

# Toward Reactive Walking: Control of Biped Robots Exploiting an Event-Based FSM

Yisoo Lee , *Member, IEEE*, Hosang Lee, Jinoh Lee , *Senior Member, IEEE*, and Jaeheung Park , *Member, IEEE*

**Abstract**—Reactivity to unforeseen disturbances is one of the most crucial characteristics for biped robots to walk robustly in the real world. Nevertheless, conventional walking methods generally have limited capability for generating rapid reactions to disturbances, because in these methods it is necessary to wait until the end of the preplanned time period to proceed to the next phase. In this study, to improve reactivity, we develop an event-based finite-state machine (E-FSM) for walking pattern generation. Reactivity is enhanced by determining the state transition conditions of the E-FSM only with time-independent events based on the present robot state. Moreover, in the E-FSM, the robot can walk robustly even when the center of mass and the swing foot motion are disturbed, by employing the capture point concept combined with a new swing foot position constraint. Finally, we propose to control the walking robot by incorporating the E-FSM with an inverse dynamics-based motion/force controller to achieve compliant behavior. This can provide safe responses to external disturbances. The developed method is verified by experiments on a 12-degrees-of-freedom torque-controlled biped robot while it locomotes under irregular external disturbances applied to the upper body or swing leg.

**Index Terms**—Humanoid and bipedal locomotion, humanoid robots, legged robots, robot walking.

## I. INTRODUCTION

OVER the past several decades, real-time 3-D walking methods for biped robots have improved substantially. Notably, studies on zero moment point [1] and the linear inverted pendulum model (LIPM) [2] have achieved breakthroughs in

Manuscript received October 18, 2020; accepted June 1, 2021. This work was supported in part by the National Research Foundation (NRF) of Korea funded by the Ministry of Education, and the Korea government (MSIT) under Grants 2019R1A6A3A03033473 and 2021R1A2C3005914 and in part by the Korea Institute of Science, and Technology Institutional Programs under Grant 2E31063. This article was recommended for publication by Associate Editor P. M. Wensing and Editor E. Yoshida upon evaluation of the reviewers' comments. (Corresponding author: Jaeheung Park.)

Yisoo Lee is with the Center for Intelligent, and Interactive Robotics, Korea Institute of Science, and Technology, Seoul 02792, South Korea, and also with the Graduate School of Convergence Science, and Technology, Seoul National University, Seoul 16229, South Korea (e-mail: yisoo.lee@kist.re.kr).

Hosang Lee is with the Graduate School of Convergence Science, and Technology, Seoul National University, Seoul 16229, South Korea (e-mail: hslee1989@snu.ac.kr).

Jinoh Lee is with the Institute of Robotics, and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany (e-mail: jinoh.lee@dlr.de).

Jaeheung Park is with the Graduate School of Convergence Science, and Technology, Seoul National University, Seoul 16229, South Korea, and also with the Advanced Institutes of Convergence Technology, Seoul 16229, South Korea (e-mail: park73@snu.ac.kr).

Color versions of one or more figures available at 10.1109/TRO.2021.3088062.

Digital Object Identifier 10.1109/TRO.2021.3088062

the enhancement of walking stability. The LIPM-based optimal control approach enables the generation of the trajectory of the center-of-mass (CoM) with preplanned footsteps [3], [4]. In addition, capture point (CP) [5] and divergent component of motion [6] are presented to describe walking stability. Walking robots can maintain balance by employing control inputs such as the center-of-pressure (CoP) or centroidal moment pivot [7] location for the walking controller.

Notwithstanding these efforts, the dynamic walking of biped robots in the real world is challenging. In particular, unforeseen physical interactions between a robot and the environment or humans occur frequently during walking, which can cause the robot to fall. Accordingly, reactivity is introduced as an immediate reaction to achieve balance and enhance the robustness of bipedal walking under external disturbances. For example, reactive walking behavior is produced by modifying the CoM trajectory [8], [9] or by adjusting the footstep location and footstep timing [10]–[12]. In [13] and [14], the estimated disturbance force is utilized to determine appropriate footsteps when an external force acts on the robot body or swing foot.

It is noteworthy that the aforementioned reactive walking methods are time-dependent approaches that generate motion trajectories as output and rely on position tracking accuracy. However, in these methods, it becomes difficult to generate an appropriate trajectory and perform accurate trajectory tracking control when an irregular and continuous external disturbance occurs. In addition, the accurate estimation of external force during walking is a complex and challenging problem for the case of a real robot. Meanwhile, time-independent approaches are also available. One of the typical time-invariant walking approaches is the hybrid zero dynamics [15] method. Its effectiveness has been demonstrated through 3-D walking experiments [16]–[18]. Interestingly, studies in the 3-D graphics field generally present bipedal walking control that applies time-independent approaches [19]–[21]. It mainly utilizes a finite-state machine (FSM), hereafter called an event-based FSM (E-FSM), with state transitions according to events such as a foot contact.

Although the abovementioned E-FSM approaches show good potential for reactive walking, they have only been tested in simulations. In robotics, efforts have been undertaken for realizing walking control of actual bipedal robot hardware in a manner similar to that using E-FSM. In particular, studies [22], [23] inspired by the virtual model approach [24] introduced practical walking methods. With sophisticated event planning, instantaneous CP control is addressed to enhance walking stability. Nevertheless, these methods need to satisfy the spatial–temporal

constraints of the swing foot. That is, the foot is to be placed on its destination at the assigned time. However, in actual robot walking, the foot may not arrive at the desired position because of reasons that include uncertain ground slope, obstacles, low performance of the position controller, and joint speed limit. Therefore, the consideration of the scenario wherein the swing foot may not arrive at the destination in the E-FSM can improve the reactivity of biped walking.

A torque-controlled robot that is similar to those used in previous studies [22], [23] is employed in this study to utilize the advantages of the force control approach for E-FSM walking. One of the simplest and computationally efficient torque control-based approaches is the use of virtual model-based methods [22]–[24]. These methods utilize an approximated relationship between the task and contact space. Their effectiveness is demonstrated by bipedal walking experiments. In the virtual model-based methods, redundant robots can be handled by introducing additional constraints or goals. The operational space-based whole-body controller (OS-WBC) [25] provides a hierarchical control structure for redundant robots. The task space (e.g., the Cartesian space of the end-effector) motion and force can be achieved by the OS-WBC while satisfying the stationary contact condition. Recently, a quadratic programming (QP) optimization-based whole-body controller (WBC) was proposed [26]–[28]. The QP-based approach can explicitly consider inequality constraints that are related to joint limits and contact constraints. In addition, multiple prioritized tasks can be addressed with soft hierarchy. The hierarchical QP (HQP) based WBC [29]–[31] was then proposed as an extended method of QP-based WBC. The HQP-based WBC produces a strict task hierarchy, while satisfying inequality constraints.

In this study, we aim to develop an E-FSM-based reactive walking method. In the E-FSM, the termination criteria for state transitions are time-independent events that do not rely on spatial-temporal constraints of the swing foot. Thus, a robot can respond to unforeseen external disturbances acting on its body, including the swing foot. In addition, compliant behavior in response to external disturbances is generated by realizing the E-FSM with an inverse dynamics-based WBC. For walking motion generation, an instantaneous force determination method is developed with the OS-WBC [25]. This method uses whole-body dynamics and a hierarchical control structure to compute operational space task references.

The authors have presented the preliminary study in [32]. It is noteworthy that in this study, the time-independent walking pattern is reformulated with the E-FSM, which clarifies the reactive walking concept. Moreover, this article strives to elaborate on details of the proposed method with the aim of enabling practical readers to implement and reproduce the control in actual experiments. In this context, an instantaneous force determination method is presented for robot motion/force control to achieve walking motion generation and balancing through the CoP control. Finally, the proposed method is verified through several experiments in three cases. Here, a torque-controlled biped robot (DYROS RED [33]) with 12 degrees of freedom (DoFs) is used. The corresponding analysis and discussion of the results are elaborated with respect to reactivity under external disturbances.

## II. OVERVIEW OF REACTIVE WALKING

For reactive walking, the one-step capturable walking concept [34] is modified in this study. In one-step capturable walking, a walking robot can come to a stop by taking a single footstep. Therefore, the CoM attains a statically balanced state after each footstep. When the swing foot can arrive at the planned position within the specified time, continuous one-step capturable walking can be developed by repeating the sequence consisting of 1) acceleration of the ground projected CoM to move toward the planned footstep and 2) deceleration of the ground projected CoM to stop its motion within the footstep area.

To enhance the reactivity of one-step capturable walking, we propose to remove a spatial-temporal constraint, i.e., the assumption that the swing foot can reach the planned footstep within the planned time. In this context, the desired behaviors B1 and B2 for the CoM's deceleration and acceleration, respectively, are proposed to develop the modified one-step capturable walking as follows:

- B1) during the deceleration phase, the ground projected CoM moves toward the center of the support polygon, and stops its motion in the support polygon;
- B2) during the acceleration phase, the ground projected CoM moves toward the ground projected position of the center of the swing foot.

Similar to one-step capturable walking, the walking robot can obtain a statically balanced state after each footstep by realizing B1. In addition, it can continue walking even when the swing foot cannot reach the planned footstep by ensuring B2. This is because the CoM moves toward the swing foot's position rather than the planned position.

We developed an E-FSM with three states to repeat the CoM acceleration phase and deceleration phase to produce a walking pattern that satisfies B1 and B2. The E-FSM provides higher reactivity by not applying the time variable as a state transition condition. Accordingly, the walking robot can respond to unforeseen external disturbances immediately without waiting for a specific time. Details on the proposed E-FSM are provided in Section III.

E-FSM-based walking is then realized by generating the desired forces and motions by an inverse dynamics-based whole-body control approach. It provides a compliant behavior to the robot for it to absorb external disturbances. Hence, the reactivity of the biped robot can be reinforced, even without measuring or estimating the disturbance force. In addition, the desired CoP from the E-FSM can be generated without the CoP measurement because a reference force for the CoP generation can be calculated based on the multibody dynamics. Details on the walking control framework are provided in Section IV.

## III. E-FSM FOR WALKING

The developed E-FSM that satisfies B1 and B2 has the following three states as follows:

- 1) a single support deceleration (SSD) state to decelerate the ground projected CoM for it to stop in the support polygon for achieving B1;

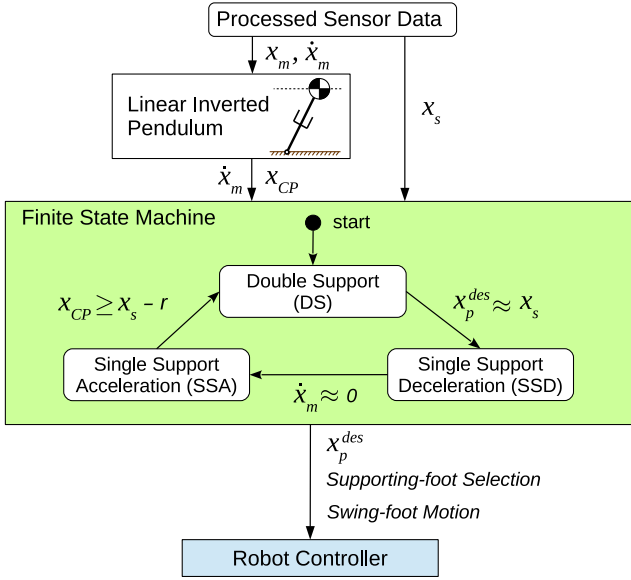


Fig. 1. Inputs and outputs of the proposed E-FSM, wherein sensor data are obtained by encoders and an inertial measurement unit. In this diagram,  $x_m$  denotes the position of the CoM,  $x_s$  denotes the position of the swing foot,  $x_{CP}$  denotes the CP,  $r$  denotes the margin parameter, and  $x_p^{des}$  denotes the desired CoP.

- 2) a single support acceleration (SSA) state to accelerate the ground projected CoM for it to move toward the ground projected position of the swing foot for achieving B2;
- 3) a double support (DS) state for the smooth transition of the supporting foot for the next footstep.

For the repetitive cycle of walking, a single sequence of a single footstep is constructed with three states in the order SSD-state, SSA-state, and DS-state, as shown in Fig. 1. Here, each state transits to the next one when a predefined termination criterion condition is satisfied. The E-FSM then can generate continuous walking by repeating the single footstep.

To accelerate and decelerate the CoM for the SSA- and SSD-state, respectively, and to determine when to transit the state, the properties from the LIPM and the CP concept are utilized. The LIPM describes the behavior of the CoM with the assumption that the robot is a point mass and that its height from the ground is constant at  $z_{m,0}$ . The LIPM dynamics is then expressed in Cartesian coordinates as follows [2]:

$$\ddot{\mathbf{x}}_m = \frac{g}{z_{m,0}}(\mathbf{x}_m - \mathbf{x}_p) \quad (1)$$

where  $\mathbf{x}_m = [x_m \ y_m]^T$  denotes the position of the CoM in the transverse plane,  $\mathbf{x}_p = [x_p \ y_p]^T$  denotes the position of the CoP in the transverse plane, and  $g$  denotes gravitational acceleration. Equation (1) reveals the first property of the CoM's acceleration as follows:

- 1) the ground projected CoM accelerates away from the CoP.

The second property of the CoM's deceleration is obtained from the definition of the CP [5] as follows:

- 2) the ground projected CoM stops at the CP when the CoP is positioned at the CP

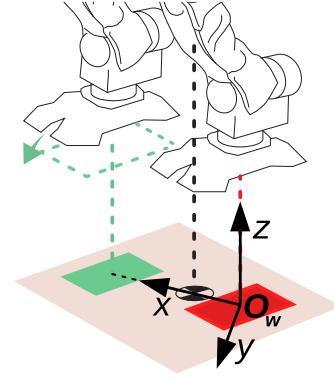


Fig. 2. Coordinate system  $\mathbf{O}_w$  for the E-FSM. The red and green colored squares are the ground projections of the supporting foot and swing foot, respectively. The origin of the coordinate system  $\mathbf{O}_w$  is located at the center of the ground projection of the supporting foot, and the positive  $z$ -direction is opposite to the direction of gravity. The  $x$ -axis of  $\mathbf{O}_w$  is parallel to the line that connects the centers of the projections of both feet.

where the CP can be calculated by the equation derived from the LIPM as follows:

$$\mathbf{x}_{CP} = \sqrt{\frac{z_{m,0}}{g}} \dot{\mathbf{x}}_m + \mathbf{x}_m \quad (2)$$

where  $\mathbf{x}_{CP} = [x_{CP} \ y_{CP}]^T$  denotes the position of the CP in the transverse plane.

To reflect the abovementioned two properties in the E-FSM, the inputs of the E-FSM include the CoM position and velocity ( $x_m, \dot{x}_m$ ), the CP ( $x_{CP}$ ), and the center of the swing foot's ground projection ( $x_s$ ). Meanwhile, the desired CoP ( $x_p^{des}$ ) is included in the output to determine whether to accelerate or decelerate the CoM's motion. In addition, for foot control, the outputs include the selection of the supporting foot and the decision of whether to raise or land the swing foot. The inputs and outputs are also shown in Fig. 1.

The following two main problems need to be resolved for applying the LIPM-based E-FSM in real robot walking: 1) for 3-D walking, the solutions in the lateral and sagittal planes should be synchronized without applying a time constraint; and 2) the properties from the simplified model are not the same as those of a real robot, although they can describe a similar dynamic tendency. Therefore, in the following sections, we first introduce a coordinate system for the former and margin parameters for the latter. We then elaborate on each state in the E-FSM.

#### A. Considerations for Real-Robot Implementation

1) *Coordinate System:* The coordinate system  $\mathbf{O}_w$  described in Fig. 2 is adopted to develop the E-FSM in 3-D space. In the coordinate system, the origin is located at the center of the supporting foot's sole. The positive  $x$ -direction is the direction of the vector that connects the center of the supporting foot to the center of the swing foot's ground projection, and the direction opposite to that of gravity is the positive  $z$ -direction. Accordingly, the coordinate system  $\mathbf{O}_w$  rotates around the  $z$ -axis when the swing foot moves because its  $x$ -direction is dependent on the swing foot's position.

The  $xz$ -plane in the coordinate system  $\mathbf{O}_w$  forms a 2-D space for the CoM motion described by B1 and B2 according to the  $x$ -axis, which is always directed toward the swing foot's position. Thus, the E-FSM creates the one-step capturable walking only in the  $xz$ -plane. On the other hand, the  $yz$ -plane forms a 2-D space, where the CoM position is constant to achieve a statically balanced state. Note that we set the CoM position to be constant on the  $y$ -axis to simplify the problem, although the CoM motion on the  $y$ -axis can be generated if it can maintain a statically balanced state. Consequently, the CoM's 3-D motion can be simplified as a 2-D problem by considering only the motion in the  $xz$ -plane of the coordinate system  $\mathbf{O}_w$ .

Reducing the 3-D walking problem to a 2-D problem is advantageous for composing the time-independent E-FSM. In general, conventional walking methods decompose the motion space into the sagittal and lateral planes, and the robot's walking motions in both planes have to be synchronized. The most common synchronization method is to apply a time constraint to the CoM or swing foot motion in the two planes to complete at the same time. Meanwhile, for the proposed coordinate system  $\mathbf{O}_w$ , the aforementioned synchronization issue can be prevented by considering the motion in only one plane (i.e., the  $xz$ -plane). Therefore, the E-FSM can be composed without a time constraint. In addition, the height of the CoM can be considered to be constant because the LIPM concept is utilized in the E-FSM. Thus, the CoM motion generation problem can be simplified to an  $x$ -axis motion problem in the E-FSM.

2) *Margin Parameters in the E-FSM*: In a real robot system, the robot's CoM behaves imperfectly as the simplified model, i.e., the LIPM. Nevertheless, the tendency of the relationship between the CoM and CoP from the LIPM can be utilized for the walking control of an actual robot. To utilize the properties from the LIPM considering the model error, we introduce conservative laws with the margin parameters  $\zeta > 0$  and  $r > 0$  as follows:

- L1) if the CoP and ground projected CoM are  $\zeta$  apart, the CoM accelerates away from the CoP [see Fig. 3(a)];
- L2) if the CP is located between the CoP and the ground projected CoM at a distance  $r$  from the CoP, the CoM becomes stationary before arriving at the CoP [see Fig. 3(b)].

The magnitudes of  $\zeta$  and  $r$  have to be determined empirically based on the similarity between the robot and the LIPM. The magnitude of the parameters has to be large if the similarity is low, and vice versa. The parameters can be determined empirically with consideration of the anticipated behaviors of the robot when  $\zeta$  and  $r$  are too small or large, as described in the following section.

The abovementioned laws (L1 and L2) are then utilized for the E-FSM implementation. In the E-FSM, the rule for determining the CoP position is obtained from L1 for both SSA- and SSD-states. Furthermore, the time when the CoM's acceleration has to be stopped, i.e., when the SSA-state has to be terminated, is obtained from L2. Further details on how laws L1 and L2 are applied to each state are described in the following section.

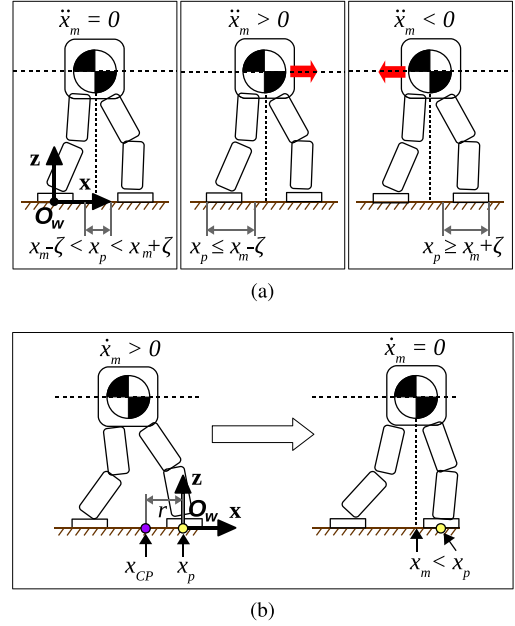


Fig. 3. Laws L1 and L2 from the LIPM with margin parameters for practical use. (a) L1: The CoM accelerates in the direction opposite to that of the CoP on the  $x$ -axis when the distance between the CoM and CoP is greater than  $\zeta$ . (b) L2: The CoM stops before reaching the CoP when the CP is located between the CoM and CoP on the  $x$ -axis. Here,  $x_m$  is the CoM's position,  $x_p$  is the CoP's position,  $x_{CP}$  is the CP, and  $\zeta$  and  $r$  are margin parameters with positive values.

TABLE I  
E-FSM FOR WALKING

	States		
	SSD	SSA	DS
Termination criterion	$\dot{x}_m \approx 0$	$x_{CP} \geq x_s - r$	$x_p^{des} \approx x_s$
Supporting-foot selection	right / left	right / left	both feet
Desired CoP ( $x_p^{des}$ )	inside of the supporting foot, Eq. (3)	inside of the supporting foot, Eq. (3)	move from one foot to another, Eq. (4)
Swing foot motion	raise	land	none
Initial condition	termination of DS-state	termination of SSD-state	termination of SSA-state, or start of the first footstep

## B. Proposed E-FSM for Walking

This section provides details of each state for the proposed E-FSM, which are summarized in Table I. Note that all spatial variables in the E-FSM are described on the  $x$ -axis of  $\mathbf{O}_w$  and the name of each state represents the desired state.

1) *SSD-State*: The SSD-state is the first state of the single footstep. During the SSD-state, the CoM's motion in the  $x$ -direction is controlled for it to decelerate until it becomes stationary, whereas the swing foot is controlled for it to rise and move to the target footstep position.

As an output of the E-FSM, the desired CoP ( $x_p^{des}$ ) has to be determined for the CoM deceleration based on laws L1 and L2.

According to L2, the CoM can become stationary before moving beyond the supporting foot when  $x_p^{\text{des}}$  is positioned at the center of the supporting foot during this state. This is because the CP is positioned between  $x_m$  and the center of the supporting foot when the previous DS-state has been successfully completed. Therefore,  $x_p^{\text{des}}$  is designed to be positioned at the center of the supporting foot, i.e.,  $x_p^{\text{des}} = 0$ . In addition, to consider the scenario wherein the CoM moves close to or beyond the center of the supporting foot, i.e., when  $x_m \leq \zeta$ ,  $x_p^{\text{des}}$  has to be positioned farther than the CoM considering the margin parameter, i.e.,  $x_p^{\text{des}} < x_m - \zeta$ , according to L1. Thus,  $x_p^{\text{des}}$  is determined by the following rule:

$$x_p^{\text{des}} = \begin{cases} 0, & \text{if } x_m > \zeta \\ x_m - \zeta, & \text{if } x_m \leq \zeta. \end{cases} \quad (3)$$

The termination criterion of the SSD-state is the stop of the CoM motion. Therefore, the SSD-state terminates when the velocity of the CoM becomes zero, i.e.,  $\dot{x}_m \approx 0$ , after which the next state, SSA, commences.

If a small value is selected for  $\zeta$ , the CoM may not be decelerated in the SSD-state as intended, and the robot may fall. Meanwhile, if the value is selected to be greater than is necessary, the CoM would stop its motion earlier, and the swing foot motion would not be generated adequately.

2) *SSA-State*: During the SSA-state, the CoM accelerates to move toward the center of the swing foot ( $x_s$ ) while the swing foot is controlled to land on the ground at the planned footstep position. According to L1, if  $x_m$  is positioned ahead of the CoP, the CoM can be accelerated toward  $x_s$ . Thus,  $x_p^{\text{des}}$  during the SSA-state can be expressed using (3), which is also the equation for the SSD-state.

The termination criterion of the SSA-state is designed in accordance with L2. The SSA-state terminates when the CP reaches the ground projected position of the swing foot with  $r$ , i.e.,  $x_{\text{CP}} \geq x_s - r$ . The DS-state then commences.

When the magnitude of  $r$  is too small, the CoM may move farther than the support polygon area and fail to stop its motion in the next SSD-state. Meanwhile, when the magnitude of  $r$  is too large, the CoM may stop its motion too early in the next SSD-state and the swing foot motion cannot be completed.

3) *DS-State*: The DS-state commences when the SSA-state terminates or when the first footstep begins. In this state,  $x_p^{\text{des}}$  has to smoothly move from one foot to the other to switch the supporting foot for the next footstep.

For this purpose, the initial value of  $x_p^{\text{des}}$  is equal to the final value of  $x_p^{\text{des}}$  in the SSA-state, and the target value is equivalent to the center of the next supporting foot. Thus, the  $x_p^{\text{des}}$  during the DS-state is determined using a simple interpolation equation as follows:

$$x_p^{\text{des}} = \begin{cases} x_{p(\kappa-1)}^{\text{des}} + \text{sgn}(\delta)\Delta x_p^{\text{des}}, & \text{if } |\delta| > \Delta x_p^{\text{des}} \\ x_s, & \text{if } |\delta| \leq \Delta x_p^{\text{des}} \end{cases} \quad (4)$$

where  $x_{p(\kappa-1)}^{\text{des}}$  is the desired CoP at the previous control period,  $\text{sgn}(\cdot)$  is the signum function,  $\delta = x_s - x_{p(\kappa-1)}^{\text{des}}$ , and  $\Delta x_p^{\text{des}} > 0$  is an arbitrary interval of the desired CoP between each sampling

time. This interpolation is required for a smooth CoP transition because real robot hardware cannot vary the CoP by a large amount within a short period due to physical limitations.

The DS-state terminates when  $x_p^{\text{des}} \approx x_s$  according to the defined termination criterion. After termination, the role of each foot is altered, and the next single footstep starts from the SSD-state.

When the term  $\Delta x_p^{\text{des}}$  is too large, the CoP cannot be generated as desired, and the contact plane can be unstable. Meanwhile, when  $\Delta x_p^{\text{des}}$  is too small, the DS-state time increases and the CoM and CP may move further. Accordingly, the CP in the SSD-state after the DS-state can move beyond the support polygon and the robot may fall over.

#### IV. WHOLE-BODY WALKING CONTROL

In this study, an inverse dynamics-based WBC and joint torque-controlled biped robot are adopted for compliant behavior in response to unforeseen external disturbances as mentioned previously. Based on the hierarchical control structure of the OS-WBC, we constructed the whole-body walking controller. Note that control frameworks other than the OS-WBC are available for the implementation. For example, HQP-based WBCs [29]–[31], which can create a hierarchy, are capable of obtaining a feasible solution that produces results that are almost identical to those of the proposed OS-WBC-based method.

##### A. Review: The OS-WBC [35]

The dynamics equation for  $n = k + 6$  DoFs (where  $k$  is the number of actuation joints) floating base robot, which has rigid contacts with the environment, can be described as follows:

$$\mathbf{A}\ddot{\mathbf{q}} + \mathbf{b} + \mathbf{g} + \mathbf{J}_c^T \mathbf{f}_c = \mathbf{S}^T \mathbf{\Gamma}$$

$$\mathbf{J}_c \ddot{\mathbf{q}} + \dot{\mathbf{J}}_c \dot{\mathbf{q}} = \mathbf{0}$$

where  $\mathbf{q} = [\mathbf{x}_f^T \ \mathbf{q}_j^T]^T \in \mathbb{R}^n$  is the vector of configuration of the robot,  $\mathbf{x}_f \in \mathbb{R}^6$  is the floating base position and orientation vector,  $\mathbf{q}_j \in \mathbb{R}^k$  is the joint position vector,  $\mathbf{A} \in \mathbb{R}^{n \times n}$  is the inertia matrix,  $\mathbf{b} \in \mathbb{R}^n$  is the Coriolis/centrifugal force vector,  $\mathbf{g} \in \mathbb{R}^n$  is the gravity force vector,  $\mathbf{J}_c \in \mathbb{R}^{c \times n}$  is the contact Jacobian matrix defined as  $\dot{\mathbf{x}}_c = \mathbf{J}_c \dot{\mathbf{q}}$ ,  $\mathbf{x}_c \in \mathbb{R}^c$  is the vector of contact positions and orientations, where the number of rows  $c$  can be determined by the contact state of the robot,  $\mathbf{f}_c \in \mathbb{R}^c$  is the contact wrench vector,  $\mathbf{S} \in \mathbb{R}^{k \times n}$  is the selection matrix for the actuated joints, and  $\mathbf{\Gamma} \in \mathbb{R}^k$  is the torque vector of the actuated joints.

Based on the abovementioned robot dynamics, the operational space task control torque ( $\mathbf{\Gamma}_t \in \mathbb{R}^k$ ) for  $m$  hierarchical tasks can be formulated as follows [25]:

$$\mathbf{\Gamma}_t = \tilde{\mathbf{J}}_1^T \mathbf{F}_1 + \sum_{i=2}^m \left\{ \left( \prod_{j=1}^{i-1} \tilde{\mathbf{N}}_j^T \right) \tilde{\mathbf{J}}_i^T \mathbf{F}_i \right\} \quad (5)$$

where the subscripts indicate the task priority,  $\tilde{\mathbf{J}}_\bullet$  is the Jacobian matrix for a contact-constrained underactuated robot,  $\tilde{\mathbf{N}}_\bullet^T$  is the corresponding null-space projection matrix, and  $\mathbf{F}_\bullet$  is the operational space force. Further details are available in Appendix A.

In the abovementioned hierarchical formulation, the commanding force  $\mathbf{F}$  can be computed without feedback (e.g., from the reference force computed by the dynamics model) or with position  $\mathbf{x}$  and velocity  $\dot{\mathbf{x}}$  feedback from the reference acceleration  $\ddot{\mathbf{x}}^*$  as

$$\begin{aligned}\mathbf{F} &= \Lambda \ddot{\mathbf{x}}^* + \boldsymbol{\mu} + \mathbf{p} \\ &= \Lambda \{k_p(\mathbf{x}^{\text{des}} - \mathbf{x}) - k_v \dot{\mathbf{x}}\} + \boldsymbol{\mu} + \mathbf{p}\end{aligned}\quad (6)$$

where  $\Lambda$ ,  $\boldsymbol{\mu}$ ,  $\mathbf{p}$ , and  $\mathbf{x}^{\text{des}}$  are the inertia matrix, Coriolis/centrifugal force vector, gravity force vector, and desired position vector, respectively, in the operational space. If the Coriolis/centrifugal and gravity forces are compensated in the joint space by the compensation torque vector  $\boldsymbol{\Gamma}_{\text{gc}} \in \mathbb{R}^k$ , the term  $(\boldsymbol{\mu} + \mathbf{p})$  can be removed from (6). The implementation of  $\boldsymbol{\Gamma}_{\text{gc}}$  is detailed in Appendix B.

In the OS-WBC, the contact wrench can be redistributed without affecting the controlling tasks by utilizing the contact null-space when the rank of the contact space is greater than six. Therefore, to obtain a continuous contact wrench and feasible solution considering the contact constraints, such as the CoP during the DS-state, the contact wrench is redistributed by the contact wrench distribution torque  $\boldsymbol{\Gamma}_c \in \mathbb{R}^k$  with the given contact wrench correction vector  $\mathbf{f}_c^{\text{cor}} \in \mathbb{R}^6$ . In this study,  $\mathbf{f}_c^{\text{cor}}$  is obtained by the method introduced in [35], which minimizes the tangential force and moments.  $\boldsymbol{\Gamma}_c$  can be calculated for continuous torque as follows:

$$\boldsymbol{\Gamma}_c = \mathbf{Y}_c(\varepsilon \mathbf{f}_c^{\text{cor}})$$

where  $\mathbf{Y}_c$  is a mapping matrix for calculating the torque that can create a contact wrench in the contact null space so as not to affect the task space. Further details regarding the calculation of  $\mathbf{Y}_c$  are described in Appendix C. In the abovementioned, the parameter  $\varepsilon$  for the smooth contact transition is multiplied by  $\mathbf{f}_c^{\text{cor}}$ . How to determine the parameter  $\varepsilon$  for the developed robot walking is described in the following section.

Finally, the reference torque for the robot can be expressed as a summation of the three torque vectors as follows:

$$\boldsymbol{\Gamma} = \boldsymbol{\Gamma}_t + \boldsymbol{\Gamma}_c + \boldsymbol{\Gamma}_{\text{gc}}.\quad (7)$$

The reference torque  $\boldsymbol{\Gamma}$  in (7) can be utilized to calculate the expected contact wrench  $\mathbf{f}_c$  with the forward dynamics as follows:

$$\mathbf{f}_c = \bar{\mathbf{J}}_c^T \mathbf{S}^T \boldsymbol{\Gamma} - \boldsymbol{\mu}_c - \mathbf{p}_c.\quad (8)$$

The resultant contact wrench vector  $\mathbf{F}_c$  can then be obtained as follows:

$$\mathbf{F}_c = \mathbf{K}(\bar{\mathbf{J}}_c^T \mathbf{S}^T \boldsymbol{\Gamma} - \boldsymbol{\mu}_c - \mathbf{p}_c)\quad (9)$$

where  $\mathbf{F}_c = [{}^x F_c \ {}^y F_c \ {}^z F_c \ {}^x M_c \ {}^y M_c \ {}^z M_c]^T \in \mathbb{R}^6$ ,  ${}^*F_c$  and  ${}^*M_c$  are the contact force and moment with respect to the  $\star$ -axis, respectively, in an arbitrary coordinate system  $\mathbf{O}$  assuming that  $\boldsymbol{\Gamma}$  is applied to the robot; and  $\mathbf{K}$  is the grasp map matrix that describes the resultant wrench in  $\mathbf{O}$ . The matrix  $\mathbf{K}$  is composed as

$$\mathbf{K} = \begin{bmatrix} {}^{\mathbf{O}}\mathbf{R}_r & \mathbf{0}_{3 \times 3} & {}^{\mathbf{O}}\mathbf{R}_l & \mathbf{0}_{3 \times 3} \\ \hat{\mathbf{P}}_r & {}^{\mathbf{O}}\mathbf{R}_r & \hat{\mathbf{P}}_l & {}^{\mathbf{O}}\mathbf{R}_l \end{bmatrix}\quad (10)$$

where  ${}^{\mathbf{O}}\mathbf{R}_r, {}^{\mathbf{O}}\mathbf{R}_l \in \mathbb{R}^{3 \times 3}$  are the rotation matrices of the right and left foot with respect to the reference coordinate system.  $\hat{\mathbf{P}}_r, \hat{\mathbf{P}}_l \in \mathbb{R}^{3 \times 3}$  are the skew-symmetric matrices of the distance vector  $\mathbf{P}_r, \mathbf{P}_l \in \mathbb{R}^3$  from the origin of the coordinate system  $\mathbf{O}$  to each foot. For the one foot contact,  $\mathbf{K}$  is constructed as  $\mathbf{I}_{6 \times 6}$  rather than (10) when the coordinate system is located at the center of the supporting foot.

### B. Contact Transition Parameter

In WBCs, the torque solution may be discontinuous due to the transition of the Jacobian matrices when the walking state switches. It is recommended to generate a continuous joint torque reference for stable robot control. If the reference joint torque is discontinuous, overshoots and oscillations in the joint torque may result in unintended abrupt movements that can adversely affect the balance of the robot. Therefore, most torque-controlled biped robots strive to create continuous torque while transitioning between contact states with a time-varying trajectory [28], [31], [35]. In this section, we propose to apply the state-transition parameter  $\varepsilon$ , which is determined based on the termination criterion of the E-FSM, to achieve continuous torque according to the time-independent walking pattern.

When the defined walking state changes from the SSA-state to the DS-state, the swing foot is switched to the contact foot. The converse occurs when the change is from the DS-state to the SSD-state. For a smooth transition, we propose to smoothly reduce the reference wrench for the corresponding foot when the robot configuration is close to the termination criterion during the SSA and DS states. For this purpose, the parameter  $d$ , which expresses the degree of similarity between the robot's present state and the termination criterion, is defined as follows:

$$d = \begin{cases} x_s - x_p^{\text{des}}, & \text{ifDS} \\ (x_s - \zeta) - x_{\text{CP}}, & \text{ifSSA.} \end{cases}$$

In the DS-state,  $d$  is the distance between  $x_p^{\text{des}}$  and  $x_s$ ; and in the SSA-state,  $d$  is the distance between  $x_{\text{CP}}$ , and  $x_s$  with consideration of  $\zeta$ .

The parameter  $\varepsilon$  is then defined based on  $d$  as follows:

$$\varepsilon = \begin{cases} 0, & \text{if } d \leq 0 \\ \frac{d}{d^{\text{thr}}}, & \text{else if } 0 < d < d^{\text{thr}} \\ 1, & \text{else if } d \geq d^{\text{thr}} \end{cases}$$

where  $d^{\text{thr}} > 0$  is an empirically determined threshold value. With  $\varepsilon$ , it can be considered that the E-FSM state is not to be altered when  $d \geq d^{\text{thr}}$ ; the robot should prepare the contact transition, which would occur earlier when  $0 < d < d^{\text{thr}}$ ; and the walking state should be altered at the instant when  $d \leq 0$ . The determined  $\varepsilon$  is then multiplied by each reference for the contact wrench and swing foot motion control during the DS- and SSA-states, respectively, to generate a continuous joint torque solution.

### C. Hierarchical Task Setting

Three prioritized tasks are configured to implement a walking controller with the proposed E-FSM. To control robot walking,

the outputs of the E-FSM have to be reconfigured as a general control input for the WBC, e.g., the specific reference acceleration or force at the task space. Also, the balancing problem that can be caused by a large control error of the feedback control tasks has to be taken into account.

To address these problems, the  $x$ -direction force on the CoM that can generate the acceleration intended by  $x_p^{\text{des}}$  is set as the highest priority task to control it without being affected by the other tasks. The task with the lowest priority is composed of the pitch and roll orientation of the upper body, and it is ceded when balancing is required. The remaining tasks constitute the second priority task. The task associated with balancing is assigned the lowest priority, so as not to affect the higher priority tasks. Therefore, only the pitch and roll orientations of the upper body are deformed for balancing without affecting the critical motion that should be maintained for walking, i.e., the CoM and swing foot motion.

From (5),  $\Gamma_t$  with the three prioritized tasks is expressed as follows:

$$\Gamma_t = \underbrace{\tilde{\mathbf{J}}_1^T \mathbf{F}_1}_{\Gamma_1} + \underbrace{\tilde{\mathbf{N}}_1^T \tilde{\mathbf{J}}_2^T \mathbf{F}_2}_{\Gamma_2} + \underbrace{\tilde{\mathbf{N}}_1^T \tilde{\mathbf{N}}_2^T \tilde{\mathbf{J}}_3^T \mathbf{F}_3}_{\Gamma_3} \quad (11)$$

where  $\Gamma_1$ ,  $\Gamma_2$ , and  $\Gamma_3$  are the task torque vectors for the first, second, and third priority tasks, respectively. The Jacobian matrices and wrench vectors of each task are composed as follows:

$$\mathbf{J}_1 = {}^x \mathbf{J}_m$$

$$\mathbf{J}_2 = \begin{cases} [y \mathbf{J}_m^T \ z \mathbf{J}_m^T \ z \mathbf{J}_u^T]^T, & \text{ifDS} \\ [y \mathbf{J}_m^T \ z \mathbf{J}_m^T \ z \mathbf{J}_u^T \ \mathbf{J}_s^T]^T, & \text{else} \end{cases}$$

$$\mathbf{J}_3 = [x \mathbf{J}_u^T \ y \mathbf{J}_u^T]^T$$

$$\mathbf{F}_1 = {}^x F_m$$

$$\mathbf{F}_2 = \begin{cases} [y F_m \ z F_m \ z M_u]^T, & \text{ifDS} \\ [y F_m \ z F_m \ z M_u \ \epsilon \mathbf{F}_s^T]^T, & \text{else} \end{cases}$$

$$\mathbf{F}_3 = [x M_u \ y M_u]^T$$

where  ${}^* \mathbf{J}_\bullet \in \mathbb{R}^{1 \times n}$  is the Jacobian matrix for the one DoF operational space task;  ${}^* F_\bullet$  is the linear force component;  ${}^* M_\bullet$  is the moment component; the subscript  $m$  denotes CoM;  $u$  denotes upper body orientation;  $x$ ,  $y$ , and  $z$  at the superscript  $\star$  denote the corresponding direction;  $\mathbf{J}_s \in \mathbb{R}^{6 \times n}$  is the Jacobian matrix of the swing foot position and orientation; and  $\mathbf{F}_s \in \mathbb{R}^6$  is the swing foot wrench vector in the operational space. Note that the state-transition parameter  $\epsilon$  is multiplied by  $\mathbf{F}_s$  to reduce the force to zero smoothly before generating a contact.

*Remark* In this study, we consider the total number of task DoFs and rank of contact ( $c$ ) for the DS-state to be 6 and 12, respectively, and those for the other states to be 12 and 6, respectively. In this case, a robot with  $n \geq 18$ , i.e.,  $k \geq 12$ , is required because a necessary but nonsufficient condition for the controllable DoFs of a robot with rigid contact is  $n - c$ . Therefore, the proposed walking controller can also be implemented on a robot with  $n > 18$ .

The reference force and acceleration of the CoM and swing foot motion required for the proposed E-FSM-based walking is determined as follows.

- (1) The  $x$ -component of the CoM force is controlled for generating  $x_p^{\text{des}}$ . The reference of the CoM force is obtained by calculating the relationship between the CoP and the CoM motion with the multibody dynamics.
- (2) The other tasks for the CoM motion along the  $y$ - and  $z$ -axis and for the swing foot motion along the  $x$ -,  $y$ -, and  $z$ -axis are controlled at the acceleration level. The reference acceleration for controlling the position  $x$  is calculated as follows:

$$\ddot{x}^* = k_p(x^{\text{des}} - x) - k_v \dot{x}$$

where  $\ddot{x}^*$  is the reference acceleration,  $x^{\text{des}}$  is the desired position,  $x$  is the measured position, and  $k_p > 0$  and  $k_v > 0$  are the gains.

For position control, according to the coordinate system  $\mathbf{O}_w$  and the LIPM, the desired position of the CoM along the  $y$ - and  $z$ -axis ( $y_m^{\text{des}}$  and  $z_m^{\text{des}}$ ) are zero and  $z_{m,0}$ , respectively. The desired position for the swing foot along the  $x$ - and  $y$ -axis ( $x_s^{\text{des}}$  and  $y_x^{\text{des}}$ ) can be determined from the footstep planner. Meanwhile, the  $z$ -axis position  $z_s^{\text{des}}$  is determined from the predefined footstep height, which is different for the raising motion for the SSD-state and the landing motion for the SSA-state. The other tasks such as the swing foot orientation and upper body orientation required for implementing walking motion have to be designed within a range that does not interfere with the successful walking motion. For example, in this study, we controlled the upper body and foot orientations to maintain their initial orientations.

An overview of the constructed whole-body walking controller for E-FSM-based walking is shown in Fig. 4. Details of how to determine the reference forces and moments ( $\mathbf{F}_1$ ,  $\mathbf{F}_2$ ,  $\mathbf{F}_3$ ) for the task torque  $\Gamma_t$  are given in Appendix D

## V. EXPERIMENTAL SETUP

### A. Hardware and Software

Walking experiments are conducted with the 12 DoFs torque-controlled biped robot DYROS RED [33] (see Fig. 5) to verify the proposed method. The robot's height, weight, foot length, and foot width are 1.43 m, 89.6 kg, 0.3 m, and 0.15 m, respectively. The joints of each leg are arranged as yaw-roll-pitch-pitch-pitch-roll rotations with respect to the upper body.

Each joint of the robot is equipped with an electric motor (Kollmorgen) with a 100:1 harmonic gear and a magnetic encoder with a resolution of 13 b mounted before the gear reduction. A motor servo drive (Elmo, Gold Solo Whistle) controls the joint torque through a low-level current controller and sends the encoder signals to the high-level control computer via EtherCAT with a real-time communication rate of 2 kHz. The inertial measurement unit (LORD, 3DM-GX3-25) in the upper body is employed to estimate the gravity direction, i.e., the  $z$ -axis. Two six DoFs F/T (force-torque)sensors (ATI, DAQ-Mini85) are mounted between each ankle and foot for monitoring the CoP rather than for feedback control.

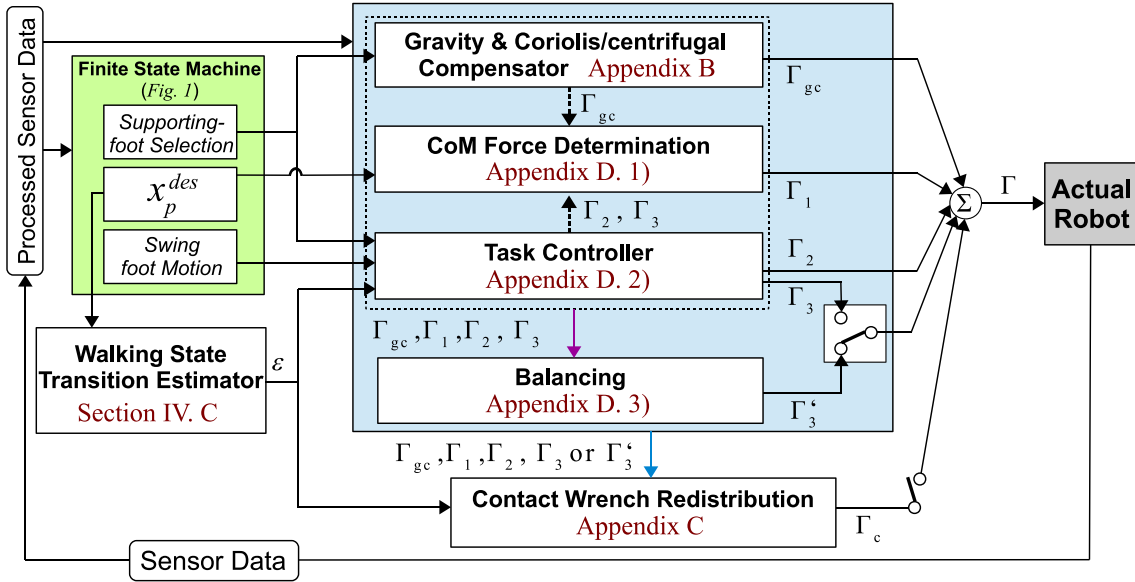


Fig. 4. Overview of the whole-body walking controller. The balancing algorithm operates only when the expected CoP is located out of the support polygon. This is illustrated as a switch. The contact wrench distribution torque  $\Gamma_c$  is considered when the contact wrench redistribution algorithm operates in the DS-state. This is illustrated as a switch after the contact null-space torque determination. Sensor data include encoders at the joints and an inertial measurement unit.

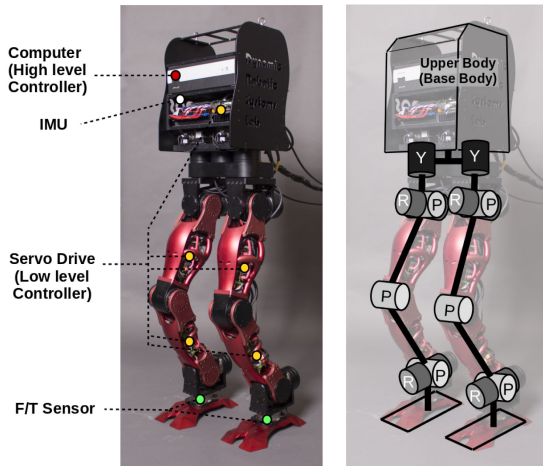


Fig. 5. Biped robot DYROS RED and schematic diagram of its kinematic structure.

The proposed controller is implemented in a high-level control computer with a real-time operating system, RTX 2009 environment, 2.66 GHz CPU (Intel i7-630 M processor), and 4 Gbyte memory. The controller is coded in the real-time software architecture RoboticsLab [36]. It integrates the reading of the sensor signal, calculation of the robot kinematics and dynamics, and writing of the torque commands calculated from the high-level controller, i.e., the proposed walking controller.

### B. Controller Implementation

The desired position and orientation for the operational space tasks are listed in Table II. As mentioned, the CoM motion is described in  $\mathbf{O}_w$ . Furthermore, the desired  $y$ -position of the

TABLE II  
REFERENCE POSITIONS AND ORIENTATIONS FOR TASKS

Task	Description	
CoM	$y$ -pos	0
	$z$ -pos	Initial value
Upper Body	Pitch	Tangential to gravity direction
	Roll	Tangential to gravity direction
	Yaw	Initial angle
Swing Foot	$x$ -pos	Pre-planned footstep position
	$y$ -pos	Pre-planned footstep position
	$z$ -pos	0.05 m for raising, 0.005m for landing
	Pitch	Tangential to gravity direction
	Roll	Tangential to gravity direction
	Yaw	Initial angle

CoM is zero, whereas the  $z$ -position of the CoM is constant for the LIPM. The desired position of the swing foot in the  $xy$ -plane is updated to the predetermined footstep position when the DS-state terminates. In addition, the height of the swing foot is updated when each walking state terminates. Note that the height of the landing foot is set to be 5 mm higher than that of the supporting foot to prevent early contact with the ground. The desired positions and orientations remain constant before each update.

For the reference forces  $\mathbf{F}_2$  and  $\mathbf{F}_3$  calculated by (6), low values of the gains for control are selected to render sufficiently compliant behavior:  $k_p = 100$  and  $k_v = 20$  for all the tasks. Nevertheless, since  $\mathbf{F}_s$  can become significantly large when the desired position is far from the actual position, an upper bound for the magnitude of  $\mathbf{F}_s$  is defined, and each component saturates when its magnitude is larger than the limit. For all



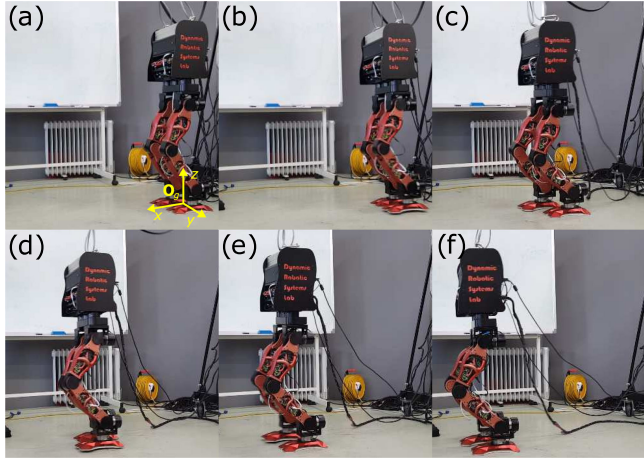


Fig. 6. Snapshots captured during forward walking experiment. (a) Initial posture of the robot. (b) First footstep. (c) Fourth footstep. (d) Seventh footstep. (e) Ninth footstep. (f) After final (11th) footstep.  $\mathbf{O}_g$  denotes the global coordinate system.

the experiments in this study, the margin parameters are set as  $\zeta = 0.03$  m and  $r = 0.08$  m, and the other empirically determined parameters are set as  $\Delta x_p^{\text{des}} = 0.0007$  m and  $d^{\text{thr}} = 0.01$  m.

## VI. EXPERIMENTAL RESULTS

In this section, three experiments are demonstrated to validate our proposed walking method. The first experiment shows the forward walking result, and the second and third experiments show the results when an unforeseen external disturbance is applied to the robot's body.

To represent the plotting results more intuitively, certain plots are described in a global coordinate system, which is obtained by assembling each local coordinate of each single footstep by postprocessing. In the following plots, the result from the global coordinate system  $\mathbf{O}_g$ , which is located at the center point between the initial positions of the two feet, is represented by the subscript  $g$ , i.e.,  $x_g$ ,  $y_g$ , and  $z_g$ . Furthermore, the result from  $\mathbf{O}_w$  is represented by the subscript  $w$ , i.e.,  $x_w$ ,  $y_w$ , and  $z_w$ .

### A. Experiment #1: Forward Walking

Forward walking with a step length of 0.1 m is conducted successfully using the proposed walking method, as shown in Fig. 6. The walking speed is approximately 0.11 m/s during the 11 steps. As shown in Fig. 7(a) and (b), the CoM motion and the time duration of each state are evidently irregular according to the time-independent walking pattern from the E-FSM.

The reference position and control result of the swing foot are plotted in Fig. 7(a)–(c). The swing foot motion has a few centimeters of control errors, which are caused mainly by low gains. Nonetheless, the compliant behavior enables a stable transition from the SSA-state to the DS-state, although the robot foot does not land completely on the ground at the end of the SSA-state. Note that the height of the swing foot can be negative because the ground is not perfectly flat.

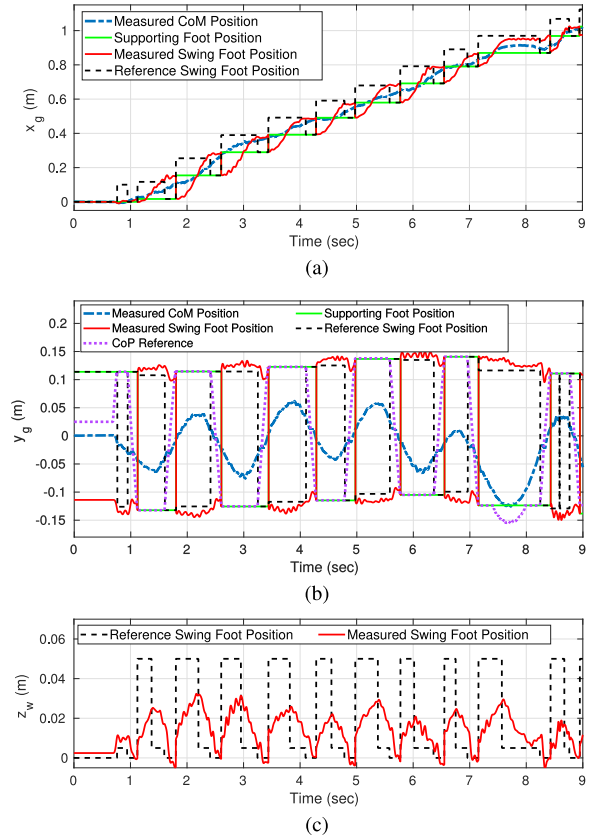


Fig. 7. CoM and foot control results of the walking experiment. (a)  $x$ -direction motion of the CoM and swing foot, supporting foot location, and reference position of swing foot in the global coordinate system. (b)  $y$ -direction motion of the CoM and swing foot, supporting foot location, reference position of swing foot, and reference CoP in the global coordinate system. (c) Swing foot motion and reference for vertical direction.

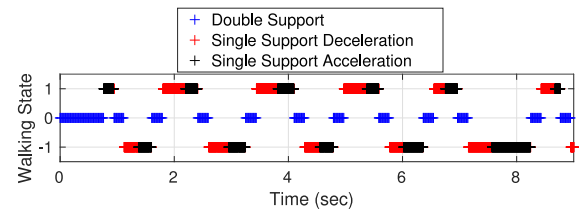


Fig. 8. Walking states of the forward walking experiment; “-1” denotes single support with right foot support and left foot swing, “0” denotes DS, and “1” denotes single support with left foot support and right foot swing. Note that each of these states denotes a state in the E-FSM, rather than the actual contact state.

The walking state results shown in Fig. 8 reveal two notable circumstances that illustrates how E-FSM operates without time variables. One is the omitted SSD-state after the first DS-state. The SSD-state cannot be observed because the CoM's velocity at the end of the first DS-state is approximately zero. That is, the SSD-state satisfies the termination criterion from the beginning, whereby it appears as if the SSD-state is omitted. The other is the different time duration for each walking state. As mentioned previously, due to the time-independent termination criterion of the E-FSM, each state during the experiment has a different time

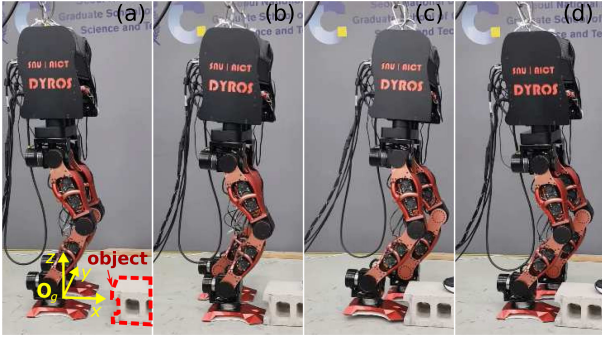


Fig. 9. Snapshots captured during the robot's walk on the surface on which a heavy obstacle is placed: (a) obstacle is placed in front of the right foot; (b) right foot swings and collides with the obstacle, wherein the proposed controller responds; (c) then, the left foot moves forward; (d) according to the disturbance from the obstacle, the right foot does not move forward, and (d) as a result, the robot keeps walking in the same place.  $\mathbf{O}_g$  denotes the global coordinate system.

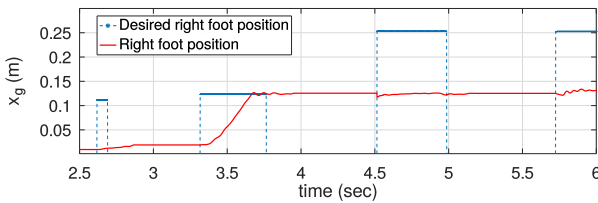


Fig. 10. Motion and reference of right foot in the global coordinate system during the walk on the surface on which a heavy obstacle is placed. Time duration without the desired position is when the right foot serves as a supporting foot.

duration. For example, during  $t \approx 7.24$ – $8.14$  s, when the distance between the CoM and CoP is marginal, as shown in Fig. 7(b), the magnitude of the CoM's deceleration becomes marginal, as can be shown from (1). This increases the time duration for the CoM velocity to become zero during the SSD-state. In contrast, during  $t \approx 6.56$ – $6.96$  s and  $t \approx 8.43$ – $8.76$  s, when the distance between the CoM and the CoP is greater than that in the other walking states, the duration of the SSD-state and the SSA-state are shorter because the magnitude of the CoM's acceleration increases.

In addition, it can be observed that the CoM successfully decelerates even when it gets close to the center of the supporting foot, as shown in Fig. 7(b). During  $t \approx 7.41$ – $8$  s, by considering the margin parameter  $\zeta$ , the desired CoP is at least 0.03 m away from the CoM location according to (3). Furthermore, the CoM stops its motion successfully before it advances beyond the supporting foot.

### B. Experiment #2: External Disturbance at Swing Foot

In the second experiment, a heavy obstacle is placed on the ground approximately 0.13 m in front of the starting position of the right foot to obstruct the swing foot motion as shown in Figs. 9 and 10. This experiment is carried out with a step length of 0.1 m, similar to Experiment #1.

During walking, the right foot collides with the obstacle while swinging. After the collision at  $t \approx 4.5$  s, it cannot move further,

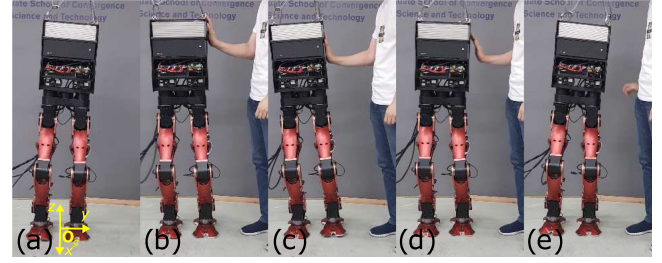


Fig. 11. Snapshots of the case wherein an external disturbance is imparted to the upper body of the robot by a human: (a) robot keeps walking in the same position; (b) human starts to push the robot from one side; (c) human keeps pushing to block the walking motion of the robot for approximately 1.8 s; (d) human releases the pushing hand from the robot by reducing the pushing force; and (e) after the release, the robot resumes walking.  $\mathbf{O}_g$  denotes the global coordinate system.

as shown in Fig. 10. Owing to the compliant behavior, the robot is stably controlled while keeping balance, even the right foot still has a reference position to move forward. The maximum force applied on the swing foot when it is completely blocked by the obstacle is approximately 32 N.

The robot walks repetitively in the same place after the right foot is blocked because the CoM moves toward the position of the swing foot which is blocked. This result reveals that stable walking can be achieved even if the swing foot cannot reach the planned location, which is aimed at by this study.

### C. Experiment #3: External Disturbance at Upper Body

In the third experiment, an external force is applied by a human to the upper body of the robot in the  $x$ -direction of  $\mathbf{O}_w$  while the robot is walking in the same position. As shown in Fig. 11, a human pushes the left side of the walking robot to block its motion for 2 s.

In this experiment, the E-FSM reacts to the external disturbance based on the measured CoM and CP. The experimental results are shown in Fig. 12. The CoM's velocity reduces to approximately zero at  $t \approx 26$  s due to the external force that blocks the robot motion during  $t \approx 25.5$ – $27.5$  s, as shown in Fig. 12(c) and (d). While the CoM is stationary, the CP moves marginally [see Fig. 12(b), and the SSA-state cannot be terminated because it cannot satisfy the termination criterion. During this period, although the force to accelerate the CoM is still being generated, the CoM can be maintained stationary because the force from the human and the generated CoM force are in equilibrium. At that time, the magnitude of the disturbance force in the  $x$ -direction is approximately 75 N. Therefore, after the person reduces the pushing force on the robot from  $t \approx 27.4$  s, the CoM begins to accelerate toward the swing foot, as intended in the SSA-state. Subsequently, the SSA-state terminates, and the DS-state commences when the CP reaches the reference position at  $t \approx 28.2$  s [see Fig. 12(a)]. A similar behavior occurs in the forward walking experiment, as shown in Fig. 13. In the experiment, the robot walks forward, and a human holds the upper body at approximately  $t \approx 6$  s to block its motion in both the  $x$ - and  $y$ -direction. Accordingly, the CoM's motion is stopped, and the transit from the SSA-state to the DS-state does not occur.

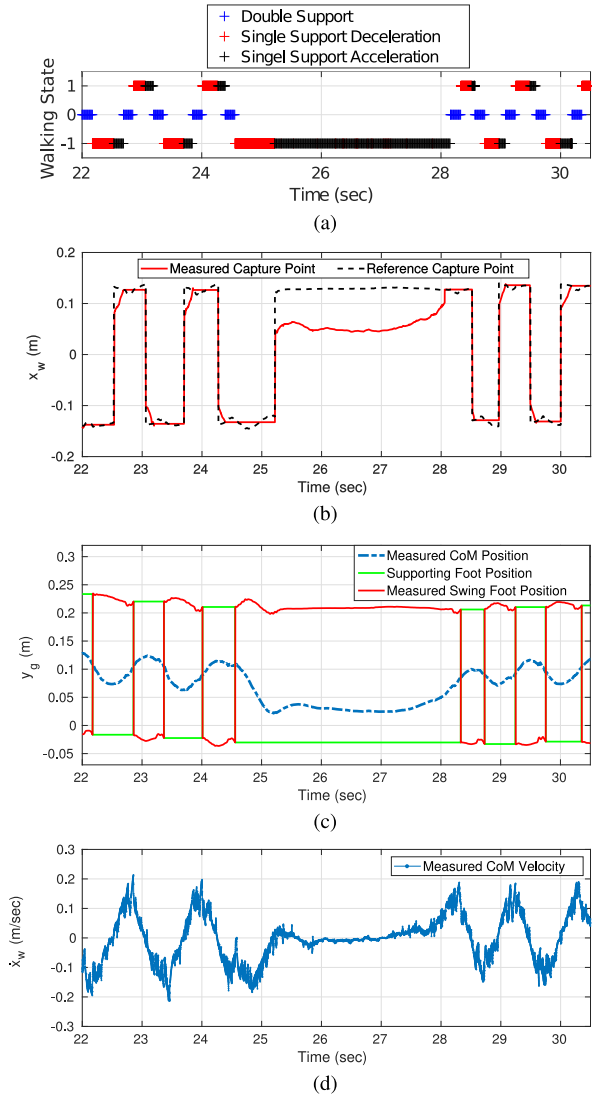


Fig. 12. (a) Walking state that occurs when a disturbance is applied on upper body during walking; “-1” denotes single support with right foot support and left foot swing, “0” denotes DS, and “1” the single support with left foot support and right foot swing. Note that each state denotes a state in the E-FSM, rather than the actual contact state. (b) Measured CP and reference CP in  $O_w$ . (c) CoM’s motion in the  $y$ -direction, the swing foot location, the supporting foot location, and the reference position of the swing foot in the global coordinate. (d) CoM’s velocity in the  $x$ -direction of  $O_w$ .

The robot resumes walking after the human releases the robot. These results show that the robot can continue to walk even when the upper body is disturbed by an external force, which is aimed at by this study.

#### D. Discussions

1) *Effectiveness of the State-Transition Parameter*: To investigate the impact of the state-transition parameter, comparative experiments with and without the use of  $\varepsilon$  are conducted under a scenario identical to that of Experiment #1. The discontinuous joint torque when  $\varepsilon$  is not applied is shown in Fig. 14(a), and the continuous torque when  $\varepsilon$  is applied is shown in Fig. 14(b). The discontinuous reference torque at  $t \approx 0.1, 0.4,$  and  $0.9$  s in

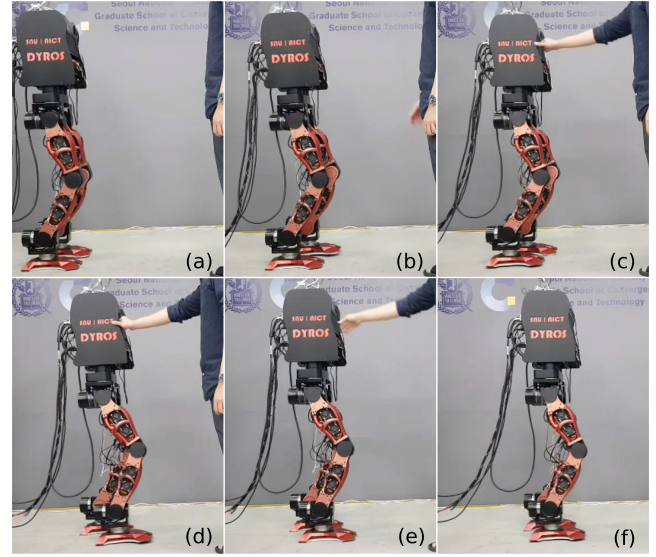


Fig. 13. Snapshots of the case wherein an external disturbance is applied by a human to the upper body of the robot: (a) robot starts walking; (b) after a few footsteps; (c) human starts to block the upper body motion; (d) swing foot moves when the upper body is disturbed; (e) after the human releases the robot, and (f) robot resumes walking.

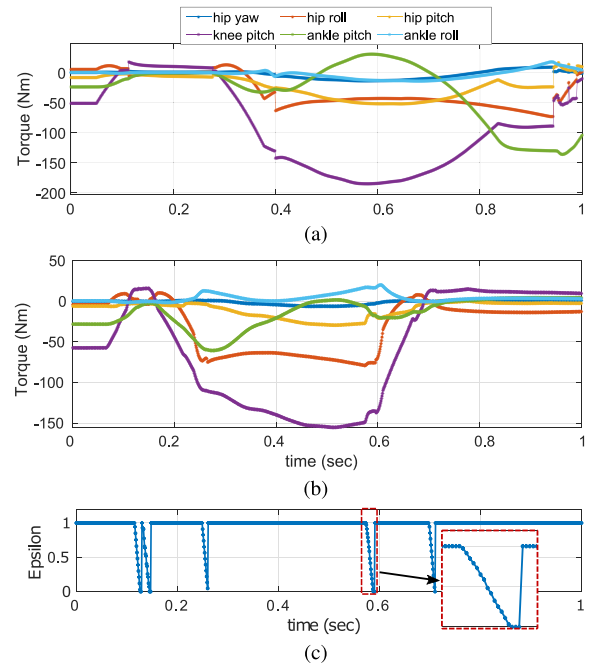


Fig. 14. Reference joint torque results of the left leg and state-transition parameter  $\varepsilon$  during the walking experiment: (a) results without the state-transition parameter; (b) results with the state-transition parameter; and (c) state-transition parameter  $\varepsilon$ .

Fig. 14(a) affects the torque controller and generates torque oscillation. This produces an unintended motion of the robot. As a result, the robot falls after  $t \approx 1.0$  s. On the other hand, when  $\varepsilon$  is applied as plotted in Fig. 14(c), continuous reference torque is achieved, and the robot walks successfully.

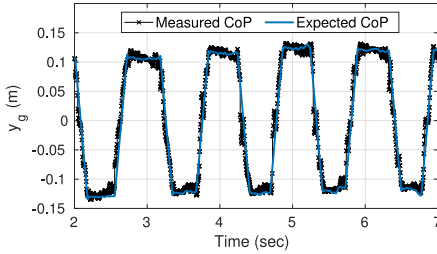


Fig. 15. CoP results obtained during walking. The measured CoP is from six DoFs force–torque sensors installed at the ankles of the robot.

2) *Accuracy Validation of the Generated CoP*: The efficacy of the multibody dynamics model-based control method is verified by comparing the calculated expected CoP and measured CoP. In Fig. 15, the CoP results during the walking experiment under a scenario identical to that of Experiment #1 are shown. Here, the expected CoP is calculated by (18) and (22) with reference torques, and the measured CoP is calculated using the measurements from the F/T sensors on each foot.

The magnitudes of the CoP errors are mostly between 0 and 0.02 m. These are relatively accurate results considering the absence of F/T sensor feedback control. These CoP errors are caused by the imperfect robot model, state estimation error, and the low-level torque control error. The results reveal that the CoP error tends not to vary noticeably even when a supporting foot moves and violates the stationary condition, in most cases, immediately after the DS-state begins. Thus, it can be considered that the violation of the stationary contact constraint in the walking controller does not have a critical effect on the CoP error. In addition, this result indirectly shows that the calculated CoM force in Appendix D is reliable because it can generate the CoP with a relatively marginal error.

## VII. CONCLUSION

In this study, we developed a reactive walking method by exploiting a walking algorithm based on the E-FSM and the operational space control approach. The developed method generates continuous walking through the circulating walking procedure in the E-FSM while producing reactivity against external disturbances. The reactivity is mainly obtained by adopting the proposed strategy wherein the CoM follows the swing foot position with the time-independent manner of the E-FSM. In addition, the compliant characteristic of the inverse dynamics-based controller realized the developed walking method in a real robot. The effectiveness of the proposed method was verified by experiments under three scenarios.

Although the developed walking method provides reactivity for stable and robust walking, the robot may fall if the unmodeled disturbance is excessive, e.g., if the CoM's position error in the  $y$ -direction is large or if the CoM is unable to move toward the swing foot. To improve the ability to maintain balance under severe unmodeled disturbances, a balancing algorithm can manage the CoP as well as the friction conditions by employing a numerical optimization technique. In addition, balancing ability can be enhanced by the CoP feedback control method with a

measured contact wrench from an F/T sensor. Reactivity of walking, as well as balancing ability, can be enhanced by a real-time footstep planning or adjusting method, which alters the swing foot position adaptively when a large control error occurs or the CoM motion is physically restricted on the  $y$ -axis of  $\mathbf{O}_w$ . In addition, the adoption of an accurate method for estimating the CoM's position and velocity can impart additional robustness to walking with the E-FSM. Our future study will be performed accordingly.

## APPENDIX

### A. Operational Space Dynamics [25]

The contact constrained dynamics in the operational space coordinate  $\mathbf{x}$  can be expressed as follows:

$$\Lambda \ddot{\mathbf{x}} + \boldsymbol{\mu} + \mathbf{p} = \mathbf{F}$$

where

$$\Lambda = \{\mathbf{J}\mathbf{A}^{-1}(\mathbf{I} - \mathbf{J}_c^T \bar{\mathbf{J}}_c^T) \mathbf{J}^T\}^{-1}$$

$$\bar{\mathbf{J}}^T = \Lambda \mathbf{J} \mathbf{A}^{-1} (\mathbf{I} - \mathbf{J}_c^T \bar{\mathbf{J}}_c^T)$$

$$\boldsymbol{\mu} = \bar{\mathbf{J}}^T \mathbf{b} - \Lambda \bar{\mathbf{J}} \dot{\mathbf{q}} + \Lambda \mathbf{J} \mathbf{A}^{-1} \mathbf{J}_c^T \Lambda_c \dot{\mathbf{J}}_c \dot{\mathbf{q}} \quad (12)$$

$$\mathbf{p} = \bar{\mathbf{J}}^T \mathbf{g} \quad (13)$$

and  $\bar{\bullet}$  denotes the dynamically consistent inverse of  $\bullet$ .  $\mathbf{J}$  is the Jacobian matrix, which is defined as  $\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}}$ .  $\Lambda$ ,  $\boldsymbol{\mu}$ ,  $\mathbf{p}$ , and  $\mathbf{F}$  are the inertia matrix, Coriolis/centrifugal force vector, gravity force vector, and wrench vector, respectively, in the operational space.

The torque solution that generates the force  $\mathbf{F}$  for a redundant case when the number of DoFs of tasks is less than  $n - c$  is obtained from the following equation [37], [38]:

$$\boldsymbol{\Gamma} = \tilde{\mathbf{J}}^T \mathbf{F} = \tilde{\mathbf{J}}^T \{\Lambda \ddot{\mathbf{x}}^* + \boldsymbol{\mu} + \mathbf{p}\}$$

where  $\tilde{\mathbf{J}}^T \triangleq \overline{(\mathbf{J}^T \mathbf{S}^T)}$ , which is given by the weighted pseudoinverse with the contact constrained inertia weighting matrix  $\mathbf{W} = \mathbf{S} \mathbf{A}^{-1} (\mathbf{I} - \mathbf{J}_c^T \bar{\mathbf{J}}_c^T) \mathbf{S}^T$  to minimize the joint acceleration energy; and  $\ddot{\mathbf{x}}^*$  is the reference acceleration vector in the operational space.

The hierarchical control structure can be formulated with the null-space projection matrix, which is defined as

$$\tilde{\mathbf{N}}^T = \mathbf{I} - \tilde{\mathbf{J}}^T \tilde{\mathbf{J}}^T = \mathbf{I} - \tilde{\mathbf{J}}^T (\mathbf{J}^T \mathbf{S}^T).$$

The hierarchical control formulation with the lower priority task torque  $\boldsymbol{\Gamma}_0$  can then be expressed as follows:

$$\boldsymbol{\Gamma} = \tilde{\mathbf{J}}^T \mathbf{F} + \tilde{\mathbf{N}}^T \boldsymbol{\Gamma}_0.$$

### B. Gravity and Coriolis/Centrifugal Force Compensation, $\boldsymbol{\Gamma}_{gc}$

In a robot having task redundancy with multiple tasks, it is efficient to compensate the gravity force  $\mathbf{g}$  and the Coriolis/centrifugal force  $\mathbf{b}$  in the joint space rather than the operational space. For the joint space compensation, the Jacobian matrix  $\mathbf{J}_{gc}$  is defined as follows:

$$\mathbf{J}_{gc} = \begin{bmatrix} \mathbf{0}_{k \times 6} & \mathbf{I}_{k \times k} \end{bmatrix}.$$

According to  $\mathbf{J}_{gc}$ , only the actual joints actuate for the gravity and Coriolis/centrifugal force compensation. The gravity and Coriolis/centrifugal force compensation torque  $\Gamma_{gc}$  can be obtained by applying  $\mathbf{J}_{gc}$  to (12) and (13) as follows:

$$\Gamma_{gc} = -\bar{\mathbf{J}}_{gc}^T \mathbf{g} - \bar{\mathbf{J}}_{gc}^T \mathbf{b} + \Lambda_{gc} \dot{\mathbf{J}}_{gc} \dot{\mathbf{q}} + \Lambda_{gc} \mathbf{J}_{gc} \mathbf{A}^{-1} \mathbf{J}_c^T \Lambda_c \dot{\mathbf{J}}_c \dot{\mathbf{q}}$$

where

$$\begin{aligned} \bar{\mathbf{J}}_{gc}^T &= \Lambda_{gc} \mathbf{J}_{gc} \mathbf{A}^{-1} (\mathbf{I} - \mathbf{J}_c^T \bar{\mathbf{J}}_c^T) \\ \Lambda_{gc} &= \{ \mathbf{J}_{gc} \mathbf{A}^{-1} (\mathbf{I} - \mathbf{J}_c^T \bar{\mathbf{J}}_c^T) \mathbf{J}_{gc}^T \}^{-1}. \end{aligned}$$

### C. Contact Wrench Redistribution Method [35]

Here, we introduce a method to obtain the contact wrench distribution torque  $\Gamma_c$  that can modify the distributed contact wrench without affecting either the tasks or the resultant contact wrench.

When we consider only plane contacts, there are always six redundant DoFs in the contact space for the DS state. This implies that  $\mathbf{W}$  in Appendix A is rank-deficient. Therefore,  $\mathbf{W}$  can be decomposed through singular value decomposition as follows:

$$\mathbf{W} \triangleq \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T = \mathbf{V} \mathbf{\Sigma} \mathbf{V}^T \quad (14)$$

where  $\mathbf{U}$  is a unitary matrix,  $\mathbf{\Sigma}$  is a diagonal matrix with singular values on the diagonal, and  $\mathbf{V}$  is a unitary matrix. Here,  $\mathbf{U}$  is equivalent to  $\mathbf{V}$  because  $\mathbf{W}$  is a symmetric matrix, and six components of  $\mathbf{\Sigma}$  are close to zero because  $\mathbf{W}$  is rank deficient. Therefore, (14) can be expressed as

$$\mathbf{W} = \begin{bmatrix} \mathbf{v}_1 & \mathbf{v}_2 \end{bmatrix} \begin{bmatrix} \mathbf{s}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \mathbf{v}_2^T \end{bmatrix}.$$

The torques in the vector space spanned by the matrix  $\mathbf{v}_2$  would neither affect the acceleration energy nor inject energy to the tasks in the operational space. Therefore, if we achieve the torque inside that space, it would modify the contact wrench without disturbing the tasks.

The contact wrench distribution torque  $\Gamma_c$  is then expressed as

$$\Gamma_c = \mathbf{v}_2 \boldsymbol{\alpha} \quad (15)$$

where  $\boldsymbol{\alpha}$  is the vector for generating the contact wrench correction vector  $\mathbf{f}_c^{\text{cor}}$ . Based on (8), the given  $\mathbf{f}_c^{\text{cor}}$  can be described as a function of  $\Gamma_c$

$$\mathbf{f}_c^{\text{cor}} = \mathbf{S}_c \bar{\mathbf{J}}_c^T \mathbf{S}^T \Gamma_c$$

where  $\mathbf{S}_c$  is the selection matrix for selecting six DoFs contact wrench components. Subsequently, the vector  $\boldsymbol{\alpha}$  can be obtained from the abovementioned equation using the dynamically consistent inverse of  $\mathbf{S}_c \bar{\mathbf{J}}_c^T \mathbf{S}^T \mathbf{v}_2$ . Finally, the contact wrench redistribution torque  $\Gamma_c$  and relation matrix  $\mathbf{Y}_c$  can be obtained by substituting the inversion into (15) as follows:

$$\begin{aligned} \Gamma_c &= \mathbf{v}_2 \underbrace{\mathbf{S}_c \bar{\mathbf{J}}_c^T \mathbf{S}^T \mathbf{v}_2}_{=\mathbf{Y}_c} \mathbf{f}_c^{\text{cor}} \\ &= \mathbf{Y}_c \mathbf{f}_c^{\text{cor}}. \end{aligned} \quad (16)$$

### D. Determination of Reference Task Torque $\Gamma_t$

1) *CoM Force Determination for  $\Gamma_1$* : To implement the proposed E-FSM-based walking algorithm, it is important to achieve a CoM force ( ${}^x F_m$ ) that can generate the given desired CoP ( $x_p^{\text{des}}$ ).

For this purpose, the relationship between  ${}^x F_m$  and the expected CoP ( $x_p^{\text{exp}}$ ) has to be achieved. The contact wrench and task force relationship is derived by substituting the torques  $\Gamma_1$ ,  $\Gamma_2$ ,  $\Gamma_3$ , and  $\Gamma_{gc}$  into (9). The resultant contact wrench yields

$$\begin{aligned} \mathbf{F}_c &= \underbrace{\mathbf{K} \bar{\mathbf{J}}_c^T \mathbf{S}^T \Gamma_1}_{\mathbf{F}_{c|1}} + \underbrace{\mathbf{K} \bar{\mathbf{J}}_c^T \mathbf{S}^T \Gamma_2}_{\mathbf{F}_{c|2}} + \underbrace{\mathbf{K} \bar{\mathbf{J}}_c^T \mathbf{S}^T \Gamma_3}_{\mathbf{F}_{c|3}} \\ &+ \underbrace{\mathbf{K} (\bar{\mathbf{J}}_c^T \mathbf{S}^T \Gamma_{gc} - \boldsymbol{\mu}_c - \mathbf{p}_c)}_{\mathbf{F}_{c|b}} \end{aligned} \quad (17)$$

where  $\mathbf{F}_{c|1}$ ,  $\mathbf{F}_{c|2}$ ,  $\mathbf{F}_{c|3}$ , and  $\mathbf{F}_{c|b}$  are the resultant contact wrench components caused by the first, second, third priority tasks, and the bias force (Coriolis/centrifugal and gravity forces), respectively. Next, the expected CoP in the  $x$ -direction ( $x_p^{\text{exp}}$ ) can be obtained from the definition of the CoP described in [39] with the components of  $\mathbf{F}_c$  as follows:

$$x_p^{\text{exp}} = -{}^y M_{c|z} / {}^z F_c \quad (18)$$

$$= -\frac{{}^y M_{c|1} + {}^y M_{c|2} + {}^y M_{c|3} + {}^y M_{c|b}}{{}^z F_{c|1} + {}^z F_{c|2} + {}^z F_{c|3} + {}^z F_{c|b}} \quad (19)$$

where  ${}^y M_{c|\bullet}$  and  ${}^z F_{c|\bullet}$  are the contact moment and force components, respectively, of  $\mathbf{F}_{c|\bullet}$ .

By excluding  ${}^y M_{c|2}$  and  ${}^y M_{c|3}$  from (19), i.e., by considering only the contact moments affected by the CoM's horizontal motion  ${}^y M_{c|1}$  and the gravity force  ${}^y M_{c|b}$ ,  $x_p^{\text{exp}}$  can be considered to be similar to the CoP described by the LIPM. Note that  $x_p^{\text{des}}$  is determined from the LIPM concept that considers only the horizontal motion of the CoM and the gravity force. In addition, it can be assumed to be  ${}^z F_{c|1} \approx 0$ , since  $\mathbf{F}_{c|1}$  regards only the  $x$ -direction motion of the CoM perpendicular to the  $z$ -direction, so that  ${}^z F_{c|1}$  can be removed from the abovementioned equation. Equation (19) can then be expressed as follows:

$$x_p^{\text{des}} = -\frac{{}^y M_{c|1} + {}^y M_{c|b}}{{}^z F_{c|2} + {}^z F_{c|3} + {}^z F_{c|b}}. \quad (20)$$

To obtain  ${}^x F_m$  from (20), a relationship between  ${}^y M_{c|1}$  and  ${}^x F_m$  is required because  ${}^y M_{c|1}$  is the only variable related to  ${}^x F_m$ . This relationship can be achieved by substituting  $\Gamma_1$  in (9) and multiplying the substituted equation with the selection vector  $\mathbf{S}_{c|1} = [0 \ 0 \ 0 \ 0 \ 1 \ 0]^T$

$$\begin{aligned} {}^y M_{c|1} &= \mathbf{S}_{c|1}^T \mathbf{K} \bar{\mathbf{J}}_c^T \mathbf{S}^T \Gamma_1 \\ &= (\mathbf{S}_{c|1}^T \mathbf{K} \bar{\mathbf{J}}_c^T \mathbf{S}^T \tilde{\mathbf{J}}_1^T) {}^x F_m \\ &= Q_1 {}^x F_m. \end{aligned} \quad (21)$$

Finally,  ${}^x F_m$  can be computed by substituting (21) into (20) as follows:

$${}^x F_m = \frac{{}^y M_{c|1}}{Q_1} = -\frac{({}^z F_{c|2} + {}^z F_{c|3} + {}^z F_{c|b}) x_p^{\text{des}} + {}^y M_{c|b}}{Q_1}.$$

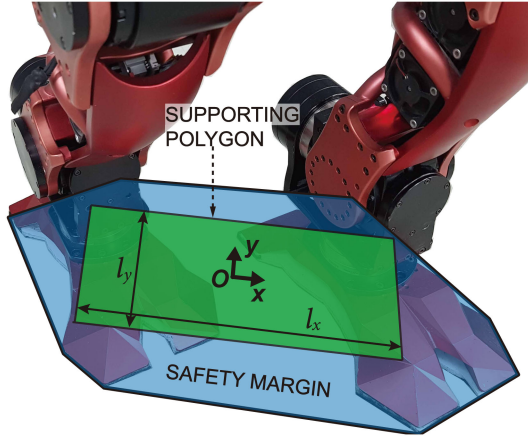


Fig. 16. Safety margin (blue colored area) applied to support polygon (green colored area) during DS-state. The support polygon is assumed to be smaller than the actual one by applying the safety margin.

Then,  $\Gamma_1$  is given as  $\Gamma_1 = \tilde{\mathbf{J}}_1^T \mathbf{x} F_m$ .

2) *Determination of force for  $\Gamma_2$  and  $\Gamma_3$* : The force vectors  $\mathbf{F}_2$  and  $\mathbf{F}_3$  are determined using (6) for the position control of the task in the operational space with the predetermined reference positions. During each state of the E-FSM, the desired positions are set as constants until the next state starts. The controlling position can thereby reach the desired position rapidly without considering a time-varying trajectory. Then, the control torque for the second and third tasks are achieved as  $\Gamma_2 = \tilde{\mathbf{N}}_1^T \tilde{\mathbf{J}}_2^T \mathbf{F}_2$  and  $\Gamma_3 = \tilde{\mathbf{N}}_1^T \tilde{\mathbf{N}}_2^T \tilde{\mathbf{J}}_3^T \mathbf{F}_3$ , respectively.

3) *Determination of Balancing Force for  $\Gamma_3'$* : In this study, the balance state is predicted in real-time by the expected CoP calculated with the given reference torque  $\Gamma$  ( $= \Gamma_1 + \Gamma_2 + \Gamma_3 + \Gamma_{gc}$ ). When balancing is considered necessary (i.e., when the expected CoP is positioned out of the support polygon) for the specified instant, the balancing force vector  $\mathbf{F}_3^{\text{bal}} \in \mathbb{R}^2$  is calculated to modify the CoP, and then added to  $\mathbf{F}_3$  to obtain the modified torque  $\Gamma_3'$ . Then,  $\Gamma_3$  is replaced with  $\Gamma_3'$  for balancing motion.

The expected CoP can be calculated using the resultant contact wrench vector  $\mathbf{F}_c$  obtained from (9) before sending it to the joint torque controller. The  $x$ -component of the CoP can be predicted using (18), and its  $y$ -component can be calculated as

$$y_p^{\text{exp}} = x M_c / z F_c. \quad (22)$$

As is widely established, if  $x_p^{\text{exp}}$  or  $y_p^{\text{exp}}$  is positioned out of the support polygon, the robot may lose plane contact, and it may fall [40]. Therefore, for balancing, the expected CoPs have to satisfy the following conditions:

$$-\frac{l_x}{2} < x_p^{\text{exp}} < \frac{l_x}{2}, \quad -\frac{l_y}{2} < y_p^{\text{exp}} < \frac{l_y}{2} \quad (23)$$

where  $l_x$  and  $l_y$  are the lengths of the square-shaped support polygon on the  $x$ - and  $y$ -axis, respectively. The area of the support polygon is set to be smaller than the actual area by applying the safety margin (see Fig. 16), to prevent the robot from falling due to the model error.

When  $x_p^{\text{exp}}$  or  $y_p^{\text{exp}}$  violates the abovementioned condition, we modify the expected CoP to establish it at the nearest boundary of the support polygon. This is similar to our previous balancing method [35]. The modified expected CoPs  $x_p^{\text{exp}'}$  and  $y_p^{\text{exp}'}$  are then expressed as

$$x_p^{\text{exp}'} = \text{sgn}(x_p^{\text{exp}}) \frac{l_x}{2}, \quad y_p^{\text{exp}'} = \text{sgn}(y_p^{\text{exp}}) \frac{l_y}{2}.$$

To modify  $x_p^{\text{exp}}$  and  $y_p^{\text{exp}}$  as  $x_p^{\text{exp}'}$  and  $y_p^{\text{exp}'}$  by the rotating motion of the upper body, the force vector  $\mathbf{F}_3^{\text{bal}} \in \mathbb{R}^2$  is added to the third priority task. From (11), the modified torque  $\Gamma_3'$  can be defined as

$$\Gamma_3' = \tilde{\mathbf{N}}_1^T \tilde{\mathbf{N}}_2^T \tilde{\mathbf{J}}_3^T (\mathbf{F}_3 + \mathbf{F}_3^{\text{bal}}). \quad (24)$$

To calculate  $\mathbf{F}_3^{\text{bal}}$ , the equation of the modified resultant contact wrench  $\mathbf{F}'_c$  is first obtained by replacing  $\Gamma_3$  with  $\Gamma_3'$  in (17). It can then be described with two terms as

$$\mathbf{F}'_c = \underbrace{\mathbf{K} \tilde{\mathbf{J}}_c^T \mathbf{S}^T \Gamma_3'}_{\mathbf{F}'_{c|3}} + \underbrace{\mathbf{K} \{ \tilde{\mathbf{J}}_c^T \mathbf{S}^T (\Gamma_1 + \Gamma_2 + \Gamma_{gc}) - \boldsymbol{\mu}_c - \mathbf{p}_c \}}_{\mathbf{F}_{c|\sigma} = \mathbf{F}_{c|1} + \mathbf{F}_{c|2} + \mathbf{F}_{c|b}} \quad (25)$$

where  $\mathbf{F}'_{c|3}$  denotes the modifying contact wrench term for balancing, and  $\mathbf{F}_{c|\sigma}$  denotes the contact wrench caused by the other tasks and the bias force. By substituting the contact force and the moment components of  $\mathbf{F}'_c$  into (18) and (22),  $x_p^{\text{exp}'}$  and  $y_p^{\text{exp}'}$  can be described as

$$x_p^{\text{exp}'} = -\frac{y M'_{c|3} + y M_{c|\sigma}}{z F'_{c|3} + z F_{c|\sigma}}, \quad y_p^{\text{exp}'} = \frac{x M'_{c|3} + x M_{c|\sigma}}{z F'_{c|3} + z F_{c|\sigma}}$$

where  $y M'_{c|3}$  and  $z F'_{c|3}$  are the contact wrench components of  $\mathbf{F}'_{c|3}$ , and  $y M_{c|\sigma}$  and  $z F_{c|\sigma}$  are those of the determined vector  $\mathbf{F}_{c|\sigma}$  in (25). Note that  $y M_{c|\sigma}$  and  $z F_{c|\sigma}$  are constant in the abovementioned equation because they are calculated with the determined vectors  $\Gamma_1$ ,  $\Gamma_2$ ,  $\Gamma_{gc}$ ,  $\boldsymbol{\mu}_c$ , and  $\mathbf{p}_c$ , as shown in (25). In addition, in the abovementioned equation, it can be assumed that  $z F'_{c|3} \approx 0$  because the upper body orientation motion causes negligible variation in the vertical force. Then, the contact moments for calculating the modified CoPs can be expressed as

$$x M'_{c|3} = y_p^{\text{exp}'} z F_{c|\sigma} - x M_{c|\sigma} \quad (26)$$

$$y M'_{c|3} = -x_p^{\text{exp}'} z F_{c|\sigma} - y M_{c|\sigma}. \quad (27)$$

To express the abovementioned equations with respect to  $\mathbf{F}_3^{\text{bal}}$ , the relationship between  $[x M'_{c|3} \ y M'_{c|3}]^T$  and  $\mathbf{F}'_{c|3}$  can be expressed as follows:

$$\begin{bmatrix} x M'_{c|3} \\ y M'_{c|3} \end{bmatrix} = \mathbf{S}_{c|3} \mathbf{F}'_{c|3}, \quad \text{where } \mathbf{S}_{c|3} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}.$$

The following equation is obtained by substituting  $\mathbf{F}'_{c|3}$

described in (25) into the abovementioned equation:

$$\begin{aligned} \begin{bmatrix} x M'_{c|3} \\ y M'_{c|3} \end{bmatrix} &= \mathbf{S}_{c|3} \mathbf{K} \mathbf{J}_c^T \mathbf{S}^T \mathbf{\Gamma}'_3 \\ &= \underbrace{(\mathbf{S}_{c|3} \mathbf{K} \mathbf{J}_c^T \mathbf{S}^T \tilde{\mathbf{N}}_1^T \tilde{\mathbf{N}}_2^T \tilde{\mathbf{J}}_3^T)}_{\mathbf{Q}_3} (\mathbf{F}_3 + \mathbf{F}_3^{\text{bal}}). \end{aligned} \quad (28)$$

Finally, when  $\mathbf{Q}_3$  is invertible, the modified task force is obtained by substituting (26) and (27) into (28) as follows:

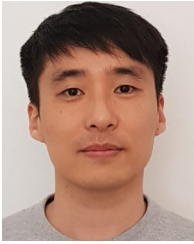
$$\mathbf{F}_3^{\text{bal}} = \mathbf{Q}_3^{-1} \begin{bmatrix} y'_p z F_{c|\sigma} - x M_{c|\sigma} \\ -x'_p z F_{c|\sigma} - y M_{c|\sigma} \end{bmatrix} - \mathbf{F}_3. \quad (29)$$

Because we can obtain  $\mathbf{\Gamma}'_3$  by substituting (29) into (24), the total task torque  $\mathbf{\Gamma}_t$  can be determined with  $\mathbf{\Gamma}'_3$  instead of  $\mathbf{\Gamma}_3$ . Finally,  $\mathbf{\Gamma}_t$  is expressed as

$$\mathbf{\Gamma}_t = \begin{cases} \mathbf{\Gamma}_1 + \mathbf{\Gamma}_2 + \mathbf{\Gamma}_3, & \text{if (23)} \\ \mathbf{\Gamma}_1 + \mathbf{\Gamma}_2 + \mathbf{\Gamma}_3', & \text{else.} \end{cases}$$

#### REFERENCES

- [1] M. Vukobratović and B. Borovac, “Zero-moment point—thirty five years of its life,” *Int. J. Humanoid Robot.*, vol. 1, no. 1, pp. 157–173, 2004.
- [2] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, “The 3 d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2001, vol. 1, pp. 239–246.
- [3] S. Kajita *et al.*, “Biped walking pattern generation by using preview control of zero-moment point,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2003, vol. 2, pp. 1620–1626.
- [4] P.-B. Wieber, “Trajectory free linear model predictive control for stable walking in the presence of strong perturbations,” in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 137–142.
- [5] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, “Capture point: A step toward humanoid push recovery,” in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 200–207.
- [6] T. Takenaka, T. Matsumoto, and T. Yoshiike, “Real time motion generation and control for biped robot-1 st report: Walking gait pattern generation,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2009, pp. 1084–1091.
- [7] M. B. Popovic, A. Goswami, and H. Herr, “Ground reference points in legged locomotion: Definitions, biological trajectories and control implications,” *Int. J. Robot. Res.*, vol. 24, no. 12, pp. 1013–1032, 2005.
- [8] 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 Trans. Robot.*, vol. 23, no. 6, pp. 1285–1293, Dec. 2007.
- [9] K. Nishiwaki and S. Kagami, “Simultaneous planning of com and ZMP based on the preview control method for online walking control,” in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2011, pp. 745–751.
- [10] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, “Online walking motion generation with automatic footstep placement,” *Adv. Robot.*, vol. 24, no. 5/6, pp. 719–737, 2010.
- [11] J. Engelsberger, C. Ott, and A. Albu-Schäffer, “Three-dimensional bipedal walking control based on divergent component of motion,” *IEEE Trans. Robot.*, vol. 31, no. 2, pp. 355–368, Apr. 2015.
- [12] M. Khadiv, A. Herzog, S. A. A. Moosavian, and L. Righetti, “Walking control based on step timing adaptation,” *IEEE Trans. Robot.*, vol. 36, no. 3, pp. 629–643, Apr. 2020.
- [13] M. Morisawa *et al.*, “Combining suppression of the disturbance and reactive stepping for recovering balance,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 3150–3156.
- [14] M. Morisawa, F. Kanehiro, K. Kaneko, S. Kajita, and K. Yokoi, “Reactive biped walking control for a collision of a swinging foot on uneven terrain,” in *Proc. RAS Int. Conf. Humanoid Robots*, 2011, pp. 768–773.
- [15] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, “Hybrid zero dynamics of planar biped walkers,” *IEEE Trans. Autom. Control*, vol. 48, no. 1, pp. 42–56, Jan. 2003.
- [16] A. Hereid, E. A. Cousineau, C. M. Hubicki, and A. D. Ames, “3-D dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1447–1454.
- [17] C. Hubicki *et al.*, “Atrias: Design and validation of a tether-free 3D-capable spring-mass bipedal robot,” *Int. J. Robot. Res.*, vol. 35, no. 12, pp. 1497–1521, 2016.
- [18] B. Griffin and J. Grizzle, “Nonholonomic virtual constraints and gait optimization for robust walking control,” *Int. J. Robot. Res.*, vol. 36, no. 8, pp. 895–922, 2017.
- [19] K. Yin, K. Loken, and M. Van de Panne, “Simbicon: Simple biped locomotion control,” in *Proc. ACM Trans. Graph.*, vol. 26, no. 3, 2007, pp. 105-es.
- [20] P. Faloutsos, M. Van de Panne, and D. Terzopoulos, “Composable controllers for physics-based character animation,” in *Proc. Annu. Conf. Comput. Graph. Interactive Techn.*, 2001, pp. 251–260.
- [21] M. De Lasa, I. Mordatch, and A. Hertzmann, “Feature-based locomotion controllers,” in *Proc. ACM Trans. Graph.*, vol. 29, no. 4, 2010, pp. 1–10.
- [22] J. Pratt *et al.*, “Capturability-based analysis and control of legged locomotion, part 2: Application to m2v2, a lower-body humanoid,” *Int. J. Robot. Res.*, vol. 31, no. 10, pp. 1117–1133, 2012.
- [23] K. Seo, J. Kim, and K. Roh, “Towards natural bipedal walking: Virtual gravity compensation and capture point control,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2012, pp. 4019–4026.
- [24] J. Pratt, C.-M. Chew, A. Torres, P. Dilworth, and G. Pratt, “Virtual model control: An intuitive approach for bipedal locomotion,” *Int. J. Robot. Res.*, vol. 20, no. 2, pp. 129–143, 2001.
- [25] J. Park and O. Khatib, “Contact consistent control framework for humanoid robots,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2006, pp. 1963–1969.
- [26] S. Kuindersma *et al.*, “Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot,” *Auton. Robots*, vol. 40, no. 3, pp. 429–455, 2016.
- [27] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, “Optimization-based full body control for the DARPA robotics challenge,” *J. Field Robot.*, vol. 32, no. 2, pp. 293–312, 2015.
- [28] G. Mesesan, J. Engelsberger, G. Garofalo, C. Ott, and A. Albu-Schäffer, “Dynamic walking on compliant and uneven terrain using DCM and passivity-based whole-body control,” in *Proc. IEEE-RAS 19th Int. Conf. Humanoid Robots*, 2019, pp. 25–32.
- [29] N. Mansard, “A dedicated solver for fast operational-space inverse dynamics,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 4943–4949.
- [30] L. Saab, O. E. Ramos, F. Keith, N. Mansard, P. Soueres, and J.-Y. Fourquet, “Dynamic whole-body motion generation under rigid contacts and other unilateral constraints,” *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 346–362, Apr. 2013.
- [31] A. Herzog, N. Rotella, S. Mason, F. Grimmering, S. Schaal, and L. Righetti, “Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid,” *Auton. Robots*, vol. 40, no. 3, pp. 473–491, 2016.
- [32] Y. Lee and J. Park, “Reactive bipedal walking method for torque controlled robot,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 395–402.
- [33] M. Schwartz, S. Hwang, Y. Lee, J. Won, S. Kim, and J. Park, “Aesthetic design and development of humanoid legged robot,” in *Proc. IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 13–19.
- [34] T. Koolen, T. DeBoer, J. Reubla, A. Goswami, and J. Pratt, “Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models,” *Int. J. Robot. Res.*, vol. 31, no. 9, pp. 1094–1113, 2012.
- [35] Y. Lee, S. Hwang, and J. Park, “Balancing of humanoid robot using contact force/moment control by task-oriented whole body control framework,” *Auton. Robots*, vol. 40, no. 3, pp. 457–472, 2016.
- [36] “Dynamic simulator RoboticsLab.” 2011, [Online]. Available: <http://www.wonikrobotics.com/RoboticsLab.htm>
- [37] J. Park, “The relationship between controlled joint torque and end-effector force in underactuated robotic systems,” *Robotica*, vol. 29, no. 4, pp. 581–584, 2011.
- [38] H. Bruyninckx and O. Khatib, “Gauss’ principle and the dynamics of redundant and constrained manipulators,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2000, vol. 3, pp. 2563–2568.
- [39] P. Sardain and G. Bessonnet, “Forces acting on a biped robot. Center of pressure-zero moment point,” *IEEE Trans. Syst., Man, Cybern.-Part A: Syst. Humans*, vol. 34, no. 5, pp. 630–637, Sep. 2004.
- [40] A. Goswami, “Postural stability of biped robots and the foot-rotation indicator (FRI) point,” *Int. J. Robot. Res.*, vol. 18, no. 6, pp. 523–533, 1999.



**Yisoo Lee** (Member, IEEE) received the B.S. and M.S. degrees in naval architecture and ocean engineering and the Ph.D. degree in intelligent convergence systems from the Department of Intelligent Convergence Systems from Seoul National University, Seoul, South Korea, in 2008, 2010 and 2017, respectively

From 2017 to 2018, he was a Postdoctoral Researcher with the Seoul National University. From 2018 to 2020, he was a Postdoc with the Humanoids and Human Centered Mechatronics lab, Istituto Italiano di Tecnologia, Genoa, Italy. He is currently a Senior Researcher with the Korea Institute of Science and Technology, Seoul, South Korea. His research interests include humanoid robot, robot locomotion, manipulation, and optimal control.



**Jino Lee** (Senior Member, IEEE) received the B.S. degree (*Summa Cum Laude*) in mechanical engineering from Hanyang University, Seoul, South Korea, in 2003, and the M.Sc. and Ph.D. degrees in mechanical engineering from the Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 2012.

He held a Postdoctoral position from 2012 to 2017 and a Research Scientist position from 2017 to 2020 with the Department of Advanced Robotics, Istituto Italiano di Tecnologia (IIT), Genoa, Italy. He is currently a Research Scientist with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Wessling, Germany. His professional is about robotics and control engineering, which include manipulation of highly redundant robots such as dual-arm and humanoids, robust control of nonlinear systems, and compliant robotic system control for safe human-robot interaction.

Dr. Lee is a member of the IEEE Robotics and Automation, the IEEE Control Systems, and the IEEE Industrial Electronics Societies; and he was an Editor on the Conference Paper Review Board (CPRB) of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS).



**Hosang Lee** received the B.S. degree in mechanical and aerospace engineering 2013 from Seoul National University, Seoul, South Korea, where he is currently working toward the Ph.D. degree in intelligent convergence systems with the Graduate School of Convergence Science and Technology.

His research interests are torque-based whole-body control and manipulation.



**Jaeheung Park** (Member, IEEE) received the B.S. and M.S. degrees in aerospace engineering from Seoul National University, Seoul, South Korea, in 1995 and 1999, respectively, and the Ph.D. degree in aeronautics and astronautics from Stanford University, Stanford, CA, USA, in 2006.

From 2006 to 2009, he was a Postdoctoral Researcher and later a Research Associate with Stanford Artificial Intelligence Laboratory. From 2007 to 2008, he worked part-time with the Hansen Medical Inc., Mountain View, CA, USA, a medical robotics company. Since 2009, he has been a Professor with the Graduate School of Convergence Science and Technology, Seoul National University. His research interests lie in the areas of robot-environment interaction, contact force control, robust haptic teleoperation, multicontact control, whole-body dynamic control, biomechanics, and medical robotics.