



ORIGINAL PAPER

Zhongqu Xie · Long Li · Xiang Luo

Optimization of the ground reaction force for the humanoid robot balance control

Received: 3 April 2020 / Revised: 6 July 2020 / Accepted: 9 June 2021 / Published online: 12 August 2021
© The Author(s), under exclusive licence to Springer-Verlag GmbH Austria, part of Springer Nature 2021

Abstract This paper proposes an effective framework to control humanoid-ground physical interaction for balance. The key concept is to control both humanoid robot linear and angular momenta, which are computed by the whole body kinematics. We firstly determine the ground reaction force (GRF) according to the momenta and then distribute the GRF to the two feet. After determining the foot GRFs, we calculate the joint torques to realize the foot GRFs. Herein, we present an effective method to set the appropriate foot GRFs and transform these forces into full-body joint torques. It operates as a force control process. The joint redundancy is solved by optimizations and constraints. Besides, we modify the classical contact model for the legged robot system. The modified contact model can provide the scientific ground reaction force. We test the controller using our full-sized humanoid robot emulational platform, including compliant balance and squatting, which demonstrates the effectiveness of the method.

1 Introduction

Humanoid robots are designed for serving humans in a large variety of tasks. To achieve an elegant and effective performance, the robot needs to adapt to variations of environments, such as shaky ground [1]. Therefore, the humanoid robot's balance in uncontrollable situations is a problem that must be solved before the humanoid robot is stepping out from the laboratory to the complex external environment.

The need for a balancing system arises from the requirement that the humanoid be capable of performing on a similar level as that of humans. Until recently, a lot of balance control techniques have attempted to maintain balance by controlling only the linear motion of a humanoid. For instance, Hyon proposed a method to calculate the joint torques to control the location of the robot's center of mass (CoM) [2]. When the CoM leaves the target position, a net GRF is determined by a simple CoM state feedback law to bring the CoM back to the target position. In fact, the rotational motion of the robot around the CoM plays an essential role in the balance. For example, when a human is walking or running, his hands and feet often move in opposite directions to each other to ensure that the angular momentum of the torso is stable. The importance of angular momentum in humanoid was firstly reported by Raibert, who is one of the founders of Boston Dynamics, in 1986, when he still was the director of MIT Leg Lab [3]. However, it went unnoticed for a long time. In 2004, Popovic performed a physiological experiment on human walking and found that the neural center of the human body intentionally regulated the angular momentum of human to maintain near-zero value while walking, which suggested that angular momentum may play a significant role in human behavior [4, 5].

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s00707-021-03025-1>.

Subsequently, different balance control strategies based on momentum have been proposed. Komura et al. [6] presented a simplified model named the angular momentum pendulum model, which augments the well-known linear pendulum model [1] with an additional capability of possessing centroidal angular momentum. Uemura et al. [7] designed a switching rule of standing or stepping according to the angular momentum around ankle. Orin et al. [8] put forward the concept of spatial momentum, divided the momenta into linear and angular momenta, and proposed the centroid momentum matrix. Macchietto et al. [9] also divided robot momenta into linear and angular momenta and employed the centroid momentum matrix to determine the joint accelerations for balance. Besides, Twan et al. [10] elaborated a momentum-based control framework for the humanoid robot "Atlas". They computed the joint acceleration vector with the limitations of GRF lying on a quadratic program. Hence, controlling the motion of the robot from the view of the momentum has become popular in recent years.

According to the basic D'Alembert's principle, the resultants of all external forces and moments are equal to the change rates of the momenta. Hence, for a force control task, it can be translated into a momentum control issue. In such a task, two levels are distinct within the control process: (i) designing the humanoid-ground interaction force, and (ii) generating the joint torques to realize the interaction force. Although the external forces and moments have a one-to-one relation with the momentum change rates, the actual operation for the task is complicated and is worth discussing. One of the issues that should be solved is how to handle redundant degrees of freedom (DoF).

For determining the GRFs applied to the feet, Hyon et al. [2] presented a method to minimize the sum of the squared norm of forces at some contact points on the boundary of each foot while satisfying the desired net GRFs. Years later, they applied the method to unknown rough terrain and nonhorizontal ground [11]. Besides, they succeeded to control a hydraulic humanoid robot maintaining balance under the interference of external force [12]. Sentis et al. [13] used a virtual-linkage model to precisely determine GRFs in a more general setting of multi-contact interaction (net GRF) between the humanoid robot and the environment. However, the determination of net GRF increases the complexity of the system, and it also may generate a large ankle torque to keep the foot from sliding and turning. Herein, we control the foot GRFs and center of pressure (CoP) by directly controlling the ankle torques for avoiding the feet sliding and turning.

It is also important to ascertain the relationship between GRFs and joint torques properly. Starting from the early work of Vukobratovic et al. [14], they have developed the relationship between the action point of GRF, i.e., ZMP, and joint motion. Kajita et al. [15] introduced a method using the pseudo-inverse of the inertia matrix to generate the whole body motion based on the linear and angular momentum. Jamisola Jr et al. [16] used the relative Jacobian matrix to control standing dual arms robot. Similarly, Ramamoorthy et al. [17] also computed joint torques by statics equivalence between joint torques and GRF. The statics equivalence is widely used for force interaction control of manipulators. It should be pointed out that replacing dynamics with statics only works in a non-high velocity situation. Lee et al. [18] defined the balance control objectives in terms of CoM and CoP, and adopted the centroidal momentum matrix to determine the joint accelerations, followed by calculating the joint torques using inverse dynamics. Determining the joint torques using centroidal momentum can also be seen in [19, 20]. However, the centroidal momentum differentiation matrix is quite complicated due to the high redundancy of the humanoid robot.

In this paper, we firstly specify the desired GRF according to the momentum states. Due to some constraints, such as the CoPs cannot run outside the feet's support domain, i.e., the feet cannot flip or slide, and GRFs need to be naturally unilateral, the desired GRFs are not always reliable. Then, we obtain the admissible GRF under each foot sole to achieve the desired GRF as close as possible. Herein, we neglect the weight of the feet and assume that the forces and torques applied to the ankles are equivalent to the GRFs. In such a process, we can directly determine admissible foot GRFs without using conventional and complicated net GRF of the robot. In the next step, we establish a dynamic equivalent relationship between joint torques and GRFs such that we computed necessary joint torques to create admissible GRFs to maintain balance. For model reality, we also present a contact model which can be applied to the shanking and moving ground. We highlight our work with various demonstrations of humanoid balance maintenance and squatting.

The main contributions and novelties of this paper are listed as follows:

1. A scientific method is proposed to determine the physically available foot GRFs for balance.
2. Dynamic mapping of GRFs to joint torques with constraints is established.
3. An effective and reliable ground contact model is developed, and it includes the main relevant contact phenomena that will affect the dynamic behavior of the robotic system.

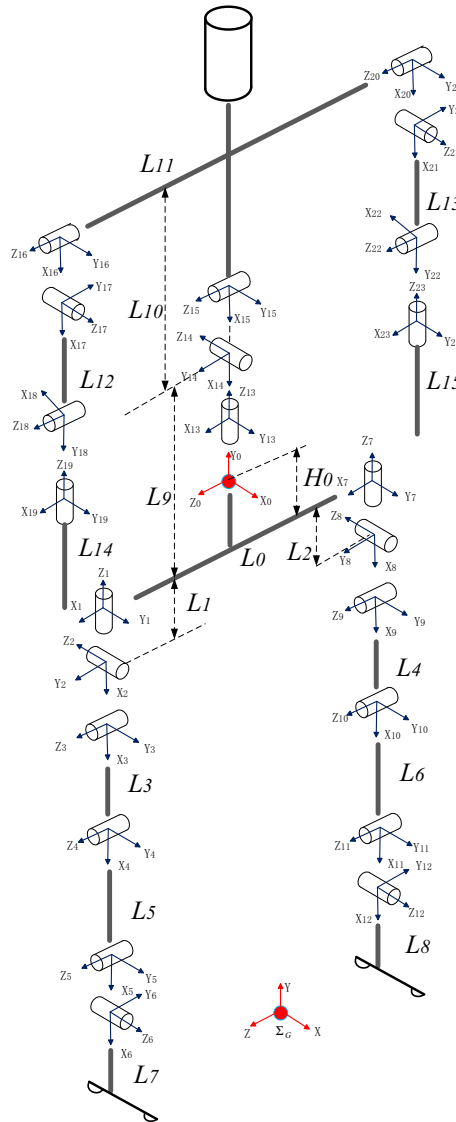


Fig. 1 Kinematic modelling

2 The momentum-based balance controller of the humanoid robot

2.1 Centroidal dynamics of a humanoid robot

The CoM is the valid location of the robot’s total mass and the point where the resultant gravity force acts. Hence, it takes a unique role in a humanoid robot in its dynamics. A humanoid robot can be simplified as a tree-structure topology with $N + 1$ links and N joints. Each link has 6° of freedom, corresponding precisely to the 6 dimensions of momentum.

Here, we adopt a humanoid robot model developed by ourselves. The height of the humanoid robot is 1.34 m, and its mass is about 40 kg. It has a total of 23 DoFs. Moreover, the freedom degree distribution of the robot is: 2 for each shoulder, 2 for each elbow, 3 for the waist, 3 for each hip, 1 for each knee, and 2 for each ankle. It is designed to have similar geometrical and dynamical parameters to those of a teenager, who is about 14 years old. Figure 1 and Table 1 gather the simplified structure and physical parameters of the biped. All the links motion states can be obtained by kinematics referring to the ground coordinate system Σ_G .

Table 1 Physical parameters of the humanoid robot

Link	L_0	L_1, L_2	L_3, L_4	L_5, L_6	L_7, L_8	L_9	L_{10}	L_{11}	L_{12}, L_{13}	L_{14}, L_{15}	H_0
Length/m	0.2	0.133	0.35	0.282	0.051	0.133	0.538	0.447	0.31	0.295	0.009
Mass/kg	1.862	4.025	3.305	1.385	0.236	5.019	8.907	0.312	2.113	1.334	

The homogeneous transformation matrix of each link CoM concerning the ground coordinate system is [1]:

$$\mathbf{T}_j = \begin{bmatrix} \mathbf{R}_j & \mathbf{c}_j \\ \mathbf{000} & 1 \end{bmatrix} \quad (1)$$

where $\mathbf{R}_j \in \mathbb{R}^{3 \times 3}$ and $\mathbf{c}_j \in \mathbb{R}^{3 \times 1}$ are the rotation matrix and the position vector of the j -th link, respectively. We also can obtain the relationship between the velocity ($\mathbf{v}_j, \boldsymbol{\omega}_j$) of the CoM of the j -th link and the velocity vector $\dot{\mathbf{q}}$ by kinematics:

$$\begin{bmatrix} \mathbf{v}_j \\ \boldsymbol{\omega}_j \end{bmatrix} = \mathbf{J}_j \dot{\mathbf{q}} \quad (2)$$

where \mathbf{J}_j is the Jacobian matrix of the j -th link. The total momentum of the robot composed of N connecting links is:

$$\mathbf{P} = \begin{bmatrix} \mathbf{L} \\ \mathbf{A} \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^N m_j \mathbf{v}_j \\ \sum_{j=1}^N \mathbf{I}_j^{cm} \boldsymbol{\omega}_j \end{bmatrix} \quad (3)$$

where m_j is the mass of the j -th link, and \mathbf{I}_j^{cm} is a 3×3 j -th link rotational inertia matrix about the CoM of the robot:

$$\mathbf{I}_j^{cm} = \mathbf{R}_j \bar{\mathbf{I}}_j \mathbf{R}_j^T + m_j \mathbf{S}(\mathbf{c}_G - \mathbf{c}_j) \mathbf{S}^T(\mathbf{c}_G - \mathbf{c}_j) \quad (4)$$

where $\bar{\mathbf{I}}_j$ represents the inertia tensor of the j -th link for its local coordinate system, $\mathbf{c}_G = \sum_{j=1}^N m_j \mathbf{c}_j / m_{tot}$ is the CoM of the robot, m_{tot} is the total mass of the robot, and the quantity $\mathbf{S}(\mathbf{p})$ is the skew-symmetric matrix that satisfies $\mathbf{S}(\mathbf{p}) \mathbf{!} = \mathbf{p} \times \mathbf{!}$ for any 3D vector $\mathbf{!}$.

Substituting (2) into (3), the robot's total momentum can be obtained as follows:

$$\mathbf{P} = \mathbf{I}(\mathbf{q}) \mathbf{J}(\mathbf{q}) \dot{\mathbf{q}} \quad (5)$$

where the Jacobian matrix $\mathbf{J}(\mathbf{q}) \in \mathbb{R}^{6N \times N}$ is the combination of the Jacobian matrix of each link:

$$\mathbf{J} = \left[\mathbf{J}_1^T, \mathbf{J}_2^T, \dots, \mathbf{J}_j^T, \dots, \mathbf{J}_N^T \right]^T, \quad (6)$$

and $\mathbf{I}(\mathbf{q})$ is the 6×6 N inertia matrix:

$$\mathbf{I} = \begin{bmatrix} \mathbf{M}_1 & \mathbf{0} & \mathbf{M}_2 & \mathbf{0} & \dots & \mathbf{M}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_1^{cm} & \mathbf{0} & \mathbf{I}_2^{cm} & \dots & \mathbf{0} & \mathbf{I}_1^{cm} \end{bmatrix} \quad (7)$$

where \mathbf{M}_j is a 3×3 mass diagonal matrix. Rin et al. [8] defined the system momentum matrix $\mathbf{N} = \mathbf{I} \mathbf{J}$. The system momentum matrix is the product of the inertia matrix and the system Jacobian matrix, and its dimension is $6 \times N$. Therefore, we can rewrite (5) as follows:

$$\mathbf{P} = \mathbf{N} \dot{\mathbf{q}}. \quad (8)$$

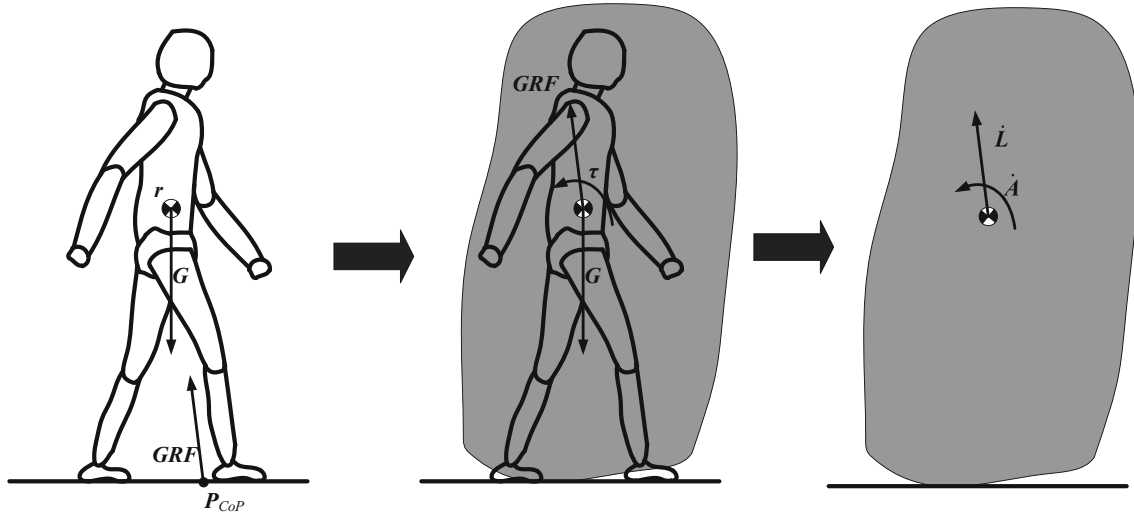


Fig. 2 Centroidal dynamics of the humanoid robot

2.2 Balance control of the robot based on centroidal momentum

Regarding the robot as a whole mass, as shown in Fig. 2, all the external forces can be turned into a force and a moment, or equivalently, the linear and angular momentum change rates, \dot{L} and \dot{A} , at the CoM [8].

Here, we can define a PD controller based on the CoM and momentum states so that it realizes the desired position and velocity of the CoM:

$$\begin{cases} \dot{L}_d = K_{p1}(\mathbf{c}_{c,d} - \mathbf{c}_G) + K_{d1}(\dot{\mathbf{c}}_{c,d} - \dot{\mathbf{c}}_G) \\ \dot{A}_d = K_{p2}(\mathbf{A}_{c,d} - \mathbf{A}) + K_{d2}(\dot{\mathbf{A}}_{c,d} - \dot{\mathbf{A}}) \end{cases} \quad (9)$$

where $\mathbf{c}_{c,d}$ and $\dot{\mathbf{c}}_{c,d}$ are the desired CoM location and velocity, respectively, and $\mathbf{A}_{c,d}$ and $\dot{\mathbf{A}}_{c,d}$ are desired CoM momentum states.

2.2.1 Determination of the GRFs

The feet of the robot occupy a small part of the robot's overall mass, so the mass of the robot's feet can be ignored. In this view, the GRF can be regarded as the force and moment applied to the ankles as shown in Fig. 3. The advantage is that we can control the angular momentum change rate of the robot by determining the positions, forces, and torques of the feet. Also, it can be easy to ensure the CoP in the interior of the foot's support polygon by controlling the ankle torque. And the momentum change rates can be expressed as follows:

$$\begin{aligned} \dot{L} &= \mathbf{f}_r + \mathbf{f}_l, \\ \dot{A} &= \dot{A}_f + \dot{A}_\tau, \\ \dot{A}_f &= (\mathbf{c}_r - \mathbf{c}_G) \times \mathbf{f}_r + (\mathbf{c}_l - \mathbf{c}_G) \times \mathbf{f}_l, \\ \dot{A}_\tau &= \boldsymbol{\tau}_r + \boldsymbol{\tau}_l \end{aligned} \quad (10)$$

where \mathbf{f}_r and \mathbf{f}_l are the GRFs acting on the right and left foot, respectively, \mathbf{c}_r and \mathbf{c}_l represent the positions of the ankles, and $\boldsymbol{\tau}_r$ and $\boldsymbol{\tau}_l$ are the torques of the right and left ankles.

To ensure stationary contacts, the CoP at each foot needs to be within the foot's support polygon. Therefore the moment $\boldsymbol{\tau}_{l,r} = [\tau_{l,r-x}, \tau_{l,r-y}, \tau_{l,r-z}]^T$ generated by the GRF and applied to the ankle can be expressed as follows:

$$\boldsymbol{\tau}_{l,r} = (\mathbf{c}_{l,r} - \mathbf{c}_{l,r-CoP}) \times \mathbf{f}_{l,r} \quad (11)$$

where $\mathbf{c}_{l,r-CoP}$ is the CoP of the left or right foot.

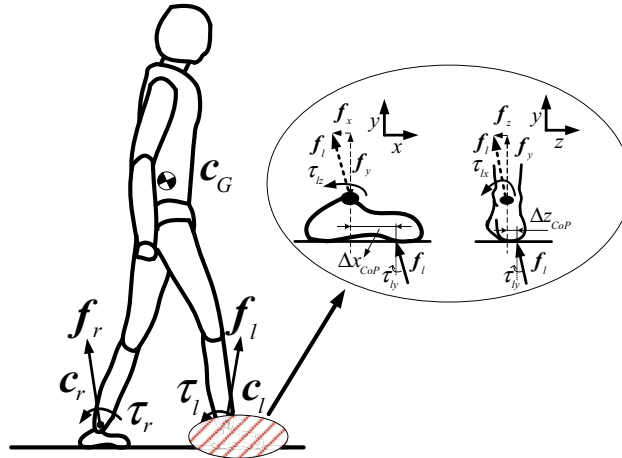


Fig. 3 Translation of the GRFs to the ankles

Except for gravity, the change of the momentum is caused by a combination of GRFs at the left and right foot. After determining \dot{L} and \dot{A} , we need to distribute the GRF to the left and right foot. The allocation rule can be accomplished by the optimization criteria as follows:

$$\begin{aligned} \min & \|\dot{L}_d - \dot{L}(f_r, f_l)\|^2 + w_k \|\dot{A}_d - \dot{A}(f_r, f_l)\|^2 + w_f (\|f_r\|^2 + \|f_l\|^2), \\ \text{s.t.} & \mathbf{F}_{l,r-n} \geq 0 \text{ and } \arctan \frac{|\mathbf{F}_{l,r-t}|}{\mathbf{F}_{l,r-n}} \leq u_s \end{aligned} \quad (12)$$

where $\dot{L}(f_r, f_l)$ and $\dot{A}(f_r, f_l)$ are described in (10), and $\dot{A}(f_r, f_l)$ is only related to the GRFs at left and right foot, which equals \dot{A}_f in (10). $\mathbf{F}_{l,r-n}$ and $\mathbf{F}_{l,r-t}$ are the normal support force and tangential friction force, respectively. Note that, once either left or right foot lifts off the ground, the GRF only applies to the standing support leg.

The reason why we did not add the ankle torque item in (12) is that it would be difficult to get a real-time optimal solution due to the nonlinearity of cross-product terms in ankle torque item [18]. Moreover, the benefit of ignoring the ankle torque item is to minimize the ankle torques as possible such that it would be easier to assure the CoPs within the foot's support polygon and avoid foot slipping.

2.2.2 Determination of foot CoPs and desired ankle torques

Actually, the foot GRFs are difficult to satisfy both the desired linear and angular momentum change rates. The difference value $\dot{A}_{d,\tau} = \dot{A}_d - \dot{A}_f$, needs to be generated by ankle torques. Furthermore, ankle torques should have upper and lower bound after determining the GRFs to avoid flipping. The CoP of each sole can be solved by an optimization problem as follows:

$$\begin{aligned} \min & \|\dot{A}_{d,\tau} - (\tau_l(\mathbf{c}_{l_CoP}) + \tau_r(\mathbf{c}_{r_CoP}))\|^2 \\ \text{s.t.} & \mathbf{r}_{l,r_CoP} \in [\mathbf{l}_{heel}, \mathbf{l}_{toe}] \end{aligned} \quad (13)$$

where $\tau_l(\mathbf{c}_{l_CoP})$ and $\tau_r(\mathbf{c}_{r_CoP})$ are described in (11). Note that: there is no yaw degree of freedom at the ankle, $\theta_{l,r-y}$ is offset by the hip. Then the desired ankle torques can be determined by substituting the optimized GRFs in (12) and the optimized CoPs in (13) into (11).

2.2.3 Determination of joint torques

In Sect. 2.1, the spatial centroidal momentum $\mathbf{P} = [\mathbf{L}^T \ \mathbf{A}^T]^T$ is expressed. And the relationship between spatial centroidal momentum and generalized joint velocities is expressed: $\mathbf{P} = \mathbf{N}\dot{\mathbf{q}}$. In [8, 9, 18], they took a derivative of spatial centroidal momentum $\dot{\mathbf{P}} = \mathbf{N}\ddot{\mathbf{q}} + \dot{\mathbf{N}}\dot{\mathbf{q}}$, and replaced $\dot{\mathbf{P}}$ referring to $[\dot{\mathbf{L}} \ \dot{\mathbf{A}}]^T$ in (9). And then they obtained the joint accelerations which satisfied the constraint $\mathbf{J}\ddot{\mathbf{q}} + \dot{\mathbf{J}}\dot{\mathbf{q}} = 0$. Finally, it is easy

to compute the torques by performing inverse dynamics. Unfortunately, the robot is a multibody mechanism. Thus, the spatial centroidal momentum differential equation is quite complicated such that it would be difficult to real-time compute the joint accelerations. Besides, in the floating dynamics, there are no actuating rotation joints equipped at the CoM (or the base) to control the robot's attitude angle. The simple substitution of GRFs and joint accelerations makes it possible to appear the robot attitude torques, which is physically impossible to achieve. We develop a different approach, with computing joint torques directly instead of the centroidal momentum differential equation.

Let $c_G = [x_c, y_c, z_c]^T \in \mathbf{R}^3$ be the position vector of the CoM in the world coordinate frame. Let $\Omega = [\theta_x, \theta_y, \theta_z]^T \in \mathbf{R}^3$ be the attitude of the robot. Thus, the generalized coordinates are described with the vector $\mathbf{x} = [c_G^T, \Omega^T, \mathbf{q}^T]^T$, and the dynamic model of the floating-base robot is established as follows:

$$\mathbf{D}(\mathbf{x})\ddot{\mathbf{x}} + \mathbf{C}(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}} + \mathbf{G}(\mathbf{x}) = \mathbf{B}\boldsymbol{\tau} + \mathbf{J}(\mathbf{x})^T \mathbf{F}, \quad (14)$$

With the constraint equations to define the contact between the stance feet and ground:

$$\mathbf{J}(\mathbf{x})\dot{\mathbf{x}} = 0, \quad \dot{\mathbf{J}}(\mathbf{x})\dot{\mathbf{x}} + \mathbf{J}(\mathbf{x})\ddot{\mathbf{x}} = 0 \quad (15)$$

where $\mathbf{D}(\mathbf{x}) \in \mathbf{R}^{(n+6) \times (n+6)}$ is a positive definite inertia matrix, $\mathbf{C}(\mathbf{x}, \dot{\mathbf{x}}) \in \mathbf{R}^{(n+6) \times (n+6)}$ contains the centrifugal and Coriolis matrix, $\mathbf{G}(\mathbf{x}) \in \mathbf{R}^{n+6}$ is the gravity term, \mathbf{B} is the constant transformation matrix, $\boldsymbol{\tau}$ represents joint torques, $\mathbf{J}(\mathbf{x}) \in \mathbf{R}^{m \times (n+6)}$ is the Jacobian constraint matrix associated with the contact, and vector $\mathbf{F} \in \mathbf{R}^m$ is the contact force, i.e., GRFs and moment applied at the left and right foot soles. Note that $\mathbf{J}(\mathbf{x})$ is the composition matrix of the left and right foot sole Jacobian matrices. $m = 6$ or 12 depending on the robot with single support or double support.

Since there is an unknown joint angular acceleration term $\ddot{\mathbf{x}}$ in (14), the joint torques cannot be calculated directly with the contact force \mathbf{F} . Considering that the inertia matrix is generally symmetric positive definite, that is, a generally invertible matrix, we multiply $\mathbf{J}\mathbf{D}^{-1}$ from the left-hand side of (14):

$$\mathbf{J}\ddot{\mathbf{x}} + \mathbf{J}\mathbf{D}^{-1}\mathbf{C}\dot{\mathbf{x}} + \mathbf{J}\mathbf{D}^{-1}\mathbf{G} = \mathbf{J}\mathbf{D}^{-1}\mathbf{B}\boldsymbol{\tau} + \mathbf{J}\mathbf{D}^{-1}\mathbf{J}^T \mathbf{F}, \quad (16)$$

$$\mathbf{J}\mathbf{D}^{-1}\mathbf{B}\boldsymbol{\tau} = \mathbf{J}\ddot{\mathbf{x}} + \mathbf{J}\mathbf{D}^{-1}\mathbf{C}\dot{\mathbf{x}} + \mathbf{J}\mathbf{D}^{-1}\mathbf{G} - \mathbf{J}\mathbf{D}^{-1}\mathbf{J}^T \mathbf{F}. \quad (17)$$

Substituting the contact constraints mentioned in (15) into the above equation, we can get:

$$\frac{\mathbf{J}\mathbf{D}^{-1}\mathbf{B}\boldsymbol{\tau}}{\mathbf{H}\mathbf{W}} = \frac{-\dot{\mathbf{J}}\dot{\mathbf{x}} + \mathbf{J}\mathbf{D}^{-1}\mathbf{C}\dot{\mathbf{x}} + \mathbf{J}\mathbf{D}^{-1}\mathbf{G} - \mathbf{J}\mathbf{D}^{-1}\mathbf{J}^T \mathbf{F}}{\mathbf{H}\mathbf{W}}, \quad (18)$$

$$\mathbf{H}\boldsymbol{\tau} = \mathbf{W}. \quad (19)$$

$\dot{\mathbf{J}}$ is only related to joint angles and angular velocities, and the sensors can obtain the current states of the joints. There is another advantage that the equation directly contains contact constraint.

Besides, to keep the upper body as upright as possible, the desired moment applied at the torso can be derived using a standard PD control with the extent to the robot falls:

$$\boldsymbol{\tau}_{l_h} - \boldsymbol{\tau}_{r_h} = \frac{\mathbf{K}_{p3}\mathbf{q}_{torso} + \mathbf{K}_{d3}\dot{\mathbf{q}}_{torso}}{\boldsymbol{\tau}_{dh}} \quad (20)$$

$$\mathbf{T}\boldsymbol{\tau} = \boldsymbol{\tau}_{dh} \quad (21)$$

where $\boldsymbol{\tau}_{l_h} \in \mathbf{R}^3$ and $\boldsymbol{\tau}_{r_h} \in \mathbf{R}^3$ are left and right hip torques, respectively, \mathbf{K}_{p3} and \mathbf{K}_{d3} are 3×3 diagonal matrices and represent the parameters of the proportional and differential term, $\mathbf{q}_{torso} \in \mathbf{R}^3$ is the inclination of the torso in the world coordinate frame, and $\mathbf{T} \in \mathbf{R}^{3 \times n}$ is the hip actuation transformation matrix.

Based on known joint states, the relationship between joint torques and contact forces is established. Furthermore, the desired torso moment to keep upright is determined. Hence, we can express the objective function as follow:

$$\begin{aligned} \boldsymbol{\tau}_a &= \min \|\mathbf{H}\boldsymbol{\tau} - \mathbf{W}_a\|^2 + \mathbf{w}_{torso} \|\mathbf{T}\boldsymbol{\tau} - \boldsymbol{\tau}_{dh}\|^2 + \mathbf{w}_\tau \|\boldsymbol{\tau}\|^2, \\ st. \quad \mathbf{q}_{\min} &\leq \mathbf{q} + \dot{\mathbf{q}} \Delta t + \frac{1}{2} \ddot{\mathbf{q}}(\boldsymbol{\tau}) \Delta t^2 \leq \mathbf{q}_{\max} \end{aligned} \quad (22)$$

where Δt is the control period of the system, \mathbf{q}_{\min} and \mathbf{q}_{\max} are the upper and lower limits of the robot joint motion, respectively, and $\ddot{\mathbf{q}}(\boldsymbol{\tau})$ is expressed in the [Appendix](#) in detail.

To avoid numerical errors, we add a kinematic feedback control term. After obtaining the joint torques, the joint accelerations can be expressed:

$$\begin{aligned}\ddot{\mathbf{x}}_a &= \mathbf{D}^{-1}(\mathbf{B}\boldsymbol{\tau}_a + \mathbf{J}^T \mathbf{F}_a - \mathbf{C}\dot{\mathbf{x}} - \mathbf{G}), \\ \ddot{\mathbf{q}}_a &= \mathbf{E}\ddot{\mathbf{x}}_a\end{aligned}\quad (23)$$

where $\ddot{\mathbf{q}}_a$ are the all joint admissible accelerations except those of the floating freedoms, \mathbf{E} is the transformation matrix for selecting the joint accelerations, and the theoretical contact force \mathbf{F}_a is:

$$\mathbf{F}_a = (\mathbf{J}\mathbf{D}^{-1}\mathbf{J}^T)^{-1}[-\dot{\mathbf{J}}\dot{\mathbf{x}} + \mathbf{J}\mathbf{D}^{-1}(\mathbf{C}\dot{\mathbf{x}} + \mathbf{G} - \mathbf{B}\boldsymbol{\tau}_a)].\quad (24)$$

Note that, if \mathbf{J} is a square matrix and reversible, (23) can be simplified by substituting (24) into (23):

$$\ddot{\mathbf{x}}_a = -\mathbf{J}^{-1}\dot{\mathbf{J}}\dot{\mathbf{x}}.\quad (25)$$

Equation (25) is consistent with the constraint equation. If the joint accelerations could be obtained by (25), then joint torques can be computed by performing inverse dynamics, and the joint torques can be acquired directly. Unfortunately, the Jacobian matrix is usually not invertible. Moreover, joint accelerations obtained from the contact constraints can only keep the foot from sliding. The robot is still highly likely to fall down without sliding. Finally, the commanded joint torques are obtainable as:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_d + \boldsymbol{\tau}_k\quad (26)$$

where $\boldsymbol{\tau}_d$ is computed by performing dynamics, and $\boldsymbol{\tau}_k$ is determined by adding feedback terms:

$$\boldsymbol{\tau}_k = \mathbf{K}_p(\mathbf{q}_d - \mathbf{q}) + \mathbf{K}_i \int (\mathbf{q}_d - \mathbf{q})dt + \mathbf{K}_d(\dot{\mathbf{q}}_d - \dot{\mathbf{q}})\quad (27)$$

where \mathbf{K}_p , \mathbf{K}_i , and \mathbf{K}_d are proportional, integral, and differential gain, respectively. The desired position and velocity are both obtained by time integration of $\ddot{\mathbf{q}}_d$.

3 Summary

The overall control procedure can be summarized as follows.

- (P1) Calculate all forward kinematics to obtain the positions and velocities of each part of the robot.
- (P2) Compute the momentum of the CoM by (1–8).
- (P3) Set the desired GRF and moment by (9).
- (P4) Ignore the weight of the feet and regard that the force is applied to the ankle as the GRF.
- (P5) Calculate the admissible GRFs, CoPs, and ankle torques by (10–13).
- (P6) Transform the admissible GRFs to joint torque commands for the active joints by (14–27).

4 Ground contact model

To demonstrate the implementation of the control scheme described in Sect. 2, we need to model the ground and GRF that will be applied to the feet of the biped robot. In this Section, we will introduce a general ground contact model for the biped robot.

4.1 The normal force

We regard the contact surface as a series of contact points. And each contact point consists of a combination of nonlinear spring and damper element in the normal direction such that the relationship between the viscoelastic deformation and the resulting force can be derived. The nonlinear spring and damper, known as the Hunt-Crossley model [21], is expressed as:

$$f_y = k_y|\Delta y|^a + c_y|\Delta y|^a \Delta \dot{y}\quad (28)$$

Table 2 Parameters of the ground contact model

k_y	c_y	ρ	κ
4.5×10^7	9.6×10^5	1.8	1.1

where k_y is the stiffness coefficient, a can be determined by the geometry of the contact surfaces, c_y represents the damping factor, and Δy is the insertion depth of the foot contact point.

Furthermore, the Hunt–Crossley model seems to be physically meaningful: the energy loss caused by the contact bodies' damping which dissipates energy in the form of heat and larger depth means closer contact of the body surface which leads to larger stiffness and damping factor. However, it should be noted that it may generate downward normal forces as long as the velocity away from contact is fast enough. It does not obey the nature of the contact. Once the two contact bodies start to separate, the contact force cannot be the pulling force to prevent them from leaving. Here, we propose a modified normal support force which is expressed as:

$$f_y = \begin{cases} k_y |\Delta y|^\rho & \Delta y \leq 0 \ \& \ \Delta \dot{y} \geq 0 \\ k_y |\Delta y|^\rho + c_y |\Delta y|^\kappa \Delta \dot{y} & \Delta y \leq 0 \ \& \ \Delta \dot{y} < 0 \\ 0 & \Delta y > 0 \end{cases} \quad (29)$$

where ρ and κ describe the influences of penetration depth on the stiffness and damping coefficients. The prime aim we piecewise calculate the normal force function is to simulate unilateral surface constraint between the two contact bodies. And the parameters of the normal contact model are shown in Table 2.

Note that: if the constant step size is adopted in the simulation, the contact force starts being applied at the first time step after the bodies come into contact. As a result, a large contact force may be generated at the beginning of the simulation due to the high stiffness. Hence, to avoid the artificial initial energy, a variable step solver is suggested or the penetration depth at the time the contact is first detected is used as a reference position to calculate a relaxed penetration depth estimate y_r as follows:

$$\Delta y_r = y - y_{r,0} \quad (30)$$

where y is the current normal position of the contact point, $y_{r,0} = -1/2gt^2$ is the penetration depth at the instant the impact is detected, and t represents the single time step.

4.2 Tangential friction force

Unlike the normal force which can be derived using scalar equations, the surface friction force is a vectorial quantity by nature. Canudas de Wit et al. [22] introduced a bristle friction force model. The tangential friction also can be modelled as a linear or nonlinear spring-damper system based on a bristle friction model. However, it is only appropriate for non-sliding or low-velocity condition. In the case of high velocity, it always performs to pull the contact point back to the initial contact position. To make the simulation closer to reality and consistent with the observation by Coulomb, whereby the tangential friction increases linearly with the normal load, we adopt the Stick–Slip friction model [23], as shown in Fig. 4.

The friction force is the product of frictional coefficient and normal force, where the coefficient of friction is a function of the relative velocity at the point of contact. The explicit tangential friction model in the x -direction is:

$$f_x = \begin{cases} -u_s / v_{th} \Delta \dot{x} f_y & \Delta y \leq 0 \ \& \ |\Delta \dot{x}| \leq v_{th} \\ -\left(u_s \operatorname{sgn}(\Delta \dot{x}) - \Delta \dot{x} \frac{(u_s - u_k)}{0.5v_{th}}\right) f_y & \Delta y \leq 0 \ \& \ v_{th} \leq |\Delta \dot{x}| \leq 1.5v_{th} \\ -u_d \operatorname{sgn}(\Delta \dot{x}) f_y \Delta y \leq 0 \ \& \ |\Delta \dot{x}| \geq v_s & \\ 0 & \Delta y > 0 \end{cases} \quad (31)$$

where $u_s = 0.7$ and $u_k = 0.5$ are static friction coefficient and kinetic friction coefficient, respectively, and $v_{th} = 5 \times 10^{-4}$ represents the threshold velocity. All the parameters are determined by a parametric analysis. The tangential force can be expressed by its two components in x and z directions. Similarly, the friction in the z -direction has the same form as the one in x -direction.

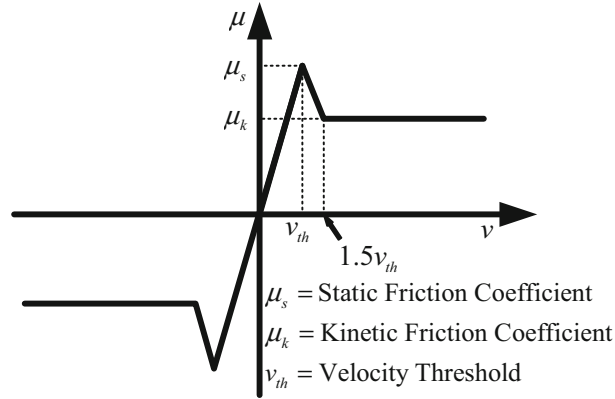


Fig. 4 The relationship between friction coefficient and sliding speed

4.3 Nonhorizontal and nonstationary ground

In reality, the contact conditions are complicated. Hence, we progressively extend the contact model to more general shapes. Here, we mainly discuss the nonhorizontal and nonstationary field to simulate the slope, earthquake, etc.

4.3.1 Determination of whether touching the surface and the last landing point

Obviously, the penetration depth and velocity in the contact model must be relative to the last contact point. In the horizontal surface, we determine whether the two bodies are in touching by $\Delta y < 0$. Nevertheless, the judgment condition is not rigorous in the nonhorizontal slope. The intuitive method to judge whether the contact point touches the surface is to see whether the contact point is below the surface or not. Here, for ease of understanding and calculation, we mainly discuss the contact condition that one of the contact surfaces is flat. We rotate and translate the contact points under each sole and the land to the initial horizontal position:

$$\mathbf{p}_{f2} = \mathbf{R}(\theta_x, \theta_y, \theta_z)^{-1}(\mathbf{p}_f - \mathbf{t} - \mathbf{p}_c) + \mathbf{p}_c \quad (32)$$

where $\mathbf{p}_{f2} = [x_{f2}, y_{f2}, z_{f2}]^T$ is the corresponding location of the contact point when the slope rotates to horizontal. \mathbf{R} represents the rotation matrix of \mathbf{l}_2 to \mathbf{l}_1 , and θ_x, θ_y , and θ_z are the rotation angles of the slope. $\mathbf{t} = [t_x, t_y, t_z]^T$ is the translation quantity of the slope. $\mathbf{p}_c = [r_x, r_y, r_z]^T$ is the center of rotation. Indeed, we can assume that the initial height of the horizontal ground is 0 such that we only need to judge y_{f2} . After rotating and translating, the criteria for determining whether to contact the land is the same as before. Note that, when $\Delta y > 0$, it means that the two bodies do not touch each other. Once Δy is less than 0 again, recording the current location $\mathbf{p}_s = [x_s, y_s, z_s]^T$, the original contact point needs to be updated. It is crucially important in the contact simulation.

4.3.2 Determination of the offset of deformation

After determining the original contact point \mathbf{p}_s , the offset of deformation $\Delta \mathbf{d} = [\Delta x, \Delta y, \Delta z]^T$ should be determined to compute the contact force. It should be noted that the original contact point is no longer the location where it was recorded before in the world frame if the land is moving or shaking. The current location $\mathbf{p}'_s = [x'_s, y'_s, z'_s]^T$ of the original contact point also needs to be updated:

$$\mathbf{p}'_s = \mathbf{R}(\theta')(\mathbf{p}_s - \mathbf{p}'_c) + \mathbf{p}'_c + \mathbf{t}' \quad (33)$$

where $\theta' = [\theta'_x, \theta'_y, \theta'_z]$ and $\mathbf{t}' = [t'_x, t'_y, t'_z]^T$ represent the angle of rotation and quantity of translation after touching, respectively. Clearly, (33) is the inverse operation of (32). So, we can obtain the offset of deformation $\Delta \mathbf{d}$:

$$\Delta \mathbf{d} = \mathbf{p}_f - \mathbf{p}'_s. \quad (34)$$

It should be pointed out that $\Delta \mathbf{d}$ is the offset of deformation in the world frame, which needs to be translated to the surface local coordinate system:

$$\Delta \mathbf{d}' = \mathbf{T}_a \Delta \mathbf{d} \quad (35)$$

where the transformation matrix \mathbf{T}_a is:

$$\mathbf{T}_a = \begin{bmatrix} c_x c_y c_z - s_x s_z & c_x s_z + c_y c_z s_x & c_z s_y \\ -c_z s_x - c_x c_y s_z & c_x c_z - c_y s_x s_z & -s_y s_z \\ -c_x s_y & -s_x s_y & c_y \end{bmatrix} \quad (36)$$

where s_x , c_x , s_y , c_y , s_z , and c_z are the sine and cosine values about the three axes, respectively. Similarly, we obtain the relative velocity $\Delta \dot{\mathbf{d}}' = [\Delta x'_s, \Delta y'_s, \Delta z'_s]^T$:

$$\Delta \dot{\mathbf{d}}' = \mathbf{T}_a \Delta \mathbf{v} \quad (37)$$

where $\Delta \mathbf{v}$ is the relative velocity in the world frame:

$$\Delta \mathbf{v} = \dot{\mathbf{p}}_f - \dot{\mathbf{p}}'_s = \dot{\mathbf{p}}_f - \dot{\mathbf{R}}(\theta')(\mathbf{p}_s - \mathbf{p}'_c) - \dot{\mathbf{t}}' \quad (38)$$

where $\dot{\mathbf{p}}'_s = [\Delta x'_s, \Delta y'_s, \Delta z'_s]^T$ is the velocity of the contact point by sensors in the world frame. After obtaining the offset of deformation $\Delta \mathbf{d}'$ and the relative velocity $\Delta \dot{\mathbf{d}}'$, the contact force is computed using (12) and (15). Physical quantities, such as location and velocity of the contact point, are obtained by sensors, and the contact force all refer to the world frame. In this approach, it is easy to integrate the contact model under a common framework. In some simulation software, the rotation motion is described by the quaternion. Here we only need to convert the quaternion into the rotation matrix, the mentioned shaking and translation principles are still applicable.

5 Experiments

Herein, we only actuate the joints that produce necessary motions on the legs and fix other joints. Different scenes are defined for verifying the feasibility and effectiveness of the ground contact model and balance controller. There is a total of fifty contact points under each toe and heel. And all the simulations are begun at the double support. Moreover, the environment such as the directions, magnitudes, and the locations of perturbations and types of the ground are unknown to the controller. Besides, only the joint angles, joint velocities, joint torques, and GRF at each sole can be measured by sensors (Fig. 5).

5.1 Maintaining balance on the stationary horizontal beam

Firstly, we test the robot while standing on a stable horizontal beam. Figure 6 shows a series of results when the robot encounters an instantaneous perturbation in the sagittal plane. The external force (150 N) is applied at 0.1 s, locates at the CoM and lasts 0.1 s. The desired position of CoM is where the robot would be standing upright, and the desired linear and angular momenta are set to zero.

The gravity of the robot is equivalent to a step response for the ground contact model. Thus, the foot support shows saltation at the beginning of the simulation. Caused by this, momentum and CoP locations also change significantly. After 0.02 s, the robot tends to be stable. Under the constant external force, the linear and angular momenta increase approximately linearly until the force disturbance vanishes (0.2 s). Figure 6a shows that the CoM location is continuing to offset from 0.2 s to nearly 0.31 s, however, the offset velocity is decreasing obviously. Meanwhile, both the linear and angular momenta decline and change the direction. Our controller distributes the support force to each foot. And the resultant support force nearly equals the gravity due to no dramatical change in the height of the robot. Hence, the support forces shown in Fig. 6e are approximately mirror pattern. Figure 6d shows that the actual angular momentum change rate is nearly identical to its desired value. They actually are slightly different because of the optimization as well as the errors carried by the ground contact model. The CoPs in the forward direction under the left and right soles are always kept in the safe support region (-0.07 – 0.13 m).

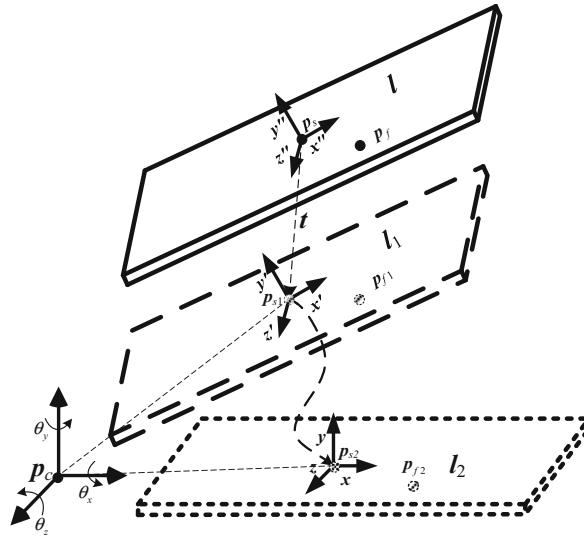


Fig. 5 For any nonhorizontal and nonstationary slopes l , they can be converted from horizontal one l_2 by translation and rotation. The center of rotation is located at p_c . p_{s1} is the interim point translated from landing point p_s and rotated by $[\theta_x, \theta_y, \theta_z]$ to get p_{s2} . Similarly, p_f , p_{f1} , and p_{f2} are the contact point locations of the robot before and after translation or rotation, respectively

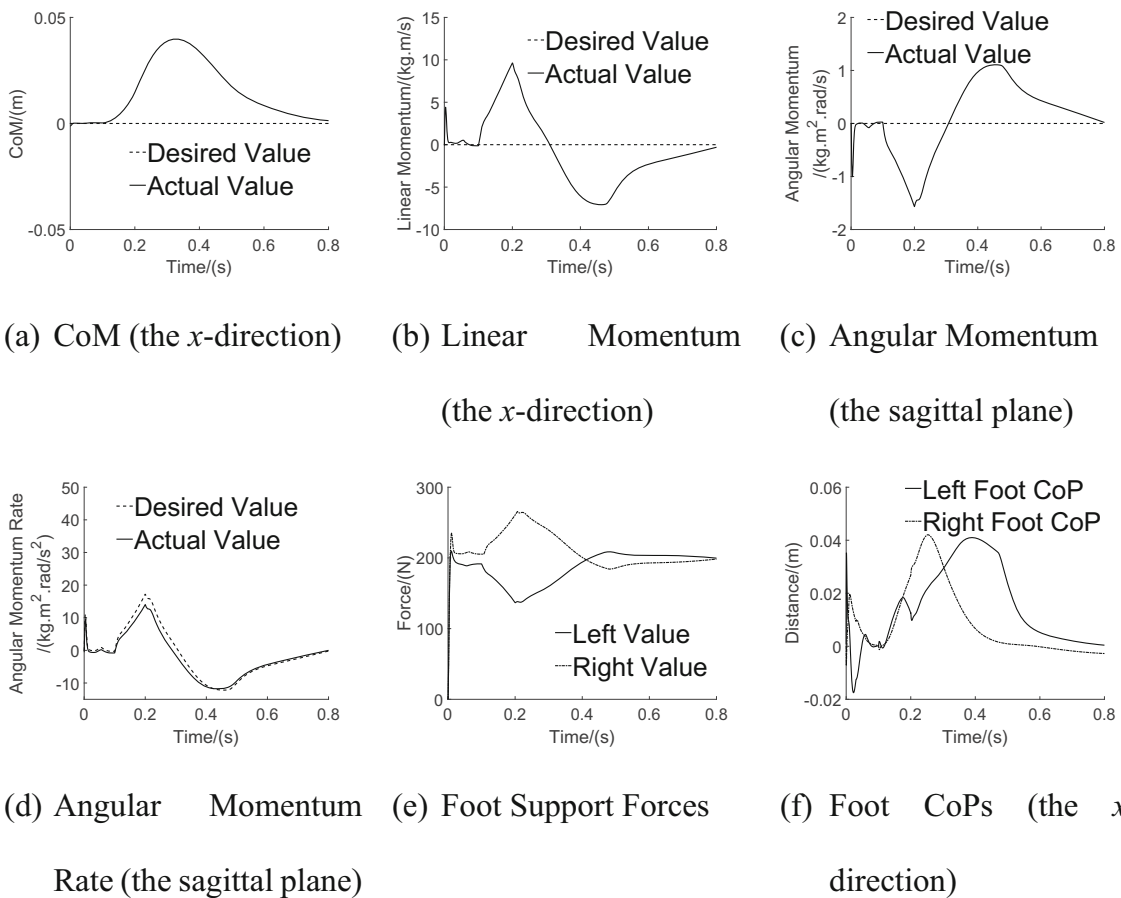


Fig. 6 Push recovery in the sagittal plane

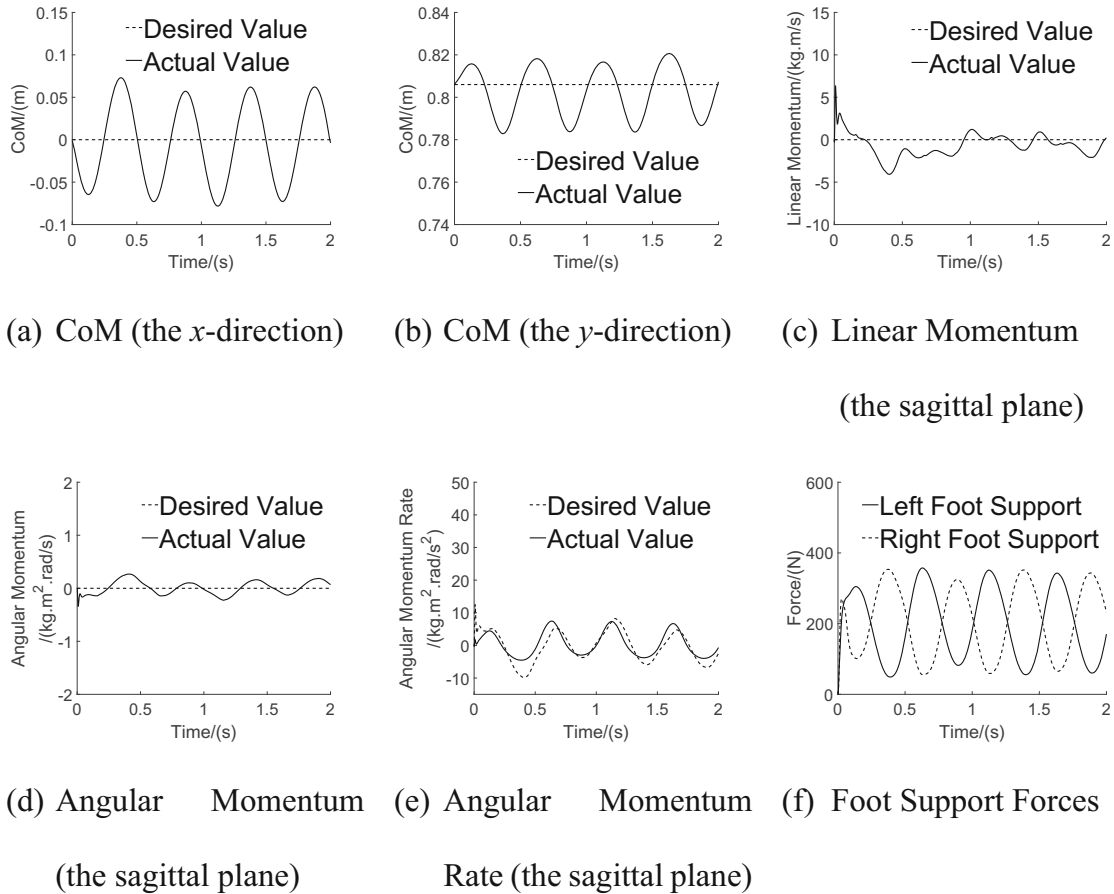


Fig. 7 Maintaining balance on moving and shaking ground

5.2 Balance on moving and shaking ground

To simulate the robot standing on a ship or a public transport, the second experiment is subjected to maintain balance on the floating ground. In this case, we specify the motions of moving and shaking as a sinusoidal signal. The magnitude of the moving sinusoidal signal is 0.1 m and frequency is 5 Hz. For shaking motion, the magnitude and frequency are 5° and 5 Hz, respectively. The ground movements start at 0 s. Besides, all the momenta exhibited here are referring to the moving or shaking ground, i.e., the momenta are zero as long as the robot is relatively stationary to the ground although the ground is moving or shaking. The target centroidal positions and momenta are the same as those in Sect. 4.1.

Under the sinusoidal signal, the overall trajectories of CoM, momenta, and GRFs exhibit patterns very similar to the sinusoidal signal. In the coronal plane, the range of the CoP location is larger during the double support compared with that in the sagittal plane. Hence, the actual angular momentum change rate is roughly the same as the desired value (Fig. 7e). Figure 7f shows that the path of the left foot support force is a mirror image of that of the right foot, and the resultant force of them is roughly equal to the gravity of the robot. Theoretically, as long as the robot is moving at the same velocity as the ground, it does not fall. Hence, our controller chooses the ground frame when calculating the momenta. Figure 7f shows the support forces of the left and right feet rise and fall alternately, and it drives the actual angular momentum change rate to follow the desired one (Fig. 7d).

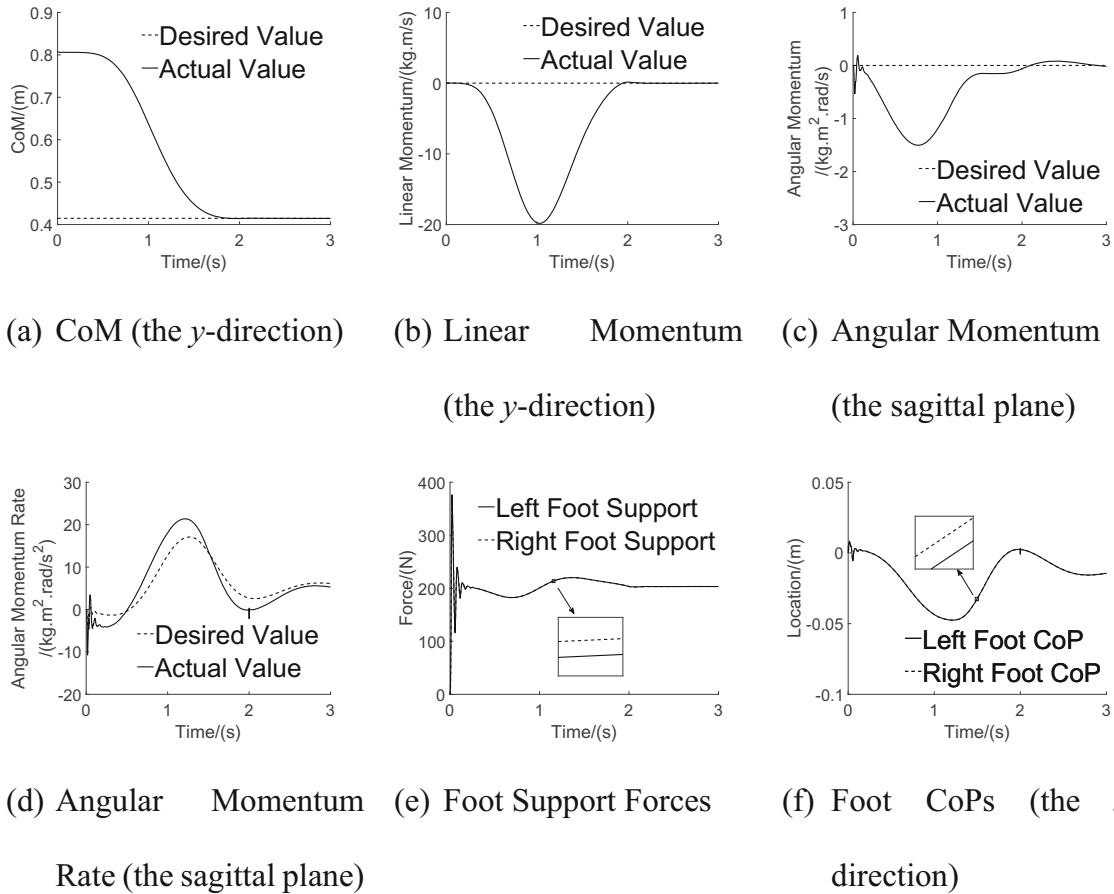


Fig. 8 Balance squatting task

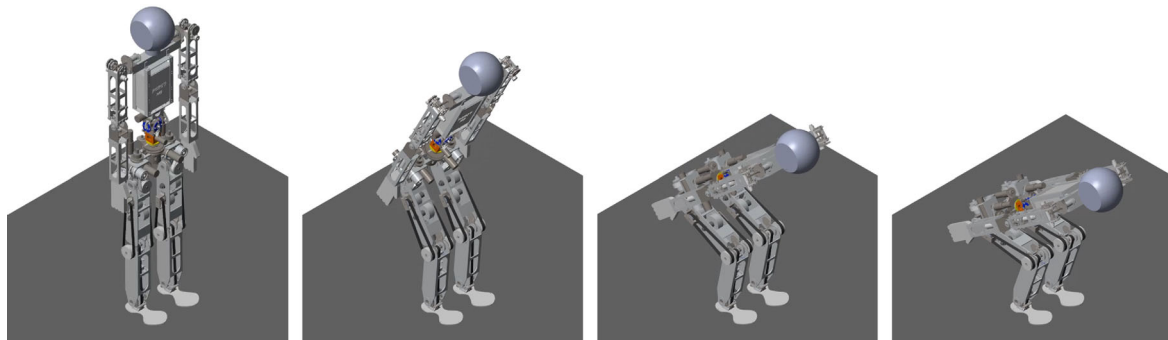


Fig. 9 Animation of a balanced squatting

5.3 Squatting down

Simulation results of balanced squatting with the linear and angular momenta controlled are shown in Fig. 8. The desired horizontal position of CoM is the center of the supporting convex hull, and the target height of the CoM is 0.418 m. Note that we do not add the extra desired torso moments in (22) due to the phenomenon that the human’s torso does not stay upright during squatting down. Figure 9 shows a series of snapshots illustrating that the robot is squatting.

Similarly, the trajectories of the CoM, linear momentum, and angular momentum in Fig. 8a–c show that they all come to the desired values smoothly. The actual angular momentum change rate roughly has a similar

path as the desired one. The support forces under each foot are highly identical. Besides, the support decreases first and then increases, which is also in line with the theoretical change of the support during the robot squatting down process. Figure 8f presents the actual CoP in the x -direction under each foot. Similarly, the CoP trajectories of the two feet are also highly consistent. There are saltations near 2 s in both Fig. 8d and 8f caused by ending of squatting down. And the total support equals gravity while the robot is motionless.

6 Conclusions and discussion

In this paper, we present our motivation for controlling humanoid robots in terms of their suitability to anthropogenic environments. The controller distributes the GRFs under the left and right feet according to the linear and angular momenta and computes the joint torques to generate the GRFs. To obtain high-fidelity simulation, we develop a ground contact model which includes all contact and impact phenomena. Our controller involves three optimization problems, and all of them are solved effectively by the least-square algorithm. Finally, we show the performance of the balance controller by several simulation experiments. The results confirm that the controller can deal with different types of ground and perturbation.

Our work mainly follows a series of centroid dynamics research [8]. We propose a practical approach to establish the dynamic equivalence relation between GRFs and joint torques. We hope that this paper not only introduces a humanoid-ground interaction force control method but also serves as an excellent example for readers who decide to build a new robotics simulation physical environment instead of some commercial robot simulation software. Also, the dynamic mapping between joint space and operational system space has a value worthy of reference for the researchers who plan the GRF to obtain the desired motion.

We have proposed an overall humanoid-ground contact framework, and it is clear that the framework can have a much wider application. In the near future, we will test the humanoid grasping tasks. Furthermore, according to our knowledge, the nature of the arm movement is not fully understood. Bassel et al. [24] explored the arm swing effects on walking bipedal gaits. Numerical results showed that passive movement of a biped due to the dynamics of the locomotor system has higher values of the sthenic criterion than that of actuated shoulders. Inspired by them, our future work will determine whether the optimal movement of the arms for balance is passive or not and verify the effect of arms on keeping balance.

Acknowledgements The work is supported by National Natural Science Foundation of China under grant 51375085.

Author's contribution A momentum-based framework is presented. It determines the desired ground reaction force and scientifically distributes the desired ground reaction force to each foot. Dynamic mapping of GRF to joint torques is proposed with constraints, which avoids the complex operation of centroid dynamics. An effective and reliable ground contact model is built to offer the ground reaction force, and it includes the main relevant contact phenomena that will affect the dynamic behavior of the robotic system.

Funding National Natural Science Foundation of China under grant 51375085.

Declarations

Conflict of interest The author declares that they have no conflict of interest.

Appendix

In double support phase, extract $m = \text{rank}(\mathbf{J})$ linearly independent rows from \mathbf{J} , and partition the generalized coordinates into m dependent variables, \dot{q}_a and $n + 6 - m$ primary variables, \dot{q}_b :

$$\mathbf{x} = \mathbf{K} \begin{bmatrix} \mathbf{x}_a \\ \mathbf{x}_b \end{bmatrix}, \dot{\mathbf{x}} = \mathbf{K} \begin{bmatrix} \dot{\mathbf{x}}_a \\ \dot{\mathbf{x}}_b \end{bmatrix} \quad (39)$$

where \mathbf{K} represents an $(n + 6) \times (n + 6)$ permutation matrix with the determinant of 1. The constraint can be written in terms of these variables:

$$[\mathbf{J}_a, \mathbf{J}_b] \begin{bmatrix} \dot{\mathbf{x}}_a \\ \dot{\mathbf{x}}_b \end{bmatrix} = \mathbf{0} \quad (40)$$

where \mathbf{J}_a is an $m \times m$ matrix formed by extracting m independent rows and columns from \mathbf{J} and \mathbf{J}_b is $m \times (n - m)$. As \mathbf{J}_a is always invertible, \mathbf{q}_b can be expressed in terms of:

$$\dot{\mathbf{x}}_a = -\mathbf{J}_a^{-1} \mathbf{J}_b \dot{\mathbf{x}}_b. \quad (41)$$

Hence, the joint angular velocities and the joint angular accelerations can be expressed as:

$$\dot{\mathbf{x}} = \mathbf{K} \begin{bmatrix} \dot{\mathbf{x}}_a \\ \dot{\mathbf{x}}_b \end{bmatrix} = \mathbf{K} \begin{bmatrix} -\mathbf{J}_a^{-1} \mathbf{J}_b, \\ \mathbf{I}_{(n+6-m) \times (n+6-m)} \end{bmatrix} \dot{\mathbf{x}}_b = \mathbf{N} \dot{\mathbf{x}}_b, \quad (42)$$

$$\ddot{\mathbf{x}} = \mathbf{N} \ddot{\mathbf{x}}_b + \dot{\mathbf{N}} \dot{\mathbf{x}}_b. \quad (43)$$

Substituting (39) and (48) into (14), the dynamics of the humanoid robot can be described as:

$$\tilde{\mathbf{I}}(\mathbf{x}_b)(\mathbf{N} \ddot{\mathbf{x}}_b + \dot{\mathbf{N}} \dot{\mathbf{x}}_b) + \tilde{\mathbf{C}}(\mathbf{x}_b, \dot{\mathbf{x}}_b) \mathbf{N} \dot{\mathbf{x}}_b + \tilde{\mathbf{G}}(\mathbf{x}_b) = \mathbf{B} \boldsymbol{\tau} + \mathbf{J}^T \mathbf{F}. \quad (44)$$

Now we multiply (44) by \mathbf{N}^T from the left-hand side and obtain:

$$\mathbf{N}^T \tilde{\mathbf{I}}(\mathbf{x}_b) \mathbf{N} \ddot{\mathbf{x}}_b + \mathbf{N}^T (\tilde{\mathbf{I}}(\mathbf{x}_b) \dot{\mathbf{N}} + \tilde{\mathbf{C}}(\mathbf{x}_b, \dot{\mathbf{x}}_b) \mathbf{N}) \dot{\mathbf{x}}_b + \mathbf{N}^T \tilde{\mathbf{G}}(\mathbf{x}_b) = \mathbf{N}^T \mathbf{B} \boldsymbol{\tau} + \mathbf{N}^T \mathbf{J}^T \mathbf{F}. \quad (45)$$

Here, we can notice that $\mathbf{JN} = \mathbf{0}$:

$$\mathbf{JN} = \mathbf{JA} \begin{bmatrix} -\mathbf{J}_a^{-1} \mathbf{J}_b \\ \mathbf{I}_{(n+6-m) \times (n+6-m)} \end{bmatrix} = [\mathbf{J}_a, \mathbf{J}_b] \begin{bmatrix} -\mathbf{J}_a^{-1} \mathbf{J}_b \\ \mathbf{I}_{(n+6-m) \times (n+6-m)} \end{bmatrix} = -\mathbf{J}_a \mathbf{J}_a^{-1} \mathbf{J}_b + \mathbf{J}_b = \mathbf{0}. \quad (46)$$

Thus, the reduced form of the dynamical model in the double support phase is given as:

$$\mathbf{I}_b(\mathbf{x}_b) \ddot{\mathbf{x}}_b + \mathbf{C}_b(\mathbf{x}_b, \dot{\mathbf{x}}_b) \dot{\mathbf{x}}_b + \mathbf{G}_b(\mathbf{x}_b) = \mathbf{B}_b \boldsymbol{\tau} \quad (47)$$

where

$$\mathbf{I}_b = \mathbf{N}^T \tilde{\mathbf{I}} \mathbf{N}, \mathbf{C}_b = \mathbf{N}^T (\tilde{\mathbf{I}} \dot{\mathbf{N}} + \tilde{\mathbf{C}} \mathbf{N}), \mathbf{G}_b = \mathbf{N}^T \tilde{\mathbf{G}}, \mathbf{B}_b = \mathbf{N}^T \mathbf{B}. \quad (48)$$

As a result, given the current state and the joint torques, the unique angular acceleration can be obtained:

$$\ddot{\mathbf{x}}_b = \mathbf{I}_b^{-1} (\mathbf{B}_b \boldsymbol{\tau} - \mathbf{C}_b \dot{\mathbf{x}}_b - \mathbf{G}_b). \quad (49)$$

References

1. Kajita, S., Hirukawa, H., Harada, K., Yokoi, K.: Introduction to humanoid robotics. In: Springer Tracts in Advanced Robotics (2014)
2. Hyon, S.H., Hale, J.G., Cheng, G.: Full-body compliant human-humanoid interaction: balancing in the presence of unknown external forces. *IEEE Trans. Rob.* **23**(5), 884–898 (2007)
3. Raibert, M.H., Tello, E.R.: Legged robots that balance. *IEEE Expert.* **1**(4), 89–89 (1986)
4. Popovic, M., Hofmann, A., Herr, H.: Zero spin angular momentum control: definition and applicability. In: 2004 4th IEEE-RAS International Conference on Humanoid Robots (2004). <https://doi.org/10.1242/jeb.008573>
5. Herr, H., Popovic, M.: Angular momentum in human walking. *J. Exp. Biol.* **211**, 467–481 (2008)
6. Komura, T., Leung, H., Kudoh, S., Kuffner, J.: A feedback controller for biped humanoids that can counteract large perturbations during gait. *Proc. IEEE Int. Conf. Robot. Autom.* (2005). <https://doi.org/10.1109/ROBOT.2005.1570405>
7. Uemura, M., Hirai, H.: Standing and stepping control with switching rules for bipedal robots based on angular momentum around ankle. *Int. J. Humanoid Robot.* **16**(5), 1950022–1–1950022–18 (2019)
8. Orin, D.E., Goswami, A., Lee, S.H.: Centroidal dynamics of a humanoid robot. *Auton. Robots.* **35**, 161–176 (2013)
9. MacChietto, A., Zordan, V., Shelton, C.R.: Momentum control for balance. *ACM Trans. Graph.* (2009). <https://doi.org/10.1145/1531326.1531386>
10. Koolen, T., Bertrand, S., Thomas, G., De Boer, T., Wu, T., Smith, J., Engelsberger, J., Pratt, J.: Design of a momentum-based control framework and application to the humanoid robot atlas. *Int. J. Humanoid Robot.* **13**(1), 1650007 (2016)
11. Hyon, S.H.: Compliant terrain adaptation for biped humanoids without measuring ground surface and contact forces. *IEEE Trans. Robot.* **25**(1), 171–178 (2009)
12. Hyon, S.H., Suewaka, D., Torii, Y., Oku, N.: Design and experimental evaluation of a fast torque-controlled hydraulic humanoid robot. *IEEE/ASME Trans. Mechatron.* **22**(2), 623–633 (2017)
13. Sentis, L., Park, J., Khatib, O.: Compliant control of multicontact and center-of-mass behaviors in humanoid robots. *IEEE Trans. Robot.* (2010). <https://doi.org/10.1109/TRO.2010.2043757>
14. Vukobratović, M., Stepanenko, J.: On the stability of anthropomorphic systems. *Math. Biosci.* **15**(1), 1–37 (1972)

15. Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., Hirukawa, H.: Resolved momentum control: humanoid motion planning based on the linear and angular momentum. *IEEE Int. Conf. Intell. Robot. Syst.* (2003). <https://doi.org/10.1109/IROS.2003.1248880>
16. Jamisola, R.S., Kormushev, P.S., Roberts, R.G., Caldwell, D.G.: Task-space modular dynamics for dual-arms expressed through a relative Jacobian. *J. Intell. Robot. Syst.* **83**, 205–218 (2016)
17. Luxman, R., Zielinska, T.: Robot motion synthesis using ground reaction forces pattern: Analysis of walking posture. *Int. J. Adv. Robot. Syst.* 1–12 (2017)
18. Lee, S.H., Goswami, A.: A momentum-based balance controller for humanoid robots on non-level and nonstationary ground. *Auton. Robots.* **33**, 399–414 (2012)
19. Del Prete, A., Tonneau, S., Mansard, N.: Zero step capturability for legged robots in multicontact. *IEEE Trans. Robot.* **34**(4), 1021–1034 (2018)
20. Carpentier, J., Mansard, N.: Multicontact locomotion of legged robots. *IEEE Trans. Robot.* **34**(6), 1441–1460 (2018)
21. Hunt, K.H., Crossley, F.R.E.: Coefficient of restitution interpreted as damping in vibroimpact. *J. Appl. Mech. Trans. ASME.* **42**(2), 440–445 (1975)
22. Canudas-de-Wit, C.: Comments on “a new model for control of systems with friction”. *IEEE Trans. Autom. Control* **43**(8), 1189–1190 (1998)
23. Bhushan, B.: *Introduction to Tribology*, Second Edition. (2013)
24. Kaddar, B., Aoustin, Y., Chevallereau, C.: Arm swing effects on walking bipedal gaits composed of impact, single and double support phases. *Rob. Auton. Syst.* **66**, 104–115 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.