Locomotion Control of Quadruped Robots With Online Center of Mass Adaptation and Payload Identification

CHAO DING [ID], **LELAI ZHOU** [ID], (Member, IEEE), **YIBIN LI** [ID], (Member, IEEE),
AND XUEWEN RONG [ID], (Member, IEEE)
Center for Robotics, School of Control Science and Engineering, Shandong University, Jinan 250061, China
Engineering Research Center of Intelligent Unmanned System, Ministry of Education, Jinan 250061, China

Corresponding author: Lelai Zhou (zhoulelai@sdu.edu.cn)

ABSTRACT Most of the controllers for quadruped robots are based on the planning of their centers of mass (CoM), with the assumption that the CoM locates at the geometrical center of the robot trunk. However, this assumption rarely meets the fact of the robot prototype and introduces apparent influences to the system. This article proposes a CoM estimation and adaptation method for quadruped robots in dynamic trot gait based on the model predictive control (MPC) of the trunk. The influences of the CoM offset on the robot states are inherently analyzed. The CoM components on the horizontal plane are taken into consideration and estimated based on the speed errors and the contact force differences, which are compensated to the robot controller. In the case of payload changes, the identification method for the mass of payload is proposed based on the estimated CoM. Simulation results verify that with the proposed approach, the robot successfully adapts to the CoM payload changes while the locomotion is not interrupted, and the state errors are significantly reduced.

INDEX TERMS Quadruped robots, model predictive control, center of mass adaptation, mass of payloads.

I. INTRODUCTION

Many controllers for legged robots [1], [2] are based on their dynamic models [3], to which the inertial parameters are of great importance, especially for those performing high dynamic locomotion [4]. For simplification, some motion controllers [5] of legged robots is based on the center of mass (CoM) trajectory planning of them [6], [7], with the assumption that the CoM locates at the geometrical center of the robot trunk [2], [8]. Although the hardware components are designed to distribute uniformly using the computer aided design (CAD) software in advance, additional factors, such as, different configurations of mission payloads, make it almost impossible for the CoM of the prototype to coincide with its geometrical center.

Inaccurate estimation of the CoM position is easy to cause the zero moment point (ZMP) running out of the support region for bipedal robots [9], [10], and thus, leads to instability. Although quadruped robots have larger stability margin than bipeds in most cases, the model errors introduce

The associate editor coordinating the review of this manuscript and approving it for publication was Okyay Kaynak [ID].

heavy disturbances to the locomotion. The stability criteria can be severely affected by the CoM offset for quadruped robots in static walking gait [11], [12] and tumbling may happen. In some cases, the CoM error leads to asynchronous leg motions and impulsive contact forces, which result in attitude oscillation and even mechanical damages [13]. Quadratic optimization is commonly used to distribute contact forces based on the virtual model control (VMC) [14] or the model predictive control (MPC) [15], [16]. The CoM offset causes state tracking errors of the robot using this kind of controller, mainly including the lateral speed error and the attitude error.

There are many factors that can affect the CoM position of quadruped robots, such as the transformation of the robot configuration, the reconstitution of different function modules, and the operating motions of the mission payloads, in which case the total mass of the robot maintains constant. Other cases exist whenever the amount and loading position of the payloads change, which have effects both on the robot mass and CoM.

The errors between the CAD model and engineering prototype bring inertial parameter errors to the robots.

Stephens [17] designed a state estimator for the humanoid robot in the presence of unknown modelling errors, but did not allow the COM offset and the external force disturbances simultaneously. Muscolo et al. [18] presented an approach to minimize the CoM error between the ideal virtual model and the real platform of the humanoid robot using force/torque sensors mounted on the feet, but did not introduce the reflecting method for payload changes. Focchi et al. [19] designed a method for the quadruped robot HyQ [20] to estimate the CoM position offset from its CAD model by collecting numerous sets of sensor feedback in static poses. Alai et al. [21] designed an estimator based on Kalman Filter to estimate the CoM position and velocity of the humanoid robot. But this method is only valid when the angular acceleration and the speed of the robot are small.

In some cases, the quadruped robot suffers the time-varying CoM. This can be caused by the incidents such as the reconfiguration of hardware components, the changes of payloads, and the consumption of the fuel. When the robot stands statically, the support region surrounded by its four legs provides large stability margin, which results in the tolerance to CoM changes. The wheel-legged robot CENTAURO [22] carried 17 Kg payloads with its dual arms in static pose. But for robots running in dynamic gaits, the stability margin decreases and the CoM variations need to be taken into consideration and compensated into the controllers.

Some researchers focus on the adaptation to payload changes of legged robots. Kim et al. [23] developed a ZMP control methods for a passenger-carrying humanoid robot that coped with variable passenger weights. Tournois et al. [24] proposed an online payload identification method for quadruped robots walking in static gait. However, the forward locomotion of the robot was suspended in the case of payload changes and resumed after the adaptations and adjustments were accomplished.

To overcome the shortcomings above, this article devotes to the adaptive control method for quadruped robots with unknown CoM offsets and payload changes. The real CoM position is estimated based on the state deviation of the robot, which implements a MPC based controller. Further, the mass of payload is calculated in the case of payload changes. Both the mass and CoM offsets are compensated to the robot controller. The novelty of this work states that, it proposes a locomotion controller for the quadruped robot while taking the real CoM into consideration, with which the robot realizes online self-adaptation to real time CoM changes. Besides, the forward locomotion of the robot is not interrupted whenever the CoM changes.

This article is organized as follows. Section II briefly introduces the robot and its simulation model. Section III presents the model and control architecture for the robot with the compensation for CoM offsets. The CoM estimation and payload identification method is described in Section IV. Section V shows the effectiveness of the proposed algorithm through simulations. Section VI gives the conclusion.

II. SYSTEM OVERVIEW
The hydraulic actuated heavy-duty quadruped robot Scalf-III was released by our group in 2018, which was designed for payload transportation tasks and had interfaces for onboard mission units to execute specific tasks. The robot is mainly composed of the power unit, the mechanical unit, the electronic control unit, and the perception unit. There are three joints in each leg, one for adduction and abduction and two for flexion and extension. This work is based on the simulation model which has the same topological structure and parameters as the Scalf-III, as shown in Fig. 1. The main dimension and inertial parameters are listed in Table 1. In the mechanical structure design of legged robots, light-weight actuators and materials are usually used to construct the legs so that the legs are as light as possible. For Scalf-III, the mass of the legs is pretty small comparing to that of the trunk, and thus we neglect the mass of the legs in the simulation model.
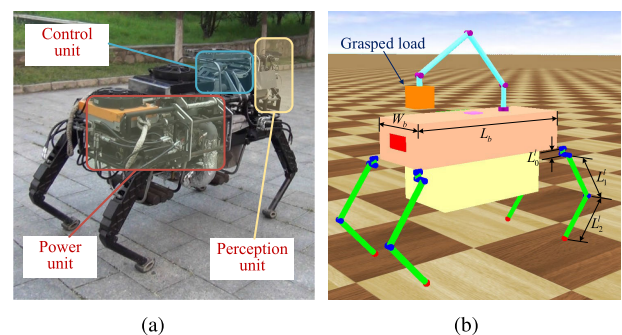

FIGURE 1. The prototype and simulation model of the quadruped robot, (a) the prototype of the Scalf-III, (b) the simulation model of the Scalf-III.

TABLE 1. Main parameters of the quadruped robot and the simulation model.

Specifications		Symbols /Units	Values	
			prototype	model
body	mass	m_r / Kg	275	300
	length	L_b / m	1.36	1.36
	width	W_b / m	0.57	0.57
leg	mass	m_{leg} / Kg	15	1
	upper link length	L_0^l / m	0.045	0.045
	middle link length	L_1^l / m	0.451	0.451
	lower link length	L_2^l / m	0.405	0.405

III. MODELING WITH CoM ADAPTATION
Most controllers for quadruped robots regard the geometrical center of the trunk as the robot CoM, but it is far from the truth for the Scalf-III. As shown in Fig. 1(a), the components of the robot, especially the power unit, are asymmetric and unstructured, which makes it impossible that the CoM coincides with the robot center. For some quadruped robots used for goods transportation, the changes of the payload mass and position may cause variations of the CoM. In some cases, quadruped robots are used as mobile platforms with mission units such as detection equipments on them. The reconfiguration of

mission payloads may cause sporadic CoM changes of the robot.

The CoM offset introduce disturbances to the system. Fig. 2 shows the mechanism of the effects. $F_i = [F_{i,x}, F_{i,y}, F_{i,z}]^T$ is the contact force of the standing leg. The identifier $i = f, h$ stands for the front leg and the hind leg. $r_{com} = [c_x, c_y, c_z]^T$ is the position of the CoM with regard to the base frame, namely the CoM offset. The x_{diag}-axis passes through the two diagonal contact points. When the robot knows nothing about the CoM offset, it regards the body center as the CoM, and regulates the locomotion while taking the body center as the force bearing point. In this case, the robot regards the condition in which the body center locates in the vertical plane passing through x_{diag}-axis as the balancing state. However, once the true CoM deviates from the body center, the resultant moment generated by contact forces and the gravity around x_{diag}-axis is nonzero since the arm of the gravity $r_g \neq 0$. This nonzero resultant moment leads the robot to tip towards one corner, which conflicts with the requirement that the robot should maintain the horizontal body posture. In this case, the robot losses either the accuracy of its posture or the precision of other states, unless the controller compensates for the CoM offset based on the value of the CoM.
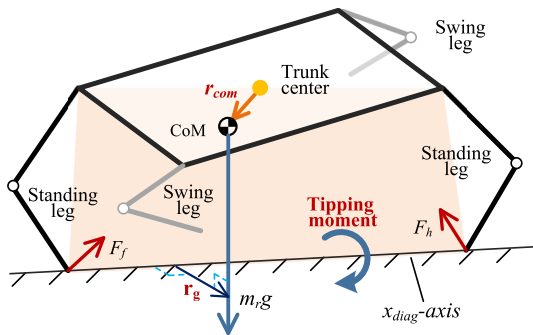

FIGURE 2. The influence of the CoM offset to the robot.

An intuitive way to compensate for the CoM offset is to adjust the positions of the standing feet to make the gravity vector pass through the connecting line of the contact points. In this way, the tipping moment generated by the gravity becomes zero. The adjustments on the sagittal plane and coronal plane are shown in Fig. 3. c_x and c_y are the horizontal components of the CoM. Since the tipping moment does not vary when the CoM shifts in the vertical direction, the z component of the CoM offset is neglected in this work.

The adaptability to CoM variations without interrupting the robot locomotion is essential for quadruped robots. This section designs the trot gait controller for the robot while compensating the CoM offset to the standing phase and swing phase, respectively.

A. MODELING OF STANDING PHASE

For the quadruped robot running in trot gait with the duty factor of 0.5, the robot has two diagonal legs standing on the
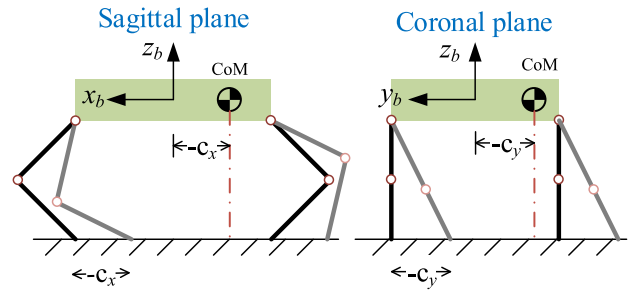

FIGURE 3. Adaptation of standing legs to the CoM offset. The legs in black color denote the positions of standing legs before adjustments, the gray ones denote the positions after adjustments, and the swing legs are not shown.
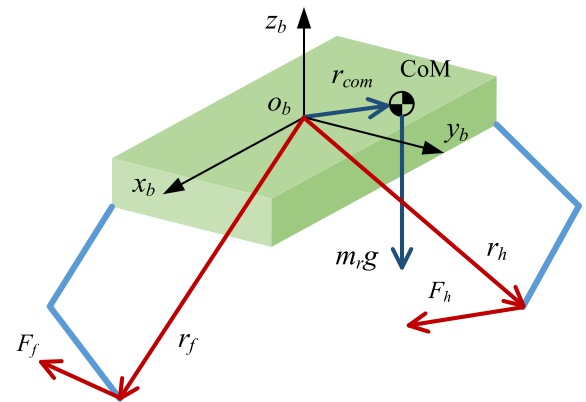

FIGURE 4. Dynamic model of the trunk in standing phase.

ground which provide driving powers. The dynamic model of the robot trunk in standing phase is depicted in Fig. 4. $\{O_b\}$ is the base frame locating at the center of the trunk. $r_i = [x_i, y_i, z_i]^T$ is the moment arm of the standing leg. g is the gravitational acceleration.

Different from most of the existing methods, this work builds the rigid body dynamic model of the robot trunk at the real CoM instead of at the origin of the base frame. The CoM offset is compensated to the arms of the contact forces as in

$$\begin{cases} \dfrac{\mathrm{d}}{\mathrm{d}t}\Theta = \omega \\ \dfrac{\mathrm{d}}{\mathrm{d}t}(I\omega) = \sum_{i=f,h}(r_i - r_{com}) \times F_i \\ \dfrac{\mathrm{d}}{\mathrm{d}t}\dot{P} = \dfrac{1}{m_r + m_{pyl}}\sum_{i=f,h}F_i + G \end{cases} \quad (1)$$

where $\Theta = [\gamma, \theta, \varphi]^T$ and $P = [x, y, z]^T$ are the attitude angle and the position of the trunk, respectively. I is the inertia tensor of the trunk. ω is the angular velocity. m_{pyl} is the mass of payloads. The pseudo roll angle γ and pseudo pitch angle θ with respect to the base frame are used in the dynamic model, which helps avoid global state observations. To achieve rough terrain adaptation, the absolute roll angle α and pitch angle β, which represent the topography, are introduced into the gravity compensation term $G = [g\cos\alpha\sin\beta, -g\sin\alpha\cos\beta, -g\cos\alpha\cos\beta]^T$.

The onboard manipulator in Fig.1(b) is mounted only for the purpose of simulating the situation in which the robot suffers time-varying CoM positions, and it is not involved in the dynamic model of the robot.

When the robot maintains the stable orientation, the term $\omega \times (I\omega)$ does not contribute significantly to the dynamics of the robot comparing to the term $I\dot{\omega}$ [19], [25], [26], which means $\frac{d}{dt}(I\omega) = \omega \times (I\omega) + I\dot{\omega} \approx I\dot{\omega}$. Encapsulating the states and inputs with the state vector $X = [\Theta^T, P^T, \omega^T, \dot{P}^T, G^T]^T$ and the input vector $U = [F_f, F_h]^T$, (1) can be rewritten in the state space formation and discretized with the extremely small time interval Δt. The iterative equation of the system state can be obtained as

$$X(k+1) = A(k)X(k) + B(k)U(k). \tag{2}$$

The system matrix and the input matrix in (2) are calculated as

$$A(k) = \begin{bmatrix} \Lambda_{3\times3} & 0_{3\times3} & \Lambda_{3\times3}\Delta t & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & \Lambda_{3\times3} & 0_{3\times3} & \Lambda_{3\times3}\Delta t & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & \Lambda_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \Lambda_{3\times3} & \Lambda_{3\times3}\Delta t \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \Lambda_{3\times3} \end{bmatrix} \tag{3}$$

$$B(k) = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \\ I^{-1}[r_f - r_{com}]_\times \Delta t & I^{-1}[r_h - r_{com}]_\times \Delta t \\ \Lambda_{3\times3}\Delta t/(m_r + m_{pyl}) & \Lambda_{3\times3}\Delta t/(m_r + m_{pyl}) \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix} \tag{4}$$

where $\Lambda_{3\times3}$ is the identity matrix and $[W]_\times$ indicates the skew-symmetric matrix of the vector W. $A(k) \in \mathbb{R}^{15\times15}$ does not vary with the time. $B(k) \in \mathbb{R}^{15\times6}$ is determined by the desired state of the robot at each sampling instant.

The MPC-based controller takes the state trajectory in the prediction horizon into consideration. The state trajectory is a sequence of the desired states of each iteration, which contains the desired motions of the robot in the prediction horizon, such as, the velocities and the attitude. The robot states can be predicted from the current one, as in

$$\mathbb{X} = \mathbb{A}^p X(0) + \mathbb{B}^p \mathbb{U} \tag{5}$$

where

$$\mathbb{X} = \begin{bmatrix} X(1)^T & X(1)^T & \cdots & X(l)^T \end{bmatrix}^T \tag{6}$$

$$\mathbb{A}^p = \begin{bmatrix} A(0) \\ A(0)^2 \\ \vdots \\ A(0)^l \end{bmatrix} \tag{7}$$

$$\mathbb{B}^p = \begin{bmatrix} B(0) & 0_{15\times6} & 0_{15\times6} & 0_{15\times6} \\ A(0)B(0) & B(1) & 0_{15\times6} & 0_{15\times6} \\ \vdots & \vdots & \ddots & 0_{15\times6} \\ A(0)^{l-1}B(0) & A(0)^{l-2}B(1) & \cdots & B(l-1) \end{bmatrix} \tag{8}$$

The force distribution of the standing legs are transferred into a quadratic programming (QP) problem with the constraints of the friction corn, as in

$$\min \ (\mathbb{X} - \mathbb{X}_{ref})^T Q (\mathbb{X} - \mathbb{X}_{ref}) + \mathbb{U}^T R \mathbb{U}$$
$$\text{s.t.} \begin{cases} 0 \leq F_{i,z}(k) \leq F_{zmax} \\ |F_{i,x}(k)| \leq \mu F_{i,z}(k) \\ |F_{i,y}(k)| \leq \mu F_{i,z}(k) \end{cases} \tag{9}$$

where $Q \in \mathbb{R}^{15l\times15l}$ and $R \in \mathbb{R}^{6l\times6l}$ are diagonal positive definite matrices indicating the weights of each state and input. μ is the frictional coefficient between the foot and the ground, which depends on the material of the foot and the terrain. F_{zmax} is the maximum output force of the leg in the z direction that the actuators can achieve.

The dimensions of $\mathbb{A}^p$ and $\mathbb{B}^p$ in (5) are determined by the length of the prediction horizon l that $\mathbb{A}^p \in \mathbb{R}^{15l\times15}$ and $\mathbb{B}^p \in \mathbb{R}^{15l\times6l}$. The larger value of l can improve the robustness of the system, since the controller finds input forces for the legs for more future steps. However, this will increase the scale of the QP problem, which leads to longer calculation time, and thus the lag of the system. Through each solving process of the QP problem, the controller gets a sequence of the input forces for the future l steps as $\mathbb{U} = [F_f(0), F_h(0), \cdots, F_f(l-1), F_h(l-1)]^T$, although only $F_f(0)$ and $F_h(0)$ are practically used as the desired forces at the current time and new prediction will be execute with the updated state $X(0)$ and reference state trajectory $\mathbb{X}_{ref}$.

B. FOOT TRAJECTORY OF SWING PHASE

The landing point of the swing phase is also the starting point of the standing phase. In order to realize the adjustment shown in Fig. 3, the landing point is compensated in x and y directions, as in

$$\begin{bmatrix} x_{land} \\ y_{land} \end{bmatrix} = \frac{T}{4} \begin{bmatrix} \dot{x}_d \\ \dot{y}_d \end{bmatrix} + K_{\dot{v}} \begin{bmatrix} \dot{x} - \dot{x}_d \\ \dot{y} - \dot{y}_d \end{bmatrix} + c_{x,y} \tag{10}$$

where x_{land} and y_{land} are the landing position and T is the gait cycle. $\dot{x}_d$ and $\dot{y}_d$ are the desired velocities of the trunk and $K_{\dot{v}} = diag(k_{\dot{x}}, k_{\dot{y}})$ is the coefficient matrix. $c_{x,y} = [c_x, c_y]^T$. The foot trajectory is designed as a curve cluster that varies with the robot state, as in

$$\begin{cases} x_{sw}(k+1) = x_{sw}(k) + \dfrac{x_{land} - x_{sw}(k)}{0.5T - t}\Delta t_{sw} \\ y_{sw}(k+1) = y_{sw}(k) + \dfrac{y_{land} - y_{sw}(k)}{0.5T - t}\Delta t_{sw} \\ z_{sw}(t) = -H_b + h_{step}\sin(2\pi t/T) \end{cases} \tag{11}$$

where t is the passed time of the current swing phase and Δt_{sw} is the servo interval. H_b and h_{step} are the stand height and step height of the robot. The swing legs are controlled to follow the designed trajectory from the inspiration of virtual model control (VMC), in which a three dimensional virtual spring-damping model is fixed on the foot. The details are not elaborated in this work for the sake of brevity.

IV. CoM ESTIMATION AND PAYLOAD IDENTIFICATION

This section analyzes the influence of the CoM offset quantitatively and qualitatively, and then gives the estimation method. With the approximation that $\frac{d}{dt}(I\omega) \approx I\dot{\omega}$, the Newton-Euler dynamic model of the trunk in (1) can be rewritten as

$$
\begin{bmatrix} I\ddot{\Theta} \\ m(\ddot{P}-G) \end{bmatrix} = \underbrace{\begin{bmatrix} [r_f - r_{com}]_\times & [r_h - r_{com}]_\times \\ \Lambda_{3\times3} & \Lambda_{3\times3} \end{bmatrix}}_{D} \begin{bmatrix} F_f \\ F_h \end{bmatrix}. \tag{12}
$$

where $m = m_r + m_{pyl}$. There are six variables with six force inputs in (12). However, the coefficient matrix $D \in \mathbb{R}^{6\times6}$ is not a full rank matrix. Generally, the rank of D is five. When $z_f = z_h$, the rank of D reduces to four, but this is practically impossible since the moment arms use the feedback positions of standing legs and have noises and errors. To solve this problem, the control of robot motions in a certain axis (usually the y-axis) is abandoned in some researches. For the MPC based control approach in this work, the solution of the QP problem finds a balance for the robot performance in all the six degrees of freedom through appropriate weight coefficient in the matrix $Q = diag(q_1(1), \cdots, q_{15}(1), q_1(2), \cdots, q_{15}(2), \cdots, q_1(l), \cdots, q_{15}(l))$.

A. ESTIMATION OF c_y

The CoM is unknown for the controller at the initial moment, and is set to be zero. It is an ideal situation that when the robot maintains upright attitude, the torque applied by the contact forces around x_b-axis satisfies

$$
T_x = -z_f F_{f,y} - z_h F_{h,y} + y_f F_{f,z} + y_h F_{h,z}. \tag{13}
$$

However, as discussed above, the CoM offset is difficult to avoid. Once the CoM shifts from the trunk center, the additional torque applied to the x_b-axis by the gravity is

$$
\Delta T_{x,c} = -m_r g c_y. \tag{14}
$$

The intuitionistic influence of this extra torque is that the trunk tends to generate a nonzero angular acceleration to break the stable pose. The weigh matrix Q in the QP problem determines the priorities of each robot state. To ensure the safety and stability of the robot, the maintaining of upright attitude usually has higher priority than the speed tracking performance. That is, the attitude angles are given more weights than the velocities in the MPC controller. To remedy the deviation of the roll angle, the robot will apply additional forces to the standing feet to satisfy that

$$
\begin{aligned}
T_x' &= \Delta T_{x,c} - z_f(F_{f,y} + \Delta F_{f,y}) - z_h(F_{h,y} + \Delta F_{h,y}) \\
&\quad + y_f(F_{f,z} + \Delta F_{f,z}) + y_h(F_{h,z} + \Delta F_{h,z}) = 0 \tag{15}
\end{aligned}
$$

where T_x' is the resultant moment applied on the x_b-axis by the contact forces and the gravity. Being restricted to the limitation that $F_{f,z} + F_{h,z} = m_r g$, the adjustable range of the contact forces in z direction is very small ($\Delta F_{f,z} \approx 0$ and

$\Delta F_{h,z} \approx 0$). Substituting (13) into (15), it can be obtained that

$$
-z_f \Delta F_{f,y} - z_h \Delta F_{h,y} = m_r g c_y. \tag{16}
$$

During the locomotion of quadruped robots, the height of the trunk H_b is commonly expected to be a constant value, which means $z_f = z_h = -H_b$. Thus, an instantaneous acceleration of the trunk in the y_b-axis appears with the value approximating to gc_y/H_b. Note that, as long as the controller is not completely compensated for the CoM, the lateral acceleration exists. Thus, the y component of the CoM position is estimated based on the errors between the feedback and desired lateral velocities, in a recursive fashion as

$$
\tilde{c}_y(k) = \tilde{c}_y(k-1) + k_{py}^c(\dot{y} - \dot{y}_d) + k_{dy}^c(\ddot{y} - \ddot{y}_d) \tag{17}
$$

where k_{py}^c and k_{dy}^c are the proportional coefficient and differential coefficient, respectively. The tilde on the top of a symbol means it is the estimated value in this work.

Compensating the controller for the estimated CoM following the way described in Fig. 3, $\tilde{c}_y$ is added to the level arms and (11) becomes

$$
T_x'' = -z_f F_{f,y}' - z_h F_{h,y}' + (y_f + \tilde{c}_y)F_{f,z} + (y_h + \tilde{c}_y)F_{h,z}. \tag{18}
$$

Once the estimated CoM converges to its true value, the item $(\tilde{c}_y F_{f,z} + \tilde{c}_y F_{h,z})$ equals to $m_r g c_y$, which appropriately balances the extra gravitational moment in (14). In this case, the contact forces in y-axis $F_{f,y}'$ and $F_{h,y}'$ resume their values when there is no CoM offset, namely $\Delta F_{f,z} = \Delta F_{h,z} = 0$. In this approach, the acceleration and velocity error in y direction converge to zero as well as the estimated CoM converges to its true value.

B. ESTIMATION OF c_x

Similar to the case of y direction, the CoM offset in x direction generates an unexpected torque around the y_b-axis that is unable to be balanced by the contact forces. The quadruped robot has larger stability margin in its longitudinal dimension than that in the lateral direction, and thus lower sensitivity to the CoM error. In practical applications, the movements of quadruped robots prefer to fore-and-aft locomotion rather than sidewise. So, the robots require higher precision of speed tracking performance in the x_b direction. With exorbitant large weight for the pitch angle, the robot tends to produce attitude oscillations when runs across rough terrains. Based on the considerations above, the controller assigns more weight for the tracking of roll angle than that for the y velocity in the lateral direction. But this emphasis is less obvious along the x_b-axis. It leads to different effects on the robot by the CoM offset, and thus different appearance of the robot states. The longitudinal CoM offset leads the robot to nod and raise its head repeatedly rather than the speed error laterally.

From this inspiration, the real x position of the CoM is estimated based on the inconformity of the contact forces of the front and hind legs. The value of the identifier is

determined by the difference between the contact forces as in

$$
S_{flag} = \begin{cases} 0, & \eta < F_{f,z} - F_{h,z} < \xi \\ \text{sgn}(F_{f,z} - F_{h,z}), & \text{otherwise} \end{cases} \quad (19)
$$

where $\xi > 0$ and $\eta < 0$ are thresholds for judgements. The comparison of the contact forces is valid only when the two standing legs both touch the ground, which represents

$$
\begin{cases} J_f^{-T}(q)\tau_f \geq F_f^{th} \\ J_h^{-T}(q)\tau_h \geq F_h^{th} \end{cases} \quad (20)
$$

where τ_f and τ_h are joint torques. F_f^{th} and F_h^{th} are thresholds. To avoid misjudgment caused by disturbances, a slide window is introduced, with short length to reduce the hysteresis. Along the window horizon, the identifiers are synthesized into the variable N, which is calculated as

$$
N_k = N_{k-1} + S_{flag} \quad (21)
$$

where k stands for the sampling sequence. The recursive estimation of the CoM position in x direction is

$$
\tilde{c}_x(k) = \tilde{c}_x(k-1) + k_{px}^c N_k + k_{dx}^c(N_k - N_{k-1}) \quad (22)
$$

where k_{px}^c and k_{dx}^c are the proportional coefficient and differential coefficient, respectively. Similar to the case of the lateral direction, $\tilde{c}_x$ is also compensated into the level arms of the contact forces. Along with the convergence of $\tilde{c}_x$ to its true value, the pitch angle oscillation of the robot trunk is weakened.

C. MASS OF PAYLOAD

There are two major situations that lead to the variation of the CoG position of quadruped robots. One is that the onboard manipulators execute motions while maintaining the total mass of the robot constant. Another case is that the payloads are added (*loading*) or cut down (*unloading*). Subjecting to the limitation of stable standing height, the variation of the robot mass has no obvious influence on the robot locomotion on flat terrain. However, when encountering slops, the inaccurate mass makes the reference values of contact forces cannot match their actual demands. The identification of the payload mass is essential. Based on the prior estimated CoM $\tilde{c}_{x,y}$, once the payload is loaded or unloaded, the CoM position drifts to

$$
\tilde{c}^{new} = \begin{cases} \dfrac{m_r \tilde{c}_{x,y} + m_{pyl} c_{pyl}}{m_r + m_{pyl}}, & \text{loading} \\[2mm] \dfrac{m_r \tilde{c}_{x,y} - m_{pyl} c_{pyl}}{m_r - m_{pyl}}, & \text{unloading} \end{cases} \quad (23)
$$

where c_{pyl} is the position of the payload with respect to the base frame. The payloads are places in the cabin which is designed at a fixed place, and thus c_{pyl} is known and maintains constant. m_r can be obtained from the contact forces when the robot stands in a static pose initially, and updated with the payload mass afterwards. The variation tendency (*load/unload*) is detected based on the comparison of the

contact forces between different sampling periods when the acceleration of the robot in the vertical direction closes to zero. By collecting a vast array of data sets, the payload mass is solved in the least square (LS) fashion, as in

$$
\min \sum_{j=1}^{n} \left[m_{pyl}(\tilde{c}_j^{new} - c_{pyl}) - m_r(\tilde{c}_{x,y} - \tilde{c}_j^{new}) \right]^2
$$
$$
\text{s.t. } m_{pyl} \begin{cases} > 0, & \text{load} \\ < 0, & \text{unload} \end{cases} \quad (24)
$$

where the subscript j means the j-th sampling. n is the scale of the QP problem.

V. SIMULATIONS AND RESULTS

To verify the effectiveness of the proposed method, simulations are executed in the simulator Webots R2018a based on the robot model described in Section II. The robot runs in trot gait with the duty factor of 0.5 and the step frequency of 1.5 Hz, which means that there are two and only two diagonal legs in contact with the ground (the standing phase) at any moment, while the other two legs are in the air. Major situations that cause the CoM errors are considered, including the design error, the reconfiguration of mission units, and the change of payloads. Note that, in this section, the CoM with respect to the base frame is constant unless when the manipulator moves or when the payload changes. In all cases, the CoM is time-varying with respect to the world coordinate frame, because the robot keep moving.

A. CoM ESTIMATION WITH PREDEFINED REFERENCE VALUES

The CoM position of the robot can be assigned manually by users in Webots. The quadruped robot is ordered to walk in situ. In order to evaluate the accuracy of the CoM detection results, the CoM of the robot is set to be at the position $r_{com} = [0.15m, -0.08m, 0.0m]^T$ with respect to the base frame, which is regarded as the real value. The estimation result is shown in Fig. 5. The estimated CoM $\tilde{c}_r$ converges to the real robot CoM within three seconds with the errors less than 0.6 cm in x direction and 0.5 cm in y direction. The step character of the x direction result is caused by the inherent dead zone of the estimation mechanism in (19) $\sim$ (22). To test the robustness of the proposed method to the external disturbances, both lateral and longitudinal impacts are exerted to the robot in this simulation. The robot is knocked by a 50 Kg pendulum bob, which falls from the horizontal position to
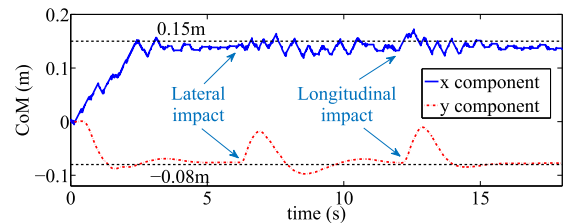
FIGURE 5. CoM estimaiton results.

the bottom with the pendulum length of 0.7 m. The estimated CoM returns to the real value within three seconds. The CoM adaptation method does not conflict with the path planning of the swing foot. As shown in Fig. 6, the oscillation of the roll and pitch angles of the robot trunk is remarkably decreased with the CoM adaptation approach.
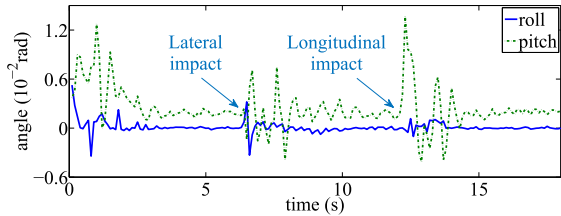

FIGURE 6. Roll and pitch angles of the robot.

B. RELIABILITY ANALYSIS

Repeated simulations are conducted with different predefined CoMs, and the variances of the estimation results are collected for one minute in the convergence horizon. The five-number summary [27] is visualized in Fig. 7. The outliers are divorced not far from the box, which indicates the reliability of the approach. The errors of the median values are within 10% of the reference, which are small enough and do not cause deviations of robot states.
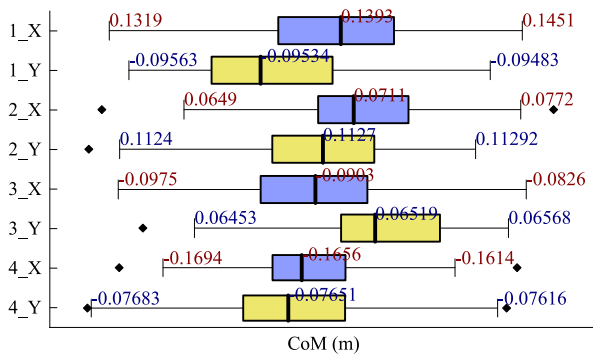

FIGURE 7. Distribution of the estimated CoM in the convergence horizon. The identifier _X at the y-axis indicates the result of $\bar{c}_x$ and _Y indicates $\bar{c}_y$. Repeated simulations with different reference CoMs are included, e.g., the name 1_ stands for the reference value $c_{x,y} = [0.15\text{ m}, -0.1\text{m}]^T$, 2_ for $c_{x,y} = [0.08\text{ m}, 0.12\text{ m}]^T$, 3_ for $c_{x,y} = [-0.1\text{ m}, 0.07\text{ m}]^T$, 4_ for $c_{x,y} = [-0.18\text{ m}, -0.08\text{ m}]^T$.

C. SENSITIVITY ANALYSIS

The CoM in x direction is estimated based on the lateral velocity, which is obtained by the differential of the foot position, and $\tilde{c}_y$ is calculated relying on the force feedbacks. To evaluate the sensitivity to sensor noises, Gaussian noises with different characteristics, namely mean (μ) and standard deviation (σ) (listed in Table 2), are set to the foot positions and the contact forces. Since the sensors are usually calibrated offline, the mean of the noise is set fixed, and different standard deviations are considered. The estimation results in the convergence horizon are analyzed based on the five-number summary, as shown in Fig. 8. The length of whiskers are

TABLE 2. Sensitivity of the method to force and position noises.

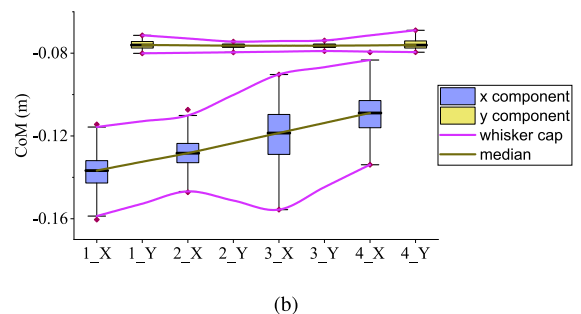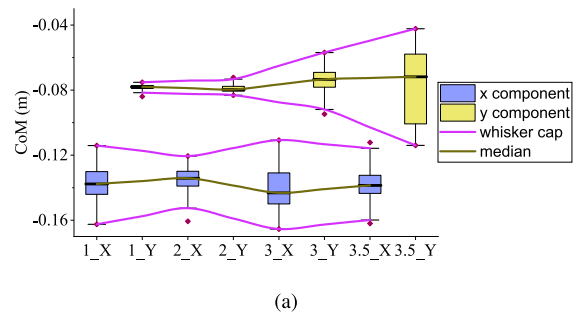| Force Noise/10^2N	$\mu=1,\sigma=1$	$\mu=1,\sigma=2$	$\mu=1,\sigma=3$	$\mu=1,\sigma=4$
$[\lVert\varepsilon_x\rVert,\lVert\varepsilon_y\rVert]$/cm	[1.33,0.39]	[2.17,0.46]	[3.14,0.36]	[4.11,0.39]
$[\lVert\varepsilon_x^{wsk}\rVert,\lVert\varepsilon_y^{wsk}\rVert]$/cm	[3.43,0.86]	[3.99,0.56]	[5.97, 0.61]	[6.67,1.11]
Position Noise/mm	$\mu=0,\sigma=1$	$\mu=0,\sigma=2$	$\mu=0,\sigma=3$	$\mu=0,\sigma=3.5$
$[\lVert\varepsilon_x\rVert,\lVert\varepsilon_y\rVert]$/cm	[1.23,0.20]	[1.58,0.04]	[0.64,0.66]	[1.14,0.82]
$[\lVert\varepsilon_x^{wsk}\rVert,\lVert\varepsilon_y^{wsk}\rVert]$/cm	[3.59,0.48]	[2.94,0.67]	[3.92,2.31]	[3.42,3.77]

(a)

(b)

FIGURE 8. Sensitivity to different force and position noises, (a) convergence horizon of the estimation results with position noises, the names at the x-axis encodes the noise characteristics and the estimated components, e.g., the name 1_X stands for the x component in the case of $\sigma_p = 1$ mm, (b) convergence horizon of the estimation results with force noises, the name 1_Y at the x-axis stands for the y component in the case of $\sigma_f = 100$ N.

set as 1.5 times the inter-quartile range. Table 2 reports the convergence errors and boundary errors in each cases. The convergence error is used to evaluate the accuracy of the results, defined as the distance between the median estimated value and the true value

$$\begin{cases} \lVert\varepsilon_x\rVert = |c_x - \tilde{c}_x| \\ \lVert\varepsilon_y\rVert = |c_y - \tilde{c}_y| \end{cases}. \quad (25)$$

The dispersion of the results in the convergence horizon is evaluated by the boundary error, which is defined as the maximum of the distances from the upper and lower whisker edges to the true value, as in

$$\begin{cases} \lVert\varepsilon_x^{wsk}\rVert = \max\left(\left|\tilde{c}_x^{up} - c_x\right|, \left|\tilde{c}_x^{lo} - c_x\right|\right) \\ \lVert\varepsilon_y^{wsk}\rVert = \max\left(\left|\tilde{c}_y^{up} - c_y\right|, \left|\tilde{c}_y^{lo} - c_y\right|\right) \end{cases} \quad (26)$$

where $\tilde{c}_x^{up}$ and $\tilde{c}_y^{up}$ are the values of the upper whisker edges, and $\tilde{c}_x^{lo}$ and $\tilde{c}_y^{lo}$ are the values of the lower whisker edges.

D. ADAPTATION TO TIME-VARYING CoMs

Although the real value of the CoM can be designated manually in simulation, it is difficult to know the real CoM of a robot prototypes, which is also the necessity of this work. In this section, we use a lightweight onboard manipulator which moves with the 50 Kg load at the end-effector to simulate the case in which the robot suffers time-varying CoMs, which may be caused by the reconfiguration of its hardware components, the changes of payloads and the consumption of the fuel. The end-effector is programmed to pass through five waypoints following the sequence of ⓪-①-②-③-④-⓪, as shown in Fig. 9.
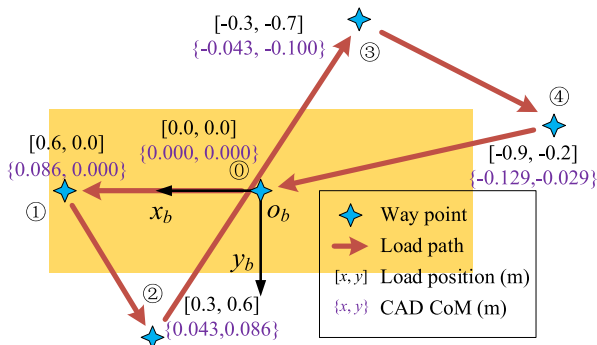
FIGURE 9. Trajectory of the grasped load.

The robot is ordered to run in trot gait at different speeds, in both x and y directions. Once the load arrives at a new waypoint, the robot takes five seconds to detect and adapt to the shifted CoM, and then the load moves to the next waypoint taking three seconds. At each waypoint, the CAD software SolidWorks is used to simulate the CoM of the combination of the robot and the grasped goods, which is regarded as the true value. The trajectory of the end-effector is limited to the horizontal plane. The waypoint and the corresponding true CoM values are shown in Fig. 9. Fig. 10 shows the CoM estimation results during the entire process. The estimated CoM value converges to the CAD model value at each waypoint, with the error less than 15% in the x direction and less than 10% in the y direction.
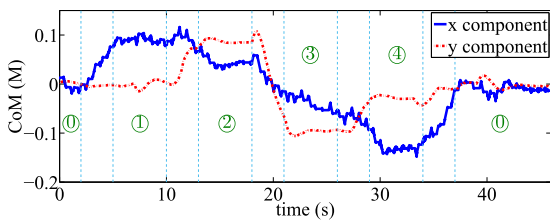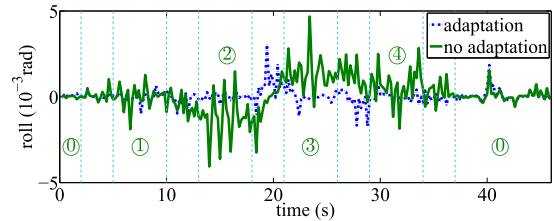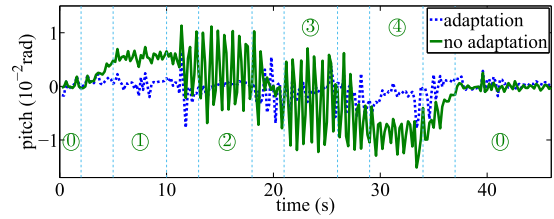
FIGURE 10. CoM estimation results when the robot suffers time-varying CoMs.
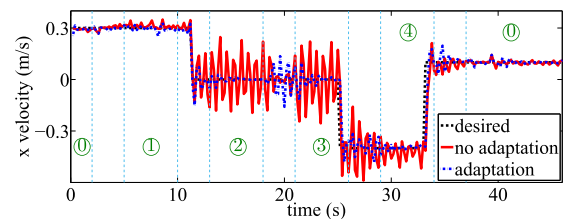
Fig. 11 shows the comparisons of the major robot states with and without the CoM adaptation approach. The oscillations and errors of the roll and pitch angles are significantly reduced when the CoM adaptation method is applied,
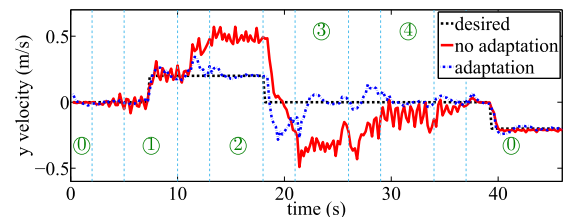
FIGURE 11. Comparisons of the robot states with/without the proposed CoM adaptation approach when the robot suffers time-varying CoMs, (a) roll angle of the robot trunk, (b) pitch angle of the robot trunk, (c) velocity in x direction, (d) velocity in y direction.

as shown in Fig. 11(a) and 11(b), which denotes that the robot possesses higher stability. The velocity deviations in Fig. 11(c) and 11(d) are remarkably decreased and the robot runs following the desired commands more precisely with the proposed method. Fig. 12 shows the comparison of contact forces in the period of $t \in [2.0\,\text{s} \sim 6.0\,\text{s}]$, during which c_x varies from zero to a positive value. The results show that when the proposed CoM adaptation method is applied, the force discrepancy between the two standing legs is remarkably decreased and the legs act more harmoniously. Fig. 13 consists of the snapshots of the robot during this simulation, showing that the locomotion is uninterrupted while adapting to the changed CoM, and the adjustments of legs can be seen. Since the dynamics of the manipulator is not involved in the robot controller, the movement velocity of the end-effector is limited to relatively low values, but it is enough for simulating the situation that the robot suffers
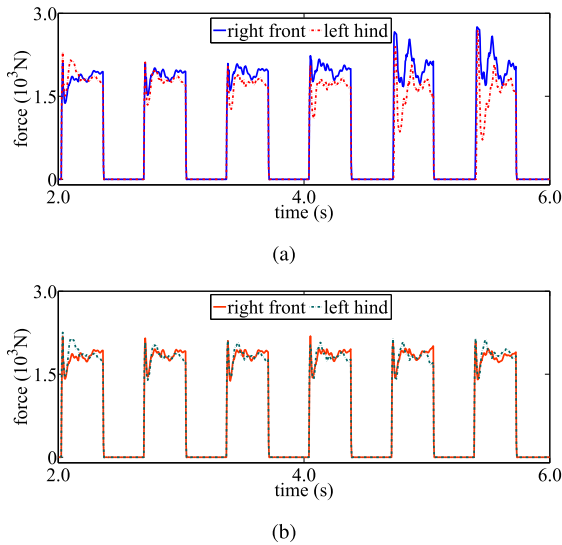
(a)

(b)

FIGURE 12. Vertical components of contact forces of the right front and left hind legs, (a) the proposed method is not applied, (b) the proposed method is applied.
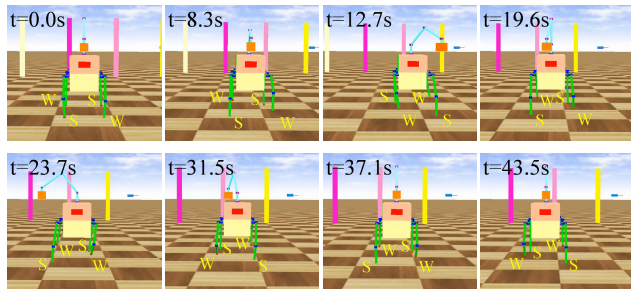

FIGURE 13. Snapshots of the simulation when the robot suffers time-varying CoMs. The phases of the legs are marked beside them, as *S* indicates the leg is in the standing phase and *W* indicates the swing phase.

time-varying CoM. When the manipulator implements high dynamic motions, it is possible that the ZMP runs out of the support region of the robot, and thus causes tumbles.

E. MASS IDENTIFICATION OF PAYLOADS

Loading of the payload is simulated in this section, with different predefined CoM positions (listed in Table 3) of the quadruped robot. The robot starts to run in situ and the payload is loaded at the position of [0.51 m, 0.20 m] (coordinates in the horizontal plane with respect to the base frame) after the estimated CoM $\tilde{c}_r$ tends to be stable. Payloads with different

mass (20 Kg, 40 Kg, 60 Kg, 80 Kg and 100 Kg) are tested in each situation. Once the robot has already adapted to the newly changed CoM, data sets are collected by one thousand continuous sampling to constitute the over determined system of equations to identify the mass of payloads. The identification results are listed in Table 3. The errors between the estimated mass and the real mass are mostly within 8 %, and the biggest error is 12.23 %. Since the robot never knows the true value of the its CoM, once the estimated value $\tilde{c}_{x,y}$ reaches its convergence horizon (the fluctuation maintains within a certain small range), it is adopted by the robot as the real CoM and employed by the controller. The mass estimation errors are mainly introduced by the error of the CoM estimation. The result shows that, the quadruped robot can accommodate to payload changes while its locomotion is not interrupted, as long as the actuators can provide sufficient forces.

VI. CONCLUSION

Most of the state-of-the-art CoM estimation methods for quadruped robot either are offline or require specific trunk poses, and thus lead to interruption of robot locomotion. This work proposed the online CoM adaptation and payload identification method for quadruped robots in trot gait based on the error of velocities and the discrepancy of contact forces. Multi-factors that may cause CoM offsets are involve in simulations. It is proved through the analysis of accuracy, reliability, and sensitivity, that the estimation results satisfy the demands of the state tracking performance of the robot. With the proposed method, the adaptation to payload changes is also realized. In all the cases above, the dynamic trot gait locomotion of the robot is not interrupted. CoM estimation approaches may vary with different robot controllers, and this work is conducted on the MPC-based one. However, it can be expanded to all controllers that are based on the rigid body Newton-Euler dynamic model of the trunk combining with the optimization theory, which has inherent under actuated characteristics.

TABLE 3. Mass identification results of the payloads.

Reference/Identified CoM		Payload Mass / Kg				
c_{CAD} / cm	$\tilde{c}_{x,y}$ / cm	20	40	60	80	100
[0.00, 0.00]	[0.15, 0.01]	19.13	37.14	52.66	72.22	91.17
[9.00, -7.00]	[8.23, -6.58]	20.06	38.39	55.65	74.02	92.22
[-10.00, 4.00]	[-9.05, 3.74]	17.94	37.16	55.70	71.79	92.90
[-15.00, -8.00]	[-14.05, -7.56]	20.09	38.45	56.82	75.80	95.17

REFERENCES

[1] X. Chen, Z. Yu, W. Zhang, Y. Zheng, Q. Huang, and A. Ming, "Bioinspired control of walking with toe-off, heel-strike, and disturbance rejection for a biped robot," *IEEE Trans. Ind. Electron.*, vol. 64, no. 10, pp. 7962–7971, Oct. 2017.

[2] X. Wang, H. Liu, B. Liang, X. Wang, and J. Yang, "Locomotion control for quadruped robot combining central pattern generators with virtual model control," in *Proc. IEEE 15th Int. Conf. Control Autom. (ICCA)*, Jul. 2019, pp. 399–404.

[3] R. Grandia, F. Farshidian, R. Ranftl, and M. Hutter, "Feedback MPC for torque-controlled legged robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 4730–4737.

[4] T. Li, L. Zhou, Y. Li, H. Chai, and K. Yang, "An energy efficient motion controller based on SLCP for the electrically actuated quadruped robot," *J. Bionic Eng.*, vol. 17, no. 2, pp. 290–302, Mar. 2020.

[5] Y.-D. Hong, C.-S. Park, and J.-H. Kim, "Stable bipedal walking with a vertical center-of-mass motion by an evolutionary optimized central pattern generator," *IEEE Trans. Ind. Electron.*, vol. 61, no. 5, pp. 2346–2355, May 2014.

[6] M. Liu, D. Qu, F. Xu, F. Zou, P. Di, and C. Tang, "Quadrupedal robots whole-body motion control based on centroidal momentum dynamics," *Appl. Sci.*, vol. 9, no. 7, pp. 1–17, 2019.

[7] A. Winkler, I. Havoutis, S. Bazeille, J. Ortiz, M. Focchi, R. Dillmann, D. Caldwell, and C. Semini, "Path planning with force-based foothold adaptation and virtual model control for torque controlled quadruped robots," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 6476–6482.

[8] G. Zhang, S. Ma, F. Liang, and Y. Li, "Quadruped locomotion control based on two bipeds jointly carrying model," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1732–1738.

[9] T. Kamioka, H. Kaneko, T. Takenaka, and T. Yoshiike, "Simultaneous optimization of ZMP and footsteps based on the analytical solution of divergent component of motion," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 1763–1770.

[10] Y.-D. Hong and B. Lee, "Real-time feasible footstep planning for bipedal robots in three-dimensional environments using particle swarm optimization," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 1, pp. 429–437, Feb. 2020.

[11] S. Ma, T. Tomiyama, and H. Wada, "Omnidirectional static walking of a quadruped robot," *IEEE Trans. Robot.*, vol. 21, no. 2, pp. 152–161, Apr. 2005.

[12] B. Chen, S. Li, and D. Huang, "Quadruped robot crawl gait planning based on DST," in *Proc. 33rd Chin. Control Conf.*, Jul. 2014, pp. 8578–8582.

[13] C. Ding, L. Zhou, X. Rong, and Y. Li, "Online center of mass detection for quadruped robots in trot gait motion," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2019, pp. 2140–2145.

[14] C. Gehring, S. Coros, M. Hutter, M. Bloesch, M. A. Hoepflinger, and R. Siegwart, "Control of dynamic gaits for a quadrupedal robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 3287–3292.

[15] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, "Dynamic locomotion in the MIT cheetah 3 through convex model-predictive control," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 1–9.

[16] J. Huang, H. An, Y. Yang, C. Wu, Q. Wei, and H. Ma, "Model predictive trajectory tracking control of electro-hydraulic actuator in legged robot with multi-scale online estimator," *IEEE Access*, vol. 8, pp. 95918–95933, 2020.

[17] B. J. Stephens, "State estimation for force-controlled humanoid balance using simple models in the presence of modeling error," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 3994–3999.

[18] G. G. Muscolo, C. T. Recchiuto, C. Laschi, P. Dario, K. Hashimoto, and A. Takanishi, "A method for the calculation of the effective center of mass of humanoid robots," in *Proc. 11th IEEE-RAS Int. Conf. Humanoid Robots*, Oct. 2011, pp. 371–376.

[19] M. Focchi, A. del Prete, I. Havoutis, R. Featherstone, D. G. Caldwell, and C. Semini, "High-slope terrain locomotion for torque-controlled quadruped robots," *Auto. Robots*, vol. 41, no. 1, pp. 259–272, Jan. 2017.

[20] N. G. T. C. Semini, E. Guglielmino, M. Focchi, F. Cannella, and D. G. Caldwell, "Design of hyq—A hydraulically and electrically actuated quadruped robot," *Proc. Inst. Mech. Eng. I, J. Syastems Control Eng.*, vol. 225, no. I6, pp. 831–849, 2011.

[21] H. Alai, F. A. Shirazi, and A. Yousefi-Koma, "New approach to center of mass estimation for humanoid robots based on sensor measurements and general LIPM," in *Proc. 6th RSI Int. Conf. Robot. Mechatronics (IcRoM)*, Oct. 2018, pp. 388–393.

[22] N. Kashiri, S. Cordasco, P. Guria, A. Margan, N. G. Tsagarakis, L. Baccelliere, L. Muratore, A. Laurenzi, Z. Ren, E. M. Hoffman, M. Kamedula, G. F. Rigano, and J. Malzahn, "CENTAURO: A hybrid locomotion and high power resilient manipulation platform," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1595–1602, Apr. 2019.

[23] J.-H. Kim, J.-Y. Kim, and J.-H. Oh, "Weight-adaptive walking of the passenger-carrying biped robot, HUBO FX-1," in *Proc. 10th IEEE-RAS Int. Conf. Humanoid Robots*, Dec. 2010, pp. 33–39.

[24] G. Tournois, M. Focchi, A. Del Prete, R. Orsolino, D. G. Caldwell, and C. Semini, "Online payload identification for quadruped robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 4889–4896.

[25] O. Villarreal, V. Barasuol, P. M. Wensing, D. G. Caldwell, and C. Semini, "MPC-based controller with terrain insight for dynamic legged locomotion," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 2436–2442.

[26] G. Bledt, P. M. Wensing, and S. Kim, "Policy-regularized model predictive control to stabilize diverse quadrupedal gaits for the MIT cheetah," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 4102–4109.

[27] J. Faigl and P. Čížek, "Adaptive locomotion control of hexapod walking robot for traversing rough terrains with position feedback only," *Robot. Auto. Syst.*, vol. 116, pp. 136–147, Jun. 2019.
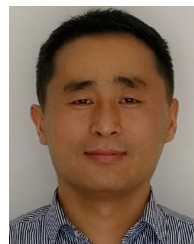
CHAO DING received the bachelor's degree from the Huazhong University of Science and Technology, China, in 2012. He is currently pursuing the Ph.D. degree with the School of Control Science and Engineering, Shandong University, China. His research interests include bionic robotics and control systems.

LELAI ZHOU (Member, IEEE) received the B.Eng. degree in mechanical engineering and automation from the University of Science and Technology Beijing, China, in 2006, the M.Eng. degree in mechanical design and theory from Beihang University, China, in 2009, and the Ph.D. degree in mechanical engineering from Aalborg University, Denmark, in 2012. He is currently an Associate Professor with the School of Control Science and Engineering, Shandong University, China. His main research interests include legged robot, robotic arm, and human-centered design.

YIBIN LI (Member, IEEE) received the B.S. degree in automation from Tianjin University, China, in 1982, the M.S. degree in electrical automation from the Shandong University of Science and Technology, China, in 1990, and the Ph.D. degree in automation from Tianjin University, in 2008. From 1982 to 2003, he has worked with the Shandong University of Science and Technology. Since 2003, he has been the Director of the Center for Robotics, Shandong University. His research interests include bionic robots, control of intelligent robots, machine learning, and intelligence architecture.

XUEWEN RONG (Member, IEEE) received the bachelor's and master's degrees from the Shandong University of Science and Technology, China, in 1996 and 1999, respectively, and the Ph.D. degree from Shandong University, China, in 2013. He is currently a Professor with the School of Control Science and Engineering, Shandong University. His research interests include bionic robotics, mechatronics, hydraulic servo transmission, and control systems.

• • •