CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Mechanism design and dynamic switching modal control of the wheel-legged separation quadruped robot

Jiandong Cao[1], Jinzhu Zhang[1,2,3] (iD), Tao Wang[1,2,3], Jiahao Meng[1], Senlin Li[1] and Miao Li[1]

[1]College of Mechanical and Vehicle Engineering, Taiyuan University of Technology, Taiyuan, 030024, China, [2]Engineering Research Center of Advanced Metal Composites Forming Technology and Equipment, Ministry of Education, Taiyuan, 030024, China, and [3]National Key Laboratory of Metal Forming Technology and Heavy Equipment, Taiyuan, 030024, China
Corresponding author: Jinzhu Zhang; E-mail: zhangjinzhu@tyut.edu.cn

Abstract

Currently, most wheel-legged robots need to complete the switching of the wheel-and-leg modal in a stationary state, and the existing algorithms of statically switching the wheel-leg modal cannot meet the control requirements of multimodal switching dynamically for robots. In this paper, to achieve efficient switching of the wheel-and-leg modal for a quadruped robot, the novel transformable mechanism is designed. Then, a multimodal coordination operation control framework based on multiple algorithms is presented, incorporating the minimum foot force distribution method (algorithm No.1), the minimum joint torque distribution method (algorithm No.2), and the method of combining the single rigid body dynamic model with quadratic programming (algorithm No.3). In the process of switching wheel-leg modal dynamically, the existing algorithm No.3 is prone to produce the wrong optimal force due to the change of the whole-body rotational inertia. Therefore, an improved algorithm No.1 and algorithm No.2 are proposed, which do not consider the change in the body's inertia. The control effects of the three algorithms are compared and analyzed by simulation. The results show that algorithm No.3 can maintain a small error in attitude angle and speed tracking regardless of whether the robot is under multilegged support or omnidirectional walking compared to the other two algorithms. However, proposed algorithms No.1 and No.2 can more accurately track the target speed when the robot is walking with wheels raising and falling. Finally, a multi-algorithm combination control scheme formulated based on the above control effects has been demonstrated to be effective for the dynamic switching of the wheel-and-leg modal.

1. Introduction

Compared with wheeled robots, legged robots have better mobility in high-risk and unstructured environments such as mountain reconnaissance [1], industrial inspection and search and rescue [2], construction inspection and monitoring [3], and underground cave exploration [4], but their running speed on flat roads is limited, and the energy consumption is significant. After comprehensively considering the advantages of wheeled robots [5, 6] and legged robots, mobile robot researchers worldwide have designed a new type of wheel-legged robot that combines fast wheeled driving and flexible leg walking functions [7–9]. Currently, the main types of wheel-legged robots include wheel-legged series-connected robots and wheel-legged metamorphic robots.

The structure of the wheel-legged series-connected robots is simple, and its technology is relatively mature. The wheeled structure of this kind of robot is usually distributed at the knee joint or the end of the mechanical leg, which leads to a correlation between the wheel and leg structure. When the wheel is distributed at the end of the mechanical leg [10–12], the wheel also acts as the foot of the leg modal. During the switching process of the wheel modal to the leg modal, the wheel suddenly changes from high-speed rotation to static, which will significantly impact the robotic stability and lifetime.

Meanwhile, the wheel installed at the end of the leg is easily stuck or damaged when facing special terrains such as stones and ruins conditions. When the wheel is distributed at the knee joint of the mechanical leg [13], the distribution of the wheel and the leg is not at the same height. In the process of modal switching, the height of the center of mass (CoM) must change rapidly to adapt to the wheeled height. Therefore, the robot needs to switch the wheel-leg modal after maintaining a stationary state, which will lead to low operational efficiency.

The wheel-legged metamorphic robot is a novel robot type proposed in recent years, which improves environmental adaptability by switching the wheel-and-leg motion modal. For example, Chen *et al.* [14] developed a deformable robot with four wheels and four legs based on a semi-circular wheel-legged structure, which realizes wheel-leg switching through the conversion between the half-wheel and the whole wheel; Kim *et al.* [15] designed a new wheel-leg hybrid robot that can transform the wheel into a legged wheel with three legs; Liu *et al.* [16] constructed a wheel-leg deformation robot by combining two single closed chain metamorphic mechanisms into a wheel-leg movement module, which can not only walk in multilegged modal but also switch the leg modal to the wheel modal when the crank and frame are collinear; Zheng *et al.* [17] presented a new adaptive wheel-and-leg transformable robot, which can rotate the gear in the center of the wheel to cause the legs to close or remain closed in a circular shape. Due to the integration of the wheel structure and leg structure, the metamorphic robot must switch the wheel-and-leg modal in a stationary state. Meanwhile, the complex wheel-leg structure will also bring many variables, increasing the control difficulty of the stable robotic operation and reducing the robot's operating efficiency.

Aiming at the problems of modal switching of wheel-legged robots caused by the correlation between wheel and leg structures, our team proposed a wheel-legged separation quadruped robot. From the mechanism perspective, the robotic wheel structure and leg structure are independent of each other and parallel connected at the body. Because the height of the wheel and foot of the robot is adjustable, it is easy to realize the dynamic and smooth switching of the wheel-and-leg motion modal. The novel robot is complementary to the functions of the existing wheel-legged robot.

In the switching process of the wheel-leg modal, the descent and recovery of the independent wheel module will affect the gravity position and inertia parameters of the wheel-legged robot in real time, which can easily lead to significant ground impulse and foot slip. Therefore, it is necessary to consider the influence of the multiple control algorithms on the ability of dynamic balance, the ability to track expected speed, and the ability of antiexternal interference force when controlling the walking of the robot.

In the field of wheeled-legged locomotion, international scholars have done much helpful exploration and gradually generated some multimodal coordination control methods for wheel-legged robots. In the early stages, a number of authors have proposed multi-algorithm control frameworks for wheeled-legged locomotion that can adapt to terrain variations and attitude disturbances by adjusting the posture of the body. For example, He *et al.* [10] proposed a terrain-adaptive control strategy for the "TAWL" wheel-legged robot consisting of contact force control, roll-and-pitch control, wheel speed allocation, steering strategy, and combined the control strategy with kinematics-based to achieve stable control of the robot in unstructured terrain. Wang *et al.* applied linear output regulation and model-based linear quadratic regulator to maintain the robot's standing state on the ground. Besides, a nonlinear controller based on the interconnection and damping assignment - passivity-based control (IDA-PBC) method is exploited to control the wheel-legged robot in more general scenarios [18]. In particular, the "Max" wheel-legged robot integrates the nonlinear model predictive control (NLMPC) algorithm, quadratic programming (QP) optimization, and compliance control algorithm to achieve dual wheel rotation, balance disturbance suppression, and wheel-leg switching control [13]. Due to the lack of consideration for the simultaneous use of wheels and legs in these methods, the robot's obstacle-crossing ability will deteriorate.

Recently, the motion planner has been presented to solve the whole-body planning problem of combining driving and stepping motions. Marko *et al*. [19] proposed an online trajectory optimization (TO)

framework for wheel-legged robots, which can operate in model predictive control (MPC) by decomposing the problem into wheel and base online TOs. The wheel online TO takes the rolling constraints of the wheels into account, while the base online TO accounts for the robot's balance during locomotion using the idea of the zero-moment point (ZMP). The presented method enables the robot to achieve wheel-and-leg hybrid motion. Subsequently, Marko *et al.* [20] also proposed a whole-body MPC allowing for online gait sequence adaptation. The "ANYmal" wheel-legged robot can dynamically select the optimal hybrid gait to fuse wheel rolling and leg stepping without the need for any cameras or LiDAR. Meanwhile, the robot can quickly transition between rolling and gait based on the perception of the terrain by the wheels. In particular, Vivian S *et al.* [21] presented an offline TO framework consisting of a terrain-aware planner and a terrain-aware controller that is an extension of a hierarchical whole-body controller. The framework allows the "ANYmal" wheel-legged robot to overcome challenging obstacles, such as a 30° slope, 0.5 m half-pipe, and stairs. All of the above approaches prove that the multi-algorithm combination control can better achieve multimodal coordinated control of wheel-legged robots, but these approaches are aimed at robots with wheel-legged series-connected structures.

Fortunately, we can integrate the wheel-leg structure and adopt multiple methods to solve the whole-body force distribution problem in multimodal transformation by drawing on the above ideas. The force distribution based on the virtual model control (VMC) is an algorithm with dynamic balance [22] and anti-interference characteristics. Many scholars have applied virtual force and torque to the CoM and adopted analytical calculation methods such as the equally proportional distribution [23, 24], the virtual force vector distribution [25], and the force distribution with increased force constraints [26] to achieve smooth and stable walking of the quadruped robot on slopes and rough terrain. Moreover, these methods can enable the quadruped robot to have a strong ability to resist lateral disturbances [27, 28]. However, compared with all the above-mentioned algorithms, the force distribution algorithm based on the single rigid body dynamics model can calculate the optimal force to realize the precise control of trajectory tracking of the CoM by introducing the whole-body balance constraint into the QP [29–32].

Due to the existing VMC force distribution algorithms based on analytical calculation not considering some factors such as ground friction, sudden force change, and foot-end output force constraints, the foot end will slip seriously on complex terrain. The force distribution algorithm based on a single rigid body dynamics model (algorithm No.3) can consider the above constraints, which are more suitable to be adapted in the robot's omnidirectional walking and attitude transformation control. Considering that the modal switching dynamically needs to ensure that the running speed of the wheels and the legs is consistent, the robot must have good speed tracking ability when walking with the wheels out of the body. However, algorithm No.3 is prone to produce the wrong optimal force due to the influence of the change of the body's inertia. In contrast, the force distribution algorithm based on a virtual model does not need to consider the problem of inertia change. Therefore, this paper proposes foot-end force minimization (algorithm No.1) and joint torque minimization (algorithm No.2) based on full rank decomposition and the least square method. Both algorithms belong to the VMC force distribution algorithm. In order to observe the respective advantages of the three algorithms in the walking and modal switching of the wheel-legged separation quadruped robot, they are compared and analyzed in the simulation. According to the performance of three algorithms, a complete walking control scheme of the wheel-legged separation quadruped robot will be formulated and verified by dynamically switching the wheel-leg modal.

The manuscript is organized in five sections, including this introductory section. Section 2 provides an overview of the mechanism design of the wheel-legged separation quadruped robot. Section 3 touches upon the multimodal coordination operation control framework and the specific solved process of three algorithms. Section 4 analyzes the simulation of static standing with multilegged support, omnidirectional walking, and walking with wheels raising and falling under three control algorithms. Meanwhile, a combined control scheme suitable for walking and modal switching of the wheel-legged robot is proposed according to the control advantages of the three algorithms, and this scheme has been proven to be significantly effective for the dynamic switching of the wheel-leg modal. Section 5 provides a summary of the entire work and a statement of future plans.
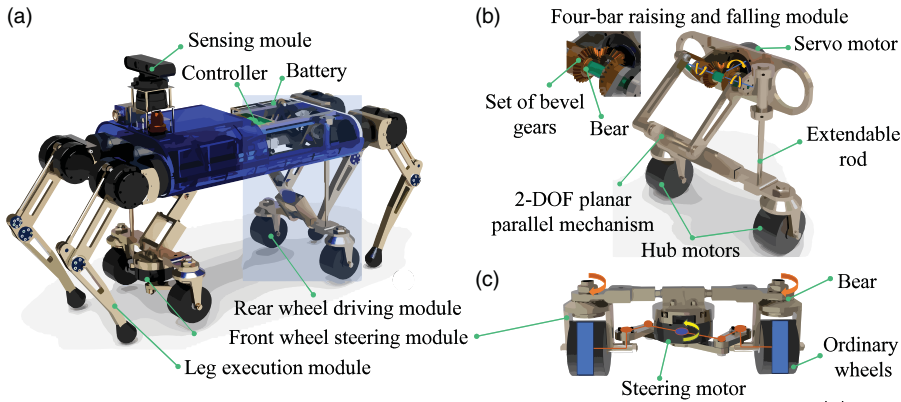
Figure 1. *Machine structure of the wheel-legged separation quadruped robot. (a) Structure components of the robot. (b) Four-bar raising and falling module. (c) Front-wheel steering module.*

2. The mechanism design of the wheel-legged separation quadruped robot

The main structure of the wheel-legged separation quadruped robot consists of the leg execution module, four-link raising and falling module, front wheel steering module, rear wheel driving module, sensing module, and controller, as shown in Fig. 1(a). The four-bar raising and lowering module in Fig. 1(b) can lower or raise the front wheel steering and the rear driving module to the specified height. Its mechanical principle is to use a servo motor to drive the set of bevel gears forward and reverse to make the 2-DOF planner parallel mechanism move vertically. Moreover, we use the extendable rod to prevent the incline of the front and rear wheel modules. The front wheel steering module in Fig. 1(c) can use the steering motor to drive the linkage mechanism to complete the specified turning angle of ordinary wheels. The sensing module mainly comprises depth cameras and laser radar, which are used to provide environmental information for the robot. All types of robot equipment are powered by a 48V Li-ion battery installed on both sides. The controller uses a combination of Upboard and two STM32F446 to control all motors.

3. Multimodal coordination control algorithm of the wheel-legged separation quadruped robot

In this section, we highlight the overview of the control architecture and critical parts of the proposed framework. The stable walking control framework of the wheel-legged separation quadruped robot is present in Fig. 2. The user inputs the desired parameters ${}^{B}V_d \in R^3$, ${}^{B}V_d \in R^3$ of the CoM, which will be applied to generate the reference trajectory of the CoM. Meanwhile, M_s is used as a switch for dynamically switching the wheel-leg modal. Assuming that the attitude of the quadruped robot is relatively stable during walking, the roll angle ϕ and pitch angle θ will approach 0. Therefore, the actual position ${}^{O}P_{com}$ and speed ${}^{O}V_{com}$ of the CoM generated by the state estimator [33–35] can be converted to frame {B} as follows.

$$\begin{cases} {}^{B}P_{com} = R_z(\psi)^{T} \cdot {}^{O}P_{com} \\ {}^{B}V_{com} = R_z(\psi)^{T} \cdot {}^{O}V_{com} \end{cases} \tag{1}$$

where $R_z(\psi) \in R^{3\times3}$ is the rotational matrix of the yaw angle ψ.

For the robotic support phase control, the expected force ${}^{B}F_{com}$ and torque ${}^{B}T_{com}$ are calculated by combining the state estimation value and the feedback three-axis angle, which is used to maintain the body's stability. Then, the joint torque τ_z of the support legs can be obtained by using the minimum foot-end force distribution method, the minimum torque distribution method, and the minimum force distribution method based on single rigid body dynamics. For the swing phase control, the results of
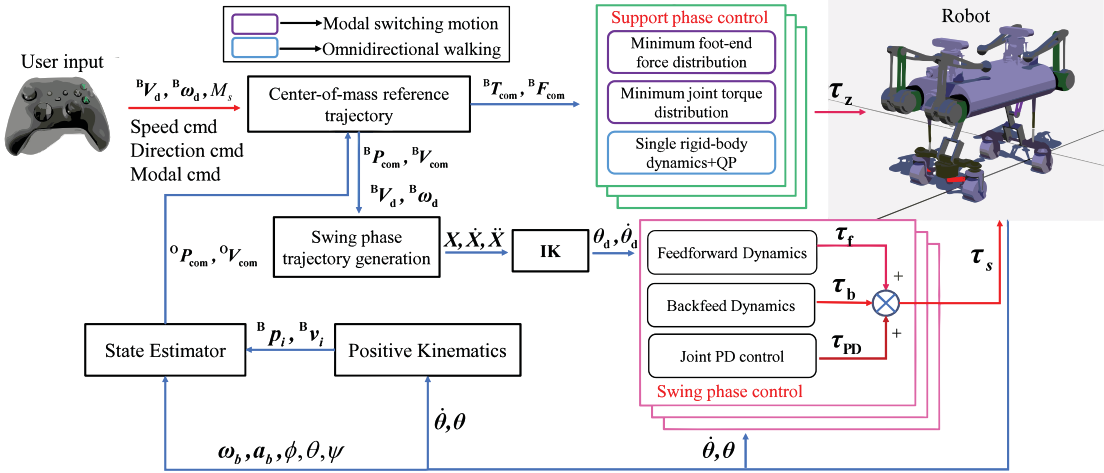
Figure 2. *The control framework of stable walking in the leg modal. $X, \dot{X}, \ddot{X}$ represent the displacement, velocity, and acceleration of the planned trajectory, respectively. $\dot{\theta}_d$ represents the expected acceleration of the joint motor, which is solved by inverse kinematics (IK). τ_s, τ_z represent the output joint torque of the swing legs and the support legs, respectively.*

combining the expected angular velocity $^B\boldsymbol{\omega}_d$ and linear velocity $^B\boldsymbol{V}_d$ with the expected body's position $^B\boldsymbol{P}_{com}$ and speed $^B\boldsymbol{V}_{com}$ are used to plan the swing trajectory. In order to accurately track the swing trajectory, dynamics of the single leg and joint PD control are fused to obtain the joint torque of the swing legs.

3.1. Minimum foot-end force distribution method based on the center-of-mass virtual model

The minimum foot-end force distribution method is used to distribute the expected force and torque of the CoM to the foot-end force vector. Due to the contact surface between the support foot end and the ground being small, it can be assumed that there is only force and no torque. So the distributed foot-end force can be directly used as the driving force to control the supporting legs of the quadruped robot. Meanwhile, to simplify the model for mechanical analysis, assuming that the CoM of the robot is concentrated in the geometric center of the body. Through the overall static analysis of the quadruped robot, the force and torque vectors of the CoM are defined as follows:

$$\boldsymbol{F}_{com} = \begin{bmatrix} ^B F_{comX} & ^B F_{comY} & ^B F_{comZ} \end{bmatrix}^T \quad \boldsymbol{M}_{com} = \begin{bmatrix} ^B T_{comX} & ^B T_{comY} & ^B T_{comZ} \end{bmatrix}^T \tag{2}$$

The foot-end position vector and the foot-end force vector of every leg in frame {B} are set as $^B\boldsymbol{r}_i = [x_i \quad y_i \quad z_i]^T$ and $^B\boldsymbol{F}_i = [^B F_{xi} \quad ^B F_{yi} \quad ^B F_{zi}]^T, i \in [1, 2, 3, 4]$, respectively. According to the schematic diagram of the single leg mechanism in Fig. 3(b), $^B\boldsymbol{r}_i$ can be calculated as below.

$$^B_i\boldsymbol{r} = \begin{bmatrix} \xi \cdot (l_1 \cos\theta_1 - l_2 \cos\theta_2) + \xi \cdot L/2 \\ \xi \cdot ((l_1 \sin\theta_1 + l_2 \sin\theta_2) \sin\theta_3) + \delta \cdot W/2 \\ -(l_1 \sin\theta_1 + l_2 \sin\theta_2) \cos\theta_3 \end{bmatrix} \tag{3}$$

where L and W are the length and width of the robotic body. ξ and δ are symbolic variables, which are defined as follows.

$$\xi = \begin{cases} 1, & i = 1 \text{ or } 2 \\ -1, & i = 3 \text{ or } 4 \end{cases} \qquad \delta = \begin{cases} 1, & i = 1 \text{ or } 4 \\ -1, & i = 2 \text{ or } 3 \end{cases} \tag{4}$$
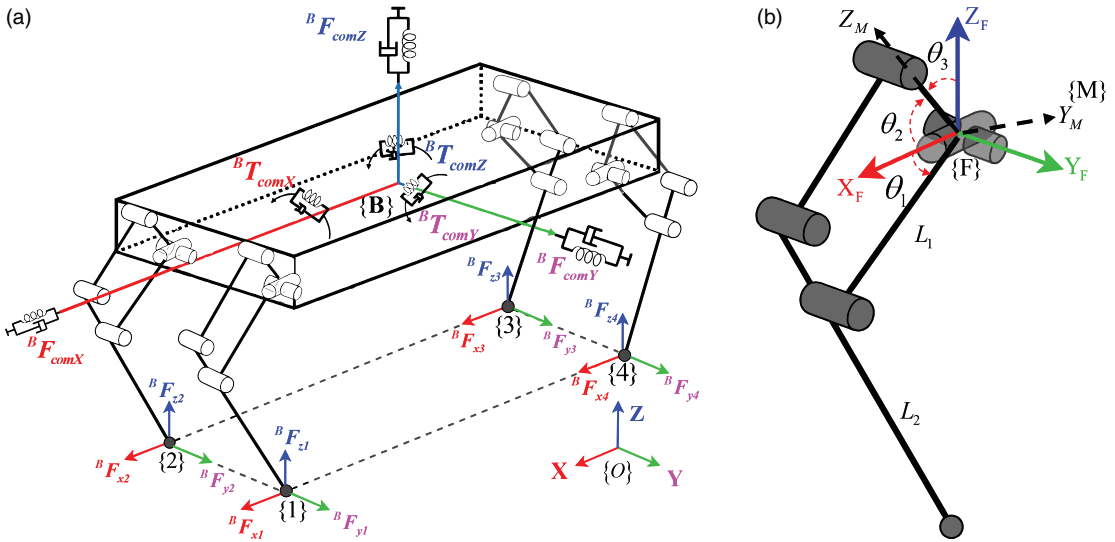
Figure 3. *Illustration of whole-body coordinate systems and the single leg coordinate systems. (a) The "spring-damping" system of the CoM. (b) Single leg mechanism diagram. {F} is the fixed coordinate system of the hip joint, while {M} is its dynamic coordinate system.*

According to the force and torque balance relationship of the quadruped robot, the center-of-mass balance equation of the robot can be obtained as follows.

$$\begin{cases} \sum_{i=1}^{n} {}^{B}\boldsymbol{F}_i = \boldsymbol{F}_{\text{com}} + \boldsymbol{G} \\ \sum_{i=1}^{n} \left({}^{B}\boldsymbol{r}_i \times {}^{B}\boldsymbol{F}_i \right) = \boldsymbol{M}_{\text{com}} \end{cases} \tag{5}$$

where n is the number of supporting legs. $\boldsymbol{G}$ is the gravitational compensation.

The position and attitude angle of the robotic body are adjusted through a virtual "spring-damping" system shown in Fig. 3(a). The system can be derived as below.

$$\begin{bmatrix} \boldsymbol{F}_{\text{com}} \\ \boldsymbol{M}_{\text{com}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_{\text{c,p}} \left(\boldsymbol{P}_{\text{com}}^{\text{d}} - {}^{B}\boldsymbol{P}_{\text{com}} \right) + \boldsymbol{D}_{\text{c,p}} \left({}^{B}\boldsymbol{V}_{\text{com}}^{\text{d}} - {}^{B}\boldsymbol{V}_{\text{com}} \right) \\ \boldsymbol{K}_{c,\omega} \left(\boldsymbol{\theta}_{\text{com}}^{\text{d}} - \boldsymbol{\theta}_{\text{com}} \right) + \boldsymbol{D}_{c,\omega} \left({}^{B}\boldsymbol{\omega}_{\text{com}}^{\text{d}} - {}^{B}\boldsymbol{\omega}_{\text{com}} \right) \end{bmatrix} \tag{6}$$

where ${}^{B}\omega_{\text{com}} = \begin{bmatrix} {}^{B}\omega_{\text{comX}} & {}^{B}\omega_{\text{comY}} & {}^{B}\omega_{\text{comZ}} \end{bmatrix}^{T}$ and $\theta_{\text{com}} = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^{T}$ represents the attitude and angular speed vector of the CoM measured by IMU. $\boldsymbol{K}_{\text{c,p}} = Diag(K_x, K_y, K_z)$, $\boldsymbol{K}_{\text{c,d}} = Diag(D_x, D_y, D_z)$, $\boldsymbol{K}_{c,\omega} = Diag(K_\phi, K_\theta, K_\psi)$, $\boldsymbol{D}_{c,\omega} = Diag(D_\phi, D_\theta, D_\psi)$ are adjustable factors.

For the four-legged support and the three-legged support, Eq. (5) can be rewritten into a specific matrix form as below.

$$\begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{I}_3 & \boldsymbol{I}_3 & \boldsymbol{I}_3 \\ {}^{B}\boldsymbol{r}_{1\times} & {}^{B}\boldsymbol{r}_{2\times} & {}^{B}\boldsymbol{r}_{3\times} & {}^{B}\boldsymbol{r}_{4\times} \end{bmatrix} \cdot \begin{bmatrix} {}^{B}\boldsymbol{F}_1 \\ {}^{B}\boldsymbol{F}_2 \\ {}^{B}\boldsymbol{F}_3 \\ {}^{B}\boldsymbol{F}_4 \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_{\text{com}} + \boldsymbol{G} \\ \boldsymbol{M}_{\text{com}} \end{bmatrix} \tag{7}$$

$$\begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{I}_3 & \boldsymbol{I}_3 \\ {}^{B}\boldsymbol{r}_{2\times} & {}^{B}\boldsymbol{r}_{3\times} & {}^{B}\boldsymbol{r}_{4\times} \end{bmatrix} \cdot \begin{bmatrix} {}^{B}\boldsymbol{F}_2 \\ {}^{B}\boldsymbol{F}_3 \\ {}^{B}\boldsymbol{F}_4 \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_{\text{com}} + \boldsymbol{G} \\ \boldsymbol{M}_{\text{com}} \end{bmatrix} \tag{8}$$

where ${}^{B}\boldsymbol{r}_{i\times}$ is the antisymmetric matrix of ${}^{B}\boldsymbol{r}_i = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^{\text{T}}$.

Algorithm 1: Solving the minimum foot-end force through FRDWLS

Initialize algorithm variables: $k, \boldsymbol{B}$
If $k = 2$ then
   Fill $\boldsymbol{A}$ by Eq. (11)
   Transform the simplest form of rows $rref(\boldsymbol{A})$ by Eq. (12)
   Select the column-full rank matrix $\boldsymbol{D} = \boldsymbol{A}(1:6, 1:5)$
   Select the row-full rank matrix $\boldsymbol{C} = rref(\boldsymbol{A})(1:5, 1:6)$
   Solving the foot-end force vector by least squares
   $\boldsymbol{D}^+ = (\boldsymbol{D}^{\mathrm{T}} \cdot \boldsymbol{D})^{-1} \boldsymbol{D}^{\mathrm{T}}$
   $\boldsymbol{C}^+ = \boldsymbol{C}^{\mathrm{T}} (\boldsymbol{C} \cdot \boldsymbol{C}^{\mathrm{T}})^{-1}$
   $\boldsymbol{A}^+ = \boldsymbol{C}^+ \cdot \boldsymbol{D}^+$
   $\boldsymbol{F} = \boldsymbol{A}^+ \cdot \boldsymbol{B}$
else if $k > 2$
   Solving the foot-end force vector by Eq. (10)
   $\boldsymbol{F} = \boldsymbol{A}^{\mathrm{T}} [\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}]^{-1} \cdot \boldsymbol{B}$
end if

Subsequently, Eqs. (7)–(8) can be summarized as follows:

$$\boldsymbol{A} \cdot \boldsymbol{F} = \boldsymbol{B} \tag{9}$$

Since $\boldsymbol{A}$ belongs to a full rank matrix, Eq. (8) can be solved by Eq. (10) directly [36, 37] when the robot is in four-legged or three-legged support.

$$\boldsymbol{F} = \boldsymbol{A}^{\mathrm{T}} \left[\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}\right]^{-1} \cdot \boldsymbol{B} \tag{10}$$

For the two-legged support of the quadruped robot, assuming that legs 2,4 are used for supporting the body. Equation (5) can be rewritten as follows.

$$\begin{bmatrix} \boldsymbol{I}_3 & \boldsymbol{I}_3 \\ {}^{\mathrm{B}}\boldsymbol{r}_{2\times} & {}^{\mathrm{B}}\boldsymbol{r}_{4\times} \end{bmatrix} \cdot \begin{bmatrix} {}^{\mathrm{B}}\boldsymbol{F}_2 \\ {}^{\mathrm{B}}\boldsymbol{F}_4 \end{bmatrix} = \begin{bmatrix} \boldsymbol{F}_{\mathrm{com}} + \boldsymbol{G} \\ \boldsymbol{M}_{\mathrm{com}} \end{bmatrix} \tag{11}$$

By comparing Eqs. (9) and (11), the rank of $\boldsymbol{A}$ can be calculated as rank$(\boldsymbol{A}) = 5$. The Eq. (10) cannot be applied in Eq. (9) directly. The minimum foot-end force vector $\boldsymbol{F}$ is solved by the method of combing full rank decomposition with least squares (FRDWLS), as shown in Algorithm 1. When the number of supporting legs k meets the requirements, filling $\boldsymbol{A}$ and transforming $\boldsymbol{A}$ into the simplest form of rows by using elementary row transformations. Then, selecting the maximum independent group of column vector of $\boldsymbol{A}$ to form the column-full rank matrix according to the ladder position elements in red of rref($\boldsymbol{A}$), shown in the purple area of Eq. (12). Taking the first rank$(\boldsymbol{A})$ rows of rref($\boldsymbol{A}$) as a row-full rank matrix, shown in the green area of Eq. (12). Finally, the foot-end force vector can be solve by least squares.

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & y_2 & z_2 & 0 & y_4 & z_4 \\ -y_2 & 0 & -x_2 & -y_4 & 0 & -x_4 \\ -z_2 & x_2 & 0 & -z_4 & x_4 & 0 \end{bmatrix} \rightarrow rref\,(\boldsymbol{A}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \frac{x_2 - x_4}{z_2 - z_4} \\ 0 & 1 & 0 & 0 & 0 & \frac{y_2 - y_4}{z_2 - z_4} \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & -\frac{x_2 - x_4}{z_2 - z_4} \\ 0 & 0 & 0 & 0 & 1 & -\frac{y_2 - y_4}{z_2 - z_4} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{12}$$

According to the action and reaction forces, the force exerted by the foot end on the ground is opposite to $\boldsymbol{F}$ in the direction. Therefore, the joint torque of support legs can be calculated as follows.

$$\boldsymbol{\tau}_z = \boldsymbol{J} \cdot (-\boldsymbol{F}) \tag{13}$$

3.2. Minimum joint torque distribution method based on the center-of-mass virtual model

Compared with the minimum foot-end force distribution method, the minimum joint torque distribution method can directly calculate the minimum joint torque of support legs. Therefore, Eq. (9) needs to be rewritten as below.

$$\boldsymbol{\tau} = \boldsymbol{J} \cdot (-\boldsymbol{F}) \tag{14}$$

$$\boldsymbol{F} = -\boldsymbol{J}^{-1} \cdot \boldsymbol{\tau} \tag{15}$$

$$\boldsymbol{A} \cdot \boldsymbol{J}^{-1} \cdot \boldsymbol{\tau} = -\boldsymbol{B} \tag{16}$$

where $\boldsymbol{J}^{-1}$ represents the inverse Jacobi matrix in the case of different supporting legs.

For the four-legged and three-legged support of the quadruped robot, the minimum joint torque vector $\boldsymbol{\tau}_{\min}$ of support legs can be obtained by Eqs. (17)–(19).

$$\boldsymbol{A}_{\mathrm{J}} = \boldsymbol{A} \cdot \boldsymbol{J}^{-1} \tag{17}$$

$$\boldsymbol{A}_{\mathrm{J}} \cdot \boldsymbol{\tau} = -\boldsymbol{B} \tag{18}$$

$$\boldsymbol{\tau}_{\min} = \boldsymbol{A}_{\mathrm{J}}^{\mathrm{T}} \left[\boldsymbol{A}_{\mathrm{J}} \boldsymbol{A}_{\mathrm{J}}^{\mathrm{T}} \right]^{-1} \cdot (-\boldsymbol{B}) \tag{19}$$

However, for the two-legged support, a full rank decomposition of the matrix is necessary, and its decomposition method is the same as in Section 3.1. Assuming that the row full rank matrix and the column full rank matrix are $\boldsymbol{D}_{\mathrm{J}} \in \mathrm{R}^{6 \times 5}$ and $\boldsymbol{C}_{\mathrm{J}} \in \mathrm{R}^{5 \times 6}$. The joint torque of the support legs can be solved by Algorithm 2. We describe the specific steps of the solution as follows: when the number of supporting legs meets the requirements, filling $\boldsymbol{A}_J$ and transforming $\boldsymbol{A}_J$ into the simplest form of rows by using elementary row transformations. Selecting the maximum independent group of column vector of $\boldsymbol{A}_J$ to form the column-full rank matrix according to the ladder position elements of rref($\boldsymbol{A}_J$). Taking the first rank($\boldsymbol{A}_J$) rows of rref($\boldsymbol{A}_J$) as a row-full rank matrix. Finally, the joint torque vector can be solved by least squares.

3.3. The frictional constraint of the foot-end force distribution and torque distribution

In order to avoid foot slippage, each force vector $^{\mathrm{B}}\boldsymbol{F}_i$ in Section 3.1 must be inside the friction cone. The ratio of the tangential component and the normal component of the ground reaction force is defined as η. Thus, $\eta \leq \mu$ respects the friction constraint [26]. The relationship of $^{\mathrm{B}}\boldsymbol{F}_i$ and the centroidal force vector $^B\boldsymbol{F}_{\mathrm{com}}$ can be written as follows.

$$\eta = \frac{\sqrt{^{\mathrm{B}}F_{\mathrm{comX}}^2 + {}^{\mathrm{B}}F_{\mathrm{comY}}^2}}{^{\mathrm{B}}F_{\mathrm{comZ}}} = \frac{\sqrt{^{\mathrm{B}}F_{ix}^2 + {}^{\mathrm{B}}F_{iy}^2}}{^{\mathrm{B}}F_{iz}} \tag{20}$$

In order to linearize Eq. (20), the cone constraint is modified to an approximate quadrangular pyramid. The ratio $\eta = [n_x, n_y]$ can be simplified as below.

Algorithm 2: Solving the minimum joint torque through FRDWLS

Initialize algorithm variables: k, B
If $k = 2$ then
   Fill A_J by Eq. (17)
   Transform the simplest form of rows rref(A_J)
   Select the column-full rank matrix $D_J = A_J(1:6, 1:5)$
   Select the row-full rank matrix $C_J = \text{rref}(A_J)(1:5, 1:6)$
   Solving the joint torque vector by least squares
   $D_J^+ = (D_J^T D_J)^{-1} D_J^T$
   $C_J^+ = C_J^T (C_J C_J^T)^{-1}$
   $A_J^+ = C_J^+ \cdot D_J^+$
   $\boldsymbol{\tau}_{\min} = A_J^+ \cdot (-B)$
else if $k > 2$
   Solving the joint torque vector by Eq. (19)
   $\boldsymbol{\tau}_{\min} = A_J^T [A_J A_J^T]^{-1} \cdot (-B)$
end if

$$\begin{cases} {}^B F_{ix} = \dfrac{{}^B F_{comX}}{{}^B F_{comZ}} {}^B F_{iz} = \dfrac{\eta_x}{\sqrt{2}} F_{iz} \\[3mm] {}^B F_{iy} = \dfrac{{}^B F_{comX}}{{}^B F_{comZ}} {}^B F_{iz} = \dfrac{\eta_y}{\sqrt{2}} F_{iz} \end{cases} \tag{21}$$

Once $\boldsymbol{F}_{com} = \begin{bmatrix} {}^B F_{comX} & {}^B F_{comY} & {}^B F_{comZ} \end{bmatrix}^T$ is generated by the virtual system of the CoM. The ratio η can then be calculated. Therefore, the forward tractive force ${}^B F_{ix}$ and the lateral tractive force ${}^B F_{iy}$ of quadruped systems can be controlled proportionally according to the normal forces ${}^B F_{iz}$ of the support foot.

However, to prevent foot slippage in extreme circumstances, the frictional constraint needs to be reinforced by an antislip factor $\xi (\xi \in (0, 1])$ as below.

$$\boldsymbol{\eta} = \begin{bmatrix} \eta_x \\ \eta_y \end{bmatrix} = \begin{bmatrix} \min \left\{ \dfrac{\sqrt{2} {}^B F_{comX}}{{}^B F_{comZ}}, \xi \mu \right\} \\[5mm] \min \left\{ \dfrac{\sqrt{2} {}^B F_{comY}}{{}^B F_{comZ}}, \xi \mu \right\} \end{bmatrix} \tag{22}$$

Since the minimum joint torque $\boldsymbol{\tau}_i$ in Section 3.2 cannot be directly constrained by the friction cone, the forward and lateral tractive forces must be recalculated as below.

$$ {}^B \boldsymbol{F}'_i = \begin{bmatrix} {}^B F'_{ix} & {}^B F'_{iy} & {}^B F'_{iz} \end{bmatrix}^T = \boldsymbol{J}_i^{-1} \boldsymbol{\tau}_i \tag{23}$$

According to Eqs. (21)~(23), the joint torque $\boldsymbol{\tau}'_i$ of the support legs can be rewritten as follows.

$$\boldsymbol{\tau}'_i = \boldsymbol{J}_i \cdot \begin{bmatrix} \dfrac{\eta_x}{\sqrt{2}} & \dfrac{\eta_y}{\sqrt{2}} & 1 \end{bmatrix}^T {}^B F'_{iz} \tag{24}$$

3.4. Minimum force distribution method based on single-rigid body dynamic model

Although the quadruped robot is a multirigid body system, the legs are often designed to be lightweight, and the servo motors are arranged near the body. For this reason, we can reasonably simplify the control

Table I. *Basic parameters of wheel-leg separation quadruped robot.*

Parameters	Symbol	Value	Unit
Mass	m	22	Kg
Inertia of body	I_{xx}, I_{yy}, I_{zz}	0.34, 2.16, 2.84	$Kg \cdot m^2$
Length of body	L	0.778	m
Width of body	W	0.224	m
Length of thighs	L_1	0.2	m
Length of calf	L_2	0.238	m
Peak torque of the motor	τ_{max}	35	$N \cdot m$
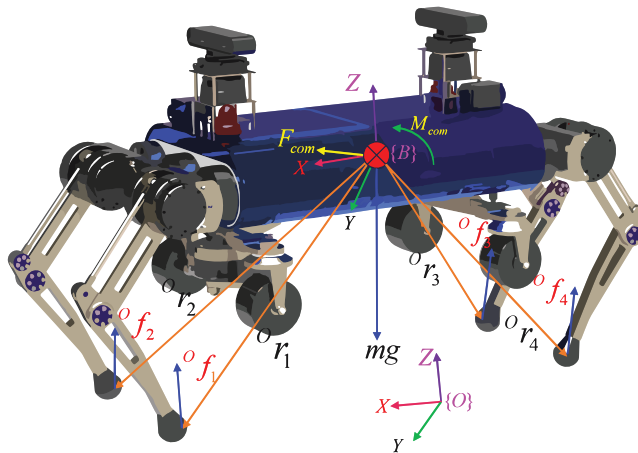
Figure 4. *Single rigid body dynamics model.*

model to a single rigid body dynamics model, as shown in Fig. 4. Set $^O\boldsymbol{f}_i = [f_{i,x}, f_{i,y}, f_{i,z}]^T$ and $^O\boldsymbol{r}_i = \boldsymbol{R} \cdot {}^B\boldsymbol{r}_i$ as the foot-end force and the foot-end position vector in frame {O}, respectively. Using Newton's second law and Euler equation, the balance equation of single rigid body of the quadruped robot can be established as follows.

$$\begin{bmatrix} \boldsymbol{I}_3 & \cdots & \boldsymbol{I}_3 \\ {}^O\boldsymbol{r}_1 \times & \cdots & {}^O\boldsymbol{r}_4 \times \end{bmatrix} f = \begin{bmatrix} m \left({}^O\ddot{\boldsymbol{p}}_{com} - \boldsymbol{a}_g \right) \\ \boldsymbol{R} \left(\psi_z \right) \boldsymbol{I}_G \boldsymbol{R} \left(\psi_z \right)^T \cdot {}^O\boldsymbol{\alpha}_{com} \end{bmatrix} \tag{25}$$

where $^O\boldsymbol{r}_i \times$ is the antisymmetric matrix of $^O\boldsymbol{r}_i$. $\boldsymbol{I}_G$ is the rotational inertia of the body. $\boldsymbol{a}_g = [0, 0, -9.8]^T$ is the gravity vector. $^O\ddot{\boldsymbol{p}}_{com}, {}^O\boldsymbol{\alpha}_{com}$ are the linear acceleration and angular acceleration of the CoM, respectively. Several parameters listed in Table I are used to calculate the foot-end position vector $^O\boldsymbol{r}_i$.

The desired linear acceleration $^O\ddot{\boldsymbol{p}}_{com}$ and angular acceleration $^O\boldsymbol{\alpha}_{com}$ of the CoM of the quadruped robot can be translated by Eqs. (26)–(27).

$$\begin{bmatrix} {}^O\ddot{\boldsymbol{P}}_{com} \\ {}^O\boldsymbol{\alpha}_{b,d} \end{bmatrix} = \begin{bmatrix} \boldsymbol{K}_{cp} \left({}^O\boldsymbol{P}_{c,d} - {}^O\boldsymbol{P}_{com} \right) + \boldsymbol{K}_{cd} \left({}^O\boldsymbol{V}_{c,d} - {}^O\boldsymbol{V}_{com} \right) \\ \boldsymbol{K}_{bp} e \left({}^O_B\boldsymbol{R}_d {}^O_B\boldsymbol{R}^T \right) + \boldsymbol{K}_{bd} \left({}^O\boldsymbol{\omega}_{c,d} - {}^O\boldsymbol{\omega}_{com} \right) \end{bmatrix} \tag{26}$$

$$\begin{cases} {}^O\boldsymbol{V}_{c,d} = \boldsymbol{R}_z(\boldsymbol{\psi}) \cdot {}^B\boldsymbol{V}_d \\ {}^O\boldsymbol{P}^k_{c,d} = {}^O\boldsymbol{P}^{k-1}_{c,d} + {}^O\boldsymbol{V}^{k-1}_{c,d} \cdot \Delta t \\ {}^O\boldsymbol{\omega}_{c,d} = {}^O_B\boldsymbol{R} \cdot {}^B\boldsymbol{\omega}_d \end{cases} \tag{27}$$

where $^O\boldsymbol{P}_{\text{c,d}}, ^O\boldsymbol{V}_{\text{c,d}}, ^O\boldsymbol{\omega}_{\text{c,d}} \in \mathbb{R}^3$ is the desired position, desired velocity, and actual angular velocity of the CoM in frame {O}. $^B\boldsymbol{V}_d$ and $^B\boldsymbol{\omega}_d \in \mathbb{R}^3$ are input by the user in Section 3. $^O_B\boldsymbol{R}_d \in \mathbb{R}^{3\times3}$ is the rotational matrix of the expected attitude angle of the CoM. $\boldsymbol{K}_{\text{cp}}, \boldsymbol{K}_{\text{cd}}, \boldsymbol{K}_{\text{bp}}, \boldsymbol{K}_{\text{bd}} \in \mathbb{R}^{3\times3}$ are modifiable factors. $\boldsymbol{e}(*)$ represents the mapping of the rotation matrix to the equivalent rotation vector, and the specific mapping equation is derived as below.

$$\varphi = acos\left(\frac{tr\left(^O_B\boldsymbol{R}_d\,^O_B\boldsymbol{R}^T\right) - 1}{2}\right) \tag{28}$$

$$\boldsymbol{eR} = \,^O_B\boldsymbol{R}_d\,^O_B\boldsymbol{R}^T \tag{29}$$

$$\boldsymbol{e}\left(^O_B\boldsymbol{R}_d\,^O_B\boldsymbol{R}^T\right) = \varphi \cdot \boldsymbol{n} \tag{30}$$

where $\boldsymbol{eR}$ represents the error rotation matrix between the desired attitude angle and the actual attitude angle. $\boldsymbol{n}, \varphi$ are the equivalent rotation axis and the equivalent rotation angle of $\boldsymbol{eR}$.

To slove the optimal foot-end force vector $\boldsymbol{f}$ by using QP and friction cone constraint, Eq. (25) can be simplified as $\boldsymbol{A}_f\boldsymbol{f} = \boldsymbol{d}$, whose mathematical model as below.

$$\boldsymbol{f}^* = \min\left(\boldsymbol{A}_f\boldsymbol{f} - \boldsymbol{d}\right)^T \boldsymbol{S}\left(\boldsymbol{A}_f\boldsymbol{f} - \boldsymbol{d}\right) + \alpha\boldsymbol{f}^T\boldsymbol{W}\boldsymbol{f} + \left(\boldsymbol{f} - \boldsymbol{f}^*_{\text{pre}}\right)^T \boldsymbol{\beta}\left(\boldsymbol{f} - \boldsymbol{f}^*_{\text{pre}}\right)$$
$$s.t. \quad \|f_\tau\| \leq \mu\,\|f_n\| \tag{31}$$
$$0 \leq f_{i,z} \leq f_{\max}$$

where the weight matrix $\boldsymbol{S} \in \mathbb{R}^{6\times6}$ is used to adjust the optimized results of the desired force and torque of the CoM, and the second term $\alpha\boldsymbol{f}^T\boldsymbol{W}\boldsymbol{f}$ is used to reduce the energy consumption and the shocks of foot-ground interaction. $\boldsymbol{W}$ is the weight matrix of the energy consumption. The last term is similar to a low-pass filter to prevent sharp changes in the foot-end force. $\|f_\tau\| \leq \mu\,\|f_n\|$ indicates that the foot-end force must be limited to the friction cone. f_τ, f_n are the tangential and normal components of the foot-end force. $\boldsymbol{f}^*_{\text{pre}}$ is the optimized value of the previous solution.

The standard form of QP optimization is below.

$$\min \quad \frac{1}{2}\boldsymbol{f}^T\boldsymbol{H}_f\boldsymbol{f} + \boldsymbol{f}^T\boldsymbol{g}_1$$
$$s.t. \quad \boldsymbol{lbA} \leq \boldsymbol{Mf} \leq \boldsymbol{ubA} \tag{32}$$

By comparing Eqs. (31) and (32), the coefficient matrix $\boldsymbol{H}_1, \boldsymbol{g}_1$ can be derived as follows.

$$\begin{cases} \boldsymbol{H}_1 = 2\boldsymbol{A}^T\boldsymbol{S}\boldsymbol{A} + 2\alpha\boldsymbol{W} + 2\boldsymbol{\beta} \\ \boldsymbol{g}_1 = -2\boldsymbol{A}^T\boldsymbol{S}\boldsymbol{d} - 2\boldsymbol{\beta}\boldsymbol{f}^* \end{cases} \tag{33}$$

The friction cone model of the foot-end force should theoretically satisfy $\sqrt{f_{i,x}^2 + f_{i,y}^2} \leq \mu f_{i,z}$, but the cone constraint needs to be modified to an approximate quadrilateral cone for the purpose of linearizing the inequality [32]. The linearized friction cone can be derived as below.

$$\underbrace{\begin{bmatrix} -\infty \\ 0 \\ -\infty \\ 0 \\ f_{\min} \end{bmatrix}}_{\boldsymbol{lbA}_i} \leq \underbrace{\begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ -\frac{\sqrt{2}}{2}\mu & \frac{\sqrt{2}}{2}\mu & -\frac{\sqrt{2}}{2}\mu & \frac{\sqrt{2}}{2}\mu & 1 \end{bmatrix}^T}_{C_i} \,^O\boldsymbol{f}_i \leq \underbrace{\begin{bmatrix} 0 \\ \infty \\ 0 \\ \infty \\ f_{\max} \end{bmatrix}}_{\boldsymbol{ubA}_i} \tag{34}$$

Algorithm 3: Solving the minimum foot-end force through the method of combing single rigid body dynamics with QP

Initialize algorithm variables: $i, S_\phi, \alpha, W, S, K_{\text{cp}}, K_{\text{cd}}, K_{\text{bp}}, K_{\text{bd}}$

Calculate $e({}^O_B R_d \, {}^O_B R^T)$ by Eqs. (28)–(30)

Calculate d by Eqs. (25)–(27)

For $i = 1:4$ # if the foot contacts the ground, $S_\phi(i) = 1$; else $S_\phi(i) = 0$

    $A_s(1:3, 1 + 3^*(i-1): 3 + 3^*(i-1)) = S_\phi(i)^* I_3$

    $A_s(4:6, 1 + 3^*(i-1): 3 + 3^*(i-1)) = S_\phi(i)^* {}^O r_{i\times}$

    $lbA(1 + 5^*(i-1): 5 + 5^*(i-1), 1) = S_\phi(i)^* lbA_i$

    $ubA(1 + 5^*(i-1): 5 + 5^*(i-1), 1) = S_\phi(i)^* ubA_i$

end for

$C = Diag(C_1, C_2, C_3, C_4)$

Obtain coefficient matrix H_1, g_1 by Eq. (33)

Solving ${}^O f_i$ through the quadprog function in MATLAB

Based on Eqs. (31)–(34), a set of expected foot-end forces can be solved, and the specific solution process is shown in Algorithm 3. Assuming that there is no slip at the foot end, the joint torque vector τ_i of the leg i can be obtained by Eq. (35).

$$\tau_i = -J_i^{TO} {}^T_B R^{TO} f_i \tag{35}$$

3.5. Whole-body inertia analysis of the wheel-legged separation quadruped robot

From Eq. (25), it can be seen that the rotational inertia I_G will affect the result of the foot-end force distribution. When the wheels fall off the body, the rotational inertia I_G of the body will change. The rotational inertia of the whole body can be analyzed as follows.

$$h = h_{\text{init}} + 0.5 h_p \left(1 - \cos(t)\right) + h_l \tag{36}$$

$$L_4 = 3L_1, L_2 = L_3 = 2L_1 \tag{37}$$

$$\theta_w = \arcsin\left(h_p / 4L_1\right) \tag{38}$$

$$\omega_1 = -\omega_2 = \frac{d\theta_w}{dt} \tag{39}$$

$$I_G = I_b + I_{\text{ef}} + I_{\text{eb}} \tag{40}$$

$$I_{\text{ef}} = I_1 + I_2 + I_3 \frac{\omega_3^2}{\omega_1^2} + I_4 \frac{\omega_4^2}{\omega_1^2} + \left(m_3 \frac{v_3^2}{\omega_1^2} + m_4 \frac{v_4^2}{\omega_1^2} + M_f \frac{v_l^2}{\omega_1^2}\right) \cdot \begin{bmatrix} 0 & & \\ & 1 & \\ & & 0 \end{bmatrix} \tag{41}$$

$$I_{\text{eb}} = I_1 + I_2 + I_3 \frac{\omega_3^2}{\omega_1^2} + I_4 \frac{\omega_4^2}{\omega_1^2} + \left(m_3 \frac{v_3^2}{\omega_1^2} + m_4 \frac{v_4^2}{\omega_1^2} + M_b \frac{v_l^2}{\omega_1^2}\right) \cdot \begin{bmatrix} 0 & & \\ & 1 & \\ & & 0 \end{bmatrix} \tag{42}$$

where h_{init} is the height of the wheels from the CoM at the beginning, h_p is the travel distance of the four-bar mechanism. h_l, h are the height of the wheel fixed frame and the CoM of the body from the lower edge of the wheel, respectively, shown in Fig. 5. $I_b, I_{\text{ef}}, I_{\text{eb}} \in \text{R}^{3 \times 3}$ are the rotational inertia of the body's constant parts, the equivalent rotational inertia of the front and the rear wheels four-bar raising and lowering module, respectively. $I_1, I_2, I_3, I_4 \in \text{R}^{3 \times 3}$ and $\omega_1, \omega_2, \omega_3, \omega_4$ are the inertia and angular velocity of
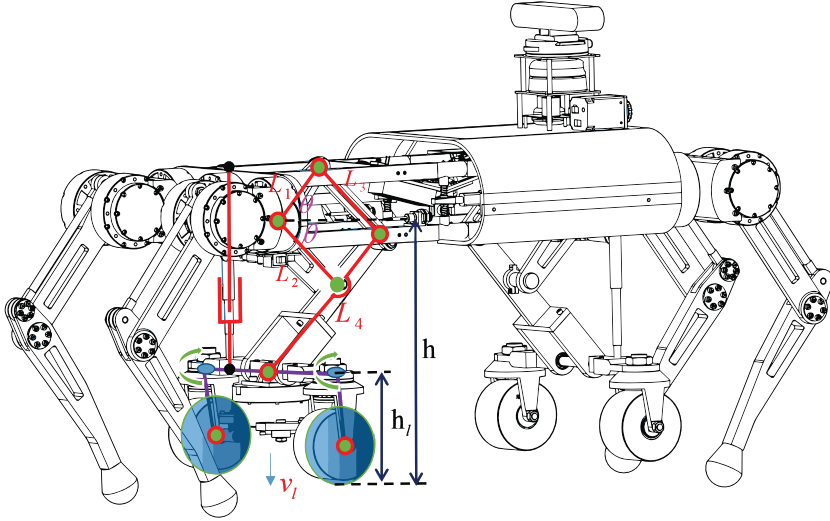
Figure 5. *Analysis of the four-bar raising and falling module.*

the connecting rods L_1, L_2, L_3, L_4, respectively. The connecting links L_1, L_2 rotate synchronously through the positive and negative rotation of bevel gears, so their rotation angles θ_w are identical. M_f and M_b are the mass of front and rear assemblies and rear wheel assemblies. m_3, m_4 are the mass of the connecting rods L_3, L_4, respectively. v_3 and v_4 is the translational speed of the connecting rods L_3, L_4 in frame {B}. v_l is the speed of wheels raising or falling.

3.6. Swing phase control of the wheel-legged separation quadruped robot

In order to better track foot-end trajectory, a control method based on the fusion of kinetic and virtual models is used, which consists of the joint PD [38], the feedforward dynamics, and the feedback dynamics. The joint PD is used to calculate the virtual joint torque of the swing legs as follows:

$$\boldsymbol{\tau}_{\text{PD}} = \boldsymbol{K}_{\text{sp}}(\boldsymbol{\theta}_{\text{d}} - \boldsymbol{\theta}) + \boldsymbol{K}_{\text{sd}}(\dot{\boldsymbol{\theta}}_{\text{d}} - \dot{\boldsymbol{\theta}}) \tag{43}$$

where $\boldsymbol{K}_{\text{sp}} \in \text{R}^{3\times3}$ and $\boldsymbol{K}_{\text{sd}} \in \text{R}^{3\times3}$ are the stiffness coefficient and damping coefficient, respectively. $\boldsymbol{\theta}_{\text{d}} = \begin{bmatrix} \theta_{1\text{d}} & \theta_{2\text{d}} & \theta_{3\text{d}} \end{bmatrix}^{\text{T}}$ represent the desired joint angle of the single leg, which is obtained by substituting the foot-end planning trajectory $\boldsymbol{\chi}_{\text{d}} = [\,x_{\text{d}} \quad y_{\text{d}} \quad z_{\text{d}}\,]^{\text{T}}$ into the single-leg inverse kinematics [39]. The desired joint angular velocity $\dot{\boldsymbol{\theta}}_{\text{d}}$ is set as 0. $\boldsymbol{\theta} = [\,\theta_1 \quad \theta_2 \quad \theta_3\,]^{\text{T}}$ and $\dot{\boldsymbol{\theta}} = \begin{bmatrix} \dot{\theta}_1 & \dot{\theta}_2 & \dot{\theta}_3 \end{bmatrix}^{\text{T}}$ represent the feedback joint angle and angular velocity, respectively.

The dynamic model of the single leg was constructed by using the Lagrangian approach, and the specific derivation of this model was published in the previous study [39]. The final dynamic model was organized in the following form.
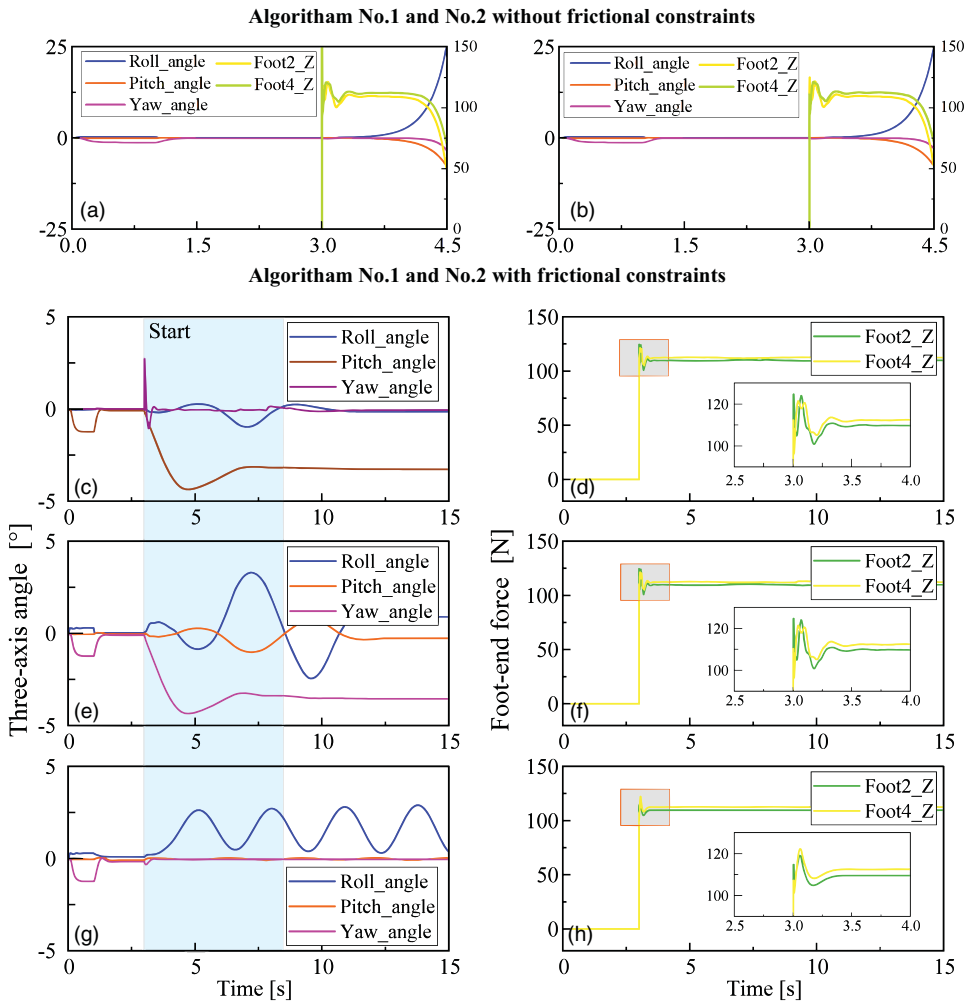
$$\boldsymbol{\tau}_{\text{f}} = \boldsymbol{M}\boldsymbol{\theta}_d + \boldsymbol{V}\dot{\boldsymbol{\theta}}_{\text{f}} + \boldsymbol{G}_f \tag{44}$$

where $\boldsymbol{M}$ indicates the inertia parameter. $\boldsymbol{V} \in \text{R}^{3\times3}$ represents the Coriolis force and centrifugal force. $\boldsymbol{G}_f \in \text{R}^3$ indicates the gravity term. The specific relationship between the acceleration $\ddot{\boldsymbol{\theta}}_{\text{d}} = \begin{bmatrix} \ddot{\theta}_{1\text{d}} & \ddot{\theta}_{2\text{d}} & \ddot{\theta}_{3\text{d}} \end{bmatrix}^{\text{T}}$ and the expected trajectory $\boldsymbol{\chi}_d$ of the foot end is as follows.

$$\ddot{\boldsymbol{\theta}}_{\text{d}} = \boldsymbol{J}_i^{-1}(\boldsymbol{\theta}_{\text{d}}) \cdot \left(\ddot{\boldsymbol{\chi}}_{\text{d}} - \dot{\boldsymbol{J}}_i(\boldsymbol{\theta}_{\text{d}}) \cdot \boldsymbol{J}_i^{-1}(\boldsymbol{\theta}_{\text{d}}) \cdot \dot{\boldsymbol{\chi}}_{\text{d}}\right) \tag{45}$$

The joint angle $\boldsymbol{\theta}$ and velocity $\dot{\boldsymbol{\theta}}_{\text{f}}$ of the feedback dynamics term are obtained from the feedback of the joint motor. The final joint torque of the swing legs is as below.

$$\boldsymbol{\tau}_{\text{b}} = \boldsymbol{\tau}_{\text{PD}} + \boldsymbol{\tau}_{\text{f}} \tag{46}$$

Algoritham No.1 and No.2 without frictional constraints

Algoritham No.1 and No.2 with frictional constraints

Figure 6. *The change of three-axis angle and foot-end force. Foot2_Fz, and Foot4_Fz represent the Z-axis force on the foot end of legs 2, 4.*

4. Simulation and analysis

In this section, we summarize the advantages of three algorithms by setting up different simulated projects and create a multi-algorithm combination control scheme. Finally, the scheme is validated in the dynamic switching of the wheel-leg modal of the wheel-legged separation quadruped robot.

4.1. The simulation of standing with multilegged support

4.1.1. The simulation of standing with two-legged support

The stability of two-legged support strongly influences the fast walking of the robot in trot gait. In the simulation, the legs 1,3 are raised to 0.2 m, and the remaining legs are used to distribute the force and torque required to maintain the balance of the body. The robot starts with two-legged support at 3 s.

When the algorithm No.1 and No.2 do not consider the frictional constraints, they can only maintain the balance of the body for 1 s, after which the CoM of the body gradually moves away from the diagonal of the support feet and loses control, shown in Fig. 6(a). The actual effects of the two algorithms in the simulation environment are shown in Fig. 7(a)(C1) and 7(a)(C2). However, when algorithms No.1
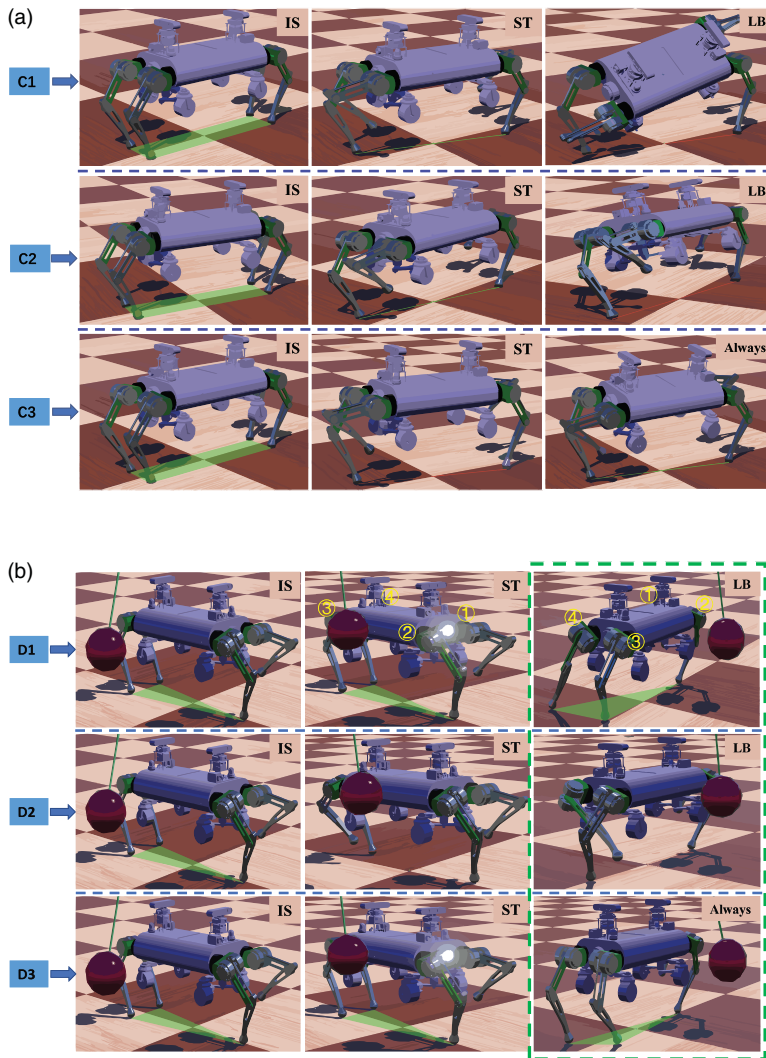
Figure 7. *The actual effects of multilegged support under three algorithms. "IS" represents the initial state of the robot. "ST" represents that the simulation of multilegged support starts. "LB" represents that the robot is losing balance. "Always" represents that the robot can continuously be standing. The green area represents the final state of the robot under three algorithms. When the robot is supported by two feet without considering the frictional constraints, (C1) and (C2) reflect the actual effects of using algorithm No.1 and algorithm No.2, respectively. When the robot is supported by two feet with considering the frictional constraints, (C3) reflect the common actual effects of when the robot adopts algorithm No.1, No.2, and No.3 to control supporting legs. (D1), (D2), and (D3) represent the actual effects of using the algorithm No.1, No.2, and No.3, respectively, when the robot is supported by three feet, both of which have frictional constraints. (a) The actual effects of two-legged support. (b) The actual effects of three-legged support.*

and No.2 add the frictional constraints proposed in Section 3.3, the quadruped robot sustainably maintains static standing of two-legged support, shown in Fig. 6(c) and 6(e). The actual effects of the two algorithms are shown in Fig. 7(a)(C3). Therefore, the frictional constraint has a significant effect on maintaining body's balance.

Algorithm No.3 also considers the friction constraints, so it can continuously maintain the sustained standing of the two-legged support, shown in Fig. 7(a)(C3). However, the Fig. 6(g) shows that algorithm No.3 can quickly maintain the stability of the pitch and yaw angles by adjusting the foot-end force. For the actual physical prototype, the center of gravity usually cannot coincide with the ideal geometric center. To make the robot stand steadily, the robot needs to independently adjust the attitude angle of its body to ensure that the CoM is always on the diagonal of the supporting feet. Hence, the roll angle of the Fig. 6(g) is is dynamically adjusted. Besides, from the perspective of foot-end force, algorithm No.3 can adjust foot force more smoothly to balance the body, as shown in Fig. 6(d), 6(f), and 6(h). That is because algorithm No.3 can distribute the optimized results of desired forces and desired torque of the CoM by setting the weight matrix S in Eq. (31). Of course, the diagonal weight value of the matrix S related to the three-axis attitude angle is adjusted to a large value, which aims to allocate enough force to stabilize the body's attitude quickly. Therefore, algorithm No.3 is more effective for maintaining the body's balance when the robot is in the two-legged support.

4.1.2. The simulation of standing with three-legged support

In addition to the two-legged support, the three-legged support is also commonly used in the quadruped robot. In order to observe the actual performance of three control algorithms, we conducted a simulation study on the anti-interference performance of the wheel-legged separation quadruped robot with a three-legged support.

Since the balance of the body can only be ensured when the CoM is located within the support polygon of the three legs, legs 1 and 2 are set to move inward in the initial standing state. Meanwhile, set legs 2, 3, and 4 as supporting legs to jointly bear the required force and torque for robotic balance. Leg 1 is raised to 0.2 m in the initial state of the robot. Besides, a pendulum weighing 1 kg is set to lateral interference, which starts swinging from the same height and time when controlling the robot with different algorithms. Based on the above conditions, the simulation process of the three-legged support of the robot has been carried out in Fig. 7(b), and the analysis results are as follows:

- When algorithms No.1 and No.2 are used to control the three-legged support of the robot, the robot mainly relies on leg 4 to dynamically adjust the support polygon. After the robot is subjected to lateral interference at 2.5 s, leg 4 exhibits poor dynamic adjustment ability, so it gradually becomes strange and eventually loses control, as shown in Fig. 7(b)(D1) and (D2).

- When algorithm No.3 is used to control the three-legged support of the robot, the position of leg 4 can be adjusted dynamically. At this point, the robotic CoM can always remain within the supporting polygon and away from the edge line, so the robot can maintain continuous standing of three-legged support, as shown in Fig. 7(b)(D3).

- Besides, comparing the joint feedback torques of support legs under three algorithms, it can be clearly seen that algorithm No.3 can quickly adjust the joint torque to stabilize the body when the robot's body is subjected to a lateral disturbance in 2.5 s. Meanwhile, the maximum torque after disturbance does not exceed the peak torque 35 N · m of the servo motor. However, when the other two algorithms are used to control the robot, the lateral interference can lead to joint torques of support legs exceeding the peak value, as shown in the torque feedback shadow of the three support legs in Fig. 8. Therefore, algorithm No.3 has strong anti-interference performance when the robot is supported by three legs.

As mentioned in Sections 4.2.1 and 4.2.2, Algorithm No.3 has better effects for the multilegged support of the robot.

4.2. The simulation of omnidirectional walking

For different algorithms, there are significant differences in the precise tracking of the robot's expected walking speed and the maintenance of dynamic stability of the body in the leg modal. In order to observe
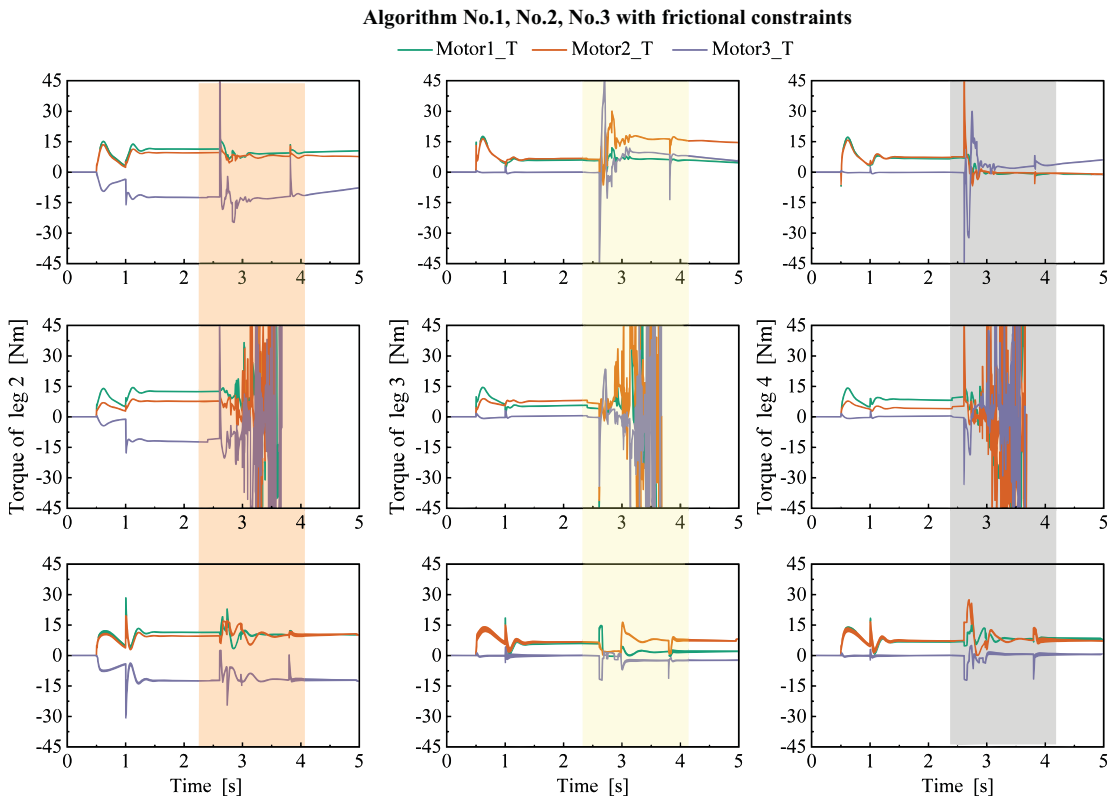
Figure 8. *The joint motor torques of support legs under three algorithms. Motor1_T, Motor2_T, and Motor3_T represent the feedback torque of the thigh, calf, and side swing motor.*

the actual control effects of three algorithms in the rotational motion, lateral motion, forward motion, and oblique motion, we have carried out the following motion planning:

1. The robot rotates $150°$ at a speed of 1.5 rad/s around the axis Z_B and always maintains the angle to complete the following motion planning.

2. The robot walks at an expected maximum lateral speed of 1 m/s, and the robotic speed in the axis Y_B changes in a sinusoidal law.

3. The robot walks at an expected maximum forward speed of 2 m/s, and the robotic speed in the axis X_B changes in a sinusoidal law.

4. The robot walks at a combined velocity of an expected maximum lateral speed of -1 m/s and expected maximum forward speed of -2 m/s, and the robotic speed in the axis X_B and Y_B changes in a sinusoidal law.

Based on the above motion planning, the actual movement situation is shown in Fig. 9(a), and the simulation results of robot motion tracking and three-axis feedback are obtained.

- Algorithm No.1 and algorithm No.2 can accurately track the desired speed of the CoM, shown in Fig. 9(c) and (e). However, the roll and pitch angle of the body fluctuates greatly, which can easily lead to the body being out of balance, shown in Fig. 9(b) and (d).
- Similarly, the control algorithm No.3 also has a small error in tracking the desired speed in the axis X_B and axis Y_B, shown in Fig. 9(c) and (e). In Eq. (31), the diagonal weight value of
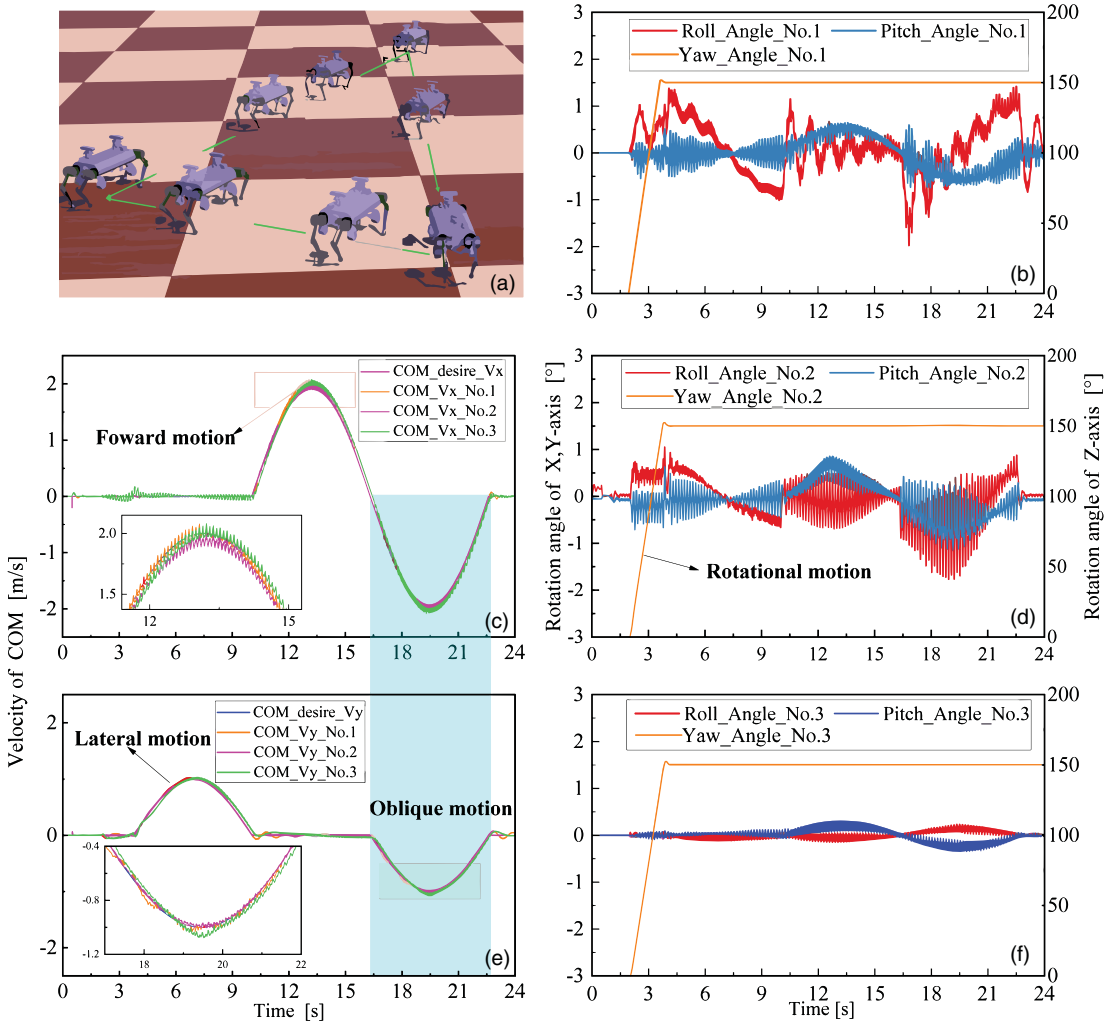
Figure 9. *Speed tracking and change of three-axis angle under three algorithms. CoM_desire_vx and CoM_desire_vy represent the expected velocity in the X_B and Y_B of the CoM, respectively. CoM_Vx_No.x and CoM_Vy_No.y are the actual velocity of the axis X_B and Y_B measured by the GPS, respectively.*

the matrix **S** related to the three-axis attitude angle is set to a considerable value, which means maintaining the body's stability is always the highest priority of control tasks when the robot walks at any speed. Therefore, algorithm No.3 enables the robot to have no more than 0.5° roll and pitch angle fluctuations, shown in Fig. 9(f).

In actual omnidirectional walking, maintaining the continuous stability of the body is the foundation for the robot to complete other motions. Therefore, algorithm No.3 is more suitable for the wheel-legged separation quadruped robot to achieve stable walking.

4.3. The simulation of walking with wheels raising and falling

Considering the need to dynamically switch between the wheel and leg modal, the simulation of walking with wheels raising and falling is performed, shown in Fig. 10. Owing to the dynamic switching of the wheel-leg modal requires that the robot can switch modal at any speed within the allowed speed range
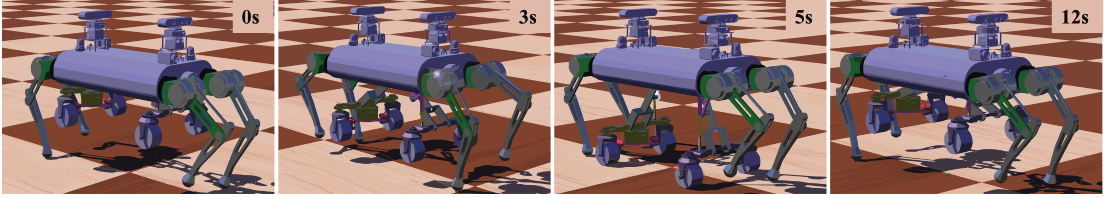
Figure 10. *The process of walking with wheels raising and falling. During the 0∼5 s process, the robot walks rapidly and is accompanied by a vertical descent of the wheels. After 5 s, the robot reaches its maximum speed, and the wheels descend to the bottom and come into contact with the ground. Then, within 12 s, the wheels retract to the set height of the vehicle's body.*
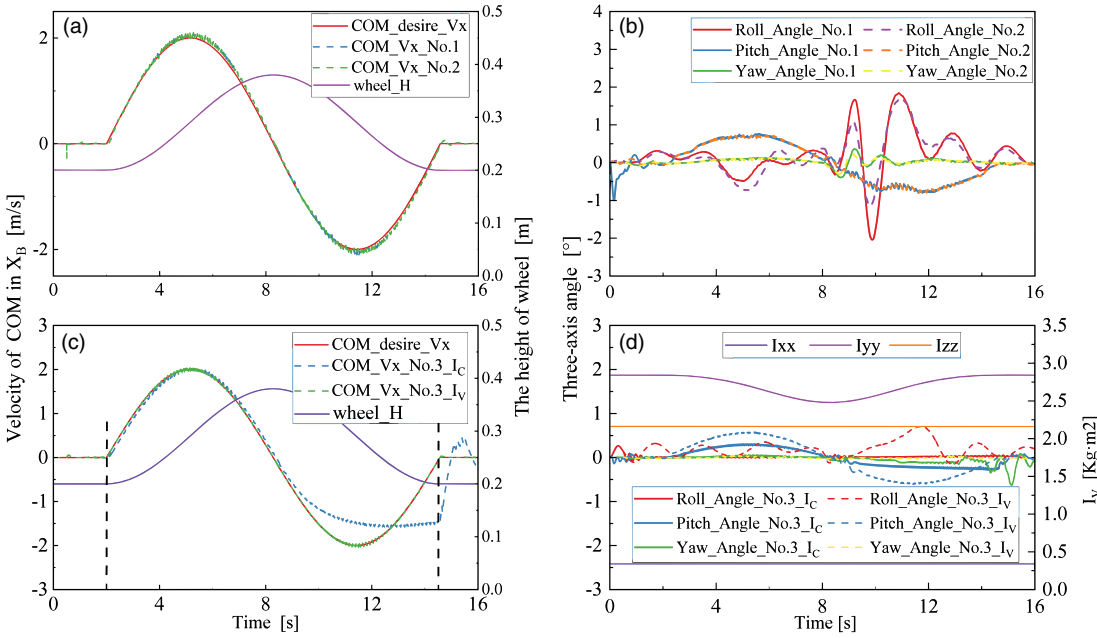
Figure 11. *Speed tracking and change of three-axis angles under three algorithms. CoM_Vx_No.3_I_C and CoM_Vx_No.3_I_V represent Algorithm 3 using constant rotational inertia and varying rotational inertia, respectively. wheel_H represents the planning height of the wheels falling and raising.*

of the gait, the body's speed in X_B is set to vary in a sinusoidal pattern when the robot walks with wheels raising and falling. In order to achieve compliance control of the wheels raising and falling, Eq. (36) is used for planning the wheeled height, and the cycle t of wheels raising and falling is consistent with the change cycle of robotic speed. Based on the above-simulated settings, the robot's speed tracking and three-axis angle changes under three algorithms are shown in Fig. 11.

- When algorithms No.1 and No.2 are used to control the robotic walking, the desired speed in the axis X_B can be tracked accurately, shown in Fig. 11(a). Besides, the roll and pitch angles of the robot fluctuate within a range of $\pm 2°$.
- When algorithm No.3 is adopted to control the robotic walking, the body has smaller angular fluctuations compared to the other two algorithms, shown in Fig. 11(d). However, the speed-tracking ability significantly deteriorates, as shown in Fig. 11(c). According to Eqs. (36)–(42), it

Table II. *All simulated results of the wheel-leg separation quadruped robot.*

Simulated motion	Comparison index	Algorithm No.1	Algorithm No.2	Algorithm No.3
Two-legged support	Attitude angle/foot-end force	Maximum 4°/rough	Maximum 4°/rough	Maximum 0.5°/smoothly
Three-legged support	Joint torque	Lose balance/ exceed 35 N · m	Lose balance/ exceed 35 N · m	Keep balance/ maximum 30 N · m
Omnidirectional walking	Speed tracking/ attitude angle	Accurate/ maximum 1.5°	Accurate/ maximum 1.5°	Accurate/ maximum 0.5°
Walking with wheels raising and falling	Speed tracking/ attitude angle	Accurate/ maximum 2°	Accurate/ maximum 1.5°	Inaccurate/ maximum 0.7°

is not difficult to observe that the inertia varies considerably with the wheels lifting and falling in practice. At this point, it is unreasonable for Algorithm 3 to continue using a constant inertia matrix $\boldsymbol{I}_G$. When we substitute the variable rotational inertia $\boldsymbol{I}_G$ into algorithm No.3, the robot's speed-tracking ability is significantly improved, and the three-axis angle of the body fluctuates within a range of $\pm 1°$, shown in Fig. 11(d).

Although the variation $\boldsymbol{I}_G$ can be derived theoretically, there will be a large inertia error in the physical prototype. That is because the impact of manufacturing and assembly errors on the inertia calculation in practical control must be considered. Otherwise, significant speed errors will be generated, which is very likely to lead to the robot collisions with the obstacle at high speed when dynamically switching the wheel-leg modal in some narrow space. However, algorithms No.1 and No.2 can still achieve better speed tracking without considering the influence of the change of the rotational inertia. Therefore, they are more suitable for the walking control of the robot at the wheel-leg modal switching moment.

Based on the effects of three algorithm for all simulation projects in Table II, we formulate a multi-algorithm combination control scheme as below: algorithm No.3 is suitable for applying to the omnidirectional walking and attitude transformation of the wheel-legged separation quadruped robot. When switching the wheel-leg modal dynamically, algorithms No.1 and No.2 can be used to track the target speed accurately, which does not need to consider the disturbance caused by the change of the body's rotational inertia.

4.4. The simulation of dynamically switching wheel-leg modal

A set of multi-algorithm combination control schemes has been summarized by analyzing the simulation results from Sections 4.1–4.3. We apply the scheme to the dynamic switching of the wheel-leg modal in the simulation. The specific process of robotic modal transformation is shown in Fig. 12.

1. At 4 s, the robot walks with wheels falling.
2. At 5 s, the four leg execution modules of the robot detach from the ground, and the robot begins to convert the leg modal to the wheel modal.
3. At 6 s, the four leg execution modules of the robot begin to contract to the top of the body according to the planned polynomial trajectory.
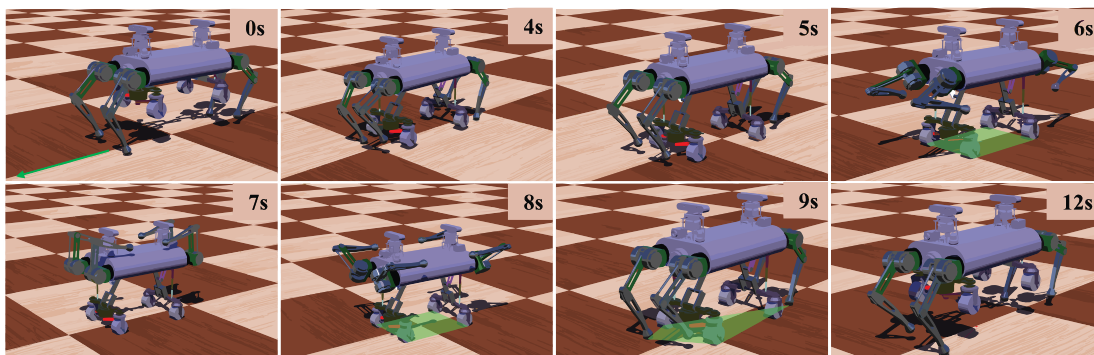4. At 7 s, the robot enters a high-speed driving modal.

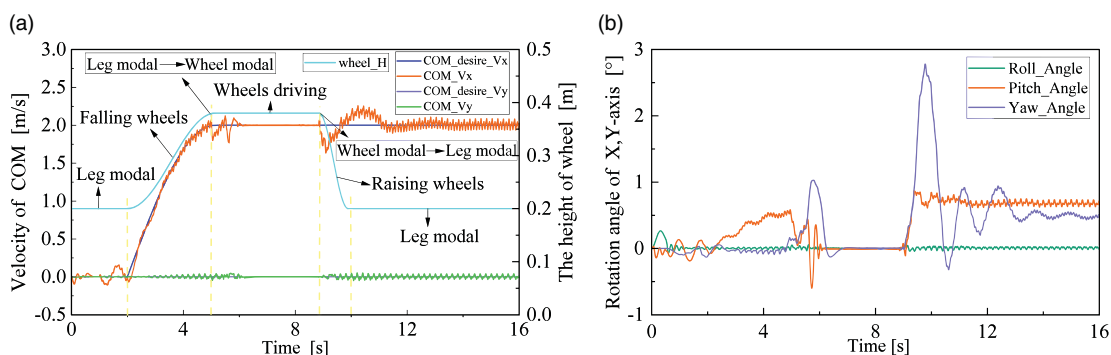**Figure 12.** *The process of dynamically switching wheel-leg modal.*

Figure 13. *Speed tracking and change of three-axis angles under the combined control scheme.*

5. At 8 s, the robot begins to recover the position of the four leg execution modules according to the planning polynomial trajectory.

6. At 9 s, the robot's feet begin to come into contact with the ground for walking, and the wheels are ready to detach from the ground. The robot converts the wheel modal to the leg modal.

7. At 12 s, the robot returns to its initial state of the leg modal.

During the entire process of the above-mentioned robotic movement, the speed-tracking situation and changes in the three-axis angle are shown in Fig. 13.

From Fig. 13(a), it can be seen that the wheel-legged separation quadruped robot can track the expected speed in the axis X_B direction. Meanwhile, at the moment of dynamic modal switching, the actual speed of the body will experience small fluctuations and eventually recover to the expected value. After the robot switches from the wheel modal to the leg modal, there is also a small velocity fluctuation in the axis Y_B to maintain the stability of the robot's yaw angle. Besides, throughout the simulation process, the roll and pitch angles of the robot's body do not change by more than 1°, shown in Fig. 13(b). The simulation results indicate that the multi-algorithm combined control scheme can enable the robot to achieve dynamic switching of the wheel-leg modal and maintain continuous stability of the body.

5. Conclusions and future work

This article presents a mechanism design scheme and a control method of switching modal dynamically for the novel wheel-legged separation quadruped robot. The overall structure of the wheel-legged robot

is different from other existing wheel-legged series or metamorphic robots. Its dual-mode interchangeability is realized by a four-bar raising and falling module (see Figs. 1 and 5), which can complete the conversion between legs and wheels without static waiting according to the modal command. In order to achieve the goal of high-speed dynamic switching between wheel and leg modal, we further propose a framework of multi-algorithm combination control to achieve achieve the coordinated operation of multimodal, which includes improved algorithm No.1 and algorithm No.2 and an existing algorithm No.3. Since algorithm No.3 has been proved to have strong advantages in speed tracking and body stability [29, 32], it is considered as a reference for robotica control methods.

Subsequently, in the simulation of multilegged standing and omni-directional walking, algorithm No.3 has better performance in speed tracking and attitude stability than the other two algorithms. However, in the simulation of walking with wheels raising and falling, algorithm No.3 degrade the speed tracking ability of the robot due to the change of the whole-body rotating inertia. Therefore, we propose improved algorithm No.1 and algorithm No.2, which do not need to consider the dynamic changes of the whole-body moment of inertia. All of them are different from the traditional idea of adding equation constraints and so on to complete the VMC force distribution [23–26]. We develop a method that combines the least square method with full rank decomposition to solve the singularity problem of the force and joint torque distribution matrix in Eqs. (11) and (18), which has low computational power requirements for hardware and fast solving speed. Meanwhile, considering the possibility of foot sliding, friction constraints are established for the two algorithms. In the walking simulation with wheels raising and falling, the proposed improved algorithm No.1 and algorithm No.2 show strong speed tracking ability without being affected by the change of whole-body inertia. According to the actual performance of the three algorithms summarized in Table II, the two algorithms and the existing algorithm No.3 constitute a combined control scheme, and which has been used to perfectly realize the dynamic switching control of the wheel-and-leg modal of the quadruped robot.

Future work can be studied in the following three aspects. The first aspect is to apply the multi-algorithm combination control in dynamically switching control of the physical prototype. The second aspect is to carry out research on wheeled high-speed driving and wheel-leg compound movement [39]. The third aspect is to achieve the purpose of crossing complex terrain quickly by researching the coordination of multimodal.

Author contributions. Jiandong Cao: Writing – original draft, methodology. Jinzhu Zhang: supervision, validation, writing, tutor – review and editing. Tao Wang: Supervision, validation. Jiahao Meng: Validation, software. Senlin Li: Manufacturing prototype. Miao Li: Validation, Methodology.

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.

Ethical approval. Not applicable.

References

[1] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Sci. Robot.* **5**(47), eabc5986 (2020).

[2] C. D. Bellicoso, M. Bjelonic, L. Wellhausen, K. Holtmann, F. Günther, M. Tranzatto, P. Fankhauser and M. Hutter, "Advances in real-world applications for legged robots," *J. Field. Robot.* **35**(8), 1311–1326 (2018).

[3] S. Halder, K. Afsari, E. Chiou, R. Patrick and K. A. Hamed, "Construction inspection & monitoring with quadruped robots in future human-robot teaming: A preliminary study," *J. Build. Eng.* **65**, 105814 (2023).

[4] D. Wisth, M. Camurri and M. F. Fallon, "VILENS: Visual, inertial, lidar, and leg odometry for all-terrain legged robots," *IEEE Trans. Robot.* **39**(1), 309–326 (2021).

[5] L. Ding, Z. Deng, H. Gao, K. Nagatani and K. Yoshida, "Planetary rovers" wheel–soil interaction mechanics: New challenges and applications for wheeled mobile robots," *Intell. Serv. Robot.* **4**(1), 17–38 (2011).

[6] L. Ding, Z. Deng, H. Gao, J. Guo, D. Zhang and K. D. Iagnemma, "Experimental study and analysis of the wheels' steering mechanics for planetary exploration wheeled mobile robots moving on deformable terrain," *Int. J. Robot. Res.* **32**(6), 712–43 (2013).

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

[8] K. Xu, Y. Lu, L. Shi, J. Li, S. Wang and T. Lei, "Whole-body stability control with high contact redundancy for wheel-legged hexapod robot driving over rough terrain," *Mech. Mach. Theory* **181**, 105199 (2023).

[9] J. Li, H. Qin, J. Wang and J. Li, "OpenStreetMap-based autonomous navigation for the four wheel-legged robot via 3D-lidar and CCD camera," *IEEE Trans. Ind. Electron.* **69**(3), 2708–2717 (2021).

[10] J. He, Y. Sun, L. Yang, J. Sun, Y. Xing and F. Gao, "Design and control of TAWL—A wheel-legged rover with terrain-adaptive wheel speed allocation capability," *IEEE/ASME Trans. Mechatron.* **27**(4), 1–12 (2022).

[11] W. Du, M. Fnadi and F. Benamar, "Rolling based locomotion on rough terrain for a wheeled quadruped using centroidal dynamics," *Mech. Mach. Theory* **153**, 103984 (2020).

[12] M. Bjelonic, C. D. Bellicoso, Y. de Viragh, D. Sako and F. D. Tresoldi, "Keep rollin'—whole-body motion control and planning for wheeled quadrupedal robots," *IEEE Robot. Autom. Lett.* **4**(2), 2116–2123 (2019).

[13] W. Chi, X. Jiang and Y. Zheng, "A Linearization of Centroidal Dynamics for the Model-Predictive Control of Quadruped Robots," **In:** *2022 International Conference on Robotics and Automation (ICRA)* (IEEE, 2022) pp. 4656–4663.

[14] S.-C. Chen, K.-J. Huang, W.-H. Chen, S.-Y. Shen and C.-H. Li, "Quattroped: A leg-wheel transformable robot," *IEEE/ASME Trans. Mechatron.* **19**(2), 730–742 (2013).

[15] Y.-S. Kim, G.-P. Jung, H. Kim, K.-J. Cho and C.-N. Chu, "Wheel transformer: A wheel-leg hybrid robot with passive transformable wheels," *IEEE Trans. Robot.* **30**(6), 1487–1498 (2014).

[16] C. Liu, X. Tan, Y. Yao and Z. Fu, "Design and analysis of a novel deformable wheel-legged robot," *J. Mech. Eng.* **58**(3), 65–74 (2022).

[17] C. Zheng, S. Sane, K. Lee, V. Kalyanram and K. Lee, α-WaLTR: Adaptive wheel-and-leg transformable robot for versatile multiterrain locomotion. *IEEE Trans. Robot.* **39**(2), 941–958 (2022).

[18] S. Wang, L. Cui, J. Zhang, J. Lai and D. Zhang, "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.

[19] M. Bjelonic, P. K. Sankar, C. D. Bellicoso, H. Vallery and M. Hutter, "Rolling in the deep - hybrid locomotion for wheeled-legged robots using online trajectory optimization," *IEEE Robot. Autom. Lett.* **5**(2), 3626–3633 (2020).

[20] M. Bjelonic, R. Grandia, O. Harley, C. Galliard and S. Zimmermann, "Whole-Body MPC and Online Gait Sequence Generation for Wheeled-Legged Robots," **In:** *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2021) pp. 8388–8395.

[21] V. S. Medeiros, E. Jelavic, M. Bjelonic, R. Siegwart and M. A. Meggiolaro, "Trajectory optimization for wheeled-legged quadrupedal robots driving in challenging terrain," *IEEE Robot. Autom. Lett.* **5**(3), 4172–4179 (2020).

[22] K.-X. Ba, Y.-H. Song, Y.-P. Shi, C.-Y. Wang, G.-L. Ma, Y. Wang, B. Yu and L.-P. Yuan, "A novel one-dimensional force sensor calibration method to improve the contact force solution accuracy for legged robot," *Mech. Mach. Theory* **169**, 104685 (2022).

[23] J. Y. He, J. P. Shao, B. W. Gao, B. Y. Miao, X. Shao and C. Aguilar-Ibanez, "Suppression of quadruped robot body disturbance by virtual spring-damping model," *Complexity* **2022**, 1–12 (2022).

[24] G. Chen, S. Guo, B. Hou and J. Wang, "Virtual model control for quadruped robots," *IEEE Access* **8**, 140736–140751 (2020).

[25] E. Mingcheng, H. Liu, X. Zhang, F. Chenglong and M. Hongxu, "Compliant gait generation for a quadruped bionic robot walking on rough terrains," *Robot* **36**(5), 584–591 (2014).

[26] P. Wang, Y. Shi, F. Zha, Z. Jiang, X. Wang and Z. Li, "An analytic solution for the force distribution based on Cartesian compliance models," *Int. J. Adv. Robot. Syst.* **16**(1), 1729881419827473 (2019).

[27] G. Zhang, X. Rong, C. Hui, Y. Li and B. Li, "Torso motion control and toe trajectory generation of a trotting quadruped robot based on virtual model control," *Adv. Robot.* **30**(4), 284–297 (2016).

[28] G. Chikun, W. Xunwei and Y. Lipeng, "Control of quadruped robot based on impedance and virtual model," *J. Syst. Simul.* **34**(10), 2152 (2022).

[29] S. Gu, F. Meng, B. Liu, Z. Zhang and N. Sun, "Stability control of quadruped robot based on active state adjustment," *Biomimetics* **8**(1), 112 (2023).

[30] K. Wang, H. Zhao, F. Meng and X. Zhang, "Research on the jumping control methods of a quadruped robot that imitates animals," *Biomimetics* **8**(1), 36 (2023).

[31] H. Qin, Y. Zhu, Y. Zhang and Q. Wei, "Terrain estimation with least squares and virtual model control for quadruped robots," *J. Phys. Conf. Ser.* **2203**(1), 012005 (2022).

[32] M. Focchi, A. D. Prete, I. Havoutis, R. Featherstone and D. G. Caldwell, "High-slope terrain locomotion for torque-controlled quadruped robots," *Auton. Robot.* **41**(1), 259–272 (2017).

[33] S. Wang, Y. Shi, X. Wang, Z. Jiang and B. Yu, "State Estimation for Quadrupedal Using Linear Inverted Pendulum Model," **In:** *2017 2nd International Conference on Advanced Robotics and Mechatronics (ICARM)* (IEEE, 2017) pp. 13–18.

[34] G. Fink and C. Semini, "Proprioceptive Sensor Fusion for Quadruped Robot State Estimation," **In:** *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2020) pp. 10914–10920.

[35] Q. Zhu, B. Yu, Z. Huang, Y. Kang, X. Kong and K. Ba, "State feedback-based impedance control for legged robot hydraulic drive unit via full-dimensional state observer," *Int. J. Adv. Robot. Syst.* **17**(3), 172988142092461 (2020).

[36] S. S. Roy, A. K. Singh and D. K. Pratihar, "Estimation of optimal feet forces and joint torques for on-line control of six-legged robot," *Robot. Comput. Integr. Manuf.* **27**(5), 8 (2011).

[37] Z. Li, S. S. Ge and S. Liu, "Contact-force distribution optimization and control for quadruped robots using both gradient and adaptive neural networks," *IEEE Trans. Neural. Netw. Learn. Syst.* **25**(8), 1460–1473 (2014).

[38] N. Rudin, H. Kolvenbach, V. Tsounis and M. Hutter, "Cat-like jumping and landing of legged robots in low gravity using deep reinforcement learning," *IEEE Trans. Robot.* **38**(1), 317–328 (2021).

[39] J. Zhang, M. Li, J. Cao, Y. Dou and X. Xiong, "Research on bionic jumping and soft landing of single leg system in quadruped robot," *J. Bionic Eng.* **20**(5), 2088–2107 (2023).