

# Humanoid Balance Control using Centroidal Angular Momentum based on Hierarchical Quadratic Programming\*

Myeong-Ju Kim<sup>1</sup>, Daegyu Lim<sup>1</sup>, Gyeongjae Park<sup>1</sup> and Jaeheung Park<sup>1,2</sup>

**Abstract**—Maintaining balance to external pushes is one of the most important features for a humanoid to walk in a real environment. In particular, methods for counteracting to pushes using the centroidal angular momentum (CAM) control have been actively developed. In this paper, a CAM control scheme based on hierarchical quadratic programming (HQP) is proposed. The scheme of the CAM control consists of CAM tracking control and initial pose return control, which is hierarchically operated based on HQP to ensure the priority of CAM tracking performance. The proposed method is implemented in a capture point (CP) feedback control framework. Through simulations and experiments, the proposed method demonstrated more stable balance control performance than the previous method when the humanoid is walking in the presence of external perturbation.

## I. INTRODUCTION

Recovering balance against external perturbation is one of the most important goals in the field of humanoid research. For a humanoid robot to perform tasks in a real environment, it must be accompanied by the ability to walk while maintaining balance in the presence of external perturbations.

Researches on maintaining balance in terms of humanoid walking have been actively developed for the past two decades. In the early stages of the study, a humanoid robot was simply abstracted into Linear Inverted Pendulum Model (LIPM) [1], [2], and the dynamic relationship between the center of mass (CoM) and zero moment point (ZMP) [3] was described based on LIPM. Since then, under the concept that the robot can walk while maintaining balance if ZMP is located inside the convex hull of the foot-support area, studies that set ZMP as a control objective and converge ZMP error to zero have been proposed. Nakaura et al. [4] designed an LQR controller with CoM input and ZMP output to improve the tracking performance of ZMP. Choi et al. [5] proposed a controller for generating CoM velocity to minimize ZMP errors. Kim et al. [6] suggested an observed state feedback-based ZMP controller considering the joint elasticity of the robot. These studies contributed to the walking stability by improving the tracking performance of ZMP, but its performance was not guaranteed on the uneven or inclined ground.

\*This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2021R1A2C3005914).

<sup>1</sup>Myeong-Ju Kim, Daegyu Lim, Gyeongjae Park and Jaeheung Park are with the Department of Intelligence and Information, Seoul National University, Republic of Korea. {myeong-ju, dgyo3784, rudwol301, park73}@snu.ac.kr

<sup>2</sup>Jaeheung Park is also with the Advanced Institutes of Convergence Technology, Republic of Korea and with ASRI, RICS, Seoul National University, Republic of Korea. He is the corresponding author of this paper.

The Capture Point (CP) (also referred to as the *extrapolated center of mass* (XCoM)) [7], [8] has begun to be widely used to cope with external perturbations. The CP has a characteristic that the CoM converges to the CP, which means that when the CP is controlled to be placed at the desired position, the CoM converges to that position. In addition, CP is also used as a criterion for evaluating the disturbed state of the robot [9], [10]. Because of these characteristics, CP has been used as a control objective. Englsberger et al. [11] proposed a method to control CP based on LIPM. The ZMP for tracking reference CP was calculated, and the CoM motion was generated based on the ZMP controller in [5]. Morisawa et al. [12] suggested a method to simultaneously control the CP and the ZMP. The control input (ZMP) is composed of the sum of the PI control of the CP error and the P control of the ZMP error and is controlled by the ZMP controller in [13]. Joe et al. [14] reduced CP error through a walking pattern generator based on the LIPM-based Model Predictive Control (MPC) scheme. These studies have improved the walking performance of robots in the presence of external perturbation. However, only the lower body was used to recover balance, and the controllable contact force or ZMP is limited to the support foot area.

Research on balance control using the upper body has also been developed. As the magnitude of the external perturbation increases, human strategies to control angular momentum using the upper body were analyzed to be effective [15], and several studies were carried out to apply this lesson to humanoids. Pratt et al. mentioned that adding rotational inertia to the LIPM allows the humanoid to control its centroidal angular momentum (CAM), just like human beings do, and proposed Linear Inverted Pendulum plus Flywheel Model (LIPFM) [7]. The LIPFM is a robust model capable of dealing with the CAM, but since humanoids cannot rotate infinitely like a flywheel, applying this model to humanoids requires consideration of the mechanical limitations of joints. To resolve this problem, studies have been introduced in which the method of rotating the torso using hip joints to generate CAM and then returning the torso to the initial pose is sequentially conducted based on time-scheduling [16], [17]. However, such a time-scheduling-based method may not be able to generate the desired CAM within a predetermined time, and it is difficult to cope with the disturbance by adjusting the time according to various situations. Jeong et al. [9] realized the desired CAM for push recovery through QP-based online optimization. Afterward, the robot was returned to the initial pose by generating

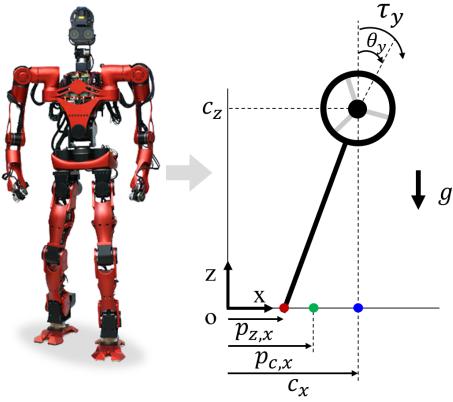


Fig. 1. Linear inverted pendulum plus flywheel model

the return torque based on the current state of the robot. The above studies assumed that the operational point of the CAM was at the hip. This assumption is a simple way to handle CAM, but it is difficult to generate accurate angular momentum in the CoM frame. Studies using the whole-body model were also introduced. In a study by Schuller et al. [18], when the ankle torque to counteract for external pushes exceeded the threshold contact torque, the additional CAM was generated from whole-body motion. In this approach, the CAM control and the initial pose return control are operated by a soft hierarchy-based QP controller, which may cause CAM tracking performance loss due to the initial pose return cost.

In this study, the CAM tracking control and the initial pose return control are proposed using HQP [19] for the improved control performance of the CAM tracking control. The HQP-based CAM controller is implemented in the previously developed CP feedback walking controller introducing LIPFM.

The main contribution of this study is the HQP-based CAM control scheme. The scheme of CAM control consists of a CAM tracking control and an initial pose return control, both of which operate hierarchically based on HQP. Here, the HQP problem was formulated so that the initial pose return task would not affect the CAM tracking task more than a tolerance. The tolerance is varied according to the magnitude of the desired CAM, ensuring high CAM tracking performance during large CAM tracking task and generating initial pose return motion during small CAM tracking task. Using the HQP-based CAM control scheme, more improved balance control performance of humanoid is presented than using the QP-based CAM controller. The proposed algorithm was implemented through simulation and experiment using our humanoid robot TOCABI, and its performance was demonstrated in the presence of external pushes.

## II. BACKGROUND

### A. Linear Inverted Pendulum plus Flywheel Model

The LIPFM was proposed as a linearized abstract model (Fig. 1) to deal with the CAM of the robot by Pratt et al. [7]. In LIPFM, the total mass is concentrated in the CoM, and the

height of the CoM from the ground is constant. In addition, the base joint is passive and the rotational joint located at the CoM can generate reaction torque, i.e., centroidal moment. The relationship between CoM, ZMP, and CMP of the robot is provided by LIPFM as follows:

$$\ddot{c}_x = \omega^2(c_x - p_{c,x}), \quad (1)$$

$$p_{c,x} = p_{z,x} + \frac{\tau_y}{mg}, \quad (2)$$

$$\tau_y = I_y \ddot{\theta}_y, \quad (3)$$

where  $c_x$ ,  $p_{z,x}$ , and  $p_{c,x}$  denote the positions of CoM, ZMP, and CMP in the x-direction, respectively.  $\omega = \sqrt{g/c_z}$  is the natural frequency,  $g$  is the gravitational acceleration, and  $c_z$  is the height of the CoM from the ground.  $\tau_y$ ,  $\ddot{\theta}_y$ , and  $I_y$  represent the torque, angular acceleration, and rotational inertia of the flywheel in the y-direction, respectively. A detailed derivation of LIPFM dynamics is introduced in [7].

In the LIPFM dynamics, the y-direction can be derived in the same way as the x-direction, and the dynamics in each direction can be dealt with independently. In this study, therefore, the derivation of equations will be dealt with only in the x-direction.

### B. Capture Point dynamics based on the LIPFM

This section briefly introduces CP dynamics based on the LIPFM. The CP was introduced in [7], [11], and it is expressed as a linear combination of horizontal position and velocity of the CoM as below

$$\xi_x = c_x + \frac{\dot{c}_x}{\omega}, \quad (4)$$

where  $\xi_x$  represents the CP in the x-direction. The CP includes the position and velocity of the CoM, which has been used as a physical indicator to estimate the magnitude of external perturbation applied to the robot in many studies [9], [12], [14]. In LIPFM, CP can also be derived as a dynamic relationship with CMP. Combining (1) with the time derivative of (4) leads to CP–CMP dynamics as

$$\dot{\xi}_x = \omega(\xi_x - p_{c,x}) = \omega(\xi_x - (p_{z,x} + \frac{\tau_y}{mg})). \quad (5)$$

Based on these dynamic relationships, the CP feedback control framework is introduced in Section III.

## III. WALKING CONTROL USING CAPTURE POINT FEEDBACK BASED ON LIPFM

This section provides a CP feedback framework for humanoid balance control. Noting that CP error caused by the position and velocity errors of CoM occurs when the external perturbation is applied to the robot, the overall framework aims to reduce CP error for balance control of the humanoid robot.

Section. III-A introduces the overall framework for CP feedback. Next, in Section III-B, control laws for CP feedback are derived. The additional CMP is generated to reduce CP error, and more precisely, the CMP is decomposed into ZMP and CAM hierarchically in Section. III-C. Details are given in the remaining subsections.

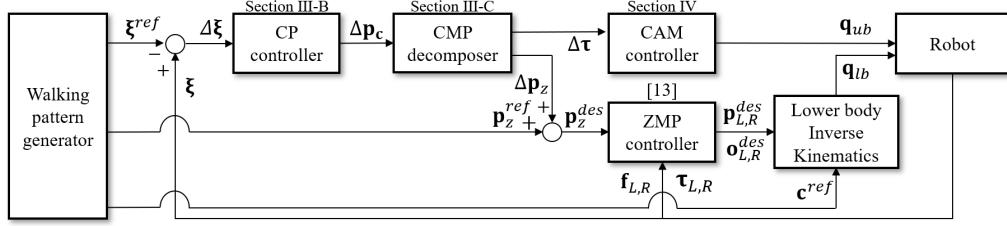


Fig. 2. Overall framework for CP feedback control

### A. Overall Framework for CP Feedback

This section briefly introduces a CP feedback control framework. Fig. 2 represents the block diagram of the overall framework. First, the CP error  $\Delta\xi$  is calculated from the reference CP,  $\xi^{ref}$ , and the current CP,  $\xi$ . Then, the additional CMP,  $\Delta p_c$ , is obtained from the CP controller to reduce the CP error. Here,  $\Delta p_c$  is decomposed in the CMP decomposer into two components,  $\Delta p_z$  and  $\Delta\tau$  which are realized by the ZMP controller and the CAM controller respectively. The ZMP controller calculates the desired position and orientation of both feet,  $\{p_{L,R}^{des}, o_{L,R}^{des}\}$ , to control the desired ZMP,  $p_z^{des}$ , using an admittance control. The lower body joint angles,  $q_{lb}$ , are obtained through the inverse kinematics of the lower body to realize the reference position of the CoM obtained from the walking pattern generator,  $c^{ref}$ , and the desired position and orientation of both feet,  $\{p_{L,R}^{des}, o_{L,R}^{des}\}$ , obtained from the ZMP controller. The joint angles of the upper body,  $q_{ub}$ , are obtained by the proposed CAM controller. Lastly, the desired joint angles are controlled using the joint PD (proportional–derivative) controller.

### B. Capture Point Feedback Control Method

To reduce CP error in this study, a feedback control law was implemented by expanding the first-order proportional CP controller in [14],

$$\dot{\xi}_x = \dot{\xi}_x^{ref} + k_{p,x}(\xi_x - \xi_x^{ref}), \quad (6)$$

where  $\dot{\xi}_x^{ref}$  and  $\xi_x^{ref}$  denote the velocity and position of the reference CP in the x-direction respectively, and  $k_{p,x}$  denotes the proportional gain. Here, CP controller (6) can be rearranged as the error dynamics of the CP, which is as follows :

$$\underbrace{\dot{\xi}_x - \dot{\xi}_x^{ref}}_{\Delta\dot{\xi}_x} = k_{p,x} \underbrace{(\xi_x - \xi_x^{ref})}_{\Delta\xi_x}. \quad (7)$$

Next, the dynamic relationship between CP and CMP, (5), also can be expressed by error dynamics,

$$\Delta\dot{\xi}_x = \omega(\Delta\xi_x - (\Delta p_{z,x} + \frac{\Delta\tau_y}{mg})). \quad (8)$$

Now, combining (7) and (8), delta CMP can be obtained as

$$\Delta p_{c,x} = \Delta p_{z,x} + \frac{\Delta\tau_y}{mg} = (1 - \frac{k_{p,x}}{\omega})\Delta\xi_x. \quad (9)$$

In (9),  $\Delta p_{c,x}$  implies an additional CMP to reduce the current CP error  $\Delta\xi_x$ . Note that if  $k_{p,x}$  in (9) is less than zero, the CP error  $\Delta\xi_x$  asymptotically converges to zero by  $\Delta p_{c,x}$ . Finally,

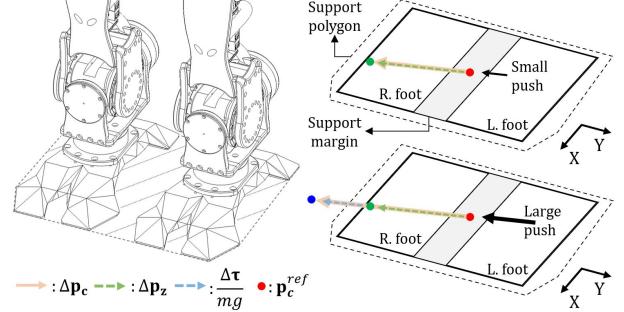


Fig. 3. Additional CMP vector decomposition based on support margin

to control the  $\Delta p_{c,x}$ ,  $\Delta p_{c,x}$  is divided into  $\Delta p_{z,x}$  and  $\Delta\tau_y$  in the CMP decomposer (Sec. III-C), and  $\Delta p_{z,x}$  and  $\Delta\tau_y$  are controlled by the ZMP controller and the CAM controller respectively.

The additional ZMP  $\Delta p_{z,x}$  is added to the reference ZMP  $p_{z,x}^{ref}$  designed based on a predefined footprint as shown in (10). Then, in our case, the desired ZMP  $p_z^{des}$  is controlled using the previously developed ZMP controller by Kajita et al. [13],

$$p_{z,x}^{des} = p_{z,x}^{ref} + \Delta p_{z,x}. \quad (10)$$

In the CAM controller (Sec. IV), the additional centroidal moment  $\Delta\tau$  is integrated into the desired CAM,  $h_y^{des}$ , which is controlled by the proposed HQP-based CAM controller,

$$h_y^{des} = \int \Delta\tau_y dt = mg \int (\Delta p_{c,x} - \Delta p_{z,x}) dt. \quad (11)$$

### C. CMP Decomposition based on Support Margin

In this section, a hierarchical decomposition of CMP based on the support margin is introduced. The support margin is defined as the sub-area of the support polygon considering the safety margin because the foot can roll over when the ZMP is controlled up to the boundary of the support polygon of the robot. In the CMP decomposition method, as shown in Fig. 3, when the robot receives external pushes, the additional CMP to reduce the CP error is decomposed with the additional ZMP,  $\Delta p_z \in \mathbb{R}^2$ , and the additional centroidal moment terms,  $\frac{\Delta\tau}{mg} \in \mathbb{R}^2$ .

In the small push case in Fig. 3, if the additional CMP,  $\Delta p_z \in \mathbb{R}^2$ , from the current reference ZMP,  $p_z^{ref} \in \mathbb{R}^2$ , exists within the support margin, no additional centroidal moment,  $\Delta\tau$ , is generated, i.e.,  $\Delta p_c = \Delta p_z$ . This implies that if the desired ZMP is modulated within the support margin, the

foot does not roll over and the CP can be stabilized only with ZMP control.

On the other hand, in the large push case in Fig. 3, since the CMP vector to reduce the CP error exceeds the range of the support margin, ZMP control is insufficient to reduce the CP error, which means that additional centroidal moment,  $\Delta\tau$ , is required. This procedure is described in Algorithm 1 where the desired ZMP,  $\mathbf{p}_z^{des} \in \mathbb{R}^2$ , and the desired CAM,  $\mathbf{h}^{des} \in \mathbb{R}^2$ , in x and y directions are finally obtained.

Note that, in this study, it is assumed that the reference centroidal moment and the reference CAM,  $\tau^{ref}$  and  $\mathbf{h}^{ref}$ , are set to zero. Therefore, the reference CMP,  $\mathbf{p}_c^{ref}$ , is identical to the reference ZMP,  $\mathbf{p}_z^{ref}$ , which is predefined by footprint.

---

**Algorithm 1** CMP Decomposition Algorithm

---

**Require:**  $\Delta\mathbf{p}_c$

- 1:  $\mathbf{p}_c^{des} \leftarrow \mathbf{p}_c^{ref} + \Delta\mathbf{p}_c$
  - 2: **if**  $\mathbf{p}_c^{des} \in \{\text{SupportMargin}\}$  **then**
  - 3:      $\Delta\mathbf{p}_z \leftarrow \Delta\mathbf{p}_c$
  - 4:      $\Delta\tau \leftarrow -k_d\mathbf{h}^{des}$
  - 5: **else**
  - 6:      $\Delta\mathbf{p}_z \leftarrow \text{size of support margin}$
  - 7:      $\Delta\tau \leftarrow mg(\Delta\mathbf{p}_c - \Delta\mathbf{p}_z)$
  - 8: **end if**
  - 9:  $\mathbf{p}_z^{des} \leftarrow \mathbf{p}_z^{ref} + \Delta\mathbf{p}_z$  ▷ ZMP controller
  - 10:  $\mathbf{h}^{des} \leftarrow \mathbf{h}^{des} + \Delta\tau \cdot dt$  ▷ CAM controller
  - 11: **return**  $\mathbf{p}_z^{des}, \mathbf{h}^{des}$
- 

#### IV. CAM CONTROLLER BASED ON HQP

In the large push case where the additional CMP to reduce the CP error exceeds the support margin, an additional centroidal moment  $\Delta\tau$  is generated as shown in Fig. 3. To control  $\Delta\tau$ ,  $\mathbf{h}^{des}$  is generated and it is controlled using the proposed CAM controller. For CAM control, eight specific joints (roll and pitch of the waist and both shoulders, and pitch of both elbows) of the upper body were selected. This is because including all joints in the problem such as the wrist and neck increases the amount of computation load while those joints do not significantly affect to the angular momentum generation.

Our CAM controller consists of two stages: 1) desired CAM generation and 2) CAM control. First, when  $\Delta\tau$  is generated for CMP control,  $\Delta\tau$  is integrated to generate  $\mathbf{h}^{des}$  as described in line 7 and 10 of the Algorithm 1. However, if  $\Delta\tau$  for CMP control is zero, the robot needs to stop rotating the upper body, so a damping reaction torque  $\Delta\tau$  is generated to decrease  $\mathbf{h}^{des}$  to zero as described in line 4 of Algorithm 1 where  $k_d$  is the positive CAM damping gain. Here, if the damping reaction torque  $\Delta\tau$  is large, the current ZMP can be rather pushed out of the support margin. Therefore, the magnitude of the damping reaction torque  $\Delta\tau$  is limited in consideration of the distance from the current ZMP to the support margin.

Second, the CAM controller controls  $\mathbf{h}^{des}$  using the selected upper body joints. However, the robot must return

to the initial pose after stabilizing the balance because the joints of the robot can not rotate infinitely unlike the ideal flywheel model to prevent self collision and avoid mechanical joint limits. Thus, initial pose return is required to store the capacity of balance control and cope with the other disturbance. To achieve these two control purposes in the previous research, [9] and [18] formulated the CAM controller as a QP problem adding all the weighted costs for both the CAM control and the initial pose return control.

$$\begin{aligned} \min_{\dot{\mathbf{q}}_{ub}} \quad & \omega_{cam} \|\mathbf{A}_{cam} \dot{\mathbf{q}}_{ub} - \mathbf{h}^{des}\|^2 + \omega_{init} \|\dot{\mathbf{q}}_{ub} - \dot{\mathbf{q}}_{ub,init}\|^2 \\ \text{s. t.} \quad & \underline{\dot{\mathbf{q}}}_{ub} \leq \dot{\mathbf{q}}_{ub} \leq \bar{\dot{\mathbf{q}}}_{ub} \\ & \mathbf{K}_q (\underline{\mathbf{q}}_{ub} - \mathbf{q}_{ub}) \leq \dot{\mathbf{q}}_{ub} \leq \mathbf{K}_q (\bar{\mathbf{q}}_{ub} - \mathbf{q}_{ub}) \end{aligned} \quad (12)$$

In (12), the cost function consists of the CAM control error term and the initial pose error term weighted by positive weighting parameters  $\omega_{cam}$  and  $\omega_{init}$ .  $\dot{\mathbf{q}}_{ub} \in \mathbb{R}^8$  is the optimization variable, i.e. the joint velocity of the selected upper body joints.  $\mathbf{A}_{cam} \in \mathbb{R}^{2 \times 8}$  is the centroidal angular momentum matrix which maps the joint velocity to the centroidal angular momentum. The inequality constraints are joint velocity and joint position limitations.  $\underline{\dot{\mathbf{q}}}_{ub}$ ,  $\bar{\dot{\mathbf{q}}}_{ub}$  are the lower and upper limits of the upper body joint velocity and  $\underline{\mathbf{q}}_{ub}$ ,  $\bar{\mathbf{q}}_{ub}$  are the lower and upper limits of the upper body joint position.  $\mathbf{K}_q \in \mathbb{R}^{8 \times 8}$  is the positive diagonal gain for the joint position limit convergence.

However, this soft hierarchy induced by the weighting parameters compromises both conflicting tasks according to the ratio of the weightings. If the ratio of  $\omega_{cam}$  over  $\omega_{init}$  increases, the CAM control performance can be improved, but still affected by the initial pose return control, and the initial pose return control deteriorates resulting in the longer return time to the initial pose. On the other hand, if the ratio of  $\omega_{cam}$  over  $\omega_{init}$  decreases, the CAM control performance decreases also resulting in less stable balance control. Consequently, if the soft hierarchy is used for both the CAM control and the initial pose return control, the CAM control performance is affected by the initial pose return control and the balance control performance decreases. Moreover, tuning the ratio between  $\omega_{cam}$  and  $\omega_{init}$  is not intuitive and time-consuming.

Therefore, for the higher CAM control performance, HQP is used by assigning the CAM control as the highest priority and assigning the return to the initial pose control as a second hierarchy. The HQP problem can be formulated as bellow.

$$\begin{aligned} \min_{\dot{\mathbf{q}}_{ub,1}} \quad & \omega_{cam} \|\mathbf{A}_{cam} \dot{\mathbf{q}}_{ub,1} - \mathbf{h}^{des}\|^2 + \omega_q \|\dot{\mathbf{q}}_{ub,1}\|^2 \\ \text{s. t.} \quad & \underline{\dot{\mathbf{q}}}_{ub} \leq \dot{\mathbf{q}}_{ub,1} \leq \bar{\dot{\mathbf{q}}}_{ub} \\ & \mathbf{K}_q (\underline{\mathbf{q}}_{ub} - \mathbf{q}_{ub}) \leq \dot{\mathbf{q}}_{ub,1} \leq \mathbf{K}_q (\bar{\mathbf{q}}_{ub} - \mathbf{q}_{ub}) \end{aligned} \quad (13)$$

$$\begin{aligned} \min_{\dot{\mathbf{q}}_{ub,2}} \quad & \|\dot{\mathbf{q}}_{ub,2} - \dot{\mathbf{q}}_{ub,init}\|^2 \\ \text{s. t.} \quad & \underline{\dot{\mathbf{q}}}_{ub} \leq \dot{\mathbf{q}}_{ub,2} \leq \bar{\dot{\mathbf{q}}}_{ub} \\ & \mathbf{K}_q (\underline{\mathbf{q}}_{ub} - \mathbf{q}_{ub}) \leq \dot{\mathbf{q}}_{ub,2} \leq \mathbf{K}_q (\bar{\mathbf{q}}_{ub} - \mathbf{q}_{ub}) \\ & \mathbf{A}_{cam} \dot{\mathbf{q}}_{ub,1}^* - \epsilon \leq \mathbf{A}_{cam} \dot{\mathbf{q}}_{ub,2} \leq \mathbf{A}_{cam} \dot{\mathbf{q}}_{ub,1}^* + \epsilon \end{aligned} \quad (14)$$

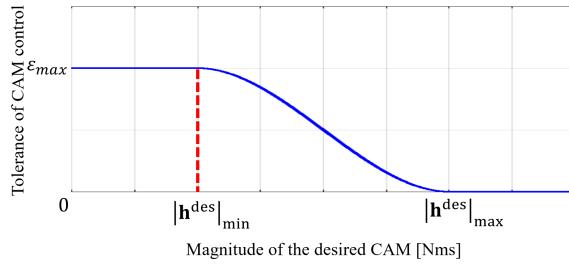


Fig. 4. Cubic function of tolerance of CAM control according to the magnitude of the desired CAM

The cost function in (13) consists of the CAM control error term and the joint velocity regularization term weighted by positive weighting parameters  $\omega_{\text{cam}}$  and  $\omega_q$ . At the first QP problem, the generated CAM is controlled considering inequality constraints. Then, the optimal velocity of the upper body from the first hierarchy of the QP,  $\dot{\mathbf{q}}_{ub,1}^* \in \mathbb{R}^8$ , is transmitted to the second QP problem to conserve the prior CAM control task. In (14), the return strategy to the initial pose is implemented minimizing the joint velocity error. The commanded joint velocity to return the initial pose can be calculated as  $\dot{\mathbf{q}}_{ub,\text{init}} = \dot{\mathbf{q}}_{ub,d} + \mathbf{K}_{\text{init}}(\mathbf{q}_{ub,d} - \mathbf{q}_{ub})$  where  $\dot{\mathbf{q}}_{ub,d}$  and  $\mathbf{q}_{ub,d}$  are the desired joint trajectory interpolating from the current joint position to the initial position, and  $\mathbf{K}_{\text{init}}$  is the positive diagonal matrix of joint position return control. Finally, the optimal joint velocity of the HQP,  $\dot{\mathbf{q}}_{ub,2}^* \in \mathbb{R}^8$ , is calculated from the (14).

However, it was observed that the robot can not generate joint velocity to return to the initial pose after CAM control because the initial pose return control conflicts with the prior CAM control. It is impossible to generate the return motion without violation of the prior CAM control only using the selected upper body joints. To resolve this problem, a tolerance of the CAM control,  $\varepsilon$ , is introduced to relax the strict hierarchy only by a specified magnitude. Because  $\varepsilon$  has the same dimension with angular momentum, it is easy to determine the value of  $\varepsilon$  approximately. Allowing the error for the CAM control, the joint velocity for the initial pose return can be generated. Larger value of  $\varepsilon$  can result in fast return to the initial pose but large errors in CAM control. Therefore, the value of  $\varepsilon$  is determined as a function of the magnitude of the desired CAM,  $\varepsilon(|\mathbf{h}^{\text{des}}|)$ , so that when the  $|\mathbf{h}^{\text{des}}|$  is large, the desired CAM is controlled as much as possible, and when the  $|\mathbf{h}^{\text{des}}|$  is small, the CAM control is compromised as the amount of  $\varepsilon$  to make possible the initial pose return control. As shown in Fig. 4,  $\varepsilon(|\mathbf{h}^{\text{des}}|)$  is designed as a cubic function of  $|\mathbf{h}^{\text{des}}|$ . The term  $\varepsilon$  has the maximum value,  $\varepsilon_{\text{max}}$ , if  $|\mathbf{h}^{\text{des}}|$  is lower than  $|\mathbf{h}^{\text{des}}|_{\text{min}}$ , and  $\varepsilon$  decreases to zero if  $|\mathbf{h}^{\text{des}}|$  is over  $|\mathbf{h}^{\text{des}}|_{\text{max}}$  so that the control performance of the CAM is conserved.

Using the proposed HQP-based CAM controller, higher CAM control performance is achieved compared with the QP-based CAM controller. Accordingly, the humanoid can maintain the balance more stably using the proposed method, and all the experimental results are supporting the proposed method in the following chapter.

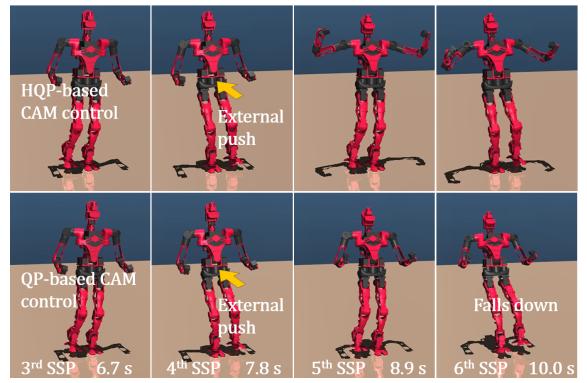


Fig. 5. Snapshots of simulations with external push of 350 N for 0.1 s

## V. RESULTS OF SIMULATIONS AND EXPERIMENTS

### A. System Overview

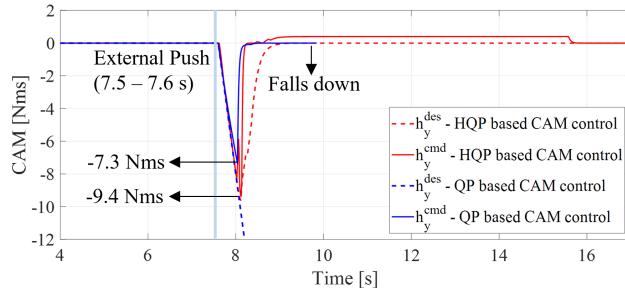
In this section, a system overview in simulations and experiments is provided. The humanoid robot, TOCABI, used in simulations and experiments consists of a total of 33 DOFs (16 in both arms, 12 in both legs, 3 in waist, and 2 in neck). The height and weight of the robot are 1.8 m and 100 kg, respectively. The foot size of the robot is 13 cm × 30 cm. The actuators consist of electric motors and harmonic gears. An encoder is attached to each joint, an IMU is located on the pelvis, and a F/T sensor is attached to each foot. The control frequency of simulation and experiment is the same with 2 kHz, and MuJoCo simulator is used for the simulation. Results also can be seen in the supplementary video.

### B. Walking Simulations

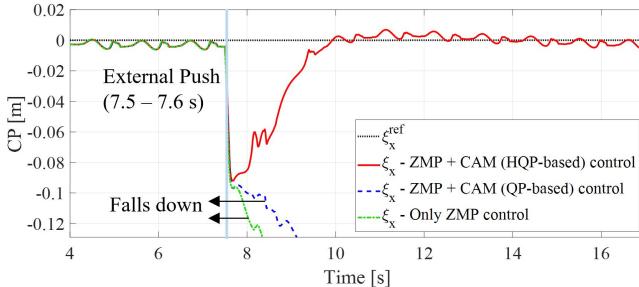
Simulation is conducted to demonstrate the performance of the proposed method through comparison with different methods. Three methods are used to analyze the results: 1) Only ZMP control 2) CAM (HQP-based proposed method) control + ZMP control 3) CAM (QP-based) control + ZMP control. Note that the QP-based method in (12) is used with the same weighting ratio,  $\omega_{\text{cam}} : \omega_{\text{init}} = 1 : 1$ .

A total of 16 in-place walking steps are planned with 1.1 s of the step duration including 0.65 s single support phase (SSP) and 0.45 s double support phase (DSP). An external push in the x-direction is applied to the pelvis of the robot for 0.1 s in the 4th SSP. In this paper, only the external push in x-direction is analyzed because the results for the external push in y-direction is similar with the results for the external push in x-direction.

Fig. 5 shows comparative snapshots of the simulation between the proposed HQP-based and the QP-based CAM controller. When the external push of 350 N is applied to the robot from 7.5 s to 7.6 s in the 4th SSP, CAM control is performed in addition to ZMP control to reduce CP error. As can be seen in the 5th SSP, 8.9 s, it is noteworthy that CAM control is performed using both waist joint and arms joints in the HQP-based method, thereby maintaining the balance of the robot. On the other hand, in the QP-based method, the arm joints are not actively used compared to the HQP-based method, and the robot lost its balance.



(a) Desired and commanded CAM in y-direction

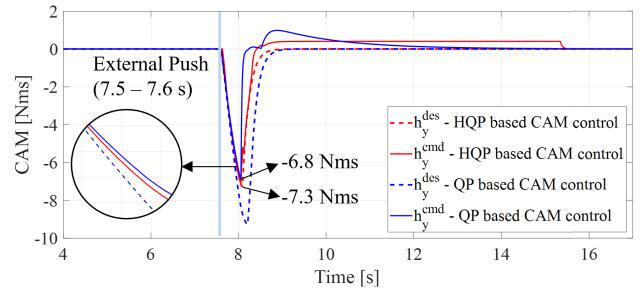


(b) Reference and measured CP in x-direction

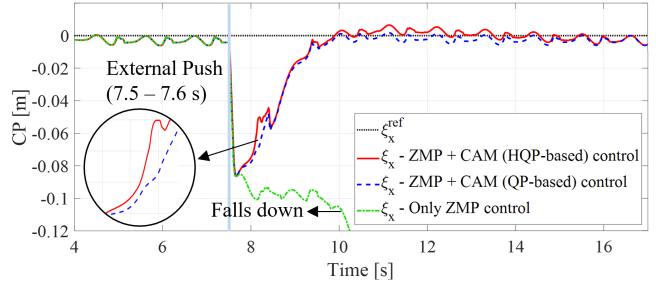
Fig. 6. Results of simulation with external push of 350 N for 0.1 s

The upper body motion for both CAM controllers in Fig. 5 can be explained from the structural difference between HQP and QP. When the waist joint approaches the joint limit for CAM tracking control, in the HQP-based method, both arms are additionally used for CAM tracking control because the CAM tracking task is hierarchically prioritized over the initial pose return task. On the other hand, in the QP-based method, the CAM tracking task using both arms and the initial pose return task conflict with each other, so that the arm motion for CAM tracking is barely emerged compared to the HQP-based method.

The differences between both CAM controllers mentioned above are once again shown as a result of the CAM and CP data in Fig. 6. Fig. 6(a) presents the CAM data when the robot receives an external push of 350 N for 0.1 s from 7.5 s (sky blue line). In Fig. 6(a), the red and blue lines indicate CAM data obtained from simulations using HQP-based and QP-based CAM controllers, respectively. There are differences in CAM control performance between both CAM controllers. In the HQP-based method, the commanded CAM calculated from  $\dot{q}_{ub}^*$  in (14) is created up to -9.4 Nms, and the RMS error of the CAM that occurs during 0.5 s after the push is 0.12 Nms. On the other hand, in the QP-based method, the CAM tracking performance deteriorates by the initial pose return task, resulting in a commanded CAM of up to -7.3 Nms, and the RMS error during 0.5 s after the push is 0.58 Nms. Thus, the HQP-based CAM control shows higher CAM tracking performance than the QP-based CAM control. In both methods, the commanded CAM is created to track the desired CAM and then reduced to zero because the robot can not generate further CAM due to the joint limits. The effect on the difference in CAM tracking



(a) Desired and commanded CAM in y-direction



(b) Reference and measured CP in x-direction

Fig. 7. Results of simulation with external push of 317 N for 0.1 s

performance can be seen from the CP data in Fig. 6(b). The black dotted line is the reference CP, and the other lines indicate the measured CP according to each control method. In the HQP-based method, the CP is pushed back up to -9.2 cm while external push is applied, but it is recovered after the push by the CAM control, and the robot walks stably. In both cases when QP-based method is used and when only ZMP is controlled without CAM control, CP could not recover from the external push and the robot loses its balance.

Another simulations are conducted with external push of 317 N for 0.1 s from 7.5 s. Fig. 7 represents the CAM and CP data when the robot receives an external push. In Fig. 7(a), since a small push is applied to the robot compared to the previous simulation, the CAM data has a small difference compared to the CAM data in Fig. 6(a). However, the HQP-based method still shows better tracking performance. In the HQP-based method, the commanded CAM is created up to -7.3 Nms, and the RMS error during 0.5 s after the push is 0.12 Nms. On the other hand, in the QP-based method, the commanded CAM is created up to -6.8 Nms, and the RMS error during 0.5 s after the push is 0.56 Nms. Note that, after approximately 9 s, in the QP method, a CAM error occurs up to 0.98 Nms due to the initial pose return control, but in the HQP method, a CAM error occurs as much as the tolerance of CAM control. The difference between both CAM controllers also can be seen in Fig. 7(b). In both CAM controllers, the CP is similarly pushed back to -8.6 cm while the external push is applied. However, there is a difference in CP during 0.5 s after push. The CP is recovered to approximately -5 cm in the HQP-based method and -7 cm in the QP-based method at approximately 8.1 s. When only ZMP is controlled without CAM control, the robot could not maintain its balance and CP diverges.

TABLE I

COMPARISON ON THE MAXIMUM EXTERNAL FORCE TO WITHSTAND ACCORDING TO THREE CONTROL METHODS AND FOOT SIZE.

Controller	Maximum external force that robot can withstand [N]		
	Foot size $\times 1.0$	Foot size $\times 0.85$	Foot size $\times 0.7$
1. ZMP only	316	266	210
2. ZMP +QP-CAM	349 (+10.4% compared to 1.)	297 (+11.7% compared to 1.)	237 (+12.9% compared to 1.)
3. ZMP +HQP-CAM	373 (+18.0% compared to 1.)	320 (+20.6% compared to 1.)	256 (+21.9% compared to 1.)

Adjusting the ratio of the weighting parameters of tasks in the QP-based method, comparative simulations are conducted for the external push of 350 N during 0.1 s to show the effect of the weighting values in QP method. When the ratio of the weighting parameters of the CAM tracking and the initial pose return task ( $\omega_{cam} : \omega_{init}$ ) is set to 5:1 and 10:1 respectively, the commanded CAM generated a maximum of -7.8 and -8.1 Nms, respectively, still lower than the -9.4 Nms generated in the HQP. This means that the two tasks in the QP are still conflicting, and CAM tracking task performance is lower than that of the HQP method. In addition, the performance of initial pose return deteriorates, and the posture is not returned to initial pose within 16 steps of completing the planned walking. In the QP formulation, it is difficult to adjust the performance of tasks by varying the weighting parameter ratio. Therefore, we improved the performance of CAM tracking by quantitatively setting the tolerance according to a specific situation using the HQP-based method, which improves the balance control performance of the robot and enables the initial pose return time adjustable indirectly.

While increasing the magnitude of the external force applied for 0.1 s by 1 N, the magnitude of external force that the robot can withstand according to the three control methods and foot size is analyzed. The reason for decreasing the size of the foot is to show that the proportion of ZMP control performance on the balance may decrease and the proportion of CAM control can increase depending on the size of the robot's foot. As can be seen from the results in Table. I, in the original foot size, the magnitude of the maximum external force that the robot can withstand increases by 10.4% and 18.0% compared to the ZMP control when the QP-CAM and the HQP-CAM controller is used respectively. Furthermore, when the foot size decreases to 0.85 times and 0.7 times smaller than the original, the maximum external force to withstand in the only ZMP control case also decreases. This similar tendency is observed in both CAM control methods, but the ratio of increase in the maximum force to withstand for both CAM control methods compared to the only ZMP control increases as the foot size is reduced. This means that for robots with small foot sizes, CAM control is increasingly important for balance control.

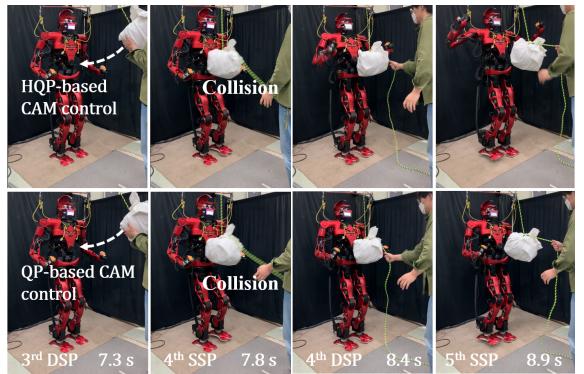
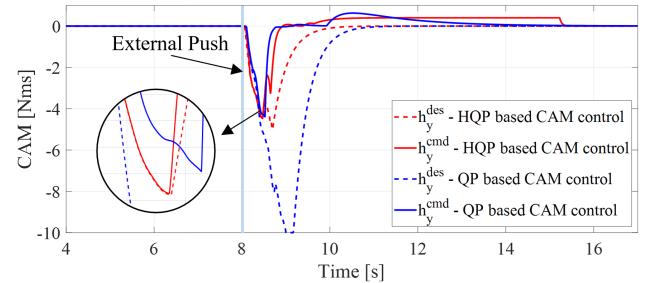
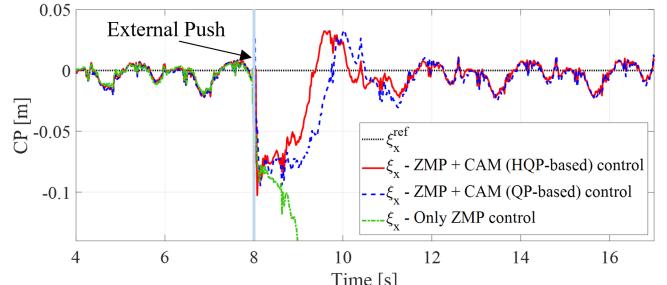


Fig. 8. Snapshots of experiment with impact push of 25.9 N·s



(a) Desired and commanded CAM in y-direction



(b) Reference and measured CP in x-direction

Fig. 9. Results of experiment with impact push of 25.9 N·s

### C. Walking Experiments

Experiment is conducted to show how the real robot recovers from impact push through the proposed algorithm. As in the simulation in Section V-B, experimental results are analyzed using the three control methods.

In the experiment, a pendulum-style weight was dropped from a certain height and collided with the robot at the lowest point of its arc. While the robot was walking in place, during the fourth SSP, the weight struck the robot in a front-to-back direction. The weight was dropped from a height of 0.33 m from the center of the robot's waist (collision position), and a weight of 10.2 kg was used. Assuming that the weight stops after the collision, the magnitude of impact applied to the robot is approximately 25.9 N·s. The step duration is 1.1 s including the SSP for 0.65 s and DSP for 0.45 s.

Fig. 8 shows comparative snapshots of the experiment when both CAM controllers are used respectively. Similar to the results of the simulation, in the last column of the

snapshot, it can be seen that the HQP-based proposed method performs CAM control by actively using the arms compared to the QP-based method.

Fig. 9(a) represents the desired and commanded CAM of both CAM controllers. In the HQP-based method, the commanded CAM is generated up to -4.5 Nms to satisfy the desired CAM until it is constrained by the joint limit, and the RMS error during 0.5 s after push is 0.17 Nms. In addition, both arms are used around 8.6 s by which commanded CAM of -3.3 Nms is additionally generated. In the QP-based method, commanded CAM is generated up to -4.3 Nms, and the RMS error is 0.5 Nms. Similar to the results of simulation, the CAM tracking performance of the QP-based method is lower than the one of the HQP-based method.

Fig. 9(b) represents the CP data. When an impact push is applied to the robot at approximately 8 s, the CP errors increase in all three control methods. After the push, the CP is pushed up to -10.1 and -9.7 cm in the proposed HQP-based and QP-based method respectively. However, in the HQP-based method after push, the CP is stabilized approximately 0.5 s faster than the QP-based method. This means that, in the HQP-CAM control, the stabilization performance of CP error is better for the same push, and more stable walking performance is guaranteed. In case of controlling ZMP only, the CP is not stabilized for the same impact push.

## VI. CONCLUSIONS AND FUTURE WORKS

In this study, a HQP-based CAM control scheme is proposed to improve the CAM tracking performance. In previous QP-based method, CAM control and initial pose return control conflict with each other. To resolve this problem, HQP-based approach is introduced to control the desired CAM with a higher priority than the initial pose return control. Additionally, the tolerance of the CAM control is introduced for the solution of the initial pose return control depending on the magnitude of the desired CAM. The proposed CAM controller is implemented in the CP feedback control framework and the stable balance control is realized when the robot is walking.

The ability to stabilize walking in the presence of external perturbations is demonstrated through simulations and experiments using our full-size humanoid robot, TOCABI. The results show that the proposed HQP-based CAM controller has better CAM tracking task performance than QP-based CAM controller, which resulted in more stable walking performance.

However, despite of proposed algorithm, the ability of the robot to withstand external forces while walking is limited by the mechanical limitations of the robot (e.g. joint angle, angular velocity, and foot size, etc.). Therefore, when the robot cannot withstand external forces, it is necessary to adjust the step position and step time of the robot. Our future work will expand the proposed algorithm so that the robot can walk stably by adjusting the step position and step time to respond to disturbances.

## REFERENCES

- [1] S. Kajita and K. Tani, "Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*. IEEE Computer Society, 1991, pp. 1405–1406.
- [2] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium (Cat. No. 01CH37180)*, vol. 1. IEEE, 2001, pp. 239–246.
- [3] M. Vukobratović and B. Borovac, "Zero-moment point—thirty five years of its life," *International journal of humanoid robotics*, vol. 1, no. 01, pp. 157–173, 2004.
- [4] S. Nakaura, M. Sampei *et al.*, "Balance control analysis of humanoid robot based on zmp feedback control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3. IEEE, 2002, pp. 2437–2442.
- [5] Y. Choi, D. Kim, Y. Oh, and B.-J. You, "Posture/walking control for humanoid robot based on kinematic resolution of com jacobian with embedded motion," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1285–1293, 2007.
- [6] J.-Y. Kim, I.-W. Park, and J.-H. Oh, "Experimental realization of dynamic walking of the biped humanoid robot khr-2 using zero moment point feedback and inertial measurement," *Advanced Robotics*, vol. 20, no. 6, pp. 707–736, 2006.
- [7] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *2006 6th IEEE-RAS international conference on humanoid robots*. IEEE, 2006, pp. 200–207.
- [8] A. L. Hof, "The ‘extrapolated center of mass’ concept suggests a simple control of balance in walking," *Human movement science*, vol. 27, no. 1, pp. 112–125, 2008.
- [9] H. Jeong, I. Lee, J. Oh, K. K. Lee, and J.-H. Oh, "A robust walking controller based on online optimization of ankle, hip, and stepping strategies," *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 1367–1386, 2019.
- [10] H.-M. Joe and J.-H. Oh, "A robust balance-control framework for the terrain-blind bipedal walking of a humanoid robot on unknown and uneven terrain," *Sensors*, vol. 19, no. 19, p. 4194, 2019.
- [11] J. Englsberger, C. Ott, M. A. Roa, A. Albu-Schäffer, and G. Hirzinger, "Bipedal walking control based on capture point dynamics," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 4420–4427.
- [12] M. Morisawa, S. Kajita, F. Kanehiro, K. Kaneko, K. Miura, and K. Yokoi, "Balance control based on capture point error compensation for biped walking on uneven terrain," in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*. IEEE, 2012, pp. 734–740.
- [13] S. Kajita, M. Morisawa, K. Miura, S. Nakao, K. Harada, K. Kaneko, F. Kanehiro, and K. Yokoi, "Biped walking stabilization based on linear inverted pendulum tracking," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2010, pp. 4489–4496.
- [14] H.-M. Joe and J.-H. Oh, "Balance recovery through model predictive control based on capture point dynamics for biped walking robot," *Robotics and Autonomous Systems*, vol. 105, pp. 1–10, 2018.
- [15] A. D. Kuo and F. E. Zajac, "Human standing posture: multi-joint movement strategies based on biomechanical constraints," *Progress in brain research*, vol. 97, pp. 349–358, 1993.
- [16] B. Stephens, "Humanoid push recovery," in *2007 7th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2007, pp. 589–595.
- [17] S.-J. Yi, B.-T. Zhang, D. Hong, and D. D. Lee, "Whole-body balancing walk controller for position controlled humanoid robots," *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1650011, 2016.
- [18] R. Schuller, G. Mesesan, J. Englsberger, J. Lee, and C. Ott, "Online centroidal angular momentum reference generation and motion optimization for humanoid push recovery," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5689–5696, 2021.
- [19] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1006–1028, 2014.