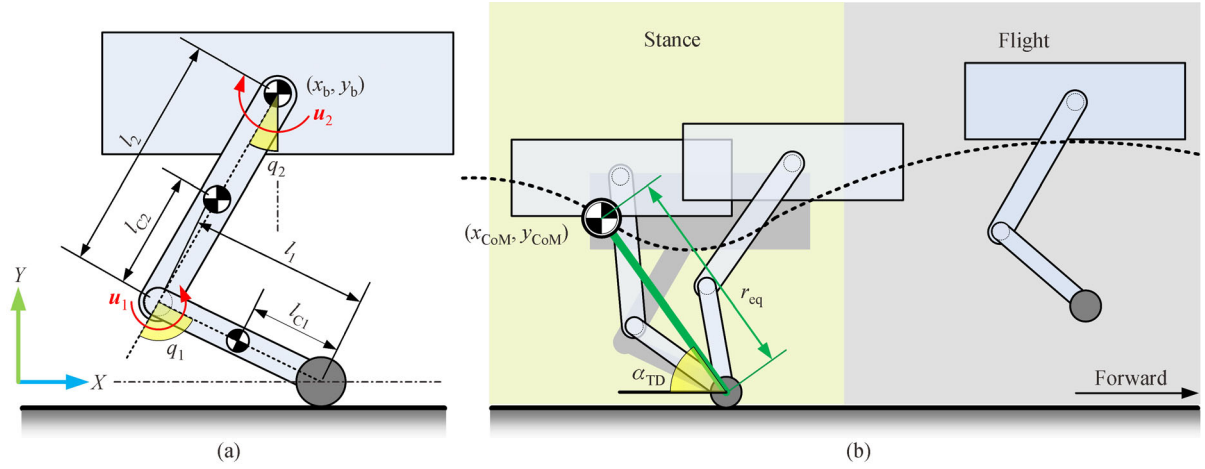
Fig. 6 Rigid model of the monopode robot. (a) Leg configuration and relevant parameters; (b) entire hopping gait cycle with sub-phase division. AoT α_{TD} and virtual equivalent leg r_{eq} are the corresponding parameters used in the SLIP model.

$$\begin{cases} x_{\mathrm{CoM}} = \dfrac{m_{\mathrm{b}}x_{\mathrm{b}} + m_1 x_1 + m_2 x_2}{m_{\mathrm{b}} + m_1 + m_2}, \\ y_{\mathrm{CoM}} = \dfrac{m_{\mathrm{b}}y_{\mathrm{b}} + m_1 y_1 + m_2 y_2}{m_{\mathrm{b}} + m_1 + m_2}, \end{cases} \qquad (18)$$

where m_{b}, m_1, and m_2 are the mass variables of the upper body, shank, and thigh, respectively. Let $q = [q_1, q_2]^{\mathrm{T}}$ and $q_{\mathrm{CoM}} = [x_{\mathrm{CoM}}, y_{\mathrm{CoM}}]^{\mathrm{T}}$ be the joint and the CoM vectors of the robot, respectively. Then, Jacobian J_{CoM} can be defined as $\dot{q}_{\mathrm{CoM}} = J_{\mathrm{CoM}}\dot{q}$. The variables in stance/flight phase can be distinguished by using the subscripts "S" and "F" to represent stance and flight in the derivations.

In the stance phase, the dynamics of the robot can be written as

$$M_{\mathrm{S}}\ddot{q}_{\mathrm{S}} + C_{\mathrm{S}}\dot{q}_{\mathrm{S}} + G_{\mathrm{S}} = S_{\mathrm{S}}^{\mathrm{T}}\tau_{\mathrm{S}} + J_{\mathrm{S}}^{\mathrm{T}}F_{\mathrm{Gnd}}, \qquad (19)$$

where M_{S}, C_{S}, G_{S}, and S_{S} denote the inertia matrix, Coriolis/centripetal vector, gravity vector, and input selection matrix for the actuated joints, respectively. $S_{\mathrm{S}}^{\mathrm{T}}\tau_{\mathrm{S}} = [u_1, u_2]^{\mathrm{T}}$ represents the actuated joint torques. F_{Gnd} is the collection of the ground reaction forces exerted on the toe. J_{S} is the Jacobian associated with F_{Gnd}. $q_{\mathrm{S}} = [x_{\mathrm{b}}, y_{\mathrm{b}}, q_{1\mathrm{S}}, q_{2\mathrm{S}}]^{\mathrm{T}}$ represents the generalized coordinate. Foot–ground contact is modeled as an inelastic impact, in which state transition at the time instant of impact satisfies

$$\dot{q}_{\mathrm{S}}^{+} = \underbrace{(I - M_{\mathrm{S}}^{-1}J_{\mathrm{S}}^{\mathrm{T}}\Lambda_{\mathrm{S}}J_{\mathrm{S}})}_{N_{\mathrm{S}}} \dot{q}_{\mathrm{S}}^{-}, \qquad (20)$$

where $\Lambda_{\mathrm{S}} = (J_{\mathrm{S}}M_{\mathrm{S}}^{-1}J_{\mathrm{S}}^{\mathrm{T}})^{-1}$, and the superscripts "+" and "–" denote an instance immediately before and after the impact, respectively.

In flight phase, the dynamics of the robot is given by

$$M_{\mathrm{F}}\ddot{q}_{\mathrm{F}} + C_{\mathrm{F}}\dot{q}_{\mathrm{F}} + G_{\mathrm{F}} = S_{\mathrm{F}}^{\mathrm{T}}\tau_{\mathrm{F}}, \qquad (21)$$

where M_{F}, C_{F}, G_{F}, and S_{F} denote the inertia matrix, Coriolis/centripetal vector, gravity vector, and input selection matrix for the actuated joints, respectively. $S_{\mathrm{F}}^{\mathrm{T}}\tau_{\mathrm{F}} = [u_1, u_2, 0, 0]^{\mathrm{T}}$ represents the actuated joint torques. $q_{\mathrm{F}} = [q_{1\mathrm{F}}, q_{2\mathrm{F}}, x_{\mathrm{b}}, y_{\mathrm{b}}]^{\mathrm{T}}$ represents the generalized coordinate. The vector $(x_{\mathrm{b}}, y_{\mathrm{b}})^{\mathrm{T}}$ is added to determine the body position in flight as the toe leaves the ground.

4.2 Finite state machine

Similar to the switching events defined in Eq. (3), the switching conditions between stance and flight of the monopode robot are given as follows:

(i) From flight to stance switching condition:

$$S_{\mathrm{F}\to\mathrm{S}} = \left\{ (y_{\mathrm{CoM}}, r_{\mathrm{eq}}) \middle| y_{\mathrm{CoM}} - r_{\mathrm{eq}}\sin\alpha_{\mathrm{TD}} = 0, \dot{r}_{\mathrm{eq}} < 0 \right\}, \qquad (22)$$

where r_{eq} is the virtual equivalent leg length that connects CoM and the toe (see Appendix for details).

(ii) From stance to flight switching condition:

$$S_{\mathrm{S}\to\mathrm{F}}^{A} = \left\{ (r_{\mathrm{eq}}, \dot{r}_{\mathrm{eq}}) \middle| r_{\mathrm{eq}}(t_A) - r_0 = 0, \dot{r}_{\mathrm{eq}}(t_A) > 0 \right\}, \quad (23)$$

$$S_{\mathrm{S}\to\mathrm{F}}^{B} = \left\{ F_{\mathrm{Gnd}} \middle| F_{\mathrm{Gnd}}(t_B) = \mathbf{0} \right\}, \qquad (24)$$

where a_{LO} is the LO angle measured from the positive X-axis to the virtual equivalent leg r_{eq}. t_A and t_B are the corresponding time instants that fulfill Eqs. (19) and (20), respectively. Two switching conditions can be adopted to determine the stance-to-flight transition, and the formulations are by $S_{\mathrm{S}\to\mathrm{F}}^{A}$ and $S_{\mathrm{S}\to\mathrm{F}}^{B}$. The former condition is enabled when the virtual equivalent leg of the corresponding SLIP model approaches rest length r_0. The latter condition is enabled when the toe of the monopode robot loses ground contact. The schematic diagram of the FSM

for the monopode robot is illustrated in Fig. 7. The actual switching time of the stance-to-flight transition depends on the value of $\min(t_A, t_B)$.
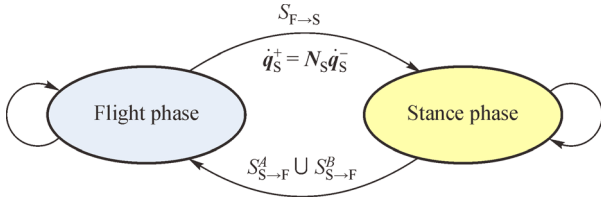

Fig. 7 FSM for monopode robot hopping.

4.3 Task space controller

Considering that the adaptive reference CoM trajectory can be generated by the sagittal SLIP model based on the devised deadbeat controller, the remaining task of this study is to enforce the monopode robot to reproduce the target SLIP dynamics by using the task space controller, as presented in Ref. [24].

First, we rewrite the target dynamics generated by the sagittal SLIP model to produce the reference CoM trajectory as follows:

$$\ddot{\boldsymbol{q}}_{\text{CoM}}^{\text{ref}} = -\frac{k_{\text{s}}}{m_{\text{s}}}(\boldsymbol{q}_{\text{CoM}}^{\text{ref}} - \boldsymbol{q}_{\text{s0}}) + \boldsymbol{g}_{\text{s}}, \tag{25}$$

where $\boldsymbol{q}_{\text{CoM}}^{\text{ref}}$ is the generated reference CoM trajectory of the sagittal SLIP model, and k_{s}, m_{s}, $\boldsymbol{g}_{\text{s}}$, and $\boldsymbol{q}_{\text{s0}}$ are defined uniformly as those in Eq. (1). The deadbeat controller with the control pair $(\alpha_{\text{TD}}, k_{\text{s}})$ can be operated synchronously with the evolution of the SLIP model.

Then, the previously defined Jacobian $\boldsymbol{J}_{\text{CoM}}$ is used to map the joint velocities into CoM space. We rearrange the stance dynamics of the monopode robot in Eq. (16) by eliminating the ground reaction force $\boldsymbol{F}_{\text{Gnd}}$ to obtain the task space formulation with

$$\boldsymbol{\Lambda}_{\text{t}}\ddot{\boldsymbol{q}}_{\text{CoM}} + \boldsymbol{b}_{\text{t}} + \boldsymbol{g}_{\text{t}} = \boldsymbol{F}_{\text{CoM}}, \tag{26}$$

where $\boldsymbol{\Lambda}_{\text{t}}$, $\boldsymbol{b}_{\text{t}}$, $\boldsymbol{g}_{\text{t}}$, and $\boldsymbol{F}_{\text{CoM}}$ are given by

$$\begin{cases} \boldsymbol{\Lambda}_{\text{t}} = (\boldsymbol{J}_{\text{CoM}}\boldsymbol{M}_{\text{S}}^{-1}\boldsymbol{J}_{\text{CoM}}^{-1})^{-1}, \\ \boldsymbol{b}_{\text{t}} = \overline{\boldsymbol{J}}_{\text{CoM}}^{-1}\boldsymbol{C}_{\text{S}}\dot{\boldsymbol{q}}_{\text{S}} + \overline{\boldsymbol{J}}_{\text{CoM}}^{\text{T}}\boldsymbol{J}_{\text{S}}^{\text{T}}\boldsymbol{\Lambda}_{\text{S}}\dot{\boldsymbol{J}}_{\text{S}}\dot{\boldsymbol{q}}_{\text{S}} - \boldsymbol{\Lambda}_{\text{t}}\dot{\boldsymbol{J}}_{\text{CoM}}\dot{\boldsymbol{q}}_{\text{S}}, \\ \boldsymbol{g}_{\text{t}} = \overline{\boldsymbol{J}}_{\text{CoM}}^{\text{T}}\boldsymbol{G}_{\text{S}}, \\ \boldsymbol{F}_{\text{CoM}} = \overline{\boldsymbol{J}}_{\text{CoM}}^{\text{T}}(\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}})^{\text{T}}\boldsymbol{\tau}_{\text{S}}, \end{cases} \tag{27}$$

where $\overline{\boldsymbol{J}}_{\text{CoM}} = \boldsymbol{M}_{\text{S}}^{-1}\boldsymbol{J}_{\text{CoM}}^{\text{T}}\boldsymbol{\Lambda}_{\text{t}}$ is a generalized inverse of the task Jacobian $\boldsymbol{J}_{\text{CoM}}$. By using Eq. (25) as the reference CoM acceleration, a proportion-differentiation (PD)-type command can be expressed as

$$\ddot{\boldsymbol{q}}_{\text{CoM}}^{\text{Cmd}} = \ddot{\boldsymbol{q}}_{\text{CoM}}^{\text{ref}} + \boldsymbol{K}_{\text{D}}(\dot{\boldsymbol{q}}_{\text{CoM}}^{\text{ref}} - \dot{\boldsymbol{q}}_{\text{CoM}}) + \boldsymbol{K}_{\text{P}}(\boldsymbol{q}_{\text{CoM}}^{\text{ref}} - \boldsymbol{q}_{\text{CoM}}), \tag{28}$$

where $\boldsymbol{K}_{\text{D}}$ and $\boldsymbol{K}_{\text{P}}$ are the diagonal PD gain matrices. Therefore, the control law for the torque-actuated monopode robot in stance phase is

$$\boldsymbol{\tau}_{\text{S}} = \underbrace{\left(\boldsymbol{J}_{\text{CoM}}\overline{\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}}}\right)^{\text{T}}}_{\boldsymbol{J}_{\text{F}}^{\text{T}}}\underbrace{\left(\boldsymbol{\Lambda}_{\text{t}}\ddot{\boldsymbol{q}}_{\text{CoM}}^{\text{Cmd}} + \boldsymbol{b}_{\text{t}} + \boldsymbol{g}_{\text{t}}\right)}_{\boldsymbol{F}_{\text{CoM}}}, \tag{29}$$

where $\overline{\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}}}$ is the Moore–Penrose inverse of $\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}}$.

$$\overline{\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}}} = \boldsymbol{M}_{\text{S}}^{-1}(\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}})^{\text{T}}\left(\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}}\boldsymbol{M}_{\text{S}}^{-1}(\boldsymbol{S}_{\text{S}}\boldsymbol{N}_{\text{S}})^{\text{T}}\right)^{-1}. \tag{30}$$

As for the swing phase, the swing leg should be pre-positioned relative to the CoM of the monopode robot; in this manner, the forthcoming TD event can be prepared with the desired AoT α_{TD}, and sufficient ground clearance can also be provided to prevent the toe from stumbling during the retraction stride. In contrast to the stance phase that employs task space in Eq. (26) and the control law Eq. (29) to reproduce the spring–mass behavior of the sagittal SLIP dynamics, the control task for the swing phase is relatively simple and has the capability to maintain the orientation of the swing leg (see Section 3.4) with the time-dependent angle $\alpha_{\text{TD}}(t_{\text{fall}})$. We choose the desired acceleration command $\ddot{q}_{i\text{F}}^{\text{Cmd}}$ for individual joints of the swing leg as follows:

$$\ddot{q}_{i\text{F}}^{\text{Cmd}} = \ddot{q}_{i\text{F}}^{\text{ref}} + k_{\text{D}, i}(\dot{q}_{i\text{F}}^{\text{ref}} - \dot{q}_{i\text{F}}) + k_{\text{P}, i}(q_{i\text{F}}^{\text{ref}} - q_{i\text{F}}),$$

$$i = 1, 2, \tag{31}$$

where $q_{i\text{F}}^{\text{ref}}$ is the reference position of the ith joint ($i = 1$ for knee and $i = 2$ for hip), while $q_{i\text{F}}$ is the actual position of the corresponding joint, and $k_{\text{D},i}$ and $k_{\text{P},i}$ are the PD gains of the ith joint, respectively. Here, $q_{i\text{F}}^{\text{ref}}$ can be acquired by directly resolving the inverse kinematics of $\boldsymbol{q} = \Gamma^{-1}(\boldsymbol{r}_{\text{eq}})$, as presented in the Appendix. Ultimately, the task space controller of the monopode robot is established to enforce the robot to behave according to the target sagittal SLIP dynamics, with prescribed apex height and velocity.

5 Simulation results

5.1 Simulation setups

Simulations are conducted to evaluate the performance of the proposed SLIP-anchored task space control method in dealing with various terrains. The model parameters of the monopode robot and the SLIP model are shown in Tables 1 and 2, respectively. The virtual equivalent leg length is chosen as 0.8 m to sufficiently provide a large workspace for the two-segmented leg of the robot. This adequate ground clearance can help determine the hopping of the robot in the upward swing sub-phase. The virtual simulation model of the robot is created in MATLAB/

Table 1 Model parameters of the monopode robot in the simulation

Parameter	Symbol	Value	Unit
Upper body mass	m_b	12	kg
Shank mass	m_1	3.5	kg
Thigh mass	m_2	3.5	kg
Shank inertia	J_1	0.08	kg·m^2
Thigh inertia	J_2	0.08	kg·m^2
Shank length	l_1	0.5	m
Thigh length	l_2	0.5	m
Shank CoM length	l_{C1}	0.25	m
Thigh CoM length	l_{C2}	0.25	m

Table 2 Model parameters of the sagittal SLIP model in the simulation

Parameter	Symbol	Value	Unit
Total mass	m_s	19	kg
Leg length	r_0	0.8	m
Leg stiffness	k_s	3200	N/m

SimMechanics environment, and the variable-step Runge–Kutta integration *ode45* is used to compute the hopping dynamics of the robot. The absolute tolerance is set to less than 10^{-8} to guarantee computational accuracy. The FSM of the control framework is encoded in MATLAB/Stateflow to schedule the stance/flight command and steer the hopping behavior. Several scenarios with diverse control goals, including stable periodic hopping on flat surface, target apex tracking, and traversing irregular terrains, are considered to verify the capability of the robot to achieve stable and robust hopping behaviors in complicated environments.

5.2 Main results

5.2.1 Periodic hopping on flat surface

The first scenario is simulated to validate the performance of the monopode robot in achieving periodic hopping on a flat surface. The simulation results are shown in Figs. 8 and 9. The desired apex height is set to $y_a = 1$ m, and the robot starts its hopping in the initial condition of $S_0 = [1.2$ m, 1.5 m/s]. Snapshots of the monopode robot indicate that the robot takes approximately two strides to approach the desired apex height, thus exhibiting a periodic hopping gait pattern.

Figure 9(a) plots joint angles q_1 and q_2 versus simulation time. The swing-leg pre-positioning in upward flight and in the stance phase are shown in different colors in this figure. The PD gains of the swing-leg controller Eq. (31) are given by $k_D = 20$ and $k_P = 100$ for both actuated joints. The swing leg has clearly attained the pre-positioned state with the desired AoT at the apex at each stride and maintained this orientation during the downward flight. This phenomenon implies that the robot with the devised swing-leg control, as proposed in Section 4.3, has sufficient time to prepare its leg for the forthcoming TD. Figure 9(b) shows the convergence process of the apex height and the AoT within five strides after the robot is initially released. The CoM of the robot approaches $y_a = 0.98$ m (equivalently relative error of 2% with respect to the desired apex height $y_a = 1$ m) after the first stride. This relative error in apex height is considerably reduced as the stride number increases. The AoT of the virtual equivalent leg is generally maintained at 83.33°, and the robot manifests stable limit cycle behavior when the SLIP model is used for hopping at the fixed point of the ARM. These scenarios
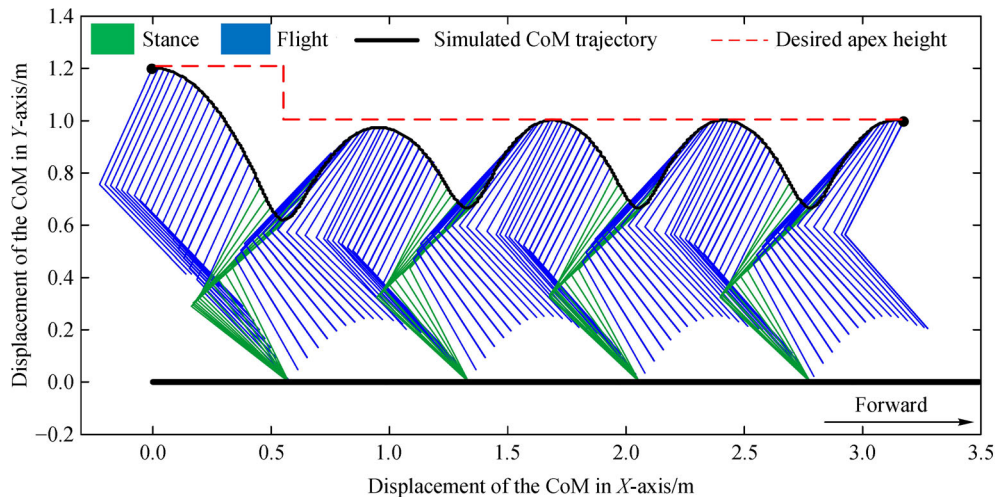
Fig. 8 Snapshots of the CoM trajectory of the monopode robot to represent periodic hopping on a flat surface.
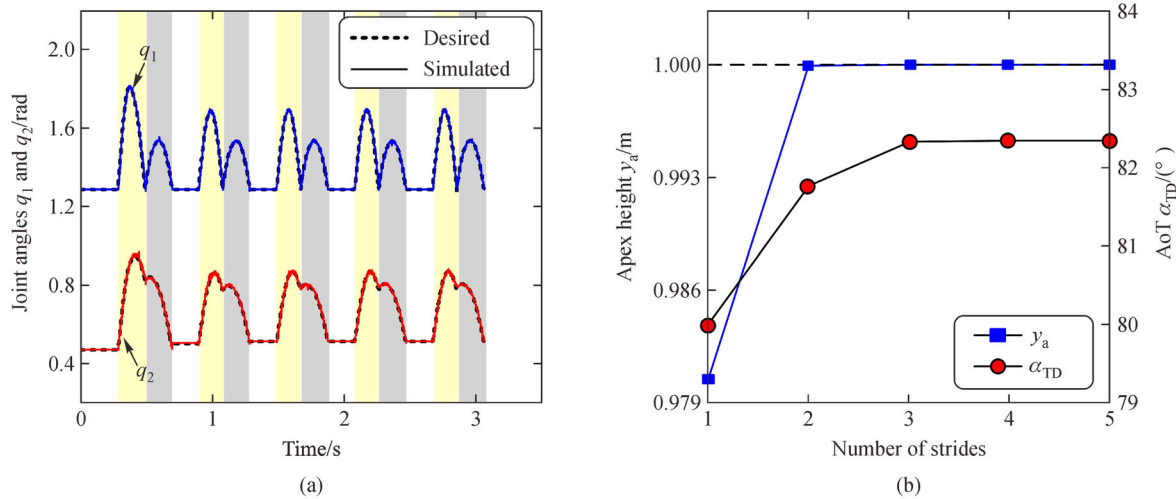
Fig. 9 Simulation results of selected variables for the monopode robot on constant apex height tracking. (a) Evolution of joint angles q_1 and q_2; (b) apex height y_a and AoT α_{TD} from stride to stride.

demonstrate the effectiveness of the proposed deadbeat controller, which operates in conjunction with the task space controller, to regulate the apex state of CoM.

5.2.2 Target apex tracking on flat surface

The second scenario is simulated to validate the tracking performance of the monopode robot with diverse target apex states. The simulation results are shown in Figs. 10 and 11. The target apex vector $\boldsymbol{S}_d$ for each stride are sequentially set to [1.0 m, 1.5 m/s], [1.0 m, 1.5 m/s], [1.3 m, 1.5 m/s], [1.3 m, 1.8 m/s], and [1.3 m, 2.0 m/s]. The tracking performance for the apex height can be directly observed from the snapshots of the CoM trajectory in Fig. 10. The stride length of the robot indirectly reveals the tracking result of apex velocity. The robot is initially released at $\boldsymbol{S}_0$ = [1.1 m, 1.5 m/s] and has exhibited stable hopping with the prescribed apex requirements.

Figure 11(a) shows the resulting joint angles q_1 and q_2 in

five strides during simulation. The PD gains of the swing-leg controller Eq. (31) are identical to that in the previous simulation. The swing leg adjusts its orientation according to the AoT generated by the deadbeat controller, a phenomenon similar to the leg movement in the periodic hopping simulation in which the robot prepared for a forthcoming TD event. Figure 11(b) plots apex height and velocity versus gait cycles. The CoM of the robot can track the target apex state with high accuracy (the maximum tracking error for apex height and velocity are 3.75% and 4.82%, respectively). Theoretically, the tracking error is twofold in that it entails the prediction error in the deadbeat controller (the derived analytical approximation is used to compute the ARM) and the convergence error in the closed-loop controller (see Eqs. (28) and (31)). The first error is sufficiently small, and its occurrence is valid for a wide range of model parameter combinations, as presented in Ref. [13]. The latter error can be restricted by properly increasing the PD gains of both stance and swing controls.
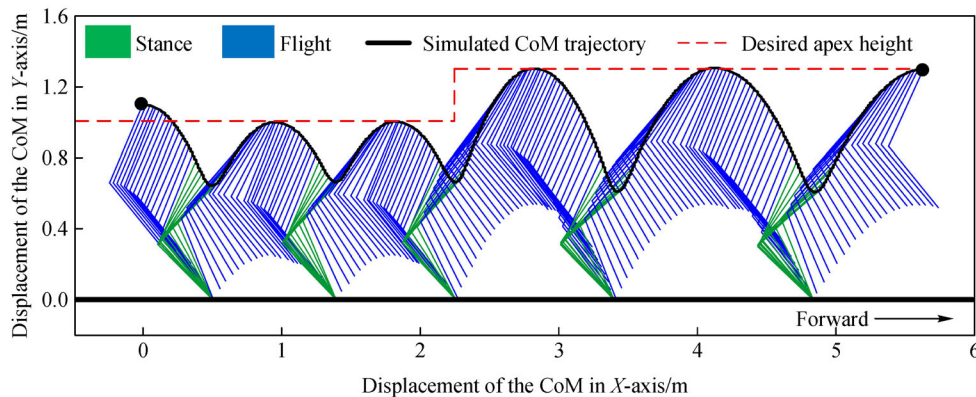

Fig. 10 Snapshots of the CoM trajectory of the monopode robot executing target apex tracking on a flat surface.
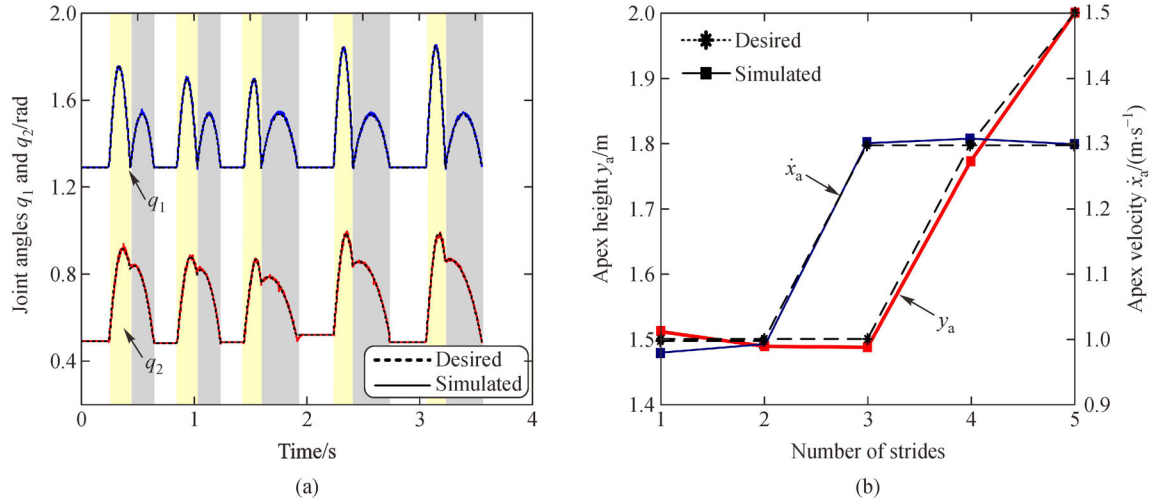
Fig. 11 Simulation results of selected variables of the monopode robot on variable apex state tracking. (a) Evolution of joint angles q_1 and q_2; (b) apex height and velocity from stride to stride.

Apex height and velocity in this scenario can be independently regulated by tuning synchronously the AoT and the virtual leg stiffness. This technique implies that the system energy level of the robot can be shifted from stride to stride owing to potential variations of the target apex vector.

5.2.3 Traversing irregular terrains

The final scenario is created by enforcing the monopode robot to hop on unknown irregular terrains. In this manner, the robustness of the proposed controller in terms of dealing with terrain perturbations can be validated. As shown in Fig. 12, the terrain profile is randomly arranged by setting the maximum step altitude to 0.17 m and the minimum step altitude to -0.2 m. An apex height preservation of $y_a = 1.0$ m is required uniformly in all strides. Moreover, we have set up a scenario in which the robot is unaware of the ground truth information; that is,

blind hopping without prior knowledge of the terrain is implemented by the control strategy. The simulation results are shown in Figs. 12 and 13. The robot is initially released at $S_0 = [1.1$ m, 1.5 m/s] and has maintained the apex height of approximately 1.0 m (the maximum relative error is 3.87%) in all strides, as shown in Fig. 12. By using the swing-leg pre-positioning strategy of the falling time-dependent AoT, as proposed in Section 3.4, the robot can achieve adaptive and robust hopping in irregular terrains. This finding demonstrates the effectiveness of the proposed controller in dealing with terrain perturbations.

The horizontal velocity and the vertical height of the CoM are shown in Fig. 13. In all simulations, the desired apex height and the velocity are fixed at 1.0 m and 1.2 m/s regardless of the terrain perturbation. The generated curves shown in the figures indicate that the robot can constantly maintain its hopping height and forward speed at each apex. Then, we plot the phase portraits of the joint angles q_1 and q_2 to determine the dynamical behaviors during
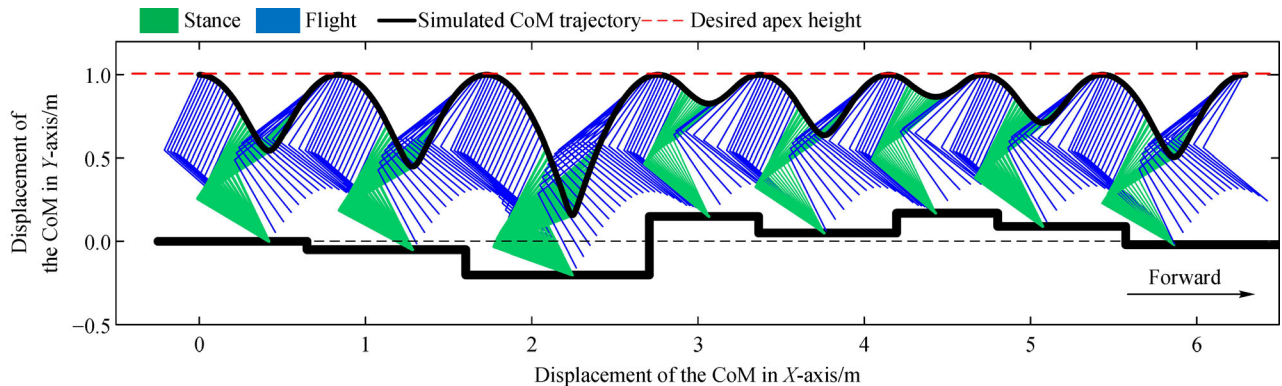

Fig. 12 Snapshots of the CoM trajectory of the monopode robot traversing an irregular terrain.
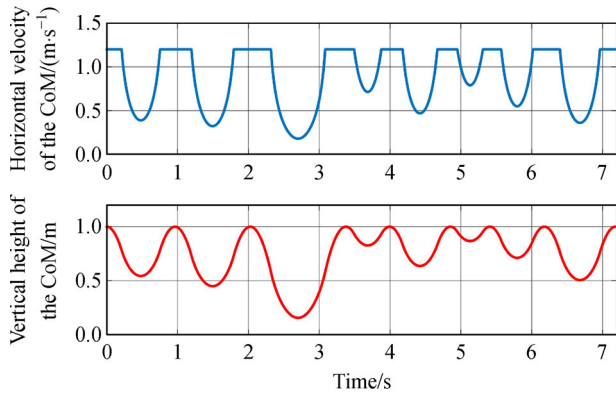
Fig. 13 Simulated horizontal velocity and vertical height of CoM of the monopode robot.

hopping. The discontinuity of the joint velocities at TD can be derived from Fig. 14; the discontinuity is due to the impulse effect of the foot–ground impact, which satisfies the transient switching condition Eq. (20). The SLIP model in the higher layer is energetically conservative based on the fixed apex state requirements, and it is invulnerable to the impact of the foot. The same scenario can be observed for the resulting reference CoM trajectory sent to the lower layer. The energy loss at TD is asymptotically compensated by the closed-loop controller Eq. (29) in the form of tracking errors within the forthcoming stance phase. Subsequently, the maximum altitude variation in the two adjacent irregular terrains approaches 0.352 m (approximately 35% of the vertical hopping height). The swing-leg controller Eq. (31) can provide sufficient ground clearance at each step, and the robot can successfully traverse complicated terrains without stumbling.
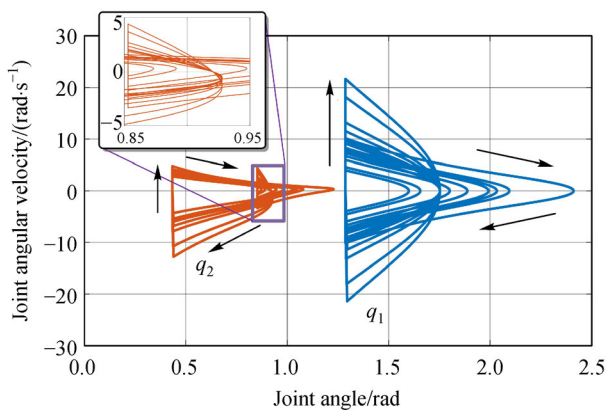
Fig. 14 Phase portrait of joint angles q_1 and q_2 of the robot traversing an irregular terrain. Arrows represent the evolution of the selected variables over time.

6　Discussions

The proposed sagittal SLIP-anchored task space controller leverages the self-stability and ease of maneuverability

advantages of the classical SLIP model by creating a two-fold deadbeat controller, which can be operated in flight and stance phases. The advantages of applying the proposed control method to steer the monopode robot in varied terrains are determined by implementing performance comparisons, including general feature synthesis and comparative simulation analysis.

6.1　General feature synthesis

The SLIP model has been extensively exploited, mainly because this model is the most common template used to describe the dynamical behaviors of legged locomotion. Traditionally, a fixed AoT policy in flight phase operating in conjunction with constant leg stiffness can be guaranteed, and the apex state during hopping can be steered from stride to stride, as reported in Ref. [10]. An issue with this scheme can be stated as follows: what is the benefit of applying the proposed deadbeat controller in contrast to the traditional SLIP control method? We implement a general feature synthesis between these two methods from the perspective of system representation, control, steering performance, and practical implementation. The detailed results are shown in Table 3. The fundamental distinction between the traditional controller and the proposed method lies in system representation. The former utilizes coupled nonlinear differential equations in formulating the SLIP dynamics (particularly in the stance phase) and inevitably downgrades mathematical tractability; in our proposed method, mathematical tractability is preserved. Therefore, the swing-leg pre-positioning policy with the stance-leg stiffness adjustment can be employed and transformed into a 1D shooting problem in the deadbeat controller. The merit of using this analytical approximation-based deadbeat controller endows the sagittal SLIP-anchored task space controller the ability of independently steering apex height and horizontal velocity. Subsequently, the reference CoM trajectory in the presence of terrain perturbations can be generated for the monopode robot.

6.2　Comparative simulation results

The superiority of the proposed deadbeat controller over the traditional SLIP controller has been established in Section 6.1. Comparative simulations are subsequently implemented to determine the performance discrepancies of the monopode robot equipped with the traditional and proposed deadbeat SLIP controllers as the high layer. Considering that the traditional controller cannot independently steer apex height and velocity, we select the apex height of CoM as the control target to execute the comparative simulation. Both cases adopt the whole-body task space controller for the 2-degree of freedom monopode robot. The traditional and the proposed SLIP controllers are adopted to generate the reference CoM

Table 3   General feature comparison between the traditional SLIP controller and the proposed controller

Comparison items	Traditional SLIP controller [10]	Proposed deadbeat controller
System representation	Nonlinear differential equations	Analytical approximations
Control input	AoT	AoT and leg stiffness
Control policy	Fixed AoT	Adjustable AoT and leg stiffness
Steering duration	Only flight phase	Both flight and stance phase
System energy	Conservation	Adding/removing energy
Steerable apex state	Height or velocity	height and velocity (independent)
Period of apex steering	Asymptotically	Only within a one-gait cycle
Terrain adaptability	Flat ground	Flat and uneven ground
Practical implementation	Fourth-order Runge–Kutta solver	Direct coding

trajectories, which are then transformed into torque commands by the task space controller to manipulate the robot. The parameters of the monopode robot with the SLIP model are identical, as shown in Tables 1 and 2.

The comparative simulation results are shown in Fig. 15. The robot is initially released at the apex height of $y_a(0) = 0.78$ m, and the target apex height is set to $y_a = 0.72$ m for both controllers. Additionally, the fixed AoT policy with $\alpha_{TD} = 64°$ is maintained throughout the simulation. The resulting height of the CoM at the convergence duration of the robot with the traditional SLIP controller is 26 gait cycles (the relative error is less than 0.58%). This duration is substantially reduced to only 1 gait cycle by using the proposed deadbeat controller (the relative error is less than 0.44%), which implies that the proposed controller can converge to the prescribed apex height more rapidly compared with the traditional method.

The underlying mechanism of the traditional SLIP controller with the fixed AoT policy is further determined by focusing on ARM and by setting the AoT range from 50° to 86°. All the fixed points (stable or unstable) of ARM are located on the diagonal line, with $y_a(i) = y_a(i+1)$. Conventionally, each ARM of a specified AoT policy has one stable and one unstable fixed point at most within the feasible motion range according to Ref. [10]. A partial view of the convergence process after the robot is released at $y_a = 0.78$ m is generated. The ARM with $\alpha_{TD} = 64°$ clearly has two fixed points: $y_{as} = 0.72$ (stable) and $y_{aus} = 0.795$ (unstable). The basin of attraction (BoA) for $y_{as} = 0.72$ is determined by $[r_0 \sin \alpha_{TD}, y_{aus}] = [0.71, 0.795]$. From this perspective, the fundamental mechanism of the traditional fixed AoT controller is on how to utilize the self-stability merit of BoA at a stable fixed point and achieve stride-to-stride convergence. However, the initial apex state must be carefully chosen to guarantee that $y_a(0)$ is embodied by the BoA, especially since the BoA region is narrow. This disadvantage implies that the hopping behavior of the robot is sensitive to initial-state perturbation. The convergence process of the traditional SLIP controller is dependent on the initial state. Moreover, the proposed deadbeat controller requires only one step to attain the target apex state; by contrast, the traditional controller conventionally requires multiple gait cycles.

7   Conclusions

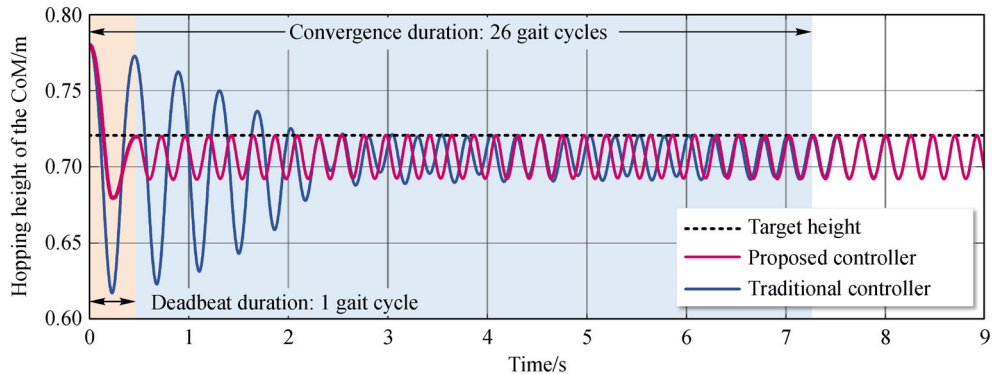This study proposes a sagittal SLIP-anchored task space

Fig. 15   Comparative simulation results of the monopode robot equipped with the traditional and proposed controllers. Shaded areas represent convergence duration for each case.
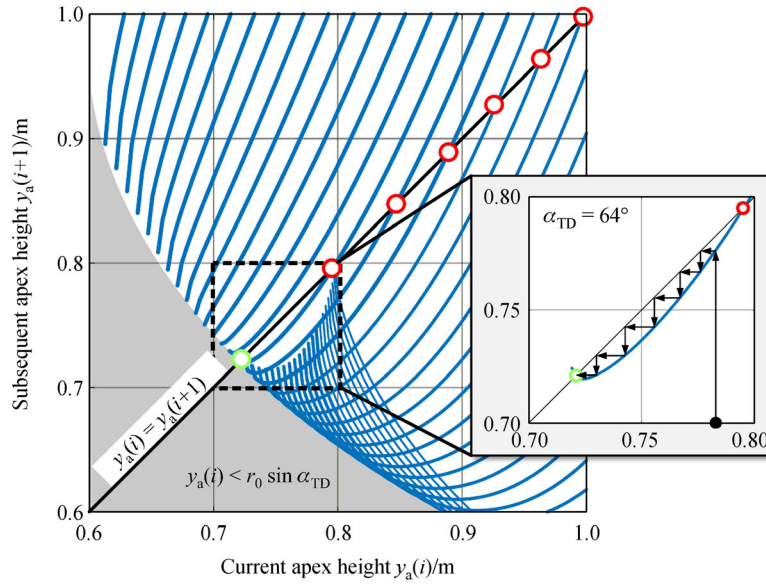
Fig. 16 ARM of the SLIP model equipped with traditional fixed AoT policy. Shaded areas represent the unfeasible region in which stumbling occurs with insufficient initial releasing height $y_a(i) < r_0\sin\alpha_{TD}$. Stable and unstable fixed points are colored green and red, respectively. The partial view shows the convergence process after initial release at AoT $\alpha_{TD} = 64°$.

control for a monopode robot traversing irregular terrains. The advantages of the SLIP model in terms of self-stability and ease of maneuverability are employed in this study. Moreover, the classical sagittal SLIP model with a deadbeat controller is developed to independently regulate apex height and velocity. The proposed scheme is based on the analytical approximate solution, which can be operated as representation of the SLIP dynamics. The deadbeat controller takes the AoT of the swing leg and the stiffness of the stance leg as the control input. In the stance phase, a variable leg stiffness policy with a piecewise-constant stiffness profile is employed to adjust the energy level of the SLIP model. In the flight phase, a falling time-dominant TD policy for the swing leg in downward flight is proposed to adapt to unknown terrain irregularities in anticipation of forthcoming TD events. Subsequently, a sagittal SLIP-anchored double-layered task space formulation is established for the monopode robot. The higher layer utilizes the SLIP model to generate the adaptive reference CoM trajectory of the robot. Then, the lower layer transfers the CoM trajectory into individual joint commands by using the task space formulation to reproduce the target SLIP dynamics on the monopode robot. The effectiveness of the proposed control strategy is verified by simulation. The robot can achieve stable periodic hopping and target apex tracking and traverse irregular terrains. The proposed control framework has the potential to be extended to extremely complicated cases, such as when biped and quadruped robots are involved. The SLIP model needs to be elaborately constructed to capture the essential dynamical behavior of legged locomotion.

Appendix

A virtual equivalent leg can be used to bridge the SLIP model and the monopode robot. Given the upper body coordinate (x_b, y_b) and the joint angles q_1 and q_2 of the robot, the coordinate (x_{toe}, y_{toe}) of the toe of the leg can be written as

$$\begin{cases} x_{toe} = x_b - l_2\sin q_2 - l_1\sin(q_2 - q_1), \\ y_{toe} = y_b - l_2\cos q_2 - l_1\cos(q_2 - q_1). \end{cases} \tag{A1}$$

The position vector of the virtual equivalent leg is defined along the direction of the toe towards the CoM of the robot with the following relation:

$$\begin{aligned} \boldsymbol{r}_{eq} &= (x_{CoM} - x_{toe}, y_{CoM} - y_{toe})^T \\ &= \begin{pmatrix} \left((1-\lambda_1+\lambda_2)l_2 - \lambda_2 l_{C2}\right)\sin q_2 + \left((\lambda_1-1)l_1 - \lambda_1 l_{C1}\right)\sin(q_1-q_2) \\ \left((1-\lambda_1-\lambda_2)l_2 + \lambda_2 l_{C2}\right)\cos q_2 + \left((1-\lambda_1)l_1 + \lambda_1 l_{C1}\right)\cos(q_1-q_2) \end{pmatrix} := \Gamma(\boldsymbol{q}), \end{aligned} \tag{A2}$$

where $r_{eq} = \Gamma(q)$ denotes the forward kinematics of the robot, and the coefficients λ_1, λ_2, and λ_3 are given by

$$\begin{cases} \lambda_1 = \dfrac{m_1}{m_b + m_1 + m_2}, \\ \lambda_2 = \dfrac{m_2}{m_b + m_1 + m_2}, \\ \lambda_3 = \dfrac{m_b}{m_b + m_1 + m_2}. \end{cases} \quad (A3)$$

The length of the virtual equivalent leg is determined by

$$r_{eq} = |r_{eq}| = \sqrt{(x_{CoM} - x_{toe})^2 + (y_{CoM} - y_{toe})^2}. \quad (A4)$$

The leg angle representing the orientation with respect to the positive X-axis is determined by

$$\alpha = \pi - \mathrm{atan2}(x_{CoM} - x_{toe}, y_{CoM} - y_{toe}). \quad (A5)$$

In particular, the leg angle at TD, which is defined as the AoT α_{TD} of the SLIP model, is utilized in the adaptive swing-leg controller Eq. (15).

References

1. Blickhan R, Full R J. Similarity in multilegged locomotion: Bouncing like a monopde. Journal of Comparative Physiology, 1993, 173: 509–517

2. Hubicki C, Grimes J, Jones M, et al. ATRIAS: Design and validation of a tether-free 3D-capable spring-mass bipedal robot. International Journal of Robotics Research, 2016, 35(12): 1497–1521

3. Sreenath K, Park H, Poulakakis I, et al. A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on MABEL. International Journal of Robotics Research, 2011, 30(9): 1170–1193

4. Haldane D W, Yim J K, Fearing R S. Repetitive extreme-acceleration (14-g) spatial jumping with Salto-1P. In: Proceedings of 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vancouver: IEEE, 2017, 3345–3351

5. Seok S, Wang A, Otten D, et al. Actuator design for high force proprioceptive control in fast legged locomotion. In: Proceedings of 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vilamoura: IEEE, 2012, 1970–1975

6. Farley C, Glasheen J, McMahon T. Running springs: Speed and animal size. Journal of Experimental Biology, 1993, 185: 71–86

7. Blickhan R. The spring-mass model for running and hopping. Journal of Biomechanics, 1989, 22(11–12): 1217–1227

8. Schwind W J, Koditschek D E. Approximating the stance map of a 2-DOF monoped runner. Journal of Nonlinear Science, 2000, 10(5): 533–568

9. Ghigliazza R M, Altendorfer R, Holmes P, et al. A simply stabilized running model. SIAM Review, 2005, 47(3): 519–549

10. Geyer H, Seyfarth A, Blickhan R. Spring-mass running: Simple approximate solution and application to gait stability. Journal of Theoretical Biology, 2005, 232(3): 315–328

11. Arslan Ö, Saranli U, Morgül Ö. Approximate stance map of the spring mass hopper with gravity correction for nonsymmetric locomotions. In: Proceedings of 2009 IEEE International Conference on Robotics and Automation (ICRA). Kobe: IEEE, 2009, 2388–2393

12. Shahbazi M, Babuška R, Lopes G A D. Unified modeling and control of walking and running on the spring-loaded inverted pendulum. IEEE Transactions on Robotics, 2016, 32(5): 1178–1195

13. Yu H, Li M, Wang P, et al. Approximate perturbation stance map of the slip runner and application to locomotion control. Journal of Bionics Engineering, 2012, 9(4): 411–422

14. Hutter M, Remy C D, Höpflinger A, et al. SLIP running with an articulated robotic leg. In: Proceedings of 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Taipei: IEEE, 2010, 4934–4939

15. Piovan G, Byl K. Approximation and control of the SLIP model dynamics via partial feedback linearization and two-element leg actuation strategy. IEEE Transactions on Robotics, 2016, 32(2): 399–412

16. Seyfarth A, Geyer H, Herr H. Swing-leg retraction: A simple control model for stable running. Journal of Experimental Biology, 2003, 206(15): 2547–2555

17. Karseen J G D, Haberland M, Wisse M, et al. The optimal swing-leg retraction rate for running. In: Proceedings of 2011 IEEE International Conference on Robotics and Automation (ICRA). Shanghai: IEEE, 2011, 4000–4006

18. Yu J, Hong D, Haberland M. Energetic efficiency of a compositional controller on a monoped with an articulated leg and SLIP dynamics. In: Proceedings of 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Madrid: IEEE, 2018, 2221–2228

19. Rustchmann M, Satzinger B, Byl M, et al. Nonlinear model predictive control for rough-terrain robot hopping. In: Proceedings of 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vilamoura: IEEE, 2012, 1859–1864

20. Garofalo O C, Albu-Schäffer A. Walking control of fully actuated robots based on the Bipedal SLIP model. In: Proceedings of 2012 International Conference on Robotics and Automation (ICRA). Saint Paul: IEEE, 2012, 1456–1463

21. Wensing P M, Orin D E. High-speed humanoid running through control with a 3D-SLIP model. In: Proceedings of 2013 IEEE/RSJ

International Conference on Intelligent Robots and Systems (IROS). Tokyo: IEEE, 2013, 5134–5140

22. Wensing P M, Orin D E. 3D-SLIP steering for high-speed humanoid turns. In: Proceedings of 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Chicago: IEEE, 2014, 4008–4013

23. Ernst M, Geyer H, Blickhan R. Extension and customization of self-stability control in compliant legged systems. Bioinspiration & Biomimetics, 2012, 7(4): 046002

24. Khatib O. A unified approach for motion and force control of robot manipulators: The operational space formulation. IEEE Journal on Robotics and Automation, 1987, 3(1): 43–53