Robotica (2024), **42**, pp. 2713–2741
doi:10.1017/S0263574724001061

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

Dynamic modeling of wheeled biped robot and controller design for reducing chassis tilt angle

Nan Mao[1,2], Junpeng Chen[2], Emmanouil Spyrakos-Papastavridis[1] and Jian S. Dai[1,2]

[1]Centre for Robotics Research, King's College London, London, UK
[2]Shenzhen Key Laboratory of Intelligent Robotics and Flexible Manufacturing Systems, Southern University of Science and Technology, Shenzhen, China
Corresponding author: Nan Mao; Email: herbmaonan@163.com

Received: 14 November 2023; **Revised:** 2 April 2024; **Accepted:** 8 May 2024; **First published online:** 2 October 2024

Keywords: controller design; dynamic modeling; wheeled biped robot; chassis tilt angle; planar parallel mechanism

Abstract

The wheeled-legged robot combines the advantages of wheeled and legged robots, making it easier to assist people in completing repetitive and time-consuming tasks in their daily lives. This paper presents a study on the kinematic and dynamic modeling, as well as the controller design, of a wheeled biped robot with a parallel five-bar linkage mechanism as its leg module. During the motion of the robot, the robot relies on the tilt angle of the inverted pendulum, and this angle often results in the tilting of the chassis of the robot, presenting challenges for the installation of upper-body payloads and sensor systems. The controller proposed in this paper, which is developed by decoupling the primary motions of the robot and designing a multi-objective, multilevel controller, addresses this issue. This controller employs the pendulum pitch angle of the equivalent inverted pendulum model as the control variable and compensates for the chassis tilt angle (CTA). This control method can effectively reduce the CTA of such robots and eliminate the need for additional counterweights. It also provides a more spacious structural design for accommodating upper-body devices. The effectiveness of this control framework is verified through variable height control, walking on flat ground, and carrying loads over rough terrain and slopes.

1. Introduction

In the last twenty years, robot technology has rapidly evolved in tandem with the development of societal productivity, encompassing the diverse development of robots that are now integral to society. Among these, ground mobile robots have found extensive applications in fields such as logistics and storage, counter-terrorism and bomb disposal, wilderness transportation, and domestic services [1]. Two-wheeled balancing robots, which constitute a subset of mobile robotics, possess a moderate number of variables and can be more closely approximated by the classical linear inverted pendulum model. These features render them ideal research platforms for validating a variety of control algorithms. Moreover, their simple structure and agile movement make them well-suited for operations in confined spaces or complex environments, enabling the realization of tasks that other multi-wheeled robots may be unable to undertake, such as space exploration and terrain reconnaissance [2].

Scholars from various countries have conducted extensive research on two-wheeled balancing robots due to their uniqueness and practicality. Kazuo Yamafuji et al. [3] initially proposed the concept of a two-wheeled balancing robot, laying the foundation for subsequent studies, even though the robot they constructed could only move on fixed tracks at the time. Felix Grasser et al. [4] achieved a significant milestone by developing the mobile inverted pendulum robot, establishing a dynamic model, and designing a full-state feedback controller. This marked a pivotal moment in the history of two-wheeled self-balancing robots, particularly for control on flat terrains.

© The Author(s), 2024. Published by Cambridge University Press.

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

In the case of the above-mentioned two-wheeled self-balancing robot during acceleration or deceleration, achieving higher acceleration requires a significant tilting of the body. If the body of the balancing robot is equipped with devices such as robotic arms or LiDAR sensors, the use of these devices is influenced by the chassis tilt angle (CTA), leading to reduced operational efficiency and collaborative capabilities [5]. Therefore, in this context, a parallel mechanism is employed that enables the drive wheels to have a two-degree-of-freedom (DOF) motion plane relative to the body. This approach facilitates the adjustment of the body's posture and height.

Leg structure design options include serial and parallel mechanisms. The parallel mechanisms have compact structures, making them more suitable for heavy-load scenarios [6]. This study adopts a two-DOF five-bar parallel mechanism. Giberti [7] applied this structure in a planar robotic arm design, derived kinematic and dynamic equations, conducted arm testing, achieved tracking of the prescribed trajectory, and determined its motion space. Kau [8] designed a quadruped mobile robot based on this structure, showing lighter leg structure and higher motor efficiency compared to serial-legged robots. Other researchers have improved robot motion performance by designing hybrid mechanisms [9] and switching mechanisms [10].

To make a robot adaptable to the various environments and diverse demands, Dai introduced the concept of the metamorphic mechanism in 1998 [11, 12] and then reconfigurable mechanisms [13] for mechanisms to possess variable mobility and topology during motion, significantly broadening the functionality and adaptability of robots [14]. Based on this, a novel robot hand with a metamorphic palm was developed [15], with the mathematical and manifold-related perspectives [16], to have higher dexterity of the hand [17]. In subsequent research, scholars delved deeper into the kinematics [18], manipulability [19], and singular value decomposition [20] of the metamorphic hand. Furthermore, such mechanisms have also found application in legged robots. Zhang et al. employed a planar six-bar closed-loop linkage as the robot's trunk, enhancing its stability margin and biomimetic performance [21]. The reconfigurable quadruped robot, Origaker [22], developed by Tang et al., utilizes a spatial eight-bar mechanism [23], offering four biomimetic configurations, thereby significantly enhancing the locomotion capabilities of the robot [24, 25]. This serves as an inspiration for robot development from both a biomimetic and mechanical design perspective. Additionally, other legged robots employing Bennett mechanisms [26], planar five-bar mechanisms [27, 28], or parallel mechanisms [29] have served as valuable references for leg designs in mobile robots.

In the field of robot control, numerous scholars have attempted to enhance the overall performance of robots from various aspects [30]. Wang et al proposed a nonlinear controller that improved the robot's balancing performance [31]. Saglia et al. delivered an inverse-kinematics control strategy that largely enhanced the robot's performance [32, 33]. Klemm et al. derived the motion loop dynamics for more complex systems, thus reducing the computational load on the controller [34]. Xin et al. introduced the Cart-Linear Inverted Pendulum Model and implemented model predictive control to achieve online dynamic hybrid locomotion in wheeled biped robots [35]. Chen et al. proposed a novel wheeled-spring-loaded inverted pendulum model to characterize the dynamics of wheeled biped robots during jumping [36]. In ref. [37], Zhang et al. introduced a wheel-to-foot transition mechanism and control strategy, addressing the challenge of transitioning between the robot's wheeled balancing and bipedal standing states. Robot parameter identification is also one of the significant challenges in dynamic control. Chang et al. effectively reduced the difficulty of parameter identification by proposing the "E-B" identification method [38]. Other works have also focused on improving the performance of wheeled-legged robots through approaches such as torque control [39], attitude compensation [40], and hybrid control frameworks [41].

In this study, we focus on the dynamics modeling and the cascaded control strategy of the wheeled biped robot based on the decoupling principle, with a compensating strategy for the CTA. This approach enables the robot to maintain a horizontal posture during the inclined motion, effectively reducing the CTA's magnitude. The paper incorporates a five-bar mechanism. The validity of the established model is verified through experiments. The main contributions of this paper are the following:

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

(1) An analytical method that is used for detailed kinematic and dynamic modeling of this wheeled-legged robot, enabling it to achieve self-balancing and locomotion.

(2) A multi-objective and multilevel controller that is designed based on decoupling the robot's motion. With both simulations and practical experiments, when using locked and unlocked leg structures, the control framework is demonstrated to significantly reduce the CTA during inclined motion and loading processes.

(3) A control framework that primarily relies on the positional control of the motors, employing direct motor drive without the need for additional transmission systems, thereby reducing the control complexity.

The remainder of this paper is organized as follows. Section 2 provides a brief overview of the overall mechanism of the robot. Section 3 elaborates on the detailed modeling and analysis of the leg module's kinematics, robot dynamics, and acceleration principles. Section 4 combines the balancing control of the robot with leg module locking/unlocking control to conduct a comprehensive controller analysis. Section 5 involves simulation and physical experiments to validate the proposed control framework, followed by an analysis of the experimental results. Lastly, Section 6 concludes the paper.

Table I shows the relevant symbols and descriptions pertaining to the paper's mathematical derivations.

2. Overview of the wheeled biped robot

2.1. Requirements for mechanism design

The wheeled biped robot needs to adapt to various terrains while ensuring variable body inclination. Building upon the foundations of traditional wheeled biped robots, a five-bar mechanism is introduced. Accordingly, this paper defines the requirements for mechanism design as follows:

(1) The robot should possess a highly symmetric overall structure, ensuring that the projection of the center of gravity in its upright state falls as close as possible to the geometric center of the support polygon on the ground [42].

(2) The leg design of the parallel mechanism must ensure a movable connection to the chassis [43]. This ensures that during motion, the driving wheels will not significantly impact the body posture.

(3) As the plane of the driving wheels is always perpendicular to the ground, body posture adjustability must be maintained, requiring the leg design to comprise at least two DOFs.

2.2. Mechanism design of the wheeled biped robot

Considering the aforementioned requirements, the conceptual design of the robot's overall configuration is depicted in Figure 1(a). The robot comprises the body module, two-sided driving wheel modules, and leg modules connecting the body and driving wheels. Regarding the design of the legs, as mentioned in section 2.1, each robot leg requires at least two DOFs to adjust the robot's posture. Therefore, a simple and effective parallel five-bar mechanism is deemed an auspicious choice. Compared to the traditional planar four-bar mechanism, the five-bar mechanism not only has an additional DOF that enables more flexible movement but also does not require a complex transmission mechanism [8]. It can be directly driven by motors, which to a certain extent, reduces the control complexity. Its configuration is illustrated in Figure 1(b), with both legs possessing the same configuration as each other. This five-bar mechanism also maintains a symmetrical configuration on each side, enhancing leg rigidity.

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

Table I. *Nomenclature.*

Symbols	Descriptions
P	Midpoint of the two hip joints of the same leg module.
h	Distance from point P to the foot end.
R	Radius of left or right wheel.
L_0	Distance between the chassis's equivalent COM and point P.
L_1	Distance between the hip joints I and II on the same side.
L_2,L_3	Length of the thigh linkage (TL) and shank linkage (SL)
D	Distance between the driving wheels on the left and right.
L	Distance from center of mass (COM) of pendulum to wheel axle.
H_l, H_r	Horizontal interaction force between the left and right wheel axis and the pendulum.
V_l, V_r	Vertical interaction force between the left and right wheel axis and the pendulum.
f_l,f_r	Static friction force of the ground against the left and right wheels.
τ_l, τ_r	Driving torque of the left and right wheel.
x_p	Forward displacement of the pendulum.
x	Forward displacement of the robot.
x_r	Desired forward displacement of the robot.
θ	PPA (in FIP model), that is the angle between the pendulum and y_B-axis.
θ_c	CTA, that is the angle between the line L_0 and the vertical line.
θ_L	The angle between the line h and the vertical line.
θ_{eq}	The angle between the line L_{eq} and the vertical line.
δ	PYA, that is the angle between the x-axis and x_B-axis.
m_p, m_w	Mass of the pendulum and a wheel.
J_p	Moment of inertia of the pendulum around the z_B axis.
J_{py}	Moment of inertia of the pendulum around the y_B axis.
J_y	Moment of inertia of the robot around the y_B-axis.
J_ω	Moment of inertia of the wheel around its axial direction.
$J_{\omega y}$	Moment of inertia of the wheel around its radial direction.
L_{eq}	Chassis equivalent COM to the wheel axle distance.
COG, *COM*	Center of gravity and center of mass.
FIP	First-order inverted pendulum.
TL, *SL*	Thigh linkage and shank linkage.
CTA, *PPA*, *PYA*	Chassis tilt angle, pendulum pitch angle and pendulum yaw angle.
YM, *PM*, *FM*	Yaw, pitch and forward motion.

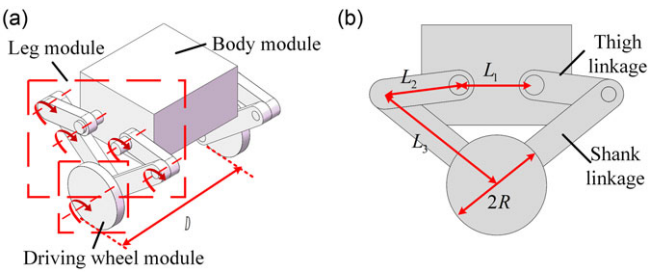


Fig. 1. *Structure diagram of the wheeled biped robot.*

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

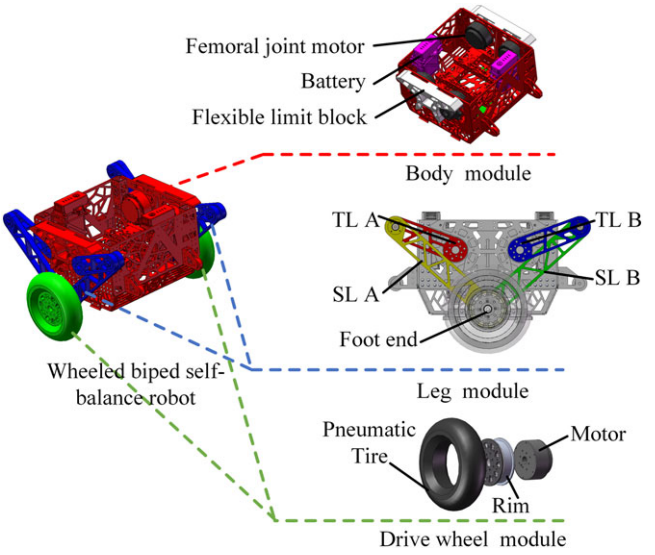


Fig. 2. *Design of main modules of the wheeled biped robot.*

The hip joints, denoted as drive joints I and II, on one side of the robot are connected to the body. The shank linkages (SLs) are attached to the driving wheels at their ends, facilitating the forward and backward movement of the robot. Due to the high symmetry requirements for balanced motion, the robot is symmetrically designed, ensuring that the symmetrical linkages have equal lengths.

The overall structure of the robot is depicted in Figure 2. The mechanical components mainly consist of three modules: the body module, leg module, and drive wheel module. The body module houses key electronic devices, batteries, femoral joint motors, and flexible limiters for protecting the leg structure. On each side, the leg can be divided into the TLs (thigh linkages), TL A and TL B, and the SLs, SL A and SL B. The leg module is predominantly constructed of carbon fiber, ensuring strength while reducing weight. The drive wheel module comprises a drive motor, rim, and pneumatic tire.

3. Kinematics and dynamics modeling of the wheeled biped robot

3.1. Kinematics modeling of the planar five-bar parallel mechanism

In general, existing approaches to robot modeling include the generalized coordinate method, and the analytical method, among others. The former can generate a simplified symbolic representation when modeling systems of low to medium complexity [36]. The latter in the analytical method, on the other hand, offers more intuitiveness in simpler systems and can achieve real-time control with less computational resource consumption. For the system discussed in this paper, we have adopted the more intuitive, analytical modeling method.

The leg module employs a planar parallel five-bar mechanism. Taking the leg module of one side as an example, if the line connecting the two driving joints of the leg module is treated as an equivalent link, a coordinate frame is established as shown in Figure 3(a) and simplified in Figure 3(b). The respective coordinate systems and defined parameters are the following:

1) The origin O_0 of the coordinate frame $\{L\}$ is located at the center of rotation of driving joint I, and the x_0 and y_0 axes are along the horizontal and vertical directions of the body, respectively.
2) The origin O_1 in frame $\{L\}$ is located at the center of the leg's tip. The x_1 and y_{A_1} axes are parallel to the x_0 and y_0 axes, respectively.

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

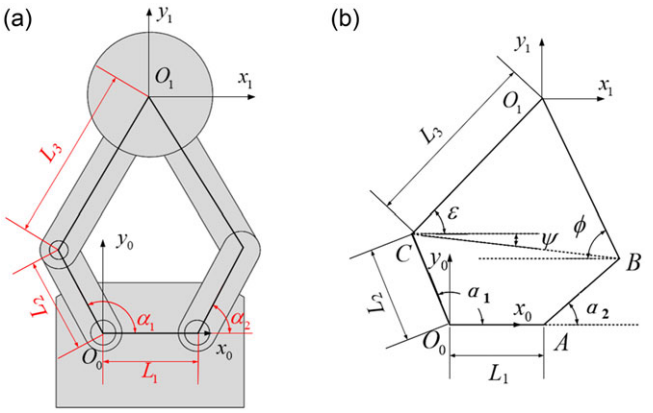


Fig. 3. *Coordinate frame and parametric annotations of the leg module.*

3) Points A, B, and C serve as the joint rotational centers; α_1 and α_2 represent the rotation angles of driving joints I and II, respectively; ε and ϕ denote the angles between the leg linkage l_{CO_1} and l_{BO_1} and the horizontal direction; and ψ indicates the angle between l_{CB} and the horizontal direction.

From the geometric relationship in Figure. 3(b), one can obtain:

$$\begin{cases} \boldsymbol{l}_{\boldsymbol{O_0C}} = (L_2 \cos\alpha_1, L_2 \sin\alpha_1) = (x_C, y_C) \\ \boldsymbol{l}_{\boldsymbol{O_0A}} = (L_1, 0) \\ \boldsymbol{l}_{\boldsymbol{AB}} = (L_2 \cos\alpha_2, L_2 \sin\alpha_2) \end{cases} \tag{1}$$

Then

$$\boldsymbol{l}_{\boldsymbol{O_0B}} = \boldsymbol{l}_{\boldsymbol{O_0A}} + \boldsymbol{l}_{\boldsymbol{AB}} = (L_1 + L_2 \cos\alpha_2, L_2 \sin\alpha_2) = (x_B, y_B) \tag{2}$$

$$|\boldsymbol{l}_{\boldsymbol{CB}}| = |\boldsymbol{l}_{\boldsymbol{O_0B}} - \boldsymbol{l}_{\boldsymbol{O_0C}}| = \sqrt{(x_B - x_C)^2 + (y_B - y_C)^2} \tag{3}$$

According to the cosine law, we have

$$\cos(\varepsilon + \psi) = \cos\varepsilon \cos\psi - \sin\varepsilon \sin\psi = \frac{{L_3}^2 + |\boldsymbol{l}_{\boldsymbol{CB}}|^2 - {L_3}^2}{2L_3|\boldsymbol{l}_{\boldsymbol{CB}}|} \tag{4}$$

where $\sin\psi = -(y_B - y_C)/|\boldsymbol{l}_{\boldsymbol{CB}}|$ and $\cos\psi = (x_B - x_C)/|\boldsymbol{l}_{\boldsymbol{CB}}|$. Substituting these values into Eq. 4 yields

$$2L_3 \cos\varepsilon\,(x_B - x_C) + 2L_3 \sin\varepsilon\,(y_B - y_C) = |\boldsymbol{l}_{\boldsymbol{CB}}|^2 \tag{5}$$

Letting $c = 2L_3\,(x_B - x_C)$, $\quad d = 2L_3\,(y_B - y_C)$, we have

$$c\cos\varepsilon + d\sin\varepsilon = |\boldsymbol{l}_{\boldsymbol{CB}}|^2 \tag{6}$$

Using the trigonometric identity, Eq. 6 can be rewritten as follows:

$$\frac{c\left(1 - \tan^2\frac{\varepsilon}{2}\right)}{1 + \tan^2\frac{\varepsilon}{2}} + \frac{2d\tan\frac{\varepsilon}{2}}{1 + \tan^2\frac{\varepsilon}{2}} = |\boldsymbol{l}_{\boldsymbol{CB}}|^2 \tag{7}$$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

Solving Eq. 7, ε can be expressed as:

$$\varepsilon = 2\arctan\left(\frac{d \pm \sqrt{c^2 + d^2 - |\boldsymbol{l_{CB}}|^4}}{c + |\boldsymbol{l_{CB}}|^2}\right) \tag{8}$$

Since $\boldsymbol{l_{CO_1}} = \left(L_3\cos\delta, L_3\sin\delta\right)$, $\boldsymbol{l_{O_0O_1}}$ can be expressed as:

$$\begin{aligned} \boldsymbol{l_{O_0O_1}} &= \boldsymbol{l_{O_0C}} + \boldsymbol{l_{CO_1}} \\ &= \left(x_C + L_3\cos\varepsilon, y_C + L_3\sin\varepsilon\right) \\ &= \left(L_2\cos\alpha_1 + L_3\cos\varepsilon, L_2\sin\alpha_1 + L_3\sin\varepsilon\right) \\ &= \left(l_{(O_0O_1)_x}, l_{(O_0O_1)_y}\right) \end{aligned} \tag{9}$$

where the precise forms of ε, c and d are provided in Appendix A.1.

Based on the aforementioned forward kinematic analysis, the relationship between the end-point of the leg module and the driving joint angles α_1 and α_2 can be obtained. After rearranging the equations, the inverse kinematics solution of this leg module can be expressed as:

$$\begin{cases} \alpha_1 = 2\arctan\left(\dfrac{f \pm \sqrt{e^2 + f^2 - g^2}}{e + g}\right) \\ \alpha_2 = 2\arctan\left(\dfrac{i \pm \sqrt{h^2 + i^2 - j^2}}{h + j}\right) \end{cases} \tag{10}$$

where the precise forms of e, f, g, h, and i are provided in Appendix A.1.

Thus far, we have established the kinematics of the leg module for the wheeled biped robot. This lays the foundation for the subsequent dynamic modeling, acceleration analysis, and trajectory planning.

3.2. Dynamic modeling for the wheeled biped robot

The robot maintains point contact with the ground, with only two contact points, which cannot form a stable contact surface. It requires additional driving forces and control algorithms to achieve body balancing. Therefore, before further investigating the parallel structure of the robot, the issue of self-balancing needs to be addressed.

When the leg modules of the robot are locked, the wheeled biped robot can be simplified into a first-order inverted pendulum (FIP) model with two drive wheels, as shown in Figure 4. In this model, the drive wheel is simplified as a uniformly distributed disk with a certain thickness, while the other parts can be simplified as a homogeneous pendulum. This robot is a typical example of an underactuated system [44, 45]. In Figure 4(a), two coordinate frames are established: the ground coordinate system, denoted as $O_G - x_G y_G z_G$, and the pendulum coordinate frame, denoted as $O_B - x_B y_B z_B$. In the figure, the direction of y_G and y_B is aligned, z_G and z_B are created by the right-hand rule, and the origin points O_G and O_B coincide with each other. The robot achieves yaw motion (YM) (y_B-axis) and pitch motion (PM) (z_B-axis) as well as forward motion (FM) through the drive wheels. The pendulum pitch angle (PPA) of the inverted pendulum is represented as θ, while the pendulum yaw angle (PYA) is denoted as δ. YM is achieved by the differential rotation of the wheels, while PM and FM are achieved through codirectional rotation of the wheels. Therefore, by decoupling the YM and the other two movements, the model can be decomposed into planar motion and rotational motion, with their respective dynamics to be discussed separately.

For the sake of clarification, a force analysis on the right wheel in Figure 4(a) is conducted, as shown in Figure 4(b). Assuming the robot is in motion, a force analysis of the wheels yields the following

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

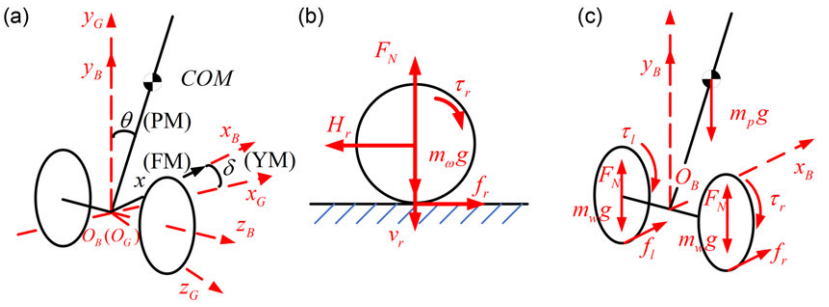


Fig. 4. *Dynamic analysis of the wheeled biped robot.*

equations for both force equilibrium and torque equilibrium:

$$\begin{cases} f_r - H_r = m_r \ddot{x}_r \\ \dfrac{J_\omega}{R} \ddot{x}_r = \tau_r - f_r R \end{cases} \tag{11}$$

where f_r represents the frictional force exerted by the ground on the right wheel, τ_r is the active torque applied to the right wheel, H_r and v_r result from the gravitational force $m_p g$ of the pendulum, leading to the horizontal and vertical components of the force exerted on the wheel by the wheel axis, m_r is the mass of the wheel, J_ω is the moment of inertia of the wheel around its axial direction, x_r is the forward displacement of the right wheel, and R is the radius of the wheel. Similar conclusions apply to the left wheel. Figure 4(c) shows the external forces applied to the entire robot system. Let the axle displacement of the FIP be denoted by x, which can be expressed as:

$$x = \frac{1}{2}\left(x_l + x_r\right) \tag{12}$$

It is assumed that the left and right wheels of the FIP have the same mass and inertia. So $m_w = m_l = m_r$, $J_\omega = J_{\omega l} = J_{\omega r}$. From Eq. 11, one can obtain:

$$\begin{cases} \left(f_l + f_r\right) - \left(H_l + H_r\right) = m_w \left(\ddot{x}_l + \ddot{x}_r\right) \\ \dfrac{J_\omega}{R}\left(\ddot{x}_l + \ddot{x}_r\right) = \left(\tau_l + \tau_r\right) - \left(f_l + f_r\right) R \end{cases} \tag{13}$$

Eq. 12 gives $\ddot{x} = \left(\ddot{x}_l + \ddot{x}_r\right)/2$. By combining Eq. 12 with Eq. 13 to eliminate the static friction force $f_l + f_r$, we can obtain:

$$2\frac{J_\omega}{R}\ddot{x} + 2m_w R\ddot{x} = \left(\tau_l + \tau_r\right) - \left(H_l + H_r\right) R \tag{14}$$

For a homogeneous pendulum, conducting a force analysis at its COM yields:

$$\begin{cases} m_p \ddot{x}_p = H_l + H_r \\ m_p \ddot{y}_p = V_l + V_r - m_p g \\ J_p \ddot{\theta} = -\left(\tau_l + \tau_r\right) - \left(H_l + H_r\right) L\cos\theta + \left(V_l + V_r\right) L\sin\theta \end{cases} \tag{15}$$

where J_p is the moment of inertia of the pendulum around the z_B axis. The relationship between the displacement of the equivalent COM of the pendulum in the horizontal direction and the wheel axis displacement can be expressed as follows:

$$\begin{aligned} x_p &= x + L\sin\theta \\ \dot{x}_p &= \dot{x} + L\dot{\theta}\cos\theta \\ \ddot{x}_p &= \ddot{x} + L\ddot{\theta}\cos\theta - L\dot{\theta}^2\sin\theta \end{aligned} \tag{16}$$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

The relationship between the two in the vertical direction can be described as:

$$\begin{aligned} y_p &= L\cos\theta \\ \dot{y}_p &= -L\dot{\theta}\sin\theta \\ \ddot{y}_p &= -L\ddot{\theta}\sin\theta - L\dot{\theta}^2\cos\theta \end{aligned} \tag{17}$$

Substituting Eqs. 16 and 17 into Eq. 15 yields:

$$\begin{aligned} J_p\ddot{\theta} &= -(\tau_l+\tau_r) - m_p\ddot{x}_p L\cos\theta + m_p\left(g+\ddot{y}_p\right)L\sin\theta \\ &= -(\tau_l+\tau_r) - m_p L\cos\theta\left(\ddot{x}+L\ddot{\theta}\cos\theta - L\dot{\theta}^2\sin\theta\right) + \\ &\quad m_p L\sin\theta\left(g - L\ddot{\theta}\sin\theta - L\dot{\theta}^2\cos\theta\right) \\ &= -(\tau_l+\tau_r) - m_p\ddot{x}L\cos\theta - m_p L^2\ddot{\theta} + m_p gL\sin\theta \end{aligned} \tag{18}$$

Additionally, by substituting Eqs. 15 and 16 into Eq. 14 and eliminating the horizontal interaction force $H_l + H_r$ between the equivalent pendulum and the wheel axle, we can derive:

$$\begin{aligned} 2\frac{J_\omega}{R}\ddot{x} + 2m_w R\ddot{x} &= (\tau_l+\tau_r) - m_p\ddot{x}_p R \\ &= (\tau_l+\tau_r) - m_p R\left(\ddot{x}+L\ddot{\theta}\cos\theta - L\dot{\theta}^2\sin\theta\right) \end{aligned} \tag{19}$$

After simplifying Eqs. 18 and 19, we obtain the dynamic equations for the PM and FM of the robot:

$$\begin{cases} \left(J_p + m_p L^2\right)\ddot{\theta} + m_p\ddot{x}L\cos\theta = -(\tau_l+\tau_r) + m_p gL\sin\theta \\ \left(2\dfrac{J_\omega}{R} + 2m_w R + m_p R\right)\ddot{x} + m_p RL\ddot{\theta}\cos\theta = (\tau_l+\tau_r) + m_p RL\dot{\theta}^2\sin\theta \end{cases} \tag{20}$$

For the robot's YM, under the assumption that the robot achieves self-balancing through a controller, we have:

$$J_y\ddot{\delta} = \frac{D}{2}(H_r - H_l) \tag{21}$$

where J_y is the moment of inertia of the robot around the y_B-axis, which can be calculated using the parallel-axis theorem, as demonstrated below:

$$J_y = 2\left(J_{\omega y} + m_w\left(\frac{D}{2}\right)^2\right) + \left(J_{py} + m_p(L\sin\theta)^2\right) \tag{22}$$

The PYA, δ, of the robot can be expressed as:

$$\delta = \frac{1}{D}(x_r - x_l) \tag{23}$$

Taking into account Eq. 11, we have:

$$(f_r - f_l) - (H_r - H_l) = m_w(\ddot{x}_r - \ddot{x}_l) = m_w D\ddot{\delta} \tag{24}$$

Then

$$\frac{J_\omega}{R}(\ddot{x}_r - \ddot{x}_l) = \frac{J_\omega}{R}D\ddot{\delta} = (\tau_r - \tau_l) - (f_r - f_l)R \tag{25}$$

By using Eqs. 21, 24, and 25 and eliminating the difference in static friction force $f_r - f_l$ and the difference in horizontal interaction force $H_r - H_l$, we obtain the following dynamic equation for the robot's YM:

$$\left(J_{\omega y}\frac{D}{R} + m_w DR + \frac{2}{D}RJ_y\right)\ddot{\delta} = (\tau_r - \tau_l) \tag{26}$$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

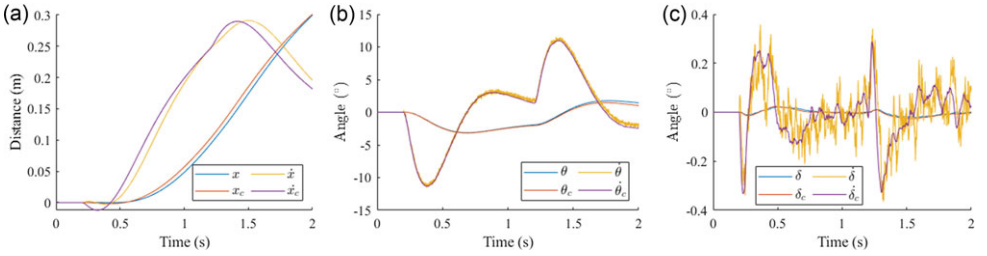


Fig. 5. *Verification of dynamics equations via Simulink.*

Eqs. 20 and 26 comprise the dynamic equations. We have built a simulation model in Simulink to validate these dynamic equations. At a body height of 0.2 m and with an acceleration of 0.3 m/s^2, we obtained two datasets to conduct a comparative analysis, as shown in Figure 5. Herein, state variables with the subscript "c" are calculated according to the dynamics equations, and those without a subscript are collected from Simulink. By inputting the torque into the dynamic equations, we calculate the system's acceleration at the current moment and then integrate the acceleration to obtain state information such as velocity and position. The results extracted from the Simulink model are congruent with those obtained using the dynamic equations, confirming the correctness of the derivation of these equations.

So far, through a Newtonian mechanics-based analysis, we have obtained dynamic expressions of the robot corresponding to the locked parallel structure (Eqs. 20 and 26). It can be observed that the robot's dynamics involve nonlinear terms such as gravitational torque, centrifugal force, and variable inertia parameters. When the robot is in a self-balancing state, the PPA, θ, has a small value, and the rate of change of the angle, denoted as $\dot{\theta}$, is also small. Therefore, the following linearization assumptions can be made: $\sin\theta \approx \theta$, $\cos\theta \approx 1$, $\dot{\theta}^2 \approx 0$, and $\sin^2\theta \approx 0$.

This assumption considers that the PPA θ state moves minimally and disregards the effect of the centrifugal force of the pendulum on the wheel axis movement, as well as the influence of the PPA θ on the YM. In this case, the dynamic expressions can be linearized as follows:

$$\begin{cases} \left(J_p + m_p L^2\right)\ddot{\theta} + m_p \ddot{x} L = -\left(\tau_l + \tau_r\right) + m_p g L\theta \\ \left(2\dfrac{J_\omega}{R} + 2m_w R + m_p R\right)\ddot{x} + m_p R L\ddot{\theta} = \left(\tau_l + \tau_r\right) \end{cases} \tag{27a}$$

$$\left(J_{\omega y}\frac{D}{R} + m_w D R + \frac{2}{D} R J_y\right)\ddot{\delta} = \left(\tau_r - \tau_l\right) \tag{27b}$$

3.3. Decoupled representation of the state space and acceleration analysis

By selecting the robot state variable $X = \left[\, x\ \dot{x}\ \theta\ \dot{\theta}\ \delta\ \dot{\delta} \,\right]^T$ and the control variable $u = \left[\, \tau_l\ \tau_r \,\right]^T$, the linearized robot dynamics can be represented in state-space form:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & A_{23} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & A_{43} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \\ \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ B_{21} & B_{22} \\ 0 & 0 \\ B_{41} & B_{42} \\ 0 & 0 \\ B_{61} & B_{62} \end{bmatrix} \begin{bmatrix} \tau_l \\ \tau_r \end{bmatrix} \tag{28a}$$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

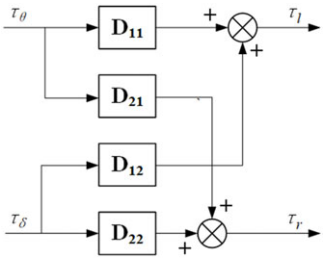


Fig. 6. *State-space model decoupling process.*

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \\ \delta \\ \dot{\delta} \end{bmatrix} \tag{28b}$$

where the precise forms of $M, A_{23}, A_{43}, B_{21}, B_{41}, B_{61}, B_{22}, B_{42}$ and B_{62} can be viewed in Appendix A.2.

In this case, from the linear state-space equations, it can be observed that the control torques applied to the left and right wheels simultaneously affect the pitch, yaw, and FM. However, the rotational states around the y_B-axis, represented via δ and $\dot{\delta}$, remain unaffected by other state variables. Based on this relationship, it is straightforward to decouple the state space into two independent state spaces and control them separately (this result relies on the assumption that θ is small and changes minimally, and that the robot carries a static load).

Two new control variables are introduced, namely τ_θ and τ_δ, that, respectively, represent the driving torques that control the PPA θ and the PYA δ of the control system. Through an intermediate decoupling link, as shown in Figure 6, a conversion from control variables τ_l and τ_r to τ_θ and τ_δ can be achieved [46].

The decoupling process can be expressed via the following equations:

$$\begin{bmatrix} \tau_l \\ \tau_r \end{bmatrix} = \begin{bmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} \tau_\theta \\ \tau_\delta \end{bmatrix} = \begin{bmatrix} 0.5 & -0.5 \\ 0.5 & 0.5 \end{bmatrix} \begin{bmatrix} \tau_\theta \\ \tau_\delta \end{bmatrix} \tag{29}$$

After decoupling, the two state-space models can be expressed as:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & A_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & A_{43} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ B_{21} \\ 0 \\ B_{41} \end{bmatrix} \tau_\theta \tag{30a}$$

$$\begin{bmatrix} \dot{\delta} \\ \ddot{\delta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta \\ \dot{\delta} \end{bmatrix} + \begin{bmatrix} 0 \\ B_{62} \end{bmatrix} \tau_\delta \tag{30b}$$

As shown by the equations above, controllers can be designed separately for the two state-space models. The first controller maintains the robot's PM within a certain range, achieving self-balancing and FM. The second controller, building upon the balanced state, controls the YM of the robot, enabling it to change its direction.

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

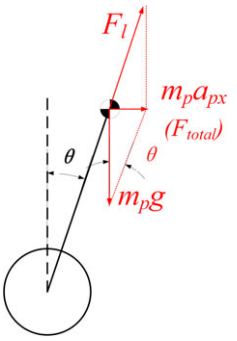


Fig. 7. *Acceleration analysis of the wheeled biped robot.*

For the simplified inverted pendulum model of the wheeled biped robot, its FM requires both the pendulum and the wheels to experience the same acceleration. For the wheels, this acceleration is provided by the static frictional force exerted by the ground. Analyzing the forces acting on the pendulum can be performed by following the approach illustrated in Figure 7.

As observed from Figure 7, the combined force F_{total} of the wheel axle provides horizontal acceleration to the pendulum [47]. When the robot maintains a constant PPA θ under the influence of the controller, the acceleration of the pendulum's COM in the vertical direction is 0. Therefore, using trigonometric analysis, the horizontal acceleration can be deduced as follows:

$$a_{px} = g \tan \theta \tag{31}$$

It can be seen that in order for the pendulum to attain a horizontal acceleration, it is necessary for the vehicle to generate and maintain a tilt angle θ. However, for traditional wheeled biped robots, due to the absence of adjustable leg modules, the generation of the equivalent pendulum tilt angle can only originate from the CTA. This is disadvantageous for subsequent installation of upper-body mechanisms, such as manipulators, cameras, and LiDAR, which, to some extent, constrains the development of self-balancing robots.

4. Cascade controller design to reduce chassis tilt angle (CTA)

When the robot needs to cover long distances on flat ground, locking the leg modules to attain a general wheeled biped self-balancing robot configuration can effectively reduce the energy consumption relating to the leg module's active joints. However, in scenarios where the robot requires frequent short-distance movements, frequent acceleration and braking can lead to oscillations in the CTA. In such cases, unlocking the leg module is necessary to maintain the horizontal orientation and to enable changes in vertical height. The overall control framework corresponding to situations in which the leg module is unlocked is depicted in Figure 8.

The entire control framework can be divided into the parallel mechanism controller, the wheel joints controller, and the measurement and equivalent parameters calculation. In the parallel mechanism controller, the approach relies upon calculating the leg joint angles based on the desired height and displacement to change the robot's configuration. The control of the wheel joints is divided into self-balancing control and turning direction control, with corresponding controllers designed for each one of these. The outputs of the controllers, τ_θ and τ_δ, are then converted into output torques, τ_l and τ_r, for the wheel joints, which are then sent to the robot. In the measurement and equivalent parameters calculation layer, the corresponding parameters are calculated based on the actual output and measurement results and then fed back to the wheel joint controllers to close the loop.

The central concepts of the control framework revolve around real-time estimation of the length L_{eq} between the chassis equivalent COM and the wheel axis, as well as the equivalent PPA, θ_{eq}, of the

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

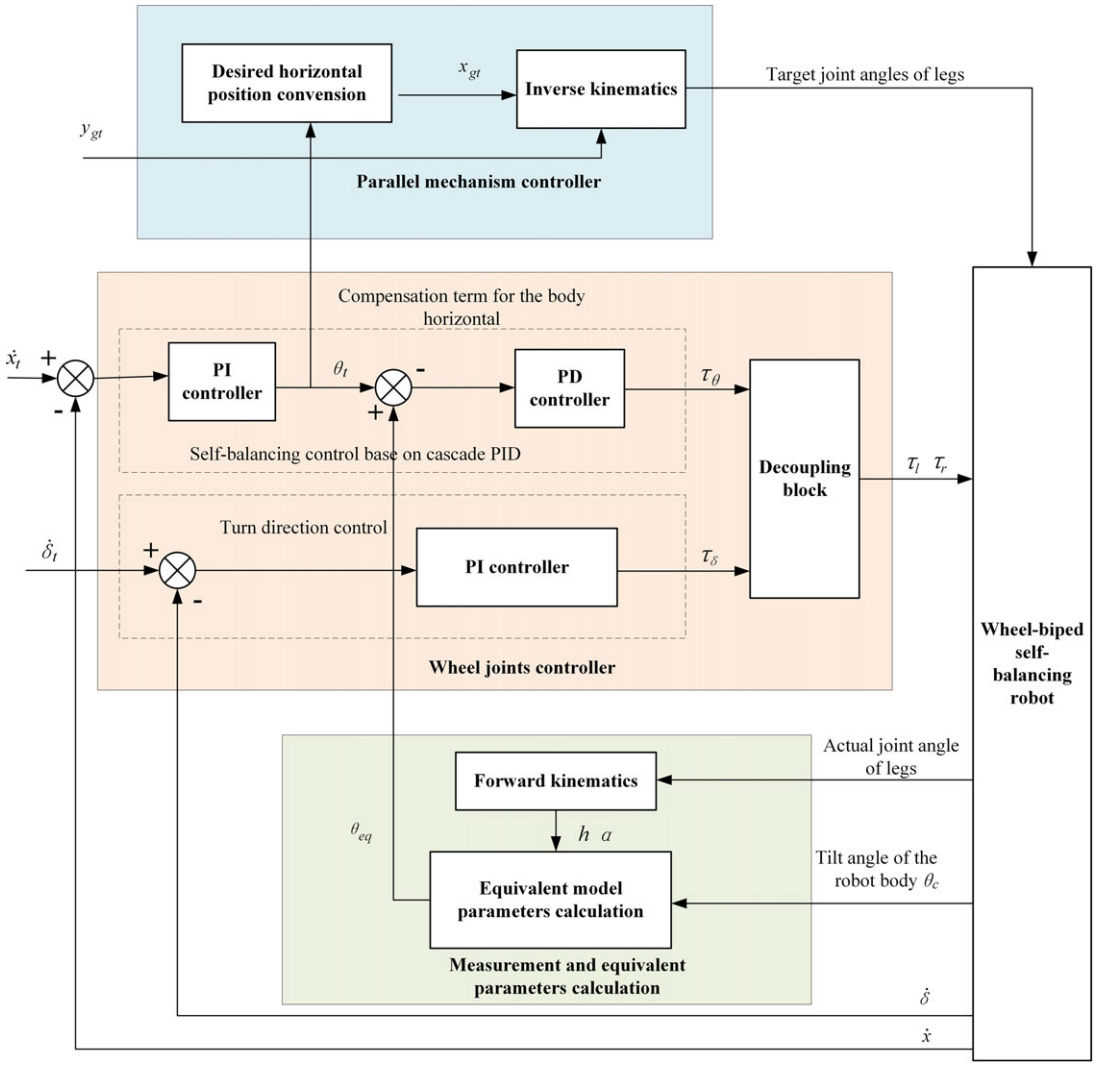


Fig. 8. *Control frame when the leg module is unlocked.*

inverted pendulum model. This treatment ensures that the entire model can be approximated by an FIP. Closed-loop control is applied to the forward velocity of the wheel axis, using the position of the wheel axis relative to the robot as the desired foot-tip position. This process ensures the maintenance of the desired horizontal orientation and vertical height.

The subsequent sub-sections sections provide specific information about the balancing, and velocity, controllers of the robot.

4.1. Balance controller design of the robot

When the leg module is locked, the robot's structure is simplified into that of a conventional wheeled biped self-balancing robot, which is then analyzed using the classical FIP model. When the leg module is unlocked, it is controlled by joint actuators, and all parts of the robot are assumed to be rigid bodies with constant masses. Even when the joint actuator speed is slow, the robot can still be abstracted as a variable-length FIP model. Based on this model, an analysis is performed to determine the left and right

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

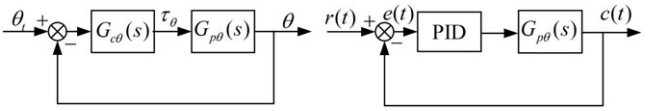


Fig. 9. *Pendulum pitch angle θ closed-loop control and single-input-single-output control system.*

wheel joint driving torques required to enable self-balancing control. Controlling the robot's balance corresponds to controlling the PPA, θ.

From Eq. 30(a), it is known that the differential equation for the PPA, θ, can be expressed as follows:

$$\ddot{\theta} = A_{43}\theta + B_{41}\tau_\theta \tag{32}$$

Furthermore, based on an earlier derivation, $A_{43} = -|A_{43}| < 0$, $B_{41} > 0$ (in Appendix A.2). Performing a Laplace transform on the above equation yields the transfer function of the PPA, θ, with respect to the driving torque τ_θ:

$$G_{p\theta}(s) = \frac{\theta(s)}{\tau_\theta(s)} = \frac{B_{41}}{s^2 + |A_{43}|} \tag{33}$$

This transfer function possesses a pair of complex conjugate poles located on the imaginary axis, indicating critical stability. This implies that if the system is at an equilibrium point, a slight disturbance could cause the system to diverge, which aligns with our intuitive understanding: perturbing the robot's wheel joint with a sufficient large force/moment, while in an upright position, will lead to the robot falling over. Considering closed-loop control of the angle θ, the concomitant feedback control system is designed as shown in Figure 9(a).

The $G_{c\theta}(s)$ controller's design is based on a classical Proportional-Integral-Derivative (PID) control scheme. In single-input-single-output continuous systems, the PID control algorithm, which functions in accordance with the proportional, integral, and derivative terms of the error, is a widely used control law in practical engineering applications. For the continuous system depicted in Figure 9(b), a PID controller of the form shown in Eq. 34 below is employed:

$$u = K_p\left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d\tau + T_d\frac{de(t)}{dt}\right) \tag{34}$$

where K_p, T_i and T_d correspond to proportional gain, integral time constant, and differential time constant, respectively, and $e(t)$ is the deviation between the desired value and the actual value of the controlled variable of the system, namely the error. In the self-balancing control of the wheeled biped robot, based on the open-loop system that is itself in a critically stable state, the concomitant unity negative feedback closed-loop system can be described via the following transfer function:

$$G_{close\theta}(s) = \frac{\theta(s)}{\theta_t(s)} = \frac{G_{c\theta}(s)G_{p\theta}(s)}{1 + G_{c\theta}(s)G_{p\theta}(s)} \tag{35}$$

If only the proportional control element is used, the characteristic equation of the closed-loop system is:

$$\Delta(s) = s^2 + |A_{43}| + B_{41}K_p \tag{36}$$

When $K_p \geq 0$, the system poles are located on the imaginary axis. However, when $K_p < 0$, the system will have positive real poles. This implies that regardless of the value of K_p, the closed-loop system under pure integral control remains unstable. Therefore, considering the adoption of PD control is necessary. Taking $G_{c\theta}(s) = K_pT_ds + K_p$, the characteristic equation of the closed-loop system becomes:

$$\Delta(s) = s^2 + B_{41}K_pT_ds + \left(|A_{43}| + B_{41}K_p\right) \tag{37}$$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

whose solution can be expressed as:

$$\begin{aligned} s_{1,2} &= \frac{-B_{41}K_pT_d}{2} \pm \frac{1}{2}\sqrt{\Delta} \\ &= \frac{-B_{41}K_pT_d}{2} \pm \frac{1}{2}\sqrt{\left(B_{41}K_pT_d\right)^2 - 4\left(|A_{43}| + B_{41}K_p\right)} \end{aligned} \tag{38}$$

To ensure stability of the closed-loop system, the solutions of the characteristic equation should have negative real parts. Taking $K_pT_d > 0$, when $\Delta < 0$, the closed-loop poles are complex conjugates with negative real components. When $\Delta \geq 0$, it is only necessary to ensure that:

$$|A_{43}| + B_{41}K_p > 0 \tag{39}$$

The choice of different values for the parameters K_p and T_d will lead to different performance of the closed-loop system. In the case of self-balancing robots, it is generally desirable for the system to exhibit as fast a response as possible, and minimal fluctuations in the pitch axis angle. Therefore, when applying this PD controller in practice, larger values of K_p and K_pT_d can be employed to fine-tune a set of parameters that provide improved self-balancing performance.

4.2. Cascaded PID-based velocity controller design for the robot

As deduced from the acceleration analysis in Section 3.3, it is apparent that for the robot to possess a non-zero velocity while maintaining self-balancing, the robot needs to generate a certain PPA, θ, to provide forward acceleration. Consequently, the problem of controlling the robot's velocity fundamentally becomes a trajectory-tracking problem for PPA θ. Treating the subsystem $G_{close\theta}(s)$, refined through the PD control law, as an inner-loop subsystem, we can derive the relationship between the forward displacement x and the desired PPA θ_t from Eq. 30 as:

$$\begin{aligned} x(s)s^2 &= A_{23}\theta(s) + B_{21}\tau_\theta \\ &= A_{23}\theta(s) + B_{21}G_{c\theta}(s)\left(\theta_t(s) - \theta(s)\right) \\ &= \left(A_{23} - B_{21}G_{c\theta}(s)\right)\theta(s) + B_{21}G_{c\theta}(s)\theta_t(s) \\ &= \left(\left(A_{23} - B_{21}G_{c\theta}(s)\right)G_{close\theta}(s) + B_{21}G_{c\theta}(s)\right)\theta_t(s) \end{aligned} \tag{40}$$

That is

$$G_{px}(s) = \frac{x(s)}{\theta_t(s)} = \frac{\left(\left(A_{23} - B_{21}G_{c\theta}(s)\right)G_{close\theta}(s) + B_{21}G_{c\theta}(s)\right)}{s^2} \tag{41}$$

The characteristic equation of the system can be expressed as:

$$\Delta(s) = s^2\left(s^2 + B_{41}K_pT_ds + \left(|A_{43}| + B_{41}K_p\right)\right) \tag{42}$$

In addition to having two poles corresponding to the PPA θ, the system also possesses two poles with values of 0. One of them arises from the pure integral of the forward velocity $\dot{x}$ to the forward displacement x, while the other is inherent to the forward velocity $\dot{x}$. If we set $v(s) = x(s)s$, then the transfer function $G_{pv}(s)$ from the target PPA $\theta_t(s)$ to the forward velocity $v(s)$ has the following characteristic equation:

$$\Delta(s) = s\left(s^2 + B_{41}K_pT_ds + \left(|A_{43}| + B_{41}K_p\right)\right) \tag{43}$$

If the forward velocity $\dot{x}$ is taken as the controlled variable and the desired forward velocity is used as the reference input $\dot{x}_t$, applying unity negative feedback to the transfer function $G_{pv}(s)$ as shown in Figure 10, it can be observed that the system can achieve the following step input commands:

$$\begin{aligned} err &= \left(\dot{x}_r(\infty) - \dot{x}(\infty)\right) \\ &= \lim_{s\to 0} sR(s)G_{pv}(s) \\ &= \lim_{s\to 0} s\frac{1}{s}G_{pv}(s) \end{aligned} \tag{44}$$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

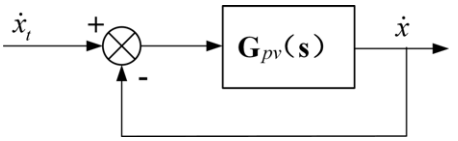


Fig. 10. *Unity negative feedback on the transfer function $G_{pv}(s)$.*

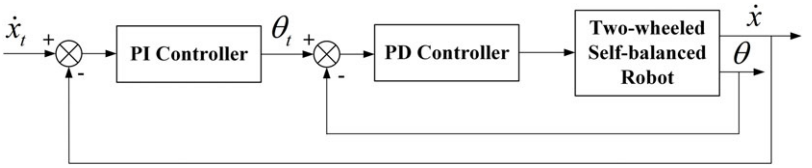


Fig. 11. *Closed-loop control system based on cascade PID for the robot.*

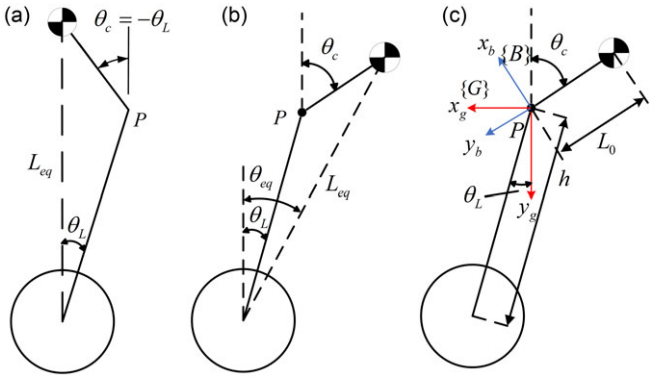


Fig. 12. *Diagram of equivalent COM.*

For robot velocity control, pure integral closed-loop control can theoretically achieve set-point regulation of the desired forward velocity. Considering the existence of unmodeled joint friction and sensor observation errors in the actual robot, practical implementation of closed-loop velocity control on the robot is achieved using a PI controller. This forms a cascade control system as shown in Figure 11.

By appropriately adjusting the parameters of the self-balancing PD controller and the velocity loop PI controller, it is possible to achieve velocity control for self-balancing and FM of the robot in both the locked and unlocked states of the leg modules.

4.3. Relationship between CTA and inverted pendulum model after unlocking of leg modules

Concerning the balancing controller of the robot, its objective is to guide the convergence of the equivalent PPA θ_{eq} of the FIP model, rather than controlling the CTA θ_c. This difference in objectives may result in the equivalent PPA θ_{eq} being zero during movement, while the CTA θ_c is not, as shown in Figure 12(a). Consequently, the control framework necessitates additional compensation for the CTA θ_c. This compensation is realized through a PD control scheme, producing a torque output that supplements the output torque of the balancing controller.

For the sake of clarity, let us define the midpoint of the two hip joints of the same leg module as point P. The distance from P to the foot end is denoted as h, and the angle between the line h and the vertical line is θ_L. The line from the chassis's equivalent COM to point P is designated as L_0, and the angle between the line L_0 and the vertical line is defined as θ_c. The distance from the chassis's equivalent COM to the foot end is represented as L_{eq}, and the angle between the line L_{eq} and the vertical line is

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

defined as θ_{eq}. At point P, we establish a two-dimensional ground coordinate frame, denoted as x_gPy_g, and a chassis coordinate frame, denoted as x_bPy_b.

As shown in Figure 12(b), the coordinates (x_b, y_b) calculated from the forward kinematics are in the body coordinate frame x_bPy_b, and they need to be transformed into (x_g, y_g) coordinates in the ground coordinate frame x_gPy_g, as follows:

$$\begin{aligned} x_g &= y_b \sin(\theta_c) + x_b \cos(\theta_c) \\ y_g &= y_b \cos(\theta_c) - x_b \sin(\theta_c) \end{aligned} \tag{45}$$

Subsequently, we can calculate θ_L and h:

$$\begin{aligned} \theta_L &= \tan^{-1}\left(\frac{x_g}{y_g}\right) \\ h &= \sqrt{{x_g}^2 + {y_g}^2} \end{aligned} \tag{46}$$

By implementing closed-loop PID control on the forward linear velocity of the wheel axis, the desired horizontal position x_{gt} of the wheel axis relative to point P can be obtained, while the desired vertical position y_{gy} of the wheel axis relative to point P will be directly specified by the command. The coordinates (x_t, y_t) in the body coordinate frame x_bPy_b can be computed using Eq. 47:

$$\begin{aligned} x_t &= x_{gt} \cos(\theta_c) - y_{gt} \sin(\theta_c) \\ y_t &= x_{gt} \sin(\theta_c) + y_{gt} \cos(\theta_c) \end{aligned} \tag{47}$$

Using the leg module's inverse kinematics model, the required joint angles are calculated and then used to control the leg motors' movement.

By continuously estimating the real-time position of the equivalent COM relative to the wheel axis, it becomes possible to calculate the angle θ_{eq} and the length L_{eq}. These calculations are subsequently integrated into the FIP model. Given that the chassis typically carries additional mechanisms, resulting in a total mass exceeding that of the leg module, the influence of the leg modules is temporarily disregarded when calculating the equivalent COM. Based on the geometric relationship illustrated in Figure 12(c), θ_{eq} and L_{eq} can be expressed as:

$$\begin{aligned} L_{eq} &= \sqrt{h^2 + L_0^2 - 2hL_0 \cos(\pi + \theta_c - \theta_L)} \\ \theta_{eq} &= \theta_L + \sin^{-1}\left(\frac{L_0}{L_{eq}}\right)(\theta_L - \theta_c) \end{aligned} \tag{48}$$

Then $\dot{\theta}_{eq}$ can be expressed as:

$$\dot{\theta}_{eq} = \dot{\theta}_L + \sin^{-1}\left(\frac{L_0}{L_{eq}}\right)\dot{\theta}_L - \sin^{-1}\left(\frac{L_0}{L_{eq}}\right)\dot{\theta}_c \tag{49}$$

Therefore, the angle θ_{eq} and its rate of change $\dot{\theta}_{eq}$, in the above equation, correspond to θ and $\dot{\theta}$ in the FIP model, respectively, which yields the input torques required for self-balancing of the robot.

5. Simulation and experiments for wheeled biped robot

5.1. Simulation of motion and control

To validate the modeling and control strategies proposed in this paper, simulations were conducted using a robot dynamics simulation platform. Webots, an open-source rigid-body robot dynamics simulation software utilizing the Ordinary differential equation (ODE) physics engine, was selected for its user-friendly interface [48]. In this study, Webots was chosen as the dynamic simulation platform, and a simplified wheeled-legged self-adaptive balancing robot was constructed, as depicted in Figure 13. In Figure 13, (a), (b), and (c) respectively depict a robot walking on flat ground, traversing rough terrain, and walking on a slope in the simulation.

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

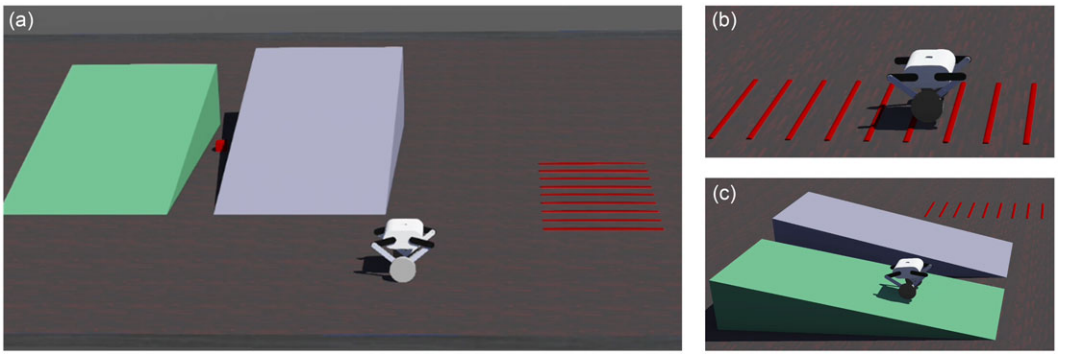


Fig. 13. *Snapshots of the wheeled-legged robot simulated using Webots.*

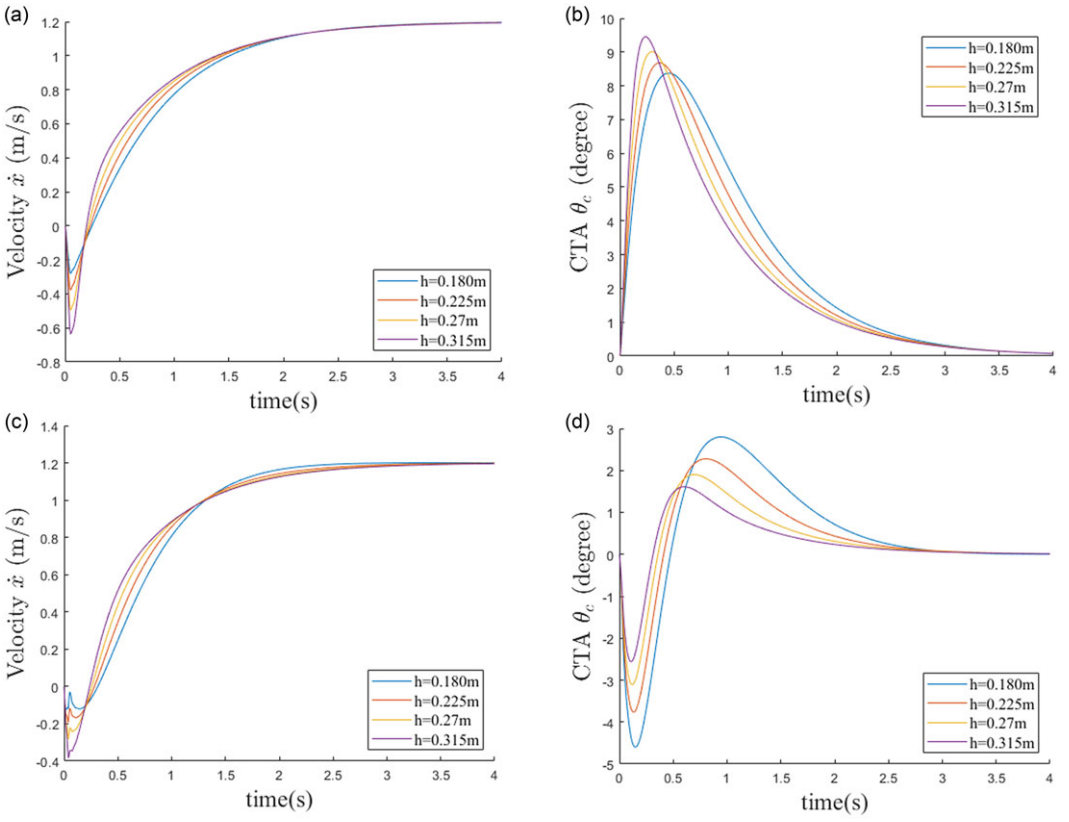


Fig. 14. *Simulation responses generated when using constant target velocity values (Top: legs locked, bottom: legs unlocked).*

The simulation utilized the dynamic parameters presented in Table II in Appendix A.3. A simulation step of 0.002 s (500 Hz) was set. The PID control parameters were chosen as detailed in Table III in Appendix A.3.

To evaluate the effectiveness of the control framework, simulation experiments were conducted under varying target velocity commands and different chassis height commands. A fixed target velocity command of 1.2 m/s was used, and the resulting translational velocity and CTA response were obtained for chassis heights of 0.18 m, 0.225 m, 0.27 m, and 0.315 m, as shown in Figure 14.

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

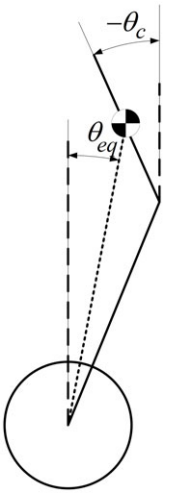


Fig. 15. *Schematic diagram of equivalent PPA θ_{eq} producing an acceleration.*

Figure 14's subfigures (a) and (b) depict the robot's speed and CTA response when the legs are locked, whereas (c) and (d) display these same variables when the legs are unlocked. In subfigures (a) and (c) of Figure 14, for different chassis heights, the controller effectively achieves following of the 1.2 m/s velocity command in the simulation environment. Acceleration of the inverted pendulum model relies on the vertical projection of the body's COM onto the wheel axis, inducing an equivalent pendulum tilt angle. When the robot needs to accelerate along the forward axis, it must initially move in the opposite direction to disrupt its balance and generate a positive tilt angle. Regarding subfigures (b) and (d), when the robot generates an equivalent PPA (θ_{eq}), the wheel axis moves relative to the chassis in the opposite direction. This will cause the entire robot to be subjected to a recovery torque due to the CTA compensator. This torque causes the chassis to exhibit a reverse tilt angle ($-\theta_c$), thus forming an equivalent PPA (θ_{eq}) used for acceleration (as depicted in Figure 15). The amplitude of the CTA's fluctuations does not exceed the 0 ° – 2.9 ° range. (Unless otherwise noted, this article discusses CTA without regard to the initial reverse tilt process.)

Using a fixed chassis target height command of 0.225 m, simulation results are obtained for various forward velocity step commands, namely: 0.8 m/s, 1.2 m/s, 1.6 m/s, 2.0 m/s, and 2.4 m/s. As shown in Figure 16, subfigures (a) and (b) display the robot's response when its legs are locked, and (c) and (d) when its legs are unlocked. Under a fixed chassis height, the robot consistently tracks velocity commands of varying magnitudes effectively. As the target velocity values are increased, the corresponding peak CTA amplitudes also increase. After unlocking the leg structure, the range of CTA changes becomes significantly smaller compared to when the legs are locked, always remaining below 3.5 ° (in the FM).

The proposed control strategy's feasibility and stability are validated in the simulation environment. It is demonstrated that, after unlocking the parallel structure, the robot can stably respond to both velocity and chassis height commands, while exhibiting very low levels of CTA oscillation as a tradeoff.

5.2. Motion experiments of the robot prototype

To further validate the effectiveness of the leg module unlocking, and its impact on the robot's motion and control strategy, we constructed a robot prototype as an experimental platform, as shown in Figure 17. Subfigures (a), (b), and (c) in Figure 17, respectively, show the robot traversing flat ground, slopes, and rough terrain. These aspects will be discussed in detail below. The prototype is equipped with a six-axis inertial sensor on the chassis, capable of estimating the ground attitude, three-axis

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

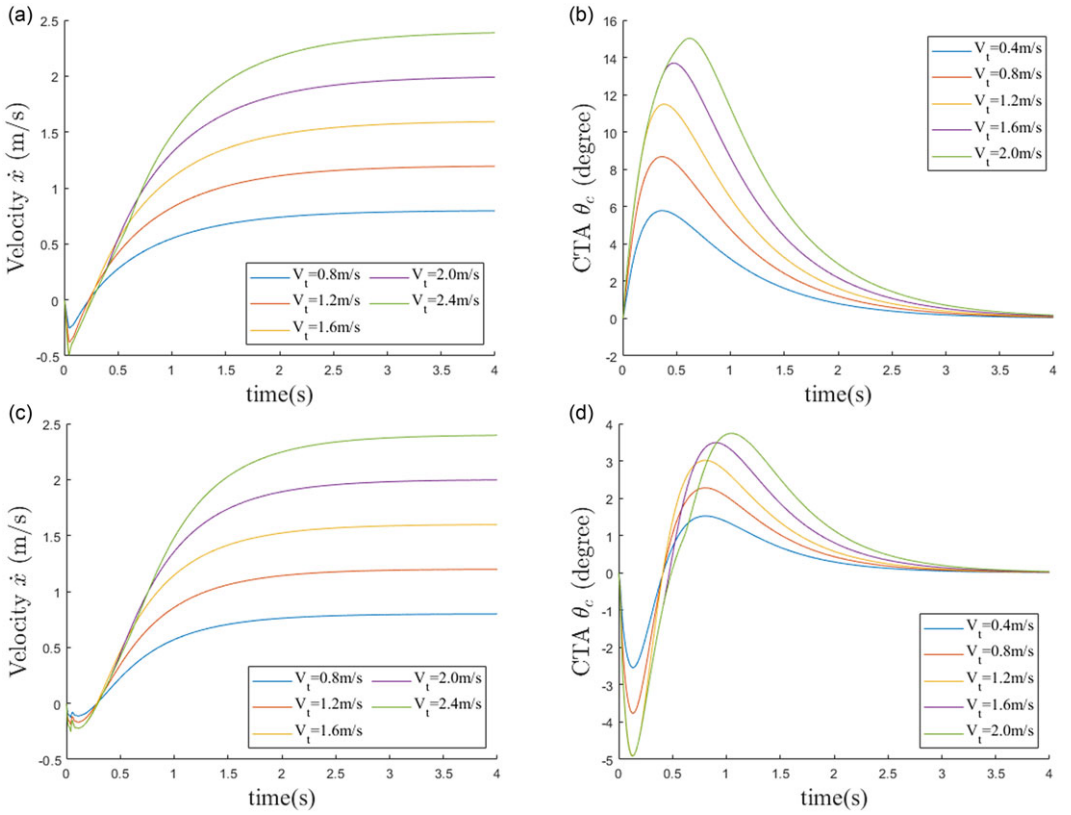


Fig. 16. *Simulation responses generated when using a fixed target height (Top: legs locked, bottom: legs unlocked).*

angular velocities, and three-axis accelerations of the robot chassis. The leg module employs high-torque FOC DC motors to power the active joints, while the wheel mechanism utilizes FOC DC motors to drive the left and right wheel joints. The dynamic parameters of the prototype are outlined in Table IV in Appendix A.3. Consistent with the simulation environment, the control frequency is set to 500 Hz, and after tuning, the PID control parameters are determined as shown in Table V in Appendix A.3.

We altered the relative heights of the chassis and wheel axis to establish a variable-length inverted pendulum, locking the leg module in place. We conducted experiments to assess the robot's FM at four target height states (namely 0.18 m, 0.225 m, 0.27 m, and 0.315 m). The velocity and CTA response curves, pertaining to situations in which the legs are unlocked/locked, are shown in Figure 18. Figure 18 illustrates the velocity and CTA responses during two distinct robot conditions: when the leg joints are locked, as shown in (a) and (b), and when the leg joints are unlocked, as presented in (c) and (d). From (a) and (c), it is evident that the robot's velocity can converge for varying values of equivalent rod length h of the legs, and the convergence speed when the legs are unlocked is faster than when the legs are locked. From (b) and (d), it can be observed that after the legs are unlocked, the robot's CTA decreases from about 7 degrees to approximately 1.5 degrees. Additionally, as the height h decreases, the amplitude of the CTA slightly increases. This is due to a reactive torque generated by the legs during swinging, which acts on the chassis. As the height h decreases, this reactive force increases, augmenting the disturbance to the chassis, thereby leading to a slight increase in the CTA.

Similarly, when the robot's height h is set to a fixed value of 0.225 m, and target velocities of 0.8 m/s, 1.2 m/s, 1.6 m/s, 2.0 m/s, and 2.4 m/s are given during the locking/unlocking of the leg mechanisms,

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

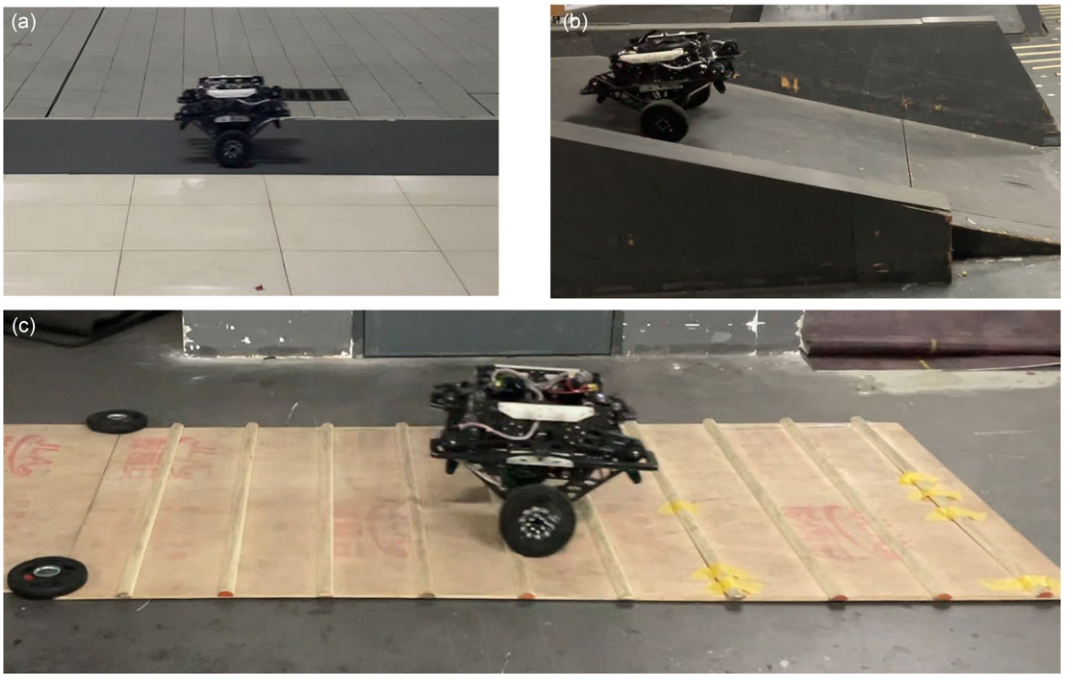


Fig. 17. *Experimental snapshots of the prototype.*

the corresponding robot responses are depicted in Figure 19. Subfigures (a) and (b) display the velocity and CTA responses when the leg mechanisms are locked, while (c) and (d) display the responses after unlocking the leg mechanisms. From (a) and (b), it is observed that as the target velocity increases, the robot requires a larger CTA to provide the necessary acceleration. At the five target velocity settings of 0.8 m/s, 1.2 m/s, 1.6 m/s, 2.0 m/s, and 2.4 m/s, the maximum CTA generated by the robot during acceleration or braking is 4 degrees, 6.3 degrees, 8.6 degrees, 10.3 degrees, and 12 degrees, respectively. By comparing subfigures (b) and (d), it is evident that the trend remains consistent, but after unlocking the leg mechanisms, the maximum CTA value decreases from 12 degrees to 2.7 degrees. This means that, in terms of peak CTA values, the maximum CTA when the leg mechanisms are unlocked is even lower than the minimum CTA when the leg mechanisms are locked. It is evident that unlocking the leg mechanisms effectively reduces the CTA, thereby maintaining the chassis's horizontal orientation.

To further verify the practicality of the proposed control framework for the robot's movement on complex terrains, an experimental environment for the robot's prototype movement was set up, as shown in Figure 17(c). Wooden planks and strips were used to create evenly-spaced obstacles (with a height of 10.3 mm, width of 40 mm, and spacing of 240 mm) protruding from the flat ground for the experiments. The robot's target velocity was set to 1.2 m/s, allowing the robot to accelerate over a 1 m distance on flat ground, then pass through a 1.25 m rugged terrain section, and finally brake at four different target height commands: 0.18 m, 0.225 m, 0.27 m, and 0.315 m. The responses when the leg mechanisms were locked and unlocked are shown in Figure 20. After unlocking the leg mechanisms, the robot's CTA could still be maintained within a 2.9-degree range, which was significantly lower than the corresponding range recorded when the leg mechanisms were locked. This demonstrates that the control framework can effectively maintain the robot's balance when traversing rugged surfaces.

As previously mentioned, reducing the CTA of the robot aids in the support of upper-body loads and completion of operations. Therefore, experiments were also conducted on the robot under different

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

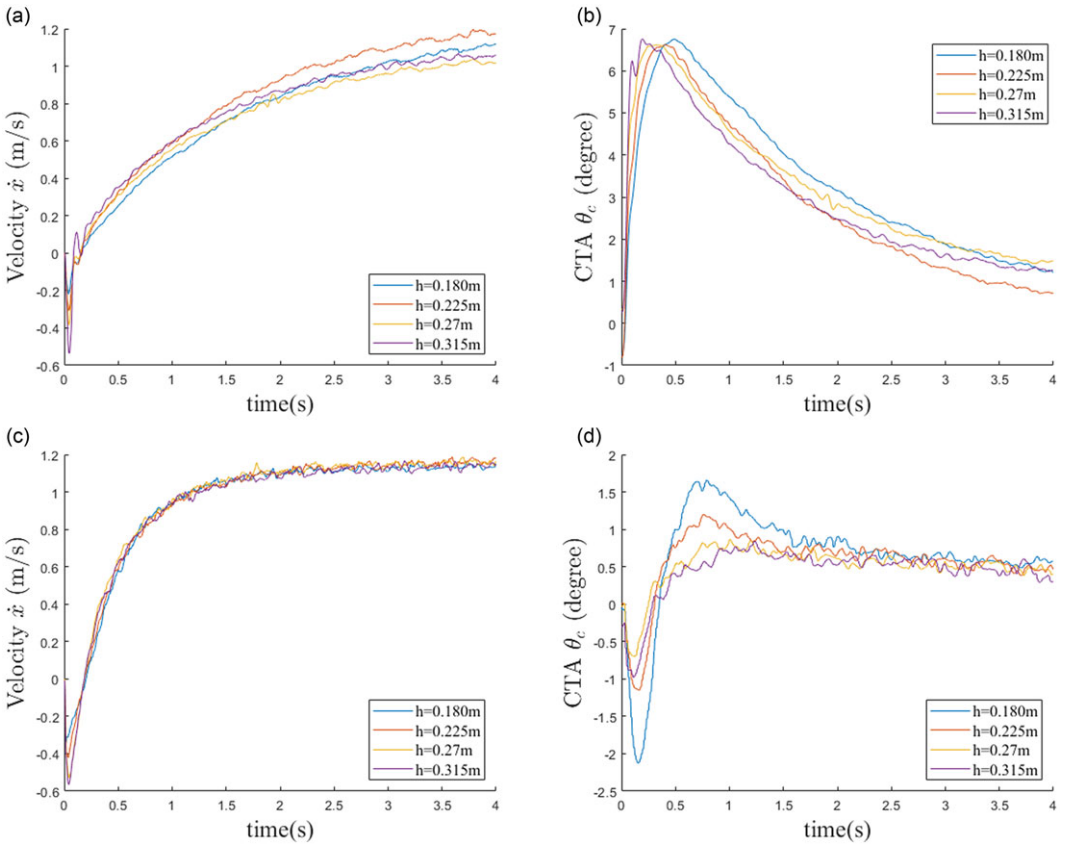


Fig. 18. *Experimental responses generated when using a fixed target velocity (Top: legs locked, bottom: legs unlocked).*

load conditions, as shown in Figure 21. In Figure 21, subfigures (a) and (b) show the robot carrying a load of 10 kg while traversing rugged terrain and while rolling up a 15-degree ramp, respectively. The red arrows in the figures indicate the direction of the robot's movement. Figure 22 displays the CTA responses recorded while the robot was traversing rough terrain and while rolling up a 15-degree slope, under different loads (it should be noted that these are static loads; dynamic loads such as those involving the operation of robotic arms require further discussion). Under loaded conditions, the robot's target speed was set to 1.2 m/s and the target height to 0.225 m, accelerating over a flat surface for 1 m before entering a 1.25 m rough terrain segment or a 15-degree slope, and finally braking. The robot's CTA responses generated when using locked/unlocked leg structures were compared over the same travel distance. Subfigures (a) and (b) respectively show the CTA responses recorded when the robot was traversing rough terrain with locked and unlocked leg structures. Comparing the two figures, it is apparent that with the leg structure unlocked, the tendency of the CTA's value to increase due to the load is less than that when the leg structure is locked. Furthermore, the amplitude of the CTA when the leg structure is unlocked is also smaller than when it is locked. Subfigures (c) and (d) respectively show the CTA responses measured when the robot is rolling up a 15-degree ramp with locked and unlocked leg structures. The main point of interest is the data variation while traversing the slope (the data between the two red vertical lines, with subsequent data showing the robot's response after braking not analyzed here). When the robot carries a smaller load, the CTA amplitude exhibited when the legs are unlocked is significantly smaller than that corresponding to the locking of the legs. As the robot's load gradually increases, the advantage of unlocking diminishes. Overall, compared to when the leg

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

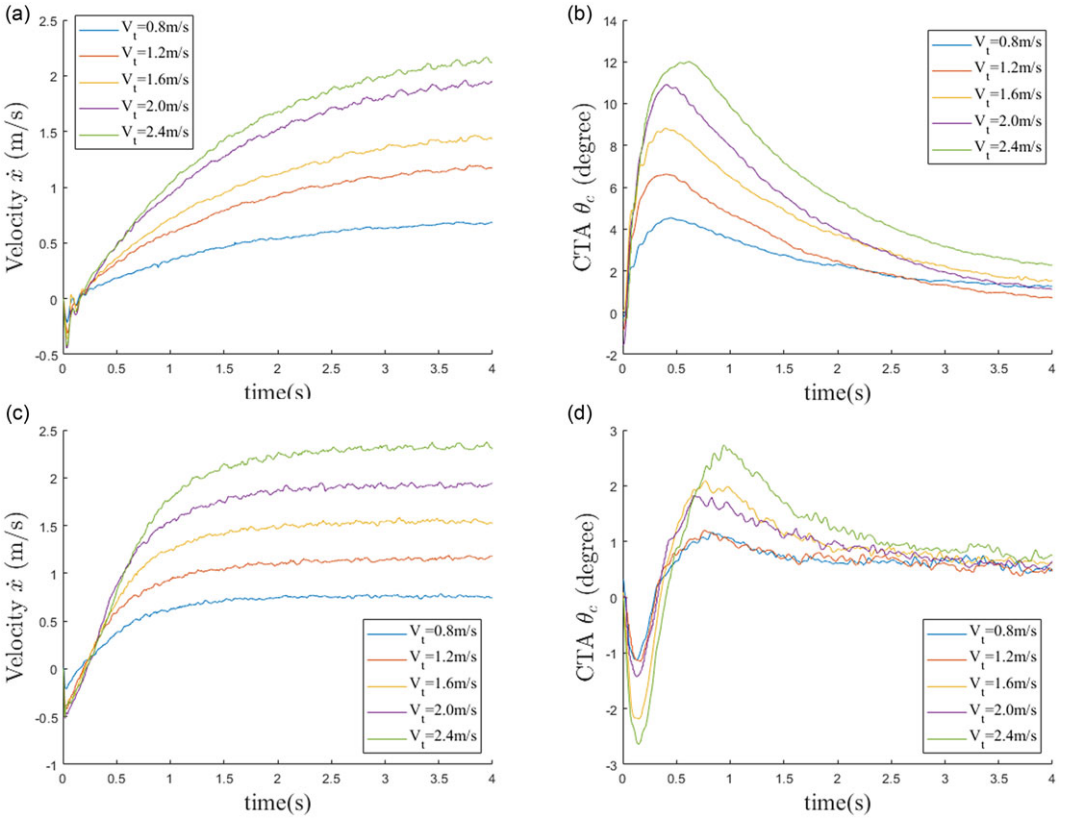


Fig. 19. *Experimental responses generated when using a fixed target height (Top: legs locked, bottom: legs unlocked).*

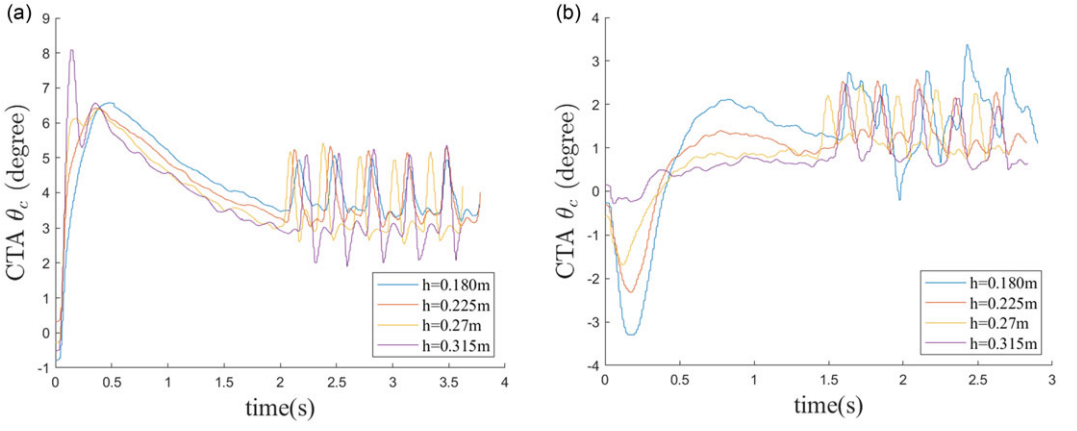


Fig. 20. *Experimental responses recorded during rough terrain experiment (Left: legs locked, right: legs unlocked).*

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

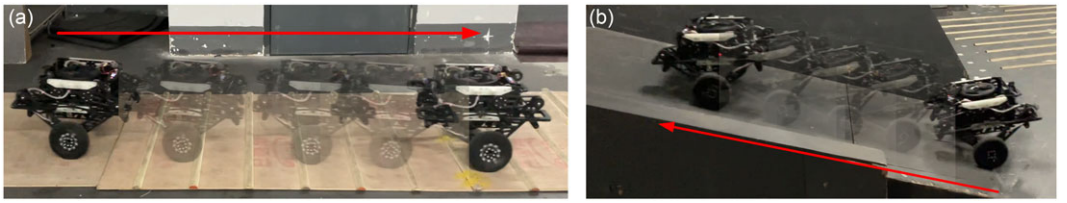


Fig. 21. *Experiment screenshots: the robot carries loads across uneven terrain and slopes.*

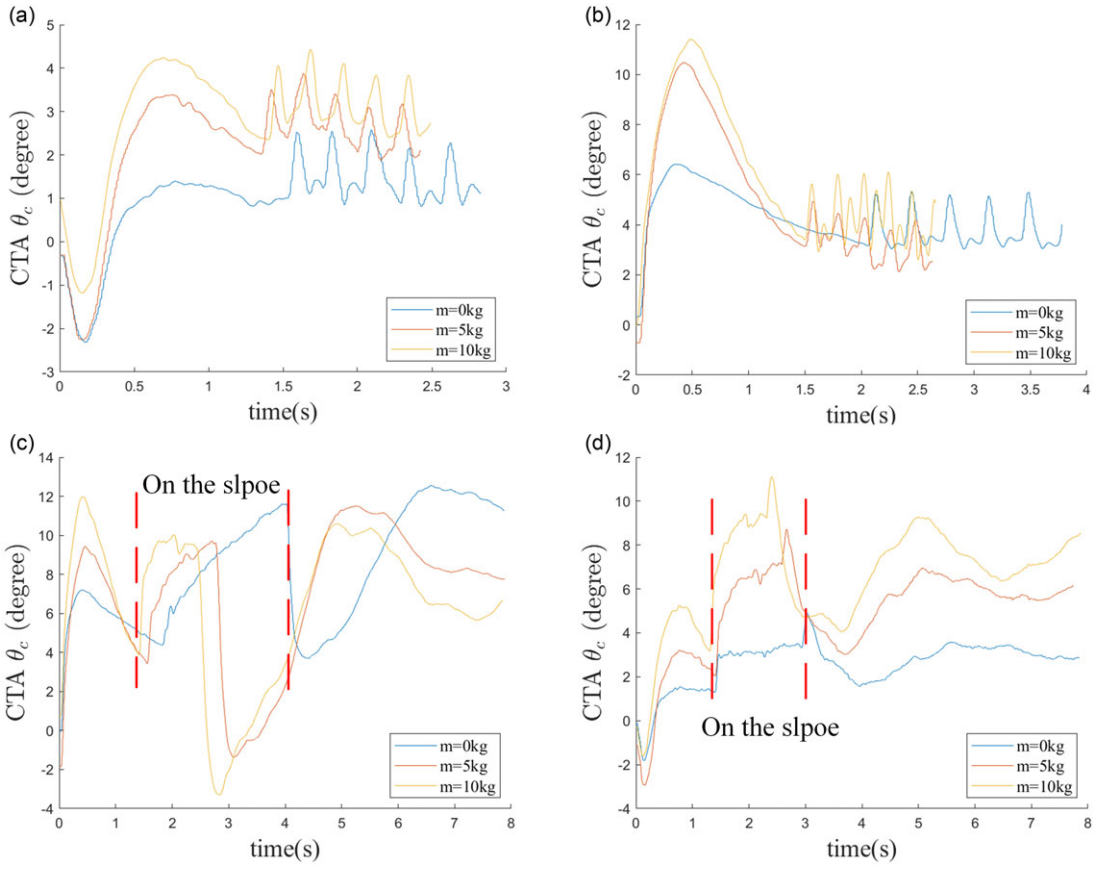


Fig. 22. *Experimental responses recorded when carrying different loads over rough terrain and slopes (Left: legs locked, right: legs unlocked).*

structures are locked, robots with unlocked leg structures exhibit a clear advantage in terms of CTA response.

In conclusion, this study has verified the feasibility of the control strategy with both simulation and a physical platform involving the locked/unlocked leg module. Experimental validations were conducted on the prototype under different configurations. Additionally, to verify the universality of the proposed control framework, experiments were conducted on robots traversing rough terrain and slopes under various load conditions. The results indicate that the control framework can effectively reduce the robot's CTA after unlocking the leg structures, confirming the effectiveness of the control framework. This has significant implications for the widespread application of such robots.

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

6. Conclusions

In order to mitigate the CTA's impact on upper-body structures during a wheeled-legged robot's motion, and to enhance its adaptability in complex environments, we developed kinematic and dynamical models of the wheeled-legged robot. The robot employs a five-bar mechanism as its leg module. This study focuses on conducting the motion-decoupling analysis and introduces an accompanying multilevel cascaded controller. The robot can keep the chassis horizontal with respect to the ground and control the height of the chassis during motion. When the leg structure is unlocked and moving on flat ground at a speed of 1.2 m/s, the maximum CTA value is about 1.2 degrees. Compared to the case when the leg structure is locked, with a maximum CTA amplitude of 6.5 degrees, the unlocked-leg-structure case exhibits a significant improvement. When the wheeled-legged robot carries a load over a rough terrain and a sloping terrain, this cascaded control method can still effectively reduce the CTA to a certain extent. However, as the load increases, the said improvements gradually diminish. Overall, this cascaded control method can effectively reduce the CTA during this type of robot's motion, thereby extending its application range.

In the future, we will further optimize the control method for robots carrying loads (especially dynamic loads like robotic arms), to enhance the robot's performance in complex conditions.

Author contribution. Nan Mao: Conceptualization, writing – original draft, investigation, and methodology. Junpeng Chen: Writing, validation, and investigation. Emmanouil Spyrakos–Papastavridis: Writing, validation, and revision. Jian S. Dai: Supervision, validation, revision, and project administration.

Financial support. We acknowledge the support of the Key Program of the National Natural Science Foundation of China (Grant No. 52335003), the Science, Technology and Innovation Commission of Shenzhen Municipality (Grant No. JCYJ20220818100417038), and the High level of special funds (G03034K003) from the Southern University of Science and Technology.

Competing interests. All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or nonfinancial interest in the subject matter or materials discussed in this manuscript.

References

[1] L. Antonyshyn, J. Silveira, S. Givigi and J. Marshall, "Multiple mobile robot task and motion planning: A survey," *ACM Comput Surv* **55**(10), 1–35 (2023).
[2] H. Tang, J. W. Zhang, L. Pan and D. Zhang, "Optimum design for a new reconfigurable two-wheeled self-balancing robot based on virtual equivalent parallel mechanism," *J Mech Design* **145**(5), 053302 (2023).
[3] K. Yamafuji and T. Kawamura, "Postural control of a monoaxial bicycle," *J Robot Soc Japan* **7**(4), 338–343 (1989).
[4] F. Grasser, A. D'arrigo, S. Colombi and A. C. Rufer, "JOE: A mobile, inverted pendulum," *IEEE Trans Ind Electron* **49**(1), 107–114 (2002).
[5] M. Prado, A. Simón, A. Pérez and F. Ezquerro, "Effects of terrain irregularities on wheeled mobile robot," *Robotica* **21**(2), 143–152 (2003).
[6] J. Luo, S. Ye, J. Su and B. Jin, "Prismatic Quasi-direct-drives for dynamic quadruped locomotion with high payload capacity," *Int J Mech Sci* **235**, 107698 (2022).
[7] H. Giberti, S. Cinquemani and S. Ambrosetti, "5R 2dof parallel kinematic manipulator-A multidisciplinary test case in mechatronics," *Mechatronics* **23**(8), 949–959 (2013).
[8] N. Kau, A. Schultz, N. Ferrante and P. Slade, "Stanford Doggo: An Open-Source, Quasi-Direct-Drive Quadruped," **In:** *2019 International Conference on Robotics and Automation (ICRA)*, (IEEE, 2019) pp. 6309–6315.
[9] E. Ottaviano and P. Rea, "Design and operation of a 2-DOF leg-wheel hybrid robot," *Robotica* **31**(8), 1319–1325 (2013).
[10] J. Cao, J. Zhang, T. Wang, J. Meng, S. Li and M. Li, "Mechanism design and dynamic switching modal control of the wheel-legged separation quadruped robot," *Robotica* **42**(3), 660–683 (2024).
[11] J. S. Dai and J. R. Jones, "Mobility in metamorphic mechanisms of foldable/Erectable kinds," *J Mech Design* **121**(3), 375–382 (1999).
[12] J. S. Dai, Z. Huang and H. Lipkin, "Mobility of overconstrained parallel mechanisms," *J Mech Design* **128**(1), 220–229 (2004).
[13] C. H. Kuo, J. S. Dai and H. S. Yan, "Reconfiguration Principles and Strategies for Reconfigurable Mechanisms," **In:** *2009 ASME/IFToMM International Conference on Reconfigurable Mechanisms and Robots*, (IEEE, 2009) pp. 1–7.
[14] F. Aimedee, G. Gogu, J. S. Dai, C. Bouzgarrou and N. Bouton, "Systematization of morphing in reconfigurable mechanisms," *Mech Mach Theory* **96**, 215–224 (2016).

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

[15] J. S. Dai and D. Wang, "Geometric analysis and synthesis of the metamorphic robotic hand,"Journal," *J Mech Des* **129**(11), 1191–1197 (2007).

[16] J. S. Dai, N. Holland and D. R. Kerr, "Finite twist mapping and its application to planar serial manipulators with revolute joints," *Proc Inst Mech Eng Pt C J Mech Eng Sci* **209**(4), 263–271 (1995).

[17] J. S. Dai, D. Wang and L. Cui, "Orientation and workspace analysis of the multifingered metamorphic hand-metahand," *IEEE Trans Robot* **25**(4), 942–947 (2009).

[18] G. Wei, J. S. Dai, S. Wang and H. Luo, "Kinematic analysis and prototype of a metamorphic anthropomorphic hand with a reconfigurable palm," *Int J Hum Robot* **8**(03), 459–479 (2011).

[19] L. Cui and J. S. Dai, "Posture, workspace, and manipulability of the metamorphic multifingered hand with an articulated palm," *J Mech Robot* **3**(2), 021001 (2011).

[20] L. Cui and J. S. Dai, "Reciprocity-based singular value decomposition for inverse kinematic analysis of the metamorphic multifingered hand," *J Mech Robot* **4**(3), 034502 (2012).

[21] C. Zhang, C. Zhang, J. S. Dai and P. Qi, "Stability margin of a metamorphic quadruped robot with a twisting trunk," *J Mech Robot* **11**(6), 064501 (2019).

[22] Z. Tang, K. Wang, E. Spyrakos-Papastavridis and J. S. Dai, "Origaker: A novel multi-mimicry quadruped robot based on a metamorphic mechanism," *J Mech Robot* **14**(6), 060907 (2022).

[23] Z. Tang and J. S. Dai, "Bifurcated configurations and their variations of an 8-bar linkage derived from an 8-kaleidocycle," *Mech Mach Theory* **121**, 745–754 (2018).

[24] J. Fu, J. Chen, Z. Tang, Z. Wei and J. S. Dai, "Stability Margin Based Gait Design on Slopes for a Novel Reconfigurable Quadruped Robot with a Foldable Trunk," **In:** 2023 IEEE International Conference on Robotics and Biomimetics (ROBIO), (IEEE, 2023) pp. 1–7.

[25] M. Li, W. Shi, J. Chen, Z. Zhuang and J. S. Dai, "Design and Analysis of 8R-Folding Metamorphic Mechanism and Metamorphic Robot," **In:** *Advances in Mechanism and Machine Science*, (M. Okada, eds.) (Springer Nature Switzerland, Cham, 2024) pp. 426–435.

[26] Y. Gu, S. Feng, Y. Guo, F. Wan, J. S. Dai, J. Pan and C. Song, "Overconstrained coaxial design of robotic legs with omni-directional locomotion," *Mech Mach Theory* **176**, 105018 (2022).

[27] J. Chen, Y. Pan, J. Yu, H. Feng and J. S. Dai, "Velocity index and wading height based design method of trajectory parameters for a coupled parallelogram legged walking robot," *Mech Mach Theory* **191**, 105464 (2024).

[28] J. Chen, Y. Pan, M. Li, R. Zhu, M. Gao, K. Wang, X. Xiao, L. Deng and J. S. Dai, "Analysis of the Obstacle-Crossing Capability for a Coupled Parallelogram Leg," **In:** *Advances in Mechanism and Machine Science*, (M. Okada, eds.) (Springer Nature Switzerland, Cham, 2023) pp. 713–722.

[29] Y. Pan and F. Gao, "Position model computational complexity of walking robot with different parallel leg mechanism topology patterns," *Mech Mach Theory* **107**, 324–337 (2017).

[30] Y. Zhou, Y. Pan, J. Chen and T. Lei, "Variable time-step physics engine with continuous compliance contact model for optimal robotic grinding trajectory planning," *Sensors* **24**(5), 1415 (2024).

[31] S. Wang, L. Cui, J. Zhang, J. Lai, D. Zhang, K. Chen, Y. Zheng, Z. Zhang and Z.-P. Jiang, "Balance Control of a Novel Wheel-Legged Robot: Design and Experiments," **In:** *2021 IEEE International Conference on Robotics and Automation (ICRA)*, (IEEE, 2021) pp. 6782–6788.

[32] J. A. Saglia, N. G. Tsagarakis, J. S. Dai and D. G. Caldwell, "Control Strategies for Ankle Rehabilitation Using a High Performance Ankle Exerciser," **In:** *2010 IEEE International Conference on Robotics and Automation*, (IEEE, 2010) pp. 2221–2227.

[33] J. A. Saglia, N. G. Tsagarakis, J. S. Dai and D. G. Caldwell, "Inverse-kinematics-based control of a redundantly actuated platform for rehabilitation proceedings of the institution of mechanical engineers," *Proc Inst Mech Eng Pt I J Syst Control Eng* **223**(1), 53–70 (2008).

[34] V. Klemm, A. Morra, L. Gulich, D. Mannhart, D. Rohr, M. Kamel, Y. d. Viragh and R. Siegwart, "LQR-assisted whole-body control of a wheeled bipedal robot with kinematic loops," *IEEE Robot Autom Lett* **5**(2), 3745–3752 (2020).

[35] S. Xin and S. Vijayakumar, "Online Dynamic Motion Planning and Control for Wheeled Biped Robots," **In:** *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, (IEEE, 2020) pp. 3892–3899.

[36] H. Chen, B. Wang, Z. Hong, C. Shen, P. M. Wensing and W. Zhang, "Underactuated motion planning and control for jumping with wheeled-bipedal robots," *IEEE Robot Autom Lett* **6**(2), 747–754 (2021).

[37] C. Zhang, T. Liu, S. Song, J. Wang and M. Q.-H. Meng, "Dynamic wheeled motion control of wheel-biped transformable robots," *Biomimet Intell Robot* **2**(2), 100027 (2022).

[38] X. Chang, H. An and H. Ma, "Modeling and base parameters identification of legged robots," *Robotica* **40**(3), 747–761 (2022).

[39] A. Siravuru, S. V. Shah and K. M. Krishna, "An optimal wheel-torque control on a compliant modular robot for wheel-slip minimization," *Robotica* **35**(2), 463–482 (2017).

[40] Y. Shen, G. Chen, Z. Li, N. Wei, H. Lu, Q. Meng and S. Guo, "Cooperative control strategy of wheel-legged robot based on attitude balance," *Robotica* **41**(2), 566–586 (2023).

[41] B. Zhu, J. He and J. Sun, "Kinematic modeling and hybrid motion planning for wheeled-legged rovers to traverse challenging terrains," *Robotica* **42**(1), 153–178 (2024).

[42] J. Zhang, G. Li, F. Liu and Y. Liu, "Design of a Two-Wheeled Self-Balance Personal Transportation Robot," **In:** *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, (IEEE, 2016) pp. 225–228.

[43] L. Zhang and J. S. Dai, "Reconfiguration of spatial metamorphic mechanisms," *J Mech Robot* **1**(1), 011012 (2008).

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

[44] S. Kim and S. J. Kwon, "Dynamic modeling of a two-wheeled inverted pendulum balancing mobile robot," *Int J Control Autom Syst* **13**(4), 926–933 (2015).
[45] H. Dong, E. Asadi, C. Qiu, J. Dai and I.-M. Chen, "Geometric design optimization of an under-actuated tendon-driven robotic gripper," *Robot Comp Int Manuf* **50**, 80–89 (2018).
[46] C. Wang, X. Jianliang and C. Zhang, "Fuzzy neural network active disturbance rejection control for two-wheeled self-balanced robot," *J Inf Proc Syst* **18**(4), 510–523 (2022).
[47] J. Zhao, J. Li and J. Zhou, "Research on two-round self-balancing robot SLAM based on the gmapping algorithm," *Sensors* **23**(5), 2489 (2023).
[48] R. Téllez and C. Angulo, "Webots Simulator 5.1. 7. "Developed and supported by cyberbotics ltd," *Artif Life* **13**(3), 313–318 (2007).

A. Appendix

A.1. Parameters in kinematic derivation

$$\begin{aligned}
&\varepsilon = 2\arctan\left(\frac{d \pm \sqrt{c^2 + d^2 - |l_{CB}^4|}}{c + |l_{CB}|^2}\right),\\
&c = 2L_3\left(x_{\mathbf{B}} - x_{\mathbf{C}}\right) = 2L_3\left(L_1 + L_2\cos\alpha_2 - L_2\cos\alpha_1\right),\\
&d = 2L_3\left(y_{\mathbf{B}} - y_{\mathbf{C}}\right) = 2L_3\left(L_2\sin\alpha_2 - L_2\sin\alpha_1\right).
\end{aligned} \tag{A1}$$

$$\begin{aligned}
e &= 2L_2 l_{(O_{A_0}O_{A_1})_x},\\
f &= 2L_2 l_{(O_{A_0}O_{A_1})_y},\\
g &= l^2_{(O_{A_0}O_{A_1})_x} + l^2_{(O_{A_0}O_{A_1})_y} + L_2^2 - L_3^2,\\
h &= 2\left(l_{(O_{A_0}O_{A_1})_x} - L_1\right)L_2,\\
i &= 2l_{(O_{A_0}O_{A_1})_y}L_2.
\end{aligned} \tag{A2}$$

A.2. Parameters in dynamic derivation

$$\begin{aligned}
M &= 2J_pJ_\omega + 2m_pL^2J_\omega + m_pR^2J_p + 2m_\omega R^2J_p + 2m_pm_\omega L^2R^2,\\
A_{23} &= -\left(m_p^2gR^2L^2\right)/M,\\
A_{43} &= m_pgL\left(2J_\omega + m_pR^2 + 2m_\omega R^2\right)/M,\\
B_{21} &= -R\left(m_pL^2 + m_pRL + J_p\right)/M,\\
B_{41} &= \left(2m_\omega R^2 + m_pR^2 + m_pRL + 2J_\omega\right)/M,\\
B_{61} &= -(RD)/\left(J_{\omega y}D^2 + m_\omega D^2R^2 + 2R^2J_y\right),\\
B_{22} &= B_{21},\quad B_{42} = B_{41},\quad B_{62} = -B_{61}.
\end{aligned} \tag{A3}$$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

Table II. *Dynamic parameters of simulation in Webots.*

Parameters	**Numerical values**
Chassis mass	7 kg
Moment of inertia of the chassis about its own pitch axis	9.498000e-02 $kg \cdot m^2$
Moment of inertia of the chassis around its own yaw axis	1.650000e-01 $kg \cdot m^2$
Relative height of chassis COM	0.05 m
Mass of TL	0.32477 kg
Mass of SL	0.56700 kg
Length of TL	0.15 m
Length of SL	0.288 m
Mass of the wheel	0.527790 kg
Moment of inertia of a wheel about its own pitch axis	1.121500e-03 $kg \cdot m^2$
Moment of inertia of a wheel about its own yaw axis	6.311500e-04 $kg \cdot m^2$

Table III. *PID parameters in webot simulation.*

Variables in PID control	**Gain coefficients**
Equivalent pitch angle θ_{eq}	30
Equivalent pitch angular velocity $\dot{\theta}_{eq}$	8
Difference of forward motion displacement for the robot $x_t - x$	0
Difference of forward motion velocity for balance control $\dot{x}_t - \dot{x}$	6
Difference of yaw angle $\delta_t - \delta$	0
Difference of yaw angle velocity $\dot{\delta}_t - \dot{\delta}$	0
Compensation for chassis tilt angle θ	10
Compensation for chassis tilt angle velocity $\dot{\theta}$	2
Difference of forward motion velocity for the leg module $\dot{x}_t - \dot{x}$	0.05 kg
Joint angle position of the leg module	8
Joint angular velocity of the leg module	0.5

Table IV. *Dynamic parameters of prototype experiments.*

Parameters	**Numerical values**
Chassis mass	6.836 kg
Moment of inertia of the chassis about its own pitch axis	9.498000e-02 $kg \cdot m^2$
Moment of inertia of the chassis around its own yaw axis	0.167716011 $kg \cdot m^2$
Relative height of chassis COM	-0.027 m
Mass of TL	0.185 kg
Mass of SL	0.279 kg
Length of TL	0.15 m
Length of SL	0.288 m
Mass of the wheel	0.432 kg
Moment of inertia of a wheel about its own pitch axis	0.001665983 $kg \cdot m^2$
Moment of inertia of a wheel about its own yaw axis	0.000913442 $kg \cdot m^2$

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press

A.3. Parameters in simulation and experiments

Table V. *PID parameters in prototype experiments.*

Variables in PID control	**Gain coefficients**
Equivalent pitch angle θ_{eq}	18.4928
Equivalent pitch angular velocity $\dot{\theta}_{eq}$	4.4873
Difference of forward motion displacement for the robot $x_t - x$	0.6
Difference of forward motion velocity for balance control $\dot{x}_t - \dot{x}$	3.3
Difference of yaw angle $\delta_t - \delta$	0
Difference of yaw angle velocity $\dot{\delta}_t - \dot{\delta}$	2
Compensation for chassis tilt angle θ	10
Compensation for chassis tilt angle velocity $\dot{\theta}$	2
Difference of forward motion velocity for the leg module $\dot{x}_t - \dot{x}$	0.06 kg
Joint angle position of the leg module	12
Joint angular velocity of the leg module	3

Cite this article: N. Mao, J. Chen, E. Spyrakos-Papastavridis and J. S. Dai (2024). "Dynamic modeling of wheeled biped robot and controller design for reducing chassis tilt angle", Robotica **42**, 2713–2741. https://doi.org/10.1017/S0263574724001061

https://doi.org/10.1017/S0263574724001061 Published online by Cambridge University Press