Journal of Intelligent & Robotic Systems (2021) 102: 66
https://doi.org/10.1007/s10846-021-01418-x

REGULAR PAPER

Whole-Body Control and Angular Momentum Regulation using Torque Sensors for Quadrupedal Robots

Young Hun Lee[1] · Yoon Haeng Lee[1] · Hyunyong Lee[1] · Hansol Kang[1] · Jun Hyuk Lee[1] · Ji Man Park[1] · Yong Bum Kim[1] · Hyungpil Moon[1] · Ja Choon Koo[1] · Hyouk Ryeol Choi[1]

Received: 16 August 2020 / Accepted: 6 May 2021 / Published online: 18 June 2021

Abstract

Ground reaction force (GRF) plays an integral role in legged robots to control interaction with the ground. However, most techniques in whole-body controller for quadrupedal robots do not explicitly take into account actual torque or force in their control loops and instead use feed-forward force to generate joint torque at every time step. In this paper, we present a closed-loop whole-body controller using the actual joint torque feedback, which regulates angular momentum of the center of mass (CoM) for quadrupedal locomotion. Using the torque measured from each torque sensor and the torque by solving the inverse dynamics, we can compute the external joint torque induced by the contact with the ground. To fully use the computed joint torque, we discuss a feasible approach and whole-body control criterion for quadrupedal robots that have constrained support polygons because of their point-feet and certain gaits using two or less legs in contact. Based on the approach, we generate a centroidal moment pivot trajectory considering the leg dynamics, linear translation, and angular rotation of the CoM, which can stabilize the robot's balance by using the actual angular momentum rate change transformed from the measured joint torque. In addition, a push recovery strategy based on capture point dynamics derived from linear momentum and a foothold generation method are integrated into the controller. The proposed controller is tested on a quadrupedal robot, called AiDIN-VI, that has a torque sensor at each joint. The proposed whole-body controller enables the robot to demonstrate several gait types such as trotting, pacing, jumping, and walking on various environments, and locomotive abilities under external pushes are verified.

Keywords Quadrupedal robot · Whole-body control · Angular momentum regulation · Balance recovery · Ground reaction force · Joint torque feedback

1 Introduction

Robotic researchers have considered dynamic and agile locomotion skills of animals as remarkable. Although improved control methods have been studied for their realization with significant advances in hardware technology, it is still difficult to replicate this biological performance into mechanical robotic systems.

To execute a robotic locomotion that resembles a natural movement, the contact force can be used to stabilize the robot's balance because locomotion is typically carried out by force interaction between the robot and the ground. Therefore, various methods have been studied that focus on acquiring actual ground reaction force (GRF). Force estimation method by using measured motor currents and joint dynamics has been researched [1, 2]. This method allows us to acquire an approximate GRF without additional attachment of a torque or force sensor. Thus, the robot can obtain a decreased weight, and a simple and compact system. However, the measured motor currents include a motor current to actuate each joint, as well as another current caused by the contact with the ground. In addition, in case of the joint dynamics, if a high gear ratio transmission, which has a non-backdrivability, is used at each joint of the robot, joint states are unchanged by the GRF. For this reason, the estimated GRF by using the motor currents and joint dynamics does not completely correspond to the actual GRF. On the other hand, embedding the torque or force sensor at each robotic leg provides a more precise GRF [3, 4]. However,

✉ Hyouk Ryeol Choi
choihyoukryeol@gmail.com

[1] School of Mechanical Engineering, Sungkyunkwan University, Suwon, 440-746, South Korea

the force sensor is vulnerable to damage because the impact with the ground is directly delivered to the sensor. Nevertheless, few quadrupedal robots with force sensors in the limbs have shown significant capabilities, but details regarding their effective use for locomotion is unavailable [5].

The GRF is strongly connected with the dynamics of the center of mass (CoM). Therefore, several approaches that use the GRF for the CoM motion control have been conducted. First, the optimal contact force that corresponds to the desired commands of the CoM is computed. Walking between two high-sloped walls [6] and jumping onto and down from a desk [7] were demonstrated by distributing the feed-forward force using given linear and angular acceleration of the CoM. In contrast to the conventional approach of identifying the CoM trajectory, which uses given footholds and generates the feed-forward force of the each leg to follow the trajectory, a method that executes quadrupedal walking by simultaneously optimizing the footholds and CoM motion was studied [8]. As a force profile design method, Bézier curve was explored to enable the robot to jump over an obstacle [9].

Another approach is the opposite to the former, which is generating the CoM motion according to the GRF. A representative example of the method is zero moment point (ZMP), which has been widely implemented for legged robots [10, 11]. It calculates the actual CoM moment caused by the GRF, and then performs feedback control [12]. As a more advanced approach, whole-body controller by using the centroidal momentum has been explored. However, although its effects have been validated [13], few locomotion controllers have considered the centoridal momentum. In previous works, under the assumption that a robot has a massless leg, desired angular momentum has been conventionally set to zero [14]. Based on this assumption, impedance control law using the centroidal moment pivot (CMP) was implemented [15]. In addition, control strategies that design the desired angular momentum trajectories by using the given commands such as lumped mass' trajectories [16] and GRF profiles [17] were studied.

This paper addresses a closed-loop whole-body controller using the actual joint torque feedback, which regulates the angular momentum of the CoM. The external joint torque is computed by using the torque measured from each torque sensor and the torque by solving the inverse dynamics. Given this computed torque, we generate a CMP trajectory, which considers the leg dynamics, linear translation, and angular rotation of the CoM. This trajectory enables the robot to stabilize its balance by using the actual angular momentum rate change, which is transformed value from the measured joint torque. Moreover, push recovery controller based on capture point (CP) and perceptive locomotion method using online foothold generation are integrated into the controller. The proposed whole-body controller is tested on a quadrupedal robot, AiDIN-VI, shown in Fig. 1, which has a torque sensor at each joint [18]. The main contributions of this work are as follows: (i) Even though locomotion is a task by the force interaction, the method to use the actual torque or force remains unclear for executing quadrupedal locomotion. Contrary to general whole-body controllers of generating the feed-forward force at each leg [6–9], we focus on using the actual torque into the controller. (ii) To fully use the actual joint torque feedback, we discuss a feasible approach and whole-body control criterion with considering the quadrupedal robot's features (i.e., pointed feet and gait types using two or less stance legs). Consequently, instead of the ZMP controller, which is a most representative control method for the CoM motion planning in bipedal as well as the quadrupedal robots, we implement the CMP controller, which uses the actual CoM state transformed from the measured joint torque. (iii) In the CMP controller, desired angular momentum rate of change has been set to zero [14, 15] or designed as a trajectory based on the predefined motion [16]. On the contrary, we generate the trajectory by using sensory feedback, which considers the leg dynamics, translation, and rotation of the CoM. Therefore, the designed CMP trajectory enables the robot to recovery its balance by reacting to disturbances occurred during locomotion. (iv) With the proposed whole-body controller, AiDIN-VI can demonstrate both four types of gaits (trotting, pacing, jumping, and walking) on various environments, and locomotive capabilities in the presence of external pushes.

The remainder of this paper is organized as follows: in Section 2, computation method of the actual GRF and the feasible approach for using the computed GRF are explained. Overall whole-body controller is specifically

Fig. 1 The proposed whole-body controller, tested on a quadrupedal robot, AiDIN-VI, that has torque sensor at each joint, allows the robot to locomote on various environments

presented in Section 3. In Section 4, experimental evaluations of the proposed controller are discussed. Finally, conclusions are provided in Section 5.

2 Ground Reaction Force Computation and Its Utilization

2.1 Quadrupedal Robot Model

A quadrupedal robot is generally regarded as a floating base system composed of $6+n$ degrees of freedom. Its dynamics model can be divided into two parts: the dynamics of the base (first 6 rows) represented by index b, and the dynamics of the joint part (last n rows) denoted by index j as

$$\underbrace{\begin{bmatrix} \boldsymbol{M}_b \\ \boldsymbol{M}_j \end{bmatrix}}_{\boldsymbol{M}} \ddot{\boldsymbol{q}} + \underbrace{\begin{bmatrix} \boldsymbol{b}_b \\ \boldsymbol{b}_j \end{bmatrix}}_{\boldsymbol{b}} = \begin{bmatrix} \boldsymbol{0}_6 \\ \boldsymbol{\tau}_j \end{bmatrix} + \sum_{i=1}^{n_c} \begin{bmatrix} \boldsymbol{J}_{b,i}^T \\ \boldsymbol{J}_{j,i}^T \end{bmatrix} \boldsymbol{f}_i \tag{1}$$

where $\boldsymbol{M} \in \mathbb{R}^{(6+n)\times(6+n)}$ represents the mass matrix, $\boldsymbol{b} \in \mathbb{R}^{6+n}$ denotes the vector of Coriolis, centrifugal and gravity force, $\boldsymbol{\tau}_j \in \mathbb{R}^n$ represents the joint torque, and $\boldsymbol{f}_i \in \mathbb{R}^m$ is the GRF of the i^{th} leg by the contact with the ground. Here, $m = 3$ assuming that the quadrupedal robots have point-foot and n_c indicates the number of contact feet. $\boldsymbol{q} \in \mathbb{R}^{6+n}$ is the stacked coordinate vector consisted of the floating base state $\boldsymbol{q}_b \in \mathbb{R}^6$ and the joint angle $\boldsymbol{q}_j \in \mathbb{R}^n$, and $\ddot{\boldsymbol{q}}$ is its second time derivative. In addition, for each contact foot, $\boldsymbol{J}_{j,i} \in \mathbb{R}^{m\times n}$ denotes the contact Jacobian and $\boldsymbol{J}_{b,i} \in \mathbb{R}^{m\times 6}$ is the base Jacobian such that $\boldsymbol{J}_{b,i} = \left[\boldsymbol{I}_3 \ [\boldsymbol{x}_{c,i}\times]\right]$, where $\boldsymbol{x}_{c,i}$ indicates i^{th} foot location from the CoM.

2.2 Ground Reaction Force Computation

The dynamics of the joint segment in Eq. 1 (last n rows) can be rewritten as

$$\boldsymbol{A} = \boldsymbol{A}_{int} + \boldsymbol{A}_{ext}, \tag{2}$$

where

$$\boldsymbol{A}_{int} = \boldsymbol{\tau}_j, \qquad \boldsymbol{A}_{ext} = \sum_{i=1}^{n_c} \boldsymbol{J}_{j,i}^T \boldsymbol{f}_i. \tag{3}$$

Here, $\boldsymbol{A}_{int} \in \mathbb{R}^n$ is the internal torque generated by joint's actuation itself, $\boldsymbol{A}_{ext} \in \mathbb{R}^n$ is the ground reaction torque induced by the contact, and $\boldsymbol{A} \in \mathbb{R}^n$ is the total joint torque. Note that the torque sensor can measure all of the torques generated at the joint. Thus, the joint torque measured from the each torque sensor $\boldsymbol{\tau}_{ts} \in \mathbb{R}^n$ includes both the internal $\boldsymbol{A}_{int}$ and external torque $\boldsymbol{A}_{ext}$ (i.e., $\boldsymbol{\tau}_{ts} = \boldsymbol{A}$).

Typically, in whole-body motion planning, the influences of the internal torque $\boldsymbol{A}_{int}$ generated from the quadrupedal robots having low-inertia robotic legs have been ignored [6, 7]. However, although the quadrupedal robot also has low-inertia, light legs manufactured using carbon fiber tubes, and the knee actuator module coaxially placed at the rotational axis of the hip joint, $\boldsymbol{A}_{int}$ is significantly detected from the torque sensor, which is because of the rapid motion of the joints to attain the dynamic mobility [18]. Therefore, to acquire purely external torque $\boldsymbol{A}_{ext}$ without the effects of $\boldsymbol{A}_{int}$, the measured joint torque $\boldsymbol{\tau}_{ts}$ must be compensated as follows:

$$\boldsymbol{A}_{ext} = \boldsymbol{\tau}_{ts} - \boldsymbol{A}_{int}. \tag{4}$$

If the quadrupedal robot has a stiff transmission at each joint by using a high ratio reduction gear, its joint can be regarded as a non-backdrivable structure. In this case, $\boldsymbol{A}_{ext}$ does not affect the actual joint states, $\boldsymbol{q}_j$, $\dot{\boldsymbol{q}}_j$, and $\ddot{\boldsymbol{q}}_j$. Hence, the internal torque $\boldsymbol{A}_{int}$ can be computed by solving $ID(\boldsymbol{q}_j, \dot{\boldsymbol{q}}_j, \ddot{\boldsymbol{q}}_j)$, where ID denotes the inverse dynamics. Given the joint torque measured by the torque sensor $\boldsymbol{\tau}_{ts}$ and the internal torque $\boldsymbol{A}_{int}$ by solving the ID, we can obtain the purely external torque by the contact $\boldsymbol{A}_{ext}$ from Eq. 4.

Using the calculated $\boldsymbol{A}_{ext}$ in Eq. 4, actual GRF of the i^{th} leg $\boldsymbol{f}_i$ can be computed with leg Jacobian $\boldsymbol{J}_i \in \mathbb{R}^{3\times m}$, which is the $\boldsymbol{J}_{j,i}$ of the i^{th} leg, as

$$\boldsymbol{f}_i = \begin{bmatrix} f_i^x \\ f_i^y \\ f_i^z \end{bmatrix} = \boldsymbol{J}_i^{-T} \boldsymbol{\tau}_i \tag{5}$$

where $\boldsymbol{\tau}_i \in \mathbb{R}^3$ is the $\boldsymbol{A}_{ext}$ of the i^{th} limb.

2.3 Feasible Approach for using Actual Ground Reaction Force

The actual GRF $\boldsymbol{f}_i$ can play an integral role to control the whole-body of legged robots because the CoM motion can be explained as a function that consists of $\boldsymbol{f}_i$ and $\boldsymbol{x}_{c,i}$ as

$$\underbrace{\begin{bmatrix} m_c(\ddot{\boldsymbol{x}}_c + \boldsymbol{g}_c) \\ \boldsymbol{I}_b \dot{\omega}_b \end{bmatrix}}_{\dot{\boldsymbol{u}}} = f(\boldsymbol{f}_i, \boldsymbol{x}_{c,i}) = \begin{bmatrix} \boldsymbol{I}_3 & \cdots & \boldsymbol{I}_3 \\ [\boldsymbol{x}_{c,1}\times] & \cdots & [\boldsymbol{x}_{c,n_c}\times] \end{bmatrix} \begin{bmatrix} \boldsymbol{f}_1 \\ \vdots \\ \boldsymbol{f}_{n_c} \end{bmatrix} \tag{6}$$

where $\boldsymbol{u} = \begin{bmatrix} \boldsymbol{L} & \boldsymbol{H} \end{bmatrix}^T$, $\boldsymbol{L} \in \mathbb{R}^3$, and $\boldsymbol{H} \in \mathbb{R}^3$ are the centroidal momentum, linear and angular momentum of the CoM, respectively. Furthermore, $m_c \in \mathbb{R}$, $\ddot{\boldsymbol{x}}_c \in \mathbb{R}^3$, $\boldsymbol{g}_c \in \mathbb{R}^3$, $\boldsymbol{I}_b \in \mathbb{R}^{3\times 3}$, and $\dot{\omega}_b \in \mathbb{R}^3$ indicate the total mass of the robot, linear acceleration of the CoM, gravity vector, centroidal rotational inertia, and angular acceleration of the base, respectively (see Fig. 2). Here, we know that Eq. 6 corresponds to the dynamics of the floating base in Eq. 1

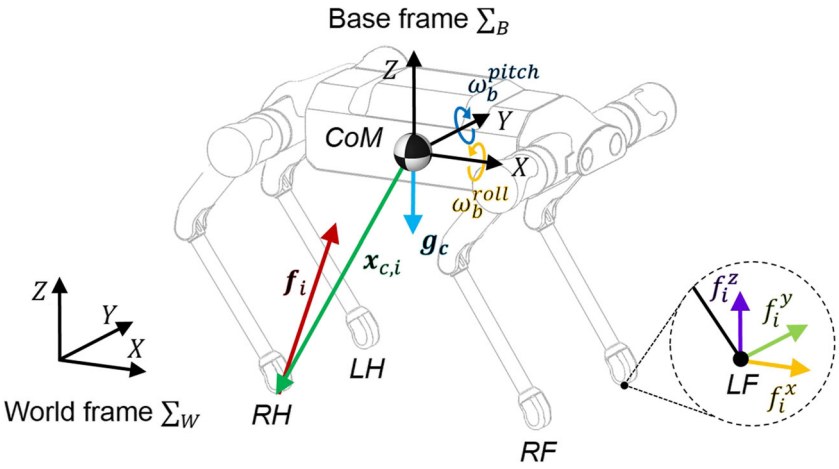

Fig. 2 Coordinate systems and literal notations used in this paper

(first 6 rows). Consequently, the quadrupedal robot with embedded torque sensors can provide (i) the actual GRF of each leg and (ii) linear and angular momentum rate change of the CoM.

As an intuitive locomotion control method, for the legged robot system that can measure or precisely estimate the actual GRF for each leg, it seems possible to compute the desired GRF of the legs, and then implement the GRF feedback control for tracking it by using the obtained actual GRF. However, this method is unsuitable to control legged robots that cannot pull on the ground. This feature makes it difficult to recover the balance while retaining the desired GRF tracking control. If one side of the robot body is slightly lifted upwards and the corresponding side's leg loses contact with the ground, the desired GRF must be generated to pull on the ground. However, the contact force with the ground is restricted to pushing motion. Thus, the desired GRF can no longer be tracked using the actual GRF and this state is assumed that the legs have strayed from their tasks. Therefore, using this control method, locomotive stability is not guaranteed. This can be further supported by a large numbers of studies [6–9] that focus on generating the feed-forward force, even though they have estimated an accurate GRF, which may be used for the desired GRF tracking control [19].

A more feasible approach is converting the actual GRF to the CoM state, and then using this CoM state feedback to control legged robots. In contrast to the former approach, which persuades exact GRF tracking, this approach using the actual CoM state feedback generates the desired position of the whole-body $\boldsymbol{q}^{des} \in \mathbb{R}^{6+n}$ to recover the robot's balance, regardless of the restriction of the GRF's direction. In addition, this approach allows the handling of the CoM according to its direct feedback. Thus, this method can more clearly adjust the CoM state, which corresponds to 6 degrees of freedom of the floating base, than the former.

3 Whole-Body Controller Using CoM State Feedback

Overall whole-body controller proposed in the paper is shown in Fig. 3, and a detailed control strategy is given in this section.

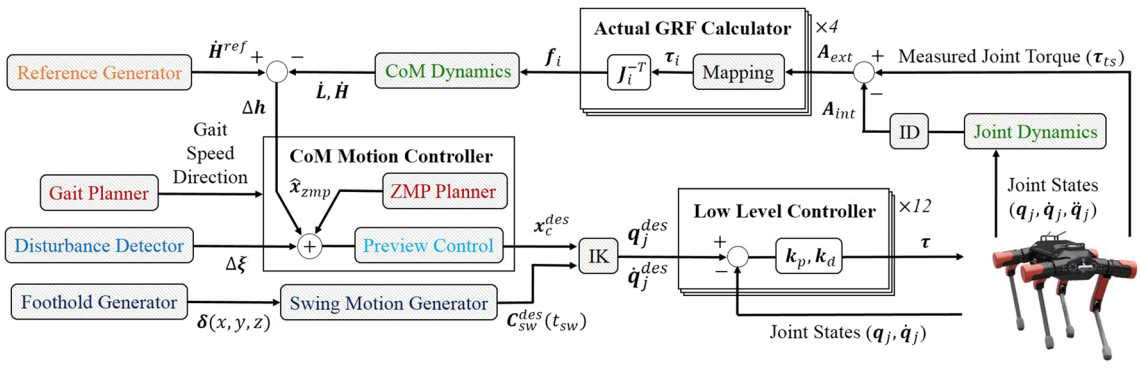

Fig. 3 Block diagram of the proposed closed-loop whole-body controller using the actual joint torque. The CoM motion controller, which generates the CoM commands by integration of the ZMP-based footstep planning, angular momentum regulator, push recovery strategy, and foothold generator, allows the robot to demonstrate both several gait types and locomotive abilities under external disturbances

3.1 Proper Whole-Body Control Criterion

Among the control methods that use the actual CoM state feedback, the ZMP has been mainly implemented for legged robots [10–12]. To employ the ZMP into the controller, measured or estimated CoM state is required. For this purpose, the actual CoM state has been generally estimated by using the robot's total mass m_c and acceleration of the CoM $\ddot{\boldsymbol{x}}_c$ [8]. However, it is difficult to obtain an accurately estimated value of the vertical CoM state because the robot's height $h_c \in \mathbb{R}$ is usually set as a constant (i.e., $\ddot{x}_c^z \approx 0$). On the contrary, using the computed GRF in Eq. 5, even if vertical motion of the robot body is fixed, the ZMP using the all three axes of the GRF can be formulated as

$$\boldsymbol{x}_{zmp} = \boldsymbol{Q}\frac{\sum_{i=1}^{n_c} \boldsymbol{x}_{c,i} \times \boldsymbol{f}_i}{\sum_{i=1}^{n_c} f_i^z}, \tag{7}$$

where

$$\boldsymbol{Q} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

The projection matrix $\boldsymbol{Q}$ enables the involvement of the vertical force of each leg f_i^z for computing the ZMP while selecting the horizontal part (i.e., $\boldsymbol{x}_{zmp} = \begin{bmatrix} x_{zmp}^x & x_{zmp}^y & 0 \end{bmatrix}^T$). The ZMP is constrained to always lie in the support polygon as

$$\hat{\boldsymbol{x}}_{zmp} = \boldsymbol{V}_{n_c}(r)\boldsymbol{x}_{zmp} \tag{8}$$

where $\hat{\boldsymbol{x}}_{zmp} \in \mathbb{R}^3$ is the constrained ZMP by the support polygon, and $\boldsymbol{V}_{n_c}(r) \in \mathbb{R}^{3\times 3}$ is the function that expresses the limited convex hull of the feet in contact such that $V_{n_c}^k(r)\lambda = r_l^k < \lambda < r_h^k$ $(k \in [x, y])$. Here, r_l and r_h are the lower and upper boundaries of the support polygon, respectively. In the case of a walking gait $(3 \leq n_c \leq 4)$, the support polygon is formed as an area. Thus, ZMP feedback control can be implemented in this area to reduce position error with the actual ZMP $\boldsymbol{x}_{zmp}$ for a more stable locomotion. In practice, we successfully demonstrated the walking gait for stair climbing in our previous work [12]. However, the foot of quadrupedal robots is typically assumed as a point and there are gaits which have two or less feet in contact. Therefore, during the trotting and pacing gaits ($n_c = 2$ in general), the convex hull of the contact points is shaped as an extremely limited line. Due to this restriction, It is difficult for the feedback control using the actual ZMP to perform the intended function, which helps stabilize the locomotion, because an output generated from the ZMP controller must lie on the limited line. It means that the CoM follows a pre-designed trajectory according to the footstep pattern without the impact of the actual ZMP feedback. Owing to the position error between the actual ZMP $\boldsymbol{x}_{zmp}$ and the constrained ZMP by the support line $\hat{\boldsymbol{x}}_{zmp}$, the generated moment around the CoM $\boldsymbol{\tau}_c \in \mathbb{R}^3$, which disturbs locomotive stability, can be expressed as follows:

$$\boldsymbol{\tau}_c = (\boldsymbol{x}_{zmp} - \hat{\boldsymbol{x}}_{zmp}) \times \sum_{i=1}^{n_c} \boldsymbol{f}_i. \tag{9}$$

For these reasons, the ZMP criterion is insufficient for the use of the actual CoM state feedback during dynamic gaits, which have two or less legs in stance, even though the ZMP has been widely employed for the CoM motion control of legged robots. To fully use the computed CoM state in Eq. 6, we need to be able to generate the CoM motion regardless of any constraints such as the support polygon. In this context, the CMP $\boldsymbol{x}_{cmp} \in \mathbb{R}^3$ can be one of the alternatives derived from the linear inverted pendulum with flywheel model (LIPFM). Figure 4 shows the dynamics of the LIPFM in sagittal plane but this model can be also explained in lateral plane by decoupling the two planes. Thus, the robot's motion with respect to the world frame $\sum_W$ can be naturally expressed. The derived CMP from the LIPFM is formulated as

$$\boldsymbol{x}_{cmp} = \hat{\boldsymbol{x}}_{zmp} + \frac{1}{f_c^z}\begin{bmatrix} \dot{H}^y & -\dot{H}^x & 0 \end{bmatrix}^T \tag{10}$$

where $f_c^z \in \mathbb{R}$ is the linear momentum rate of change for z-direction, which corresponds to the sum of the vertical GRFs of the stance legs (i.e., $f_c^z = \sum_{i=1}^{n_c} f_i^z$), and $\dot{\boldsymbol{H}}^k \in \mathbb{R}^3$ is the horizontal part of the angular momentum rate change. The CMP is defined as a point where a line parallel to the GRF and passing through the CoM, intersects the ground [20]. In contrast with the ZMP, the CMP allows the CoM to leave the support polygon by the second term

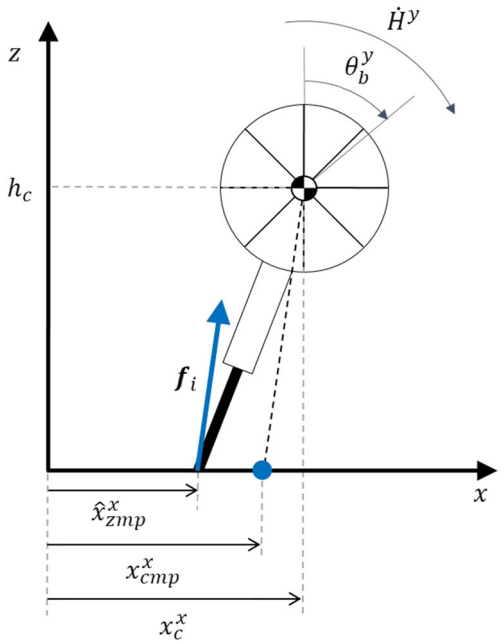

Fig. 4 Linear inverted pendulum with flywheel model

in Eq. 10. Hence, the CoM controller using the CMP criterion enables to recover the robot's balance, even if the quadrupedal robot has the constrained support line.

3.2 Stabilized Balance by Using Angular Momentum

In general, actual angular momentum has been computed by using the centroidal momentum matrix, which requires actual joint angle $\boldsymbol{q}_j$ and joint velocity $\dot{\boldsymbol{q}}_j$ [21]. Because the joint states, $\boldsymbol{q}_j$ and $\dot{\boldsymbol{q}}_j$, are changed by the internal torque $\boldsymbol{A}_{int}$, the computed actual angular momentum also includes the effects of $\boldsymbol{A}_{int}$. Therefore, during locomotion, the computed value through the method is unstable, which is because dynamic motion requires the significant internal torque $\boldsymbol{A}_{int}$ of each joint. On the other hand, we compute the actual GRF $\boldsymbol{f}_i$ in Eq. 5, and this value excludes $\boldsymbol{A}_{int}$. Thus, using $\boldsymbol{f}_i$, we can compute the actual angular momentum rate change without the effects of $\boldsymbol{A}_{int}$, and its horizontal part can be expressed as follows:

$$\dot{\boldsymbol{H}}^k = \boldsymbol{Q} \sum_{i=1}^{n_c} \begin{bmatrix} x_{c,i}^x \\ x_{c,i}^y \\ x_{c,i}^z \end{bmatrix} \times \begin{bmatrix} f_i^x \\ f_i^y \\ f_i^z \end{bmatrix}. \tag{11}$$

In LIPFM shown in Fig. 4, the angular momentum rate change around the CoM can be generated by the two parts: (i) rotation of the robot body and (ii) position error between the desired and actual location of the CoM. In addition, we include an additional term to handle the angular momentum rate of change generated by the swing motion. The designed reference for angular momentum regulation is formulated with the following:

$$\dot{\boldsymbol{H}}^{ref} = \dot{\boldsymbol{H}}_b + \dot{\boldsymbol{H}}_c + \dot{\boldsymbol{H}}_{ff}. \tag{12}$$

The first term is used to maintain the actual angle of the CoM with respect to the desired angle of the CoM as

$$\dot{\boldsymbol{H}}_b = -\boldsymbol{k}_{p,b}\boldsymbol{Q}(\boldsymbol{\theta}_b^{des} - \boldsymbol{\theta}_b) - \boldsymbol{k}_{d,b}\boldsymbol{Q}(\dot{\boldsymbol{\theta}}_b^{des} - \dot{\boldsymbol{\theta}}_b) \tag{13}$$

where $\boldsymbol{\theta}_b \in \mathbb{R}^3$, $\dot{\boldsymbol{\theta}}_b \in \mathbb{R}^3$ and $\boldsymbol{k}_b \in \mathbb{R}^{3\times3}$ are angular position and velocity of the robot body, and diagonal matrix of the gain, respectively. The second term is presented for manipulating the angular momentum generated by the position error between the desired and actual location of the CoM as

$$\dot{\boldsymbol{H}}_c = \boldsymbol{Q}(\boldsymbol{x}_c^{des} - \boldsymbol{x}_c) \times \sum_{i=1}^{n_c} \boldsymbol{f}_i \tag{14}$$

where $\boldsymbol{x}_c^{des} \in \mathbb{R}^3$ is the desired CoM's position that has zero angular momentum rate change. The third term in Eq. 12 is the feed-forward term to control the angular momentum rate change generated from each swing leg. During swing motion, despite the absence of collision, the internal torque $\boldsymbol{A}_{int}$ is significantly detected from the torque sensor. To handle its effects, this third term is employed as follows:

$$\dot{\boldsymbol{H}}_{ff} = \boldsymbol{Q} \sum_{i=1}^{n_{sw}} \boldsymbol{x}_{c,i} \times \boldsymbol{J}_i^{-T} \boldsymbol{\tau}_{ff,i}. \tag{15}$$

Here, $\boldsymbol{\tau}_{ff,i} \in \mathbb{R}^3$ is the feed-forward torque generated by the dynamic swing motion, which corresponds to $\boldsymbol{A}_{int}$ of the i^{th} leg during swing phase, and n_{sw} is the number of swing legs. We exclude the effects of $\boldsymbol{A}_{int}$ to compute $\dot{\boldsymbol{H}}^k$ in Eq. 11, whereas $\boldsymbol{A}_{int}$ of the i^{th} leg during the swing phase (i.e., $\boldsymbol{\tau}_{ff,i}$) is included in the feed-forward term in Eq. 12 to generate the reference trajectory of the angular momentum rate change $\dot{\boldsymbol{H}}^{ref}$. If the actual angular momentum rate of change $\dot{\boldsymbol{H}}^k$ is computed with including $\boldsymbol{A}_{int}$, even though $\dot{\boldsymbol{H}}^{ref}$ has a smooth trajectory, its tracking control is difficult because $\dot{\boldsymbol{H}}^k$ is unstable because of the significant $\boldsymbol{A}_{int}$. Meanwhile, if $\boldsymbol{A}_{int}$ is included in $\dot{\boldsymbol{H}}^{ref}$, the tracking control can be implemented according to the control method, which is explained in Section 3.4.

The desired angular momentum rate change has been conventionally set to zero under the assumption that there is no angular momentum produced by the swing motion [14, 15], or it has been designed based on the commanded motion such as lumped mass' trajectories [16] and the GRF profiles [17]. On the contrary, we design the reference trajectory by using sensory feedback of the robot, and this is not predefined in advance. Therefore, the reference trajectory in Eq. 12 allows the robot to react to disturbances occurred during locomotion. In addition, the second term in Eq. 12 is induced by the linear state error of the CoM (i.e., $\boldsymbol{x}_c^{des}$ and $\boldsymbol{x}_c$). Therefore, the designed reference for angular momentum regulation $\dot{\boldsymbol{H}}^{ref}$ can adjust both the linear translation and angular rotation of the CoM while handling the angular components generated by the swing motion.

3.3 Stepping Motion with Capture Point Dynamics

When the robot cannot stabilize its balance in place because of strong disturbances, generating the stepping motion can be one of the solutions to recover its balance. To do this, we employ the CP, which is defined as a point with respect to the world frame $\sum_W$ to determine where to take steps [22, 23]. Using this point, the CP controller enables the robot to realize the stable balance through the stepping motion. Using the horizontal position and velocity of the CoM, $\boldsymbol{x}_c^k$ and $\dot{\boldsymbol{x}}_c^k$, the CP $\boldsymbol{\xi} \in \mathbb{R}^3$ can be defined as

$$\boldsymbol{\xi} = \boldsymbol{x}_c^k + \frac{\dot{\boldsymbol{x}}_c^k}{\omega} \tag{16}$$

where $\omega = \sqrt{g/h_c}$. Given $\boldsymbol{\xi}$, we can then compute the CP error with the gain such that $\Delta\boldsymbol{\xi} = \boldsymbol{k}_{cp}(\boldsymbol{\xi} - \boldsymbol{\xi}^{ref})$, which is used to implement the CP controller for push response.

Using the centroidal momentum rate of change for x and y axes, we derive the dynamic relationship between the CoM, CMP, and CP. Based on the linear CoM state, horizontal part of the linear momentum rate of change, $\dot{\boldsymbol{L}}^k = \begin{bmatrix} \dot{L}^x & \dot{L}^y & 0 \end{bmatrix}$, can be expressed as

$$\dot{\boldsymbol{L}}^k = m_c \boldsymbol{Q}\ddot{\boldsymbol{x}}_c + \cancelto{\mathbf{0}^3}{m_c \boldsymbol{Q}\boldsymbol{g}_c} \tag{17}$$

where $\mathbf{0}^3$ is the 3-dimensional zero vector. In addition, given the height of the CoM h_c, the angular momentum rate of change for x and y directions can be represented as follows:

$$\dot{\boldsymbol{H}}^k = \dot{\boldsymbol{L}}^k h_c = m_c g(\boldsymbol{Q}\boldsymbol{x}_c - \boldsymbol{x}_{cmp}). \tag{18}$$

Substituting Eq. 17 into Eq. 18 provides the horizontal acceleration of the CoM:

$$\ddot{\boldsymbol{x}}_c^k = \omega^2(\boldsymbol{Q}\boldsymbol{x}_c - \boldsymbol{x}_{cmp}). \tag{19}$$

We obtain the first time derivative of $\boldsymbol{\xi}$ by substituting Eq. 16 and Eq. 19 into differentiated (16) with the following:

$$\dot{\boldsymbol{\xi}} = \omega(\boldsymbol{\xi} - \boldsymbol{x}_{cmp}). \tag{20}$$

Here, Eq. 20 yields the CP dynamics. The CP has the unstable first-order dynamics and it has a feature to diverge from the CMP, whereas the CoM converges to the CMP. Thus, if the CP is stably controlled, we can naturally obtain the stable CoM state.

3.4 CoM Motion Controller

Using the actual angular momentum rate of change in Eq. 11, reference trajectory for angular momentum regulation in Eq. 12, and the CP for the stepping motion in Eq. 16, we can generate the desired CoM motion $\boldsymbol{x}_c^{des}(k) \in \mathbb{R}^3$ by using preview control method [27], which is formulated as

$$\boldsymbol{x}_c^{des}(k) = \sum_{r=1}^{N} G_z(r)\hat{\boldsymbol{x}}_{zmp}(k+r) + \sum_{r=1}^{N} G_h(r)\Delta\boldsymbol{h} + \sum_{r=1}^{N} G_\xi(r)\Delta\boldsymbol{\xi} \tag{21}$$

where each $G(r)$ is the preview gain for each component, which is pre-calculated by considering sampling time [27], and N is the future period, which is set to 0.8 times the gait period. The ZMP and angular momentum regulator can be separately controlled in the CMP planning. The role of the first term in Eq. 21 is generating the CoM trajectory according to the ZMP-based footstep pattern. In the case where $n_c = 2$, generated area of the support polygon σ is significantly limited as the line, as shown in Fig. 5. Therefore, stabilizing the robot's balance via feedback control of the ZMP, which is constrained to always lie on σ, is difficult. For this reason, the ZMP controller is implemented according to the footstep pattern of the gait type without the ZMP feedback control. The second term is used to balance the robot by the angular momentum regulation, where $\Delta\boldsymbol{h}$ is the error of the CMP's second term. By the second term in Eq. 21, the reference trajectory $\dot{\boldsymbol{H}}^{ref}$ is tracked by using the actual angular momentum rate of change $\dot{\boldsymbol{H}}^k$, which is converted value from the measured joint torque $\boldsymbol{\tau}_{ts}$. The reference trajectory $\dot{\boldsymbol{H}}^{ref}$ can be suddenly changed, which is because it is designed by the sensory feedback. Therefore, we employ the preview control method, which is able to predict future motion of the CoM, as well as generate the smooth CoM commands. The third term represents the stepping controller, which is generated based on the CP error with the gain $\Delta\boldsymbol{\xi}$. The stepping motion affects the CoM outside the region ψ, and its boundary is defined by using the threshold $\boldsymbol{x}_{thre}$ as follows:

$$\|\boldsymbol{x}_{cmp} - \hat{\boldsymbol{x}}_{zmp}\| \geq \boldsymbol{x}_{thre}. \tag{22}$$

If the actual CP $\boldsymbol{\xi}$ is located in the kinematic workspace, the robot is able to stabilize its balance by the body motion without taking steps [22]. However, the CP uses the linear CoM state, $\boldsymbol{x}_c^k$ and $\dot{\boldsymbol{x}}_c^k$, to realize the stable balance. Hence, we implement the angular momentum regulator, which considers both the linear and angular state of the CoM, when

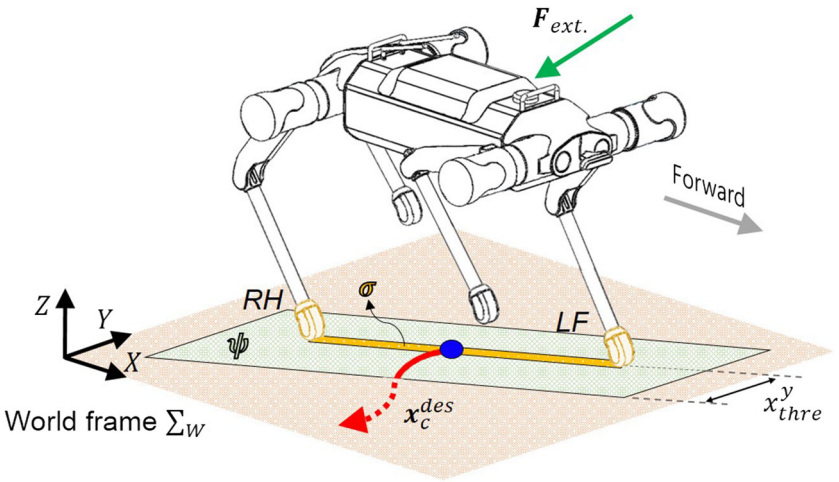

Fig. 5 Balancing strategies proposed in the paper. σ and ψ represent the support line by two feet in contact and the area that can be balanced by the angular momentum regulation during trotting gait, respectively. The projection of the desired CoM position (blue dot) with respect to the world frame $\sum_W$ can be adjusted by two components: body motion (the red solid line) and the stepping motion (the red dashed line)

the robot can recover its balance inside the region ψ without help of the stepping. On the other hand, if the desired CoM position is generated outside the region ψ, this state can be regarded that the stepping motion is required. Thus, the desired CoM motion $\boldsymbol{x}_c^{des}(k)$ is generated by all of the three terms in Eq. 21.

3.5 Jumping Motion Including Aerial Phase

In [24], a jumping gait is classified into three phases: (i) launching phase, (ii) aerial phase, and (iii) landing phase. Among these phases, the launching phase is the most important part because it directly offers the jumping action that has full flight phases. In related works of generating the launching motion, under the assumption that there is no energy loss by the air resistance, launching trajectory that satisfies the desired jump height and take-off velocity [24], and vertical acceleration profile to realize the aerial phase were documented [25]. Based on the similar approach, we generate the feed-forward linear momentum rate of change for z-axis, which is the sum of slightly modified vertical force analyzed in [25]. To interact with the ground during the landing phase, in the event of an impact, switching method to landing control was studied by using the threshold of the knee joint velocity [7]. On the contrary, in case of the quadrupedal robot having a stiff transmission at each joint, if landing control is carried out after the contact is detected, the landing force has already been delivered to the robot. Therefore, we generate the desired motion to mitigate the landing impact in advance, after which f_c^z is maintained close to zero for a set duration in the aerial phase.

3.6 Foothold Generation

For perceptive locomotion, which enables the robot to traverse over various environments, foothold score values $\delta(x, y, z)$ are computed to select an optimal foothold among the valid footholds $\boldsymbol{x}_{f,i} \in \mathbb{R}^3$. The optimal foothold is chosen by two components: (i) a distance between $\boldsymbol{x}_{f,i}$ and a foothold predicted by the robot's actual speed $\boldsymbol{x}_{v,i} \in \mathbb{R}^3$, and (ii) another distance between $\boldsymbol{x}_{f,i}$ and the actual CoM position $\boldsymbol{x}_c$ as

$$\delta(x, y, z) = min(w_1 \frac{\boldsymbol{x}_{v,i} - \boldsymbol{x}_{f,i}}{l_{crit,v}} + w_2 \frac{\boldsymbol{x}_c - \boldsymbol{x}_{f,i}}{l_{crit,c}}) \tag{23}$$

where w is the weight factor, l_{crit} is the critical value to prevent $\delta(x, y, z)$ from exceeding a value of 1. The first term in Eq. 23 is used to minimize the former distance. By doing so, the robot is able to traverse over various terrains while maintaining its actual speed close to the desired speed. The second term is applied to reduce the displacement of the CoM that must be moved.

Given a selected foothold, we generate the swing trajectory to reach the foothold during swing phase. To do this, non uniform basis spline curve is employed, which has several advantages: 1) local modification, 2) tracking velocity control, and 3) a low degree, and its equation can be defined as

$$\boldsymbol{C}_{sw}^{des}(t_{sw}) = \sum_{l=0}^{n} \boldsymbol{B}_{l,p}(t_{sw}) \boldsymbol{W}_l \tag{24}$$

where $\boldsymbol{W}_l = \left[w_0, w_1, ..., w_n\right]$ is the set of control points, and $\boldsymbol{B}_{l,p}(t_{sw})$ is the basis function of degree p according to the preset period of swing phase t_{sw}. Its effectiveness has been verified from several tasks including obstacle avoidance and stair climbing [12, 26].

3.7 Low-Level Controller

From the CoM motion controller and the swing motion generator, we obtain the desired CoM position $\boldsymbol{x}_c^{des}$ and the desired position of the swing legs $\boldsymbol{C}_{sw}^{des}(t_{sw})$. Using the inverse kinematics (IK), both desired positions are computed to the desired joint angle $\boldsymbol{q}_j^{des}$, and we can calculate the desired joint velocity $\dot{\boldsymbol{q}}_j^{des}$ by the first time derivative of $\boldsymbol{q}_j^{des}$. The feedback torque $\boldsymbol{\tau}$ from the joint PD controller is generated as

$$\boldsymbol{\tau} = \boldsymbol{k}_{p,j}(\boldsymbol{q}_j^{des} - \boldsymbol{q}_j) - \boldsymbol{k}_{d,j}(\dot{\boldsymbol{q}}_j^{des} - \dot{\boldsymbol{q}}_j) \tag{25}$$

where $\boldsymbol{k}_{p,j}$ and $\boldsymbol{k}_{d,j}$ are the positive-definite diagonal matrices of proportional and derivative gains for the joint coordinate, respectively.

4 Experiments

The proposed closed-loop whole-body controller by using the actual joint torque feedback is tested on a quadrupedal robot, AiDIN-VI. It weights 43 kg and includes two PCs in the body, that control the robot motion (Intel Celeron Processor, Quad-core 1.6 GHz), and recognize the terrain's information with the foothold generation (Intel NUC). The actuator module has a torque sensor that provides the joint torque at 1 kHz, and this value is computed to actual GRF at 500 Hz. The CoM motion controller has a sampling time of 1.5 ms and the control loop for the low-level controller run at 1 kHz. The gain setup of the controller is listed in Table 1. A more detailed description of the robot is available in [18].

With the proposed controller, AiDIN-VI can demonstrate both several gait types such as trotting, pacing, jumping, and walking and balance recovery reaction to external pushes. Figure 6 shows its locomotive abilities on various environments. The robot enables to trot while pulling a

Table 1 Gain setup of the controller

Parameter	Value	Parameter	Value
$\boldsymbol{K}_{p,b}$ (Nm/rad)	diag(1000,1000,500)	$\boldsymbol{k}_{d,b}$ (sNm/rad)	diag(50,50,25)
$k_{p,j}$ (Nm/rad)	500	$k_{d,j}$ (sNm/rad)	10
$\boldsymbol{k}_{cp}$ (m/m)	diag(250,250)		

customized cart with two passengers ($\approx 150\ kg$) at a speed of 0.5 m/s (4 Hz). Its speeds can be increased from 0 m/s up to a maximum speed of 1.2 m/s during the trot gait on flat terrain. The pace gait is able to run at a maximum frequency of 4 Hz on several surfaces. In addition, AiDIN-VI can perform the jumping motion, which has a jump height of 0.05 m. In the case of the walking gait, we tested its traversability over various terrains including flat terrain, slope, and stair. Through the implementation of the proposed controller, AiDIN-VI can ascend the stair, which has 26 ° inclination and 0.16 m height (23.5% of its maximum leg length). Furthermore, we can obtain the terrain's inclination from the embedded camera, and using this information, the robot is able to walk on the slope up to 25 °. Figure 6e shows the reaction to external disturbances. Through the trot gait-based stepping motion, the robot can stabilize its balance in the presence of external pushes. A related video is available at https://youtu.be/yNwGWnu_R6o.

To obtain the external joint torque $\boldsymbol{A}_{ext}$, the joint torque measured by the torque sensor $\boldsymbol{\tau}_{ts}$ is compensated by using the internal torque $\boldsymbol{A}_{int}$ from Eq. 4. Subsequently, $\boldsymbol{A}_{ext}$ is transformed to the actual GRF $\boldsymbol{f}_i$ based on Eq. 5. Figure 7 shows (a) the measured joint torque (τ^s, τ^h, and τ^k) and (b) the force (f^x, f^y, and f^z) with and without the compensation of the internal torque $\boldsymbol{A}_{int}$. All of the data were measured during the trajectory tracking, which achieves the speeds from 0 m/s up to 1 m/s (4 Hz) without contact, and the used trajectory is designed by using the NUBS curve [12]. The increased speed v_x^{des} is represented in the upper left graph with τ^{s_c} and τ^s. Here, joint indexes s, h, and k indicate the scapula, hip, and knee joint, and the white and shaded boxes indicate the stance and swing phases, respectively. In addition, the index c denotes the compensated value, in the absence of c, it refers to the value without the compensation of $\boldsymbol{A}_{int}$. During the experiment, the maximum absolute value of the measured joint torque (τ^s, τ^h, and τ^k) was 5.5, 8.1, and 3.3 Nm, and the maximum absolute value of the force (f^x, f^y, and f^z) was 21.2, 8.9, and 26.3 N, which are directly marked as the dots in each graph. Through the acquired value, we can verify that the significant internal torque $\boldsymbol{A}_{int}$ was detected from the torque sensor during dynamic motion, even though the robot has low inertia legs and even in non-contact. This is because the measured value includes both the internal $\boldsymbol{A}_{int}$ and external torque $\boldsymbol{A}_{ext}$. To obtain the purely external torque and force, we solve the inverse dynamics, which corresponds approximately to $\boldsymbol{A}_{int}$, and then compute $\boldsymbol{A}_{ext}$. The maximum absolute value of the compensated torque (τ^{s_c}, τ^{h_c}, and τ^{k_c}) was 1.1, 1.6, and 0.9 Nm. In addition, the maximum absolute value of the compensated force, which was computed by the relationship: $\left[f^{x_c}\ f^{y_c}\ f^{z_c}\right]^T = (\boldsymbol{J}_i^T)^{-1}\left[\tau^{s_c}\ \tau^{h_c}\ \tau^{k_c}\right]^T$, was 3.5, 2.0, and 4.3 N. Here, each compensated value is not completely converged to zero because of the modeling error. In practice, it is difficult to perfectly compensate

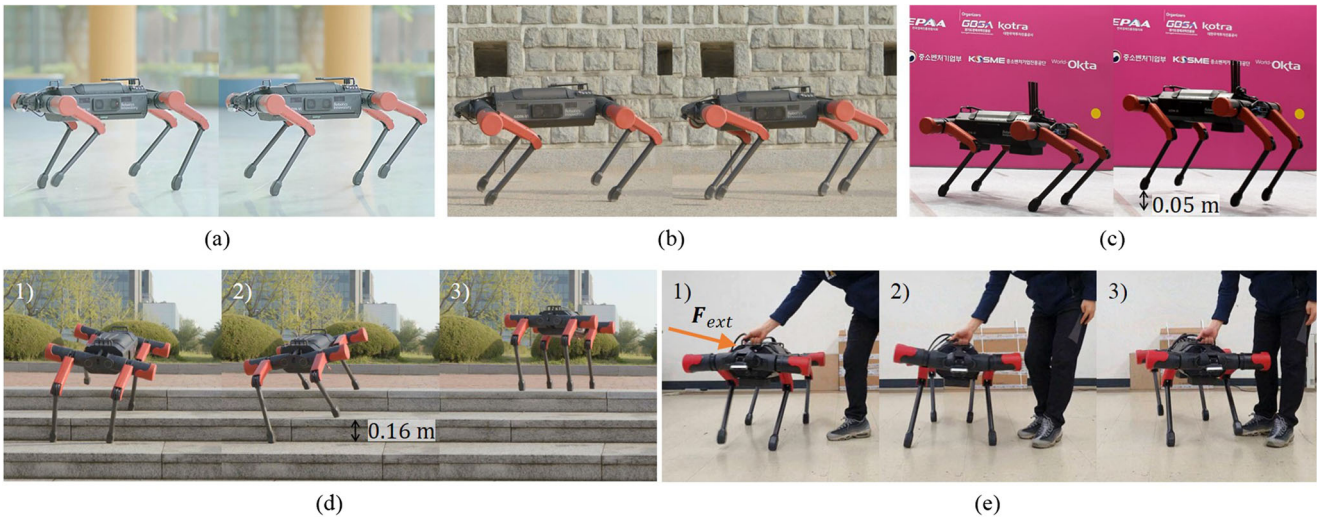

Fig. 6 The proposed controller, tested on a quadrupedal robot, AiDIN-VI, can demonstrate several gait types including **a** trotting, **b** pacing, **c** jumping, and **d** walking on various environments, as well as **e** locomotive abilities with the occurrence of external pushes. A full video is available at https://youtu.be/yNwGWnu_R6o

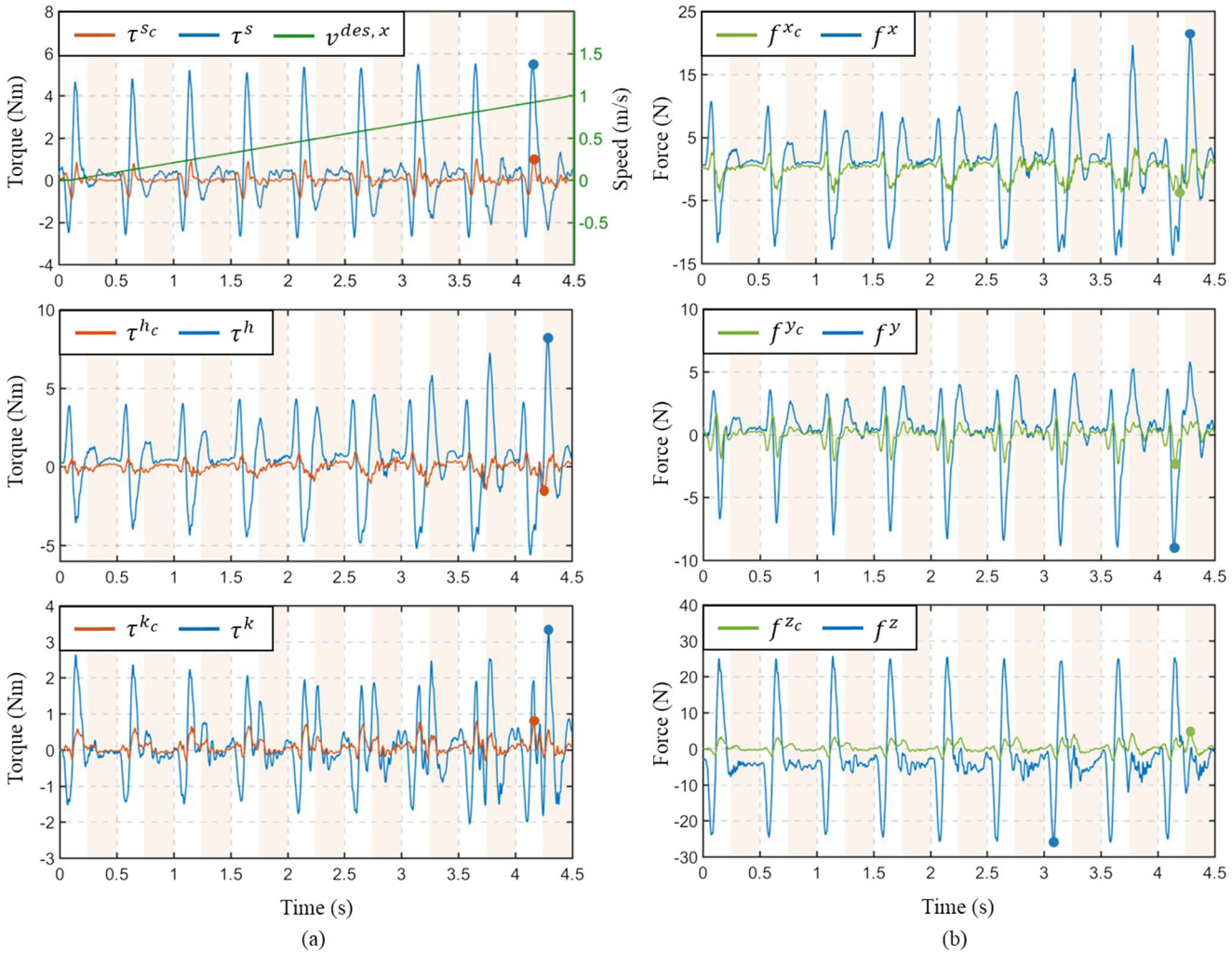

Fig. 7 **a** The measured joint torque (τ^s, τ^h, and τ^k) and **b** the force (f^x, f^y, and f^z) with and without the compensation of the internal torque $\boldsymbol{A}_{int}$ during trajectory tracking that satisfies the speeds from 0 *m/s* to 1 *m/s* and in non-contact

the effects of $\boldsymbol{A}_{int}$. However, through this compensation method, we enabled to obtain the significantly reduced values of the torque and force, and using the compensated force (f^{x_c}, f^{y_c}, and f^{z_c}), we can compute the actual angular momentum $\dot{\boldsymbol{H}}$ without affecting $\boldsymbol{A}_{int}$, which is used to track the desired angular momentum rate of change in Eq. 12.

Figure 8a shows the reference trajectory for x-axis ($\dot{H}^{ref,x}$) and desired CoM motion for y-axis ($x_c^{des,y}$). To validate that the proposed controller reacts to the swing leg dynamics, as well as translation and rotation of the CoM, we intended to generate $\dot{H}^{ref,x}$ and $x_c^{des,y}$ by one term in Eq. 12 at a certain time. Thus, for first 3 *s*, $\dot{H}_{ff}^x$, which was generated while the robot trotted on the ground, affected the CoM's motion and the others ($\dot{H}_c^x$ and $\dot{H}_b^x$) were set to zero. Similarly, $\dot{H}_c^x$ and $\dot{H}_b^x$ solely regulate the angular momentum for next 3 *s* and last 3 *s*, respectively, as shown in Fig. 8b. Here, $\dot{H}_{ff}^x$, $\dot{H}_c^x$, and $\dot{H}_b^x$ were the generated angular momentum rate of change for x-axis by the swing leg dynamics, translation ($\Delta x_c^y = x_c^{des,y} - x_c^y$), and rotation of the CoM ($\Delta\theta_b^x = \theta_b^{des,x} - \theta_b^x$), respectively. By integrating the three components ($\dot{H}_{ff}^x$, $\dot{H}_c^x$, and $\dot{H}_b^x$), we obtained the reference CoM motion for x-axis $\dot{H}^{ref,x}$ and, based on this reference with the preview controller, $x_c^{des,y}$ was generated to recover the robot's balance as shown in Fig. 8a. Through this experiment, we verify that the designed reference trajectory $\dot{\boldsymbol{H}}^{ref}$ can react to the generated angular momentum rate of change by the swing leg dynamics, as well as translation and rotation of the CoM during locomotion.

To evaluate the effectiveness of the proposed whole-body controller, we compared the values of the actual angular state of the CoM while the robot paced at a speed of 0.2 *m/s* on flat ground, which was separately performed by the proposed controller and the ZMP controller. Figure 9a shows the desired CoM position for y-axis generated by the ZMP controller ($\hat{x}_{zmp}^{des,y}$), and the reference and desired

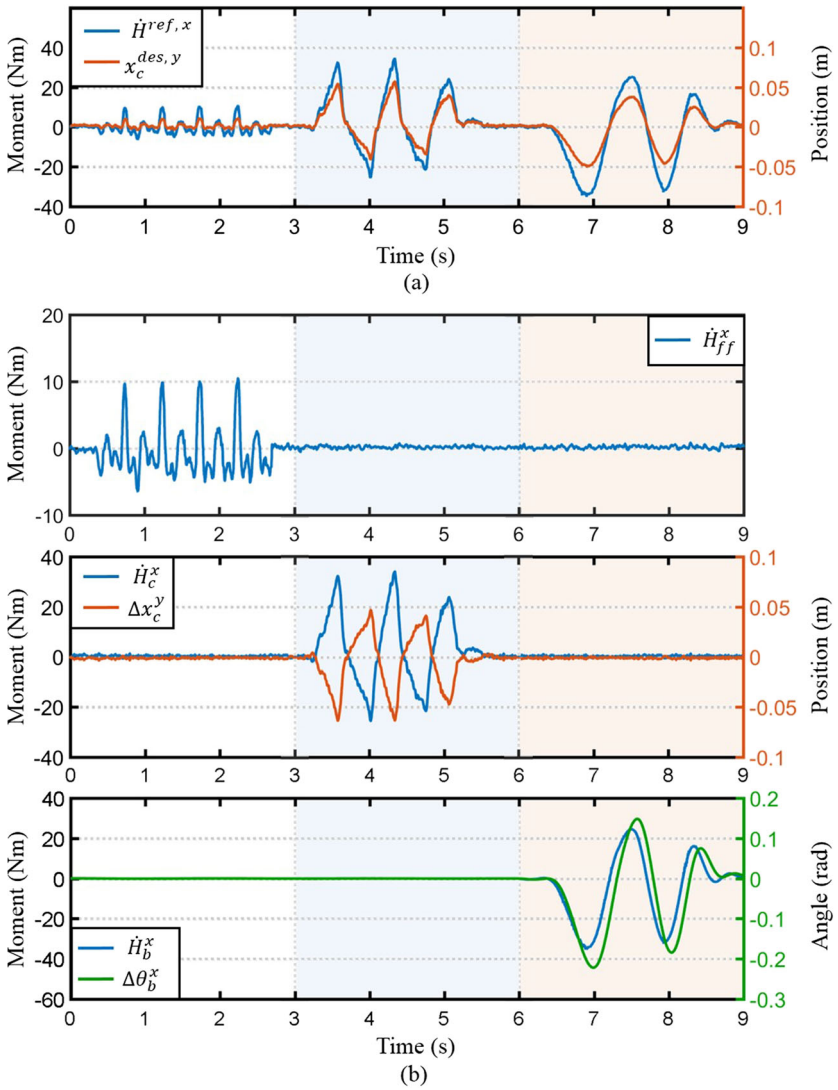

Fig. 8 **a** The desired CoM motion and reference trajectory for angular momentum regulation generated by integrating **b** the each term ($\dot{\boldsymbol{H}}_{ff}$, $\dot{\boldsymbol{H}}_c$, and $\dot{\boldsymbol{H}}_b$)

CoM trajectory for y-direction by the proposed controller ($x_c^{ref,y}$ and $x_c^{des,y}$). Here, $x_c^{ref,y}$ represents the sum of the each component in Eq. 21 before the previewing, and this value is rapidly changed in the graph because it is designed by using the sensory feedback. Thus, it is difficult to directly follow $x_c^{ref,y}$, and exact tracking control of $x_c^{ref,y}$ may rather disturb to realize the stable balance. For this reason, we employ the preview control method to obtain the desired CoM trajectory $x_c^{des,y}$ transformed from $x_c^{ref,y}$. The preview controller allows us to predict the future CoM motion, as well as generate the smooth CoM trajectory as $x_c^{des,y}$ in the graph.

Figure 9b shows the actual angular momentum rate of change ($\dot{H}^x$), angular position and velocity of the CoM (θ^x and $\dot{\theta}^x$) for x-axis. These values can be used as the indices to discern the locomotive stability of the robot because $\dot{H}^x$, θ^x, and $\dot{\theta}^x$ are related to the robot's balance. Here, right subscript zmp indicates the measured values during the pace gait by the ZMP controller, and $\dot{H}^x$, θ_b^x, and $\dot{\theta}_b^x$ are the values acquired during the proposed controller-based pace gait. The ZMP controller cannot generate the desired CoM position $\hat{x}_{zmp}^{des,y}$ outside the support polygons. On the contrary, the proposed controller can generate the desired CoM position x_c^{des} regardless of the limited support lines during the pace gait. Consequently, during the experiment by the proposed controller, the average values of $\dot{H}^x$, θ_b^x, and $\dot{\theta}_b^x$ were within $\pm 19.8\ Nm$, $\pm 2.4\ °$, $\pm 20.0\ °/s$, respectively. In contrast, during the pace gait by the ZMP controller, the angular position and velocity of the CoM, as well as the angular momentum rate of change significantly deviated from their respective average values measured when the robot paced by the proposed controller, and maximum absolute values of $\dot{H}_{zmp}^x$, θ_{zmp}^x, and $\dot{\theta}_{zmp}^x$ were $40.5\ Nm$, $4.6\ °$, $46.3\ °/s$, which are expressed as the dots in the graphs, respectively. Especially, contrary to θ_b^x that was maintained within its average value ($\pm 2.4\ °$), θ_{zmp}^x was

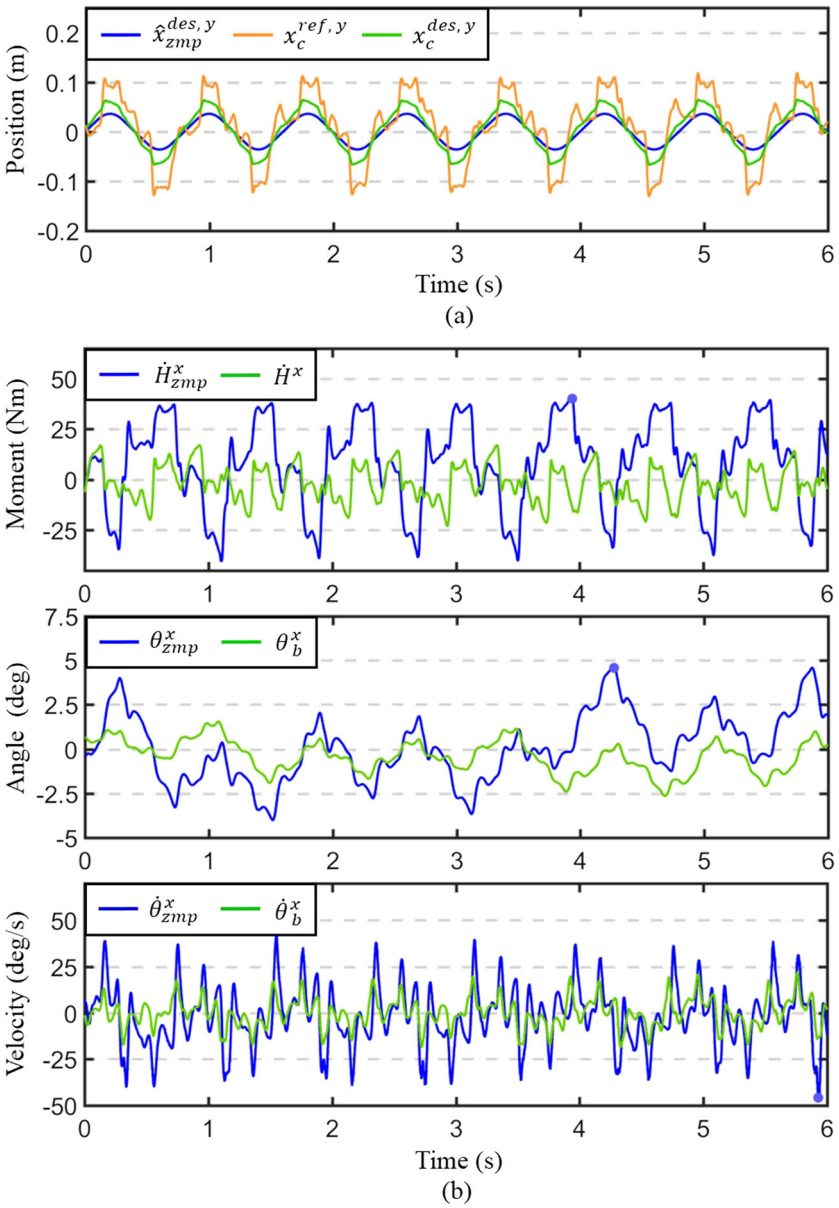

Fig. 9 Experimental results of the robot pacing by using the proposed whole-body controller and ZMP controller. **a** The desired and reference CoM motion and **b** the actual angular state of the CoM for x-axis

decreased for 3 s, then its value was increased for the next 3 s, which means that the robot's balance was not stabilized by the ZMP controller. The experimental results validate that the proposed controller can improve the locomotive stability of the robot compared to the ZMP controller.

To verify the proposed whole-body controller that reacts to external pushes, we applied three pushes while the robot was on flat surface. Figure 10 shows the desired CoM position with the stepping motion generation. During the experiment, 1^{st} and 2^{nd} pushes caused the CoM's motion to stabilize the robot's balance. However, both the pushes did not induce the stepping motion because the desired CoM position for y-direction $x_c^{des,y}$ was within the threshold x_{thre}^{y}, which was set to 0.05 m. Thus, the robot was able to react to the pushes by the body motion in place. In contrast, when the 3^{rd} push was applied to the robot, the controller generated the desired CoM motion for y-axis $x_c^{des,y}$ over x_{thre}^{y}, which means that the robot becomes unable to recover its balance in place and stepping motion is required. Therefore, the CoM motion controller generated the stepping motion, and the robot trotted to stabilize its balance with the angular momentum regulator.

5 Conclusion

In this paper, contrary to the general whole-body control strategies, which compute the feed-forward force to generate the joint torque at every time step, we presented a feasible approach, that fully uses the actual joint torque

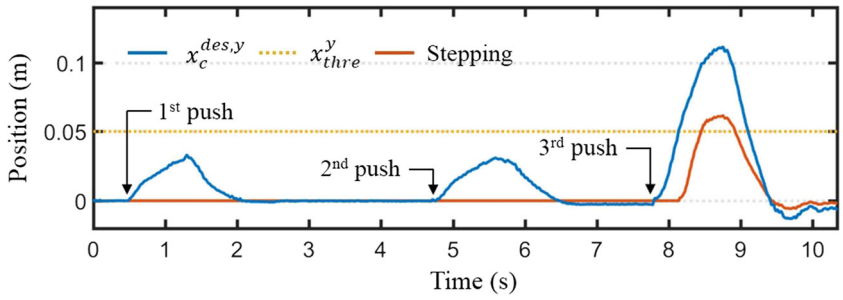

Fig. 10 The desired CoM position and the stepping motion according to the threshold

to demonstrate dynamic locomotion, with a compensation method to obtain the external torque, GRF, and CoM state. In addition, a reference trajectory to regulate the angular momentum, CP controller for the stepping motion, and foothold generation method were presented. The proposed controller enabled a quadrupedal robot, AiDIN-VI, to demonstrate the four gait types (trotting, pacing, jumping, and walking) on various environments, as well as the locomotive capabilities under external pushes were validated.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s10846-021-01418-x.

Acknowledgements This work was supported by the Robot Industry Core Technology Development Program, 20000944, funded by Ministry of Trade, Industry and Energy (MI, Korea).

Author Contributions The first draft of the manuscript was written by Young Hun Lee and all authors commented on previous versions of the manuscript. Young Hun Lee, Yoon Haeng Lee, Hyunyong Lee, Hansol Kang, Jun Hyuk Lee, and Hyouk Ryeol Choi developed a quadrupedal robot to test the study. All authors read and approved the final manuscript.

Funding This work was supported by the Robot Industry Core Technology Development Program, 20000944, funded by Ministry of Trade, Industry and Energy (MI, Korea).

Availability of Data and Materials The datasets used and analysed during the study are available from the first author on reasonable request.

Declarations

Ethics Approval Not applicable. (This study does not involve human participants, their data or biological material).

Competing interests The authors declare that they have no competing interests.

References

1. Nagamatsu, Y., Shirai, T., Suzuki, H., Kakiuchi, Y., Okada, K., Inaba, M.: Distributed torque estimation toward low-latency variable stiffness control for gear-driven torque sensorless humanoid. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 5239–5244 (2017)
2. Jin, B., et al.: Joint torque estimation toward dynamic and compliant control for gear-driven torque sensorless quadruped robot. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4630–4637 (2019)
3. Chuah, M.Y., Kim, S.: Enabling force sensing during ground locomotion: a bio-inspired, multi-axis, composite force sensor using discrete pressure mapping. IEEE Sensors J. **14**(5), 1693–1703 (2014)
4. Kaslin, R., Kolvenbach, H., Paez, L., Lika, K., Hutter, M.: Towards a passive adaptive planar foot with ground orientation and contact force sensing for legged robots. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2707–2714 (2018)
5. Unitree: http://www.unitree.cc (2017)
6. Focchi, M., del, P.A., Havoutis, I., Featherstone, R., Caldwell, D.G., Semini, C.: High-slope terrain locomotion for torque controlled quadruped robots. Auton. Robots **41**(1), 259–272 (2017)
7. Nguyen, Q., Powell, M.J., Katz, B., Carlo, J.D., Kim, S.: Optimized jumping on the MIT Cheetah 3 robot. In: 2019 IEEE International Conference on Robotics and Automation, pp. 7448–7454 (2019)
8. Winkler, A.W., Farshidian, F., Neunert, M., Pardo, D., Buchli, J.: Online walking motion and foothold optimization for quadruped locomotion. In: 2017 IEEE International Conference on Robotics and Automation, pp. 5308–5313 (2017)
9. Park, H.-W., Wensing, P.M., Kim, S.: High-speed bounding with the MIT Cheetah 2: Control design and experiments. Int. J. Robot. Res. **36**(2), 167–192 (2017)
10. Liu, C., Yang, J., An, K., Chen, Q.: Rhythmic-Reflex Hybrid adaptive walking control of biped robot. J. Intell. Robot. Syst. **94**, 603–619 (2019)
11. Mason, S., Rotella, N., Schaal, S., Righetti, L.: An MPC walking framework with external contact forces. In: 2018 IEEE International Conference on Robotics and Automation, pp. 1785–1790 (2018)
12. Lee, Y.H., et al.: Whole-body motion and landing force control for quadrupedal stair climbing. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4746–4751 (2019)
13. Herr, H., Popovic, M.: Angular momentum in human walking. J. Exp. Biol. **211**, 467–481 (2008)
14. Shafiee-Ashtiani, M., Yousefi-Koma, A., Shariat-Panahi, M.: Robust bipedal locomotion control based on model predictive control and divergent component of motion. In: 2017 IEEE International Conference on Robotics and Automation, pp. 3505–3510 (2017)
15. Chang, C., Huang, H., Hsu, H., Cheng, C.: Humanoid robot push-recovery strategy based on CMP criterion and angular momentum regulation. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 761–766 (2015)
16. Seyde, T., Shrivastava, A., Englsberger, J., Bertrand, S., Pratt, J., Griffin, R.J.: Inclusion of angular momentum during planning

for capture point based walking. In: 2018 IEEE International Conference on Robotics and Automation, pp. 1791–1798 (2018)
17. Herzog, A., Rotella, N., Schaal, S., Righetti, L.: Trajectory generation for multi-contact momentum control. In: 15Th IEEE-RAS International Conference on Humanoid Robots, pp. 874–880 (2006)
18. Lee, Y.H., et al.: Force-controllable quadruped robot system with capacitive-type joint torque sensor. In: 2019 IEEE International Conference on Robotics and Automation, pp. 6777–6782 (2019)
19. Wensing, P.M., Wang, A., Seok, S., Otten, D., Lang, J., Kim, S.: Proprioceptive actuator design in the MIT cheetah: Impact mitigation and high-bandwidth physical interaction for dynamic legged robots. IEEE Trans. Robot. **33**(3), 509–522 (2017)
20. Popovic, M.B., Goswami, A., Herr, H.: Ground reference points in legged locomotion: Definitions, biological trajectories and control implications. Int. J. Robot. Res. **24**(12), 1013–1032 (2005)
21. Orin, D.E., Goswami, A.: Centroidal momentum matrix of a humanoid robot: Structure and Properties. In: 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 653–659 (2008)
22. Pratt, J., Carff, J., Drakunov, S., Goswami, A.: Capture Point: a step toward humanoid push recovery. In: 6Th IEEE-RAS International Conference on Humanoid Robots, pp. 200–207 (2006)
23. Englsberger, J., Ott, C.: Integration of vertical com motion and angular momentum in an extended capture point tracking controller for bipedal walking. In: 12Th IEEE-RAS International Conference on Humanoid Robots, pp. 183–189 (2012)
24. Bergonti, F., Fiorio, L., Pucci, D.: Torque and velocity controllers to perform jumps with a humanoid robot: Theory and Implementation on the iCub Robot. In: 2019 IEEE International Conference on Robotics and Automation, pp. 3712–3718 (2019)
25. Linthorne, N.P.: Analysis of standing vertical jumps using a force platform. Amer. J. Phys. **69**(11), 1198–1204 (2001)
26. Lee, Y.H., et al.: Trajectory design and control of quadruped robot for trotting over obstacles. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 4897–4902 (2017)
27. Kajita, S., et al.: Biped walking pattern generation by using preview control of zero-moment point. In: 2003 IEEE International Conference on Robotics and Automation, pp. 1620–1626 (2003)

Young Hun Lee received the B.S. degree in mechanical engineering from Kyonggi University, Suwon, South Korea, in 2014. He is currently working toward the Ph.D. degree at the School of Mechanical Engineering, Sungkyunkwan University, Suwon, South Korea.

Since 2014, he has been with the Robotics Innovatory, School of Mechanical Engineering, Sungkyunkwan University, and has been working on the development and control of quadrupedal robots. His research interests include legged robots, optimal control, and motion planning.

Yoon Haeng Lee received the B.S. degree in mechanical engineering from Wonkwang University, Iksan, South Korea, in 2012 and the M.S. and Ph.D. degrees in mechanical engineering from the Sungkyunkwan University, Suwon, South Korea, in 2012 and 2014, respectively.

He has been working with the Robotics Innovatory, Sungkyunkwan University during his studies on development and control for various quadruped robots since 2012. He is Cofounder of AIDIN ROBOTICS Inc. Suwon, Suwon, South Korea. His research interests include actuation system, force/torque control, system integration, humanrobot collaboration, safety sensor, and legged robot systems.

Hyunyong Lee received the B.S. degree in mechanical engineering from Sungkyunkwan University, Suwon, South Korea, in 2016. Currently, he is studying dynamic whole-body control and power management system toward the Ph.D. degree at Robotics Innovatory, School of Mechanical Engineering, Sungkyunkwan University, Suwon.

He is Cofounder of AIDIN ROBOTICS Inc. Suwon, Suwon. His current research interest includes Coanda effect, drones, robot system and mechanism, and quadruped robot.

Hansol Kang received the B.S. degree in mechanical engineering from Ajou University, Suwon, South Korea, in 2017. He is currently working toward the Ph.D. degree at the School of Mechanical Engineering, Sungkyunkwan University, Suwon.

He has been with the Robotics Innovatory, School of Mechanical Engineering, Sungkyunkwan University, since 2017. After joining the lab, he has been working on the development and control of quadruped robots. His research interests include legged robot system and reinforcement learning.

Jun Hyuk Lee received the B.S. degree in mechanical design engineering in 2019 from Korean Polytechnic University, Siheung, South Korea, where he is currently working toward the M.S. degree at the School of Mechanical Engineering, Sungkyunkwan University, Suwon, South Korea.

He has been with the Robotics Innovatory, School of Mechanical Engineering, Sungkyunkwan University, since 2019. His research interests include quadruped walking.

Ji Man Park received the B.S. degree in mechanical engineering from Sungkyunkwan University, Suwon, South Korea, in 2020. He is currently working toward the Ph.D. degree at the School of Mechanical Engineering, Sungkyunkwan University, Suwon.

Since 2020, he has been with the Robotics Innovatory, School of Mechanical Engineering, Sungkyunkwan University, and has been working on the development and control of quadrupedal robots. His research interests include legged robot systems, mechanism design, and whole-body control.

Yong Bum Kim received the B.S. degree in robotics from Ritsumeikan University, Kyoto, Japan, in 2012, and the Ph.D. degree from the School of Mechanical Engineering, Sungkyunkwan University, Seoul, South Korea, in 2020.

He has been working with the Robotics Innovatory, Sungkyunkwan University during his studies on development of robot-sensing systems with six axis F/T sensor and torque sensor since 2012. He is Cofounder of AIDIN ROBOTICS Inc. Suwon, Suwon, South Korea. His research interests include the control of a precision robot hand manipulator, surgical robots, and robot-sensing systems.

Hyungpil Moon received the B.S. and M.S. degrees in mechanical engineering from the Pohang University of Science and Technology (POSTECH), Pohang, South Korea, in 1996 and 1998, respectively. He received the Ph.D. degree in mechanical engineering from the

University of Michigan, Ann Arbor, MI, USA, in 2005. He was a Postdoctoral Fellow with Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA.

He is currently a Professor with the Department of Mechanical Engineering, Sungkyunkwan University, Suwon, South Korea. His research interests are robotic manipulation, task-motion planning, autonomous drone racing, SLAM, hydraulic/pneumatic robots, and polymer-based sensors and actuators.

Ja Choon Koo received the B.S. degree in mechanical engineering from Hanyang University, Seoul, South Korea, in 1989 and the M.S. and Ph.D. degrees in mechanical engineering from the University of Texas, Austin, TX, USA, in 1992 and 1997, respectively.

He is a Professor with School of Mechanical Engineering, Sungkyunkwan University, Suwon, South Korea. His primary research interests are in the field of design, analysis, and control of dynamics systems with emphasis on mechatronic and robotic applications. Formerly, he was a Research Engineer with IBM, San Jose, CA, USA, and an Engineering Staff Member with SISA, San Jose, CA, USA. He was a Visiting Scholar with the University of California, Oakland, CA, USA, and with the IBM Research.

Dr. Koo is an Editor for Journal of Mechanical Science and Technology and a Technical Editor for IEEE Transactions on Mechatronics. He has also served for many editorial boards including IEEE Transactions on Robotics, Microsystem Technologies, and IEEE Robotics and Automation Letters.

Hyouk Ryeol Choi (Fellow, IEEE) received the B.S. degree from Seoul National University, Seoul, Korea, in 1984, the M.S. degree from the Korea Advanced Institute of Science and Technology, Daejeon, Korea, in 1986, and the Ph.D. degree from Pohang University of Science and Technology, Pohang, Korea, in 1994, respectively, all in mechanical engineering.

From 1986 to 1989, he was an Associate Research Engineer with the IT Research Center, LG Electronics. From 1993 to 1995, he was a Postdoctoral Researcher with Kyoto University, Kyoto, Japan. From 1999 to 2000, he visited the National Institute of Advanced Industrial Science and Technology, Japan, as a JSPS Fellow. From 2008 to 2009, he was a Visiting Professor with the University of Washington, Seattle, WA, USA. Since 1995, he has been a Professor with the School of Mechanical Engineering, Sungkyunkwan University, Suwon, South Korea. His research interests include soft robotics, robotic mechanisms, field applications of robots, dexterous robotic hands, and manipulation.

Dr. Choi was an Associate Editor for IEEE Transactions on Robotics, Technical Editor for the IEEE/ASME Transactions on Mechatronics, and the Senior Editor for Journal of Intelligent Service Robotics. He was a founding Co-chair of the IEEE RAS Technical Committee "Robot Hand, Grasping and Manipulation." He was the General Chair of the 2012 IEEE Conference on Automation Science and Engineering (CASE), Seoul, South Korea.