

Robust Locomotion Control of a Self-Balancing and Underactuated Bipedal Exoskeleton: Task Prioritization and Feedback Control

Ahmed Fahmy Soliman¹, and Barkan Ugurlu¹

Abstract—This paper presents a study to understand how an underactuated bipedal exoskeleton with an arbitrary human user can exhibit robust dynamic walking behavior despite severe parameter uncertainty and external disturbances. Unlike in the case of classical bipedal robots where each leg possesses 6 or more active joints, it is very challenging to satisfy multiple constraints simultaneously to ensure robust dynamic walking. To overcome this problem, we first propose an optimization algorithm that makes use of a prioritized stack of tasks to satisfy the constraints hierarchically. Furthermore, we synthesized a locomotion controller named ZMP impedance feedback and identified two other state-of-the-art locomotion controllers (admittance control, centroidal momentum control) all of which were built on top of the proposed task prioritization algorithm. In order to verify the validity and robustness of these controllers for a thorough benchmarking, a series of simulation experiments were conducted via MSC ADAMS in which a human-robot coupled model with a 40 kg underactuated exoskeleton and 12 distinct anthropomorphic subjects (66 ~ 102 kg) was considered. As the result, all 3 controllers showed adequate performances to address balanced locomotion behavior when used with the proposed task priority-based optimization algorithm. In addition, the proposed ZMP impedance controller showed statistically significant results, indicating its comparatively more robust feature.

Index Terms—Prosthetics and Exoskeletons, Humanoid and Bipedal Locomotion, Legged Robots.

I. INTRODUCTION

THE advent of wearable technology led to the development of various lower-body exoskeletons to provide robot-aided gait assistance. To contain the weight and energy consumption, a great majority of devices have only 2 active joints along the hip and knee F/E (Flexion/Extension) joints, thus, the user's upper body must be actively engaged via crutches to maintain balance [1]–[3]. With the emergence of self-balancing exoskeletons, researchers aimed to increase the number of active joints and achieved quasi-static walking [4]–[6], active disturbance rejection in the sagittal plane [7], and

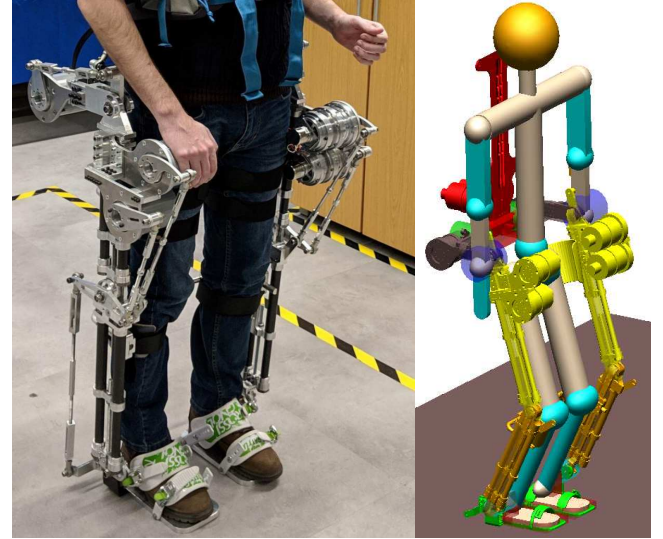


Fig. 1: Left: Self-balancing lower body exoskeleton Co-Ex [10]. Right: Its MSC ADAMS simulation model.

eventually 3-D dynamic walking [8]. In particular, the work presented in [8] makes use of a fully-actuated exoskeleton (6 active joints per leg) and with this *wearable humanoid*, hands-free dynamic walking was demonstrated.

Obviously, the more we increase the number of active joints the greater the motion flexibility is. That being said, we argue that the full actuation may not be a necessary condition to achieve self-balancing walking for exoskeletons. Indeed, in our previous work, we presented a purely kinematics-based approach in which an underactuated biped with 4 active joints along hip A/A (Abduction/Adduction), hip F/E, knee F/E and ankle D/PF (Dorsi/Plantar Flexion) could address 3-D dynamic walking [9]. While the reduced number of active joints enabled us to comparatively reduce the total weight and energy demand, the underactuated nature of the leg introduced extra challenges. For instance, even the kinematic mapping between Cartesian trajectories (Center of Mass, swing leg) and joint trajectories could not be handled without an optimization routine to prevent oblique foot landing.

The literature in underactuated biped robot control includes several inspiring studies; yet, a great majority of them focus on planar systems [11] or 3-D bipeds with point feet [12], [13], rendering their applicability to exoskeletons challenging. Meanwhile, the whole-body control approaches emerged and

Manuscript received: July, 12, 2020; Revised March, 29, 2021; Accepted April, 29, 2021.

This paper was recommended for publication by Editor Abderrahmane Kheddar upon evaluation of the Associate Editor and Reviewers' comments.

This work is supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK), with the project 215E138. The authors thank J. Babic and P. Sendur for their help.

The algorithms, data, and statistical analyses used in this paper can be accessed via <https://github.com/AhmedFahmySoliman/RobustExoControl>.

¹The authors are with the Biomechatronics Laboratory, Department of Mechanical Engineering, Ozyegin University, 34794 Istanbul, Turkey. fahmy.ahmed@ozu.edu.tr, barkan.ugurlu@ozyegin.edu.tr

Digital Object Identifier (DOI): see top of this page.

they exhibited enhanced balancing and walking behaviors for fully-actuated legged robots [14], [15]. Nevertheless, replicating such locomotion capabilities in underactuated systems is quite challenging. The *floating base* of a legged robot can move in 3-D space with any orientation when the robot leg has 6 active joints or more and when there are sufficient contact forces [16]. Likewise, contact force optimization is possible by manipulating the floating base in different orientations [17], [18]. In the case of underactuation, the joint space does not have enough variables to fully satisfy the task space constraints, and thus, the solution space is much more limited.

Inspired by the efficacy of task prioritization methods [19]–[23], we propose an optimization algorithm that employs a prioritized stack of tasks to satisfy the constraints hierarchically. Differing from the aforementioned methods that mostly aim for multi-objective task realization for systems with 6 or more active joints per leg, we aim to achieve dynamically equilibrated 3D walking using a bipedal exoskeleton in an underactuated joint configuration, e.g., 4 active joints per leg. Since achieving the exact predesignated trajectories is not possible due to underactuation, the algorithm makes use of a prioritized stack of tasks approach to sort CoM (Center of Mass), left and right foot task variables based on the contact phase. Considering the recent trend on the development of underactuated and self-balancing exoskeletons [6], [10], the paper chiefly contributes toward this direction.

The proposed task prioritization algorithm can be flexibly configured in Cartesian trajectory mode or centroidal angular momentum mode, allowing the implementation of various feedback controllers to enhance robustness against perturbations and parameter uncertainties. To this end, we constructed a ZMP (Zero Moment Point)-based controller using a floating base multibody dynamics approach where a fictitious spring-damper couple keeps the actual ZMP within the support polygon. Furthermore, we additionally synthesized two other state-of-the-art controllers based on admittance control [1], and centroidal momentum control [15] for thorough benchmarking. All the feedback controllers take the proposed task prioritization algorithm as the basis.

The main contributions of this work are three-fold: i) we propose a task prioritization algorithm that allows feasible trajectory planning to address 3-D dynamic walking for underactuated bipedal exoskeletons, even within the presence of parameter uncertainty and perturbations, ii) we propose a controller that employs multibody dynamics to satisfy the ZMP constraint by means of a fictitious spring-damper couple, iii) three state-of-the-art controllers were thoroughly compared in a realistic simulation environment where a human-robot coupled model with a 40 kg underactuated exoskeleton and 12 distinct anthropomorphic subjects (66 ~ 102 kg) were utilized.

The remainder of the paper is organized as follows. In section II, the human-robot coupled biped model is succinctly described. In section III, the proposed locomotion control methodology is explained. Simulation experiments are presented in IV and the paper is concluded in section V.

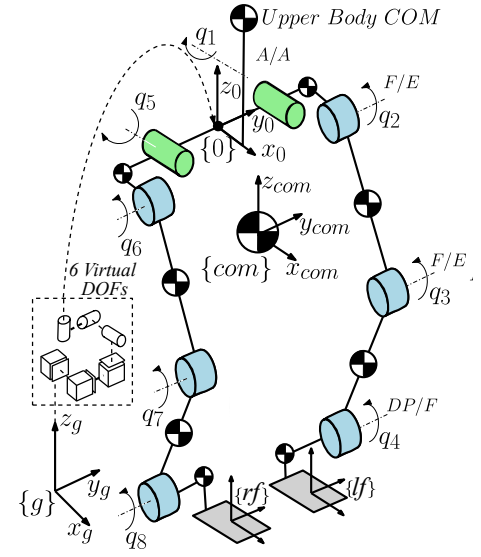


Fig. 2: Joint configuration of the bipedal exoskeleton Co-Ex and the definition of key frames.

II. HUMAN-ROBOT COUPLED MODEL

The human-robot coupled model includes a bipedal exoskeleton model and a human model which are mechanically connected via cuffs at the waist, lower leg, and feet. The underactuated biped model was constructed based on the actual lower-body exoskeleton Co-Ex that was built *in-house* [10]. Its joint configuration is illustrated in Fig. 2. Each leg possesses 4 DoFs, namely, a 2 DoF hip joint along the A/A and F/E axes, a 1 DoF knee joint along the F/E axis, and a 1 DoF ankle joint along the DP/F axis.

The human model consists of 26 DoFs (8 DoF leg, 5 DoF arm) and it is considered to be passive as the target population is spinal cord injury patients. To prevent unexpected excitations, a certain amount of rotational damping is introduced at human joints. For a thorough analysis, 12 anthropomorphic subjects with distinct inertial parameters were constructed using [24] and [25]. The parameters of the human subjects are unknown to the controllers. The robot link lengths were kept the same for every subject, therefore, robot and human joints were not aligned as the controllers were expected to handle the misalignment problem.

A. Floating Base Dynamics

The floating base dynamics play a key role in the proposed locomotion controller to compute the CMM (Centroidal Momentum Matrix) and multibody ZMP via contact forces. As explained, the inertial parameters of human models are unknown to the controllers. Yet, by only supplying the total mass and height of a given human subject, an offline algorithm constructs a lumped human-robot model in an approximate fashion using the predictive model proposed in [26]. In doing so, the level of uncertainty is relatively decreased; however, the user-dependent parameter uncertainty definitely persists. The dynamics of a floating-base system can be expressed as in the following [27],

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{S}^T \boldsymbol{\tau} + \mathbf{J}_c(\mathbf{q})^T \boldsymbol{\lambda}, \quad (1)$$

where, $\mathbf{H}(\mathbf{q}) \in \mathbb{R}^{(6+n) \times (6+n)}$ is the inertia matrix, n is the number of actuated joints, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{(6+n) \times (6+n)}$ is Coriolis and centrifugal matrix, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^{(6+n) \times 1}$ stands for the gravitational terms, $\mathbf{S}$ is the selection matrix for actuated joints ($\mathbf{S} = [\mathbf{0}_{n \times 6} \ \mathbf{I}_{n \times n}]$), $\mathbf{J}_c(\mathbf{q}) \in \mathbb{R}^{6k \times (6+n)}$ is the Jacobian matrix concerning the contact frame, k is the total number of contacts, and $\boldsymbol{\lambda} \in \mathbb{R}^{6k \times 1}$ is the vector that stores spatial forces at contacts. For multiple contacts, $\boldsymbol{\lambda}$ is expressed as $\boldsymbol{\lambda} = [\boldsymbol{\lambda}_1^T \ \boldsymbol{\lambda}_2^T \ \dots \ \boldsymbol{\lambda}_k^T]^T$. Each contact force component is defined as $\lambda_{c,i} \in \{m_{c,x}, m_{c,y}, m_{c,z}, \lambda_{c,x}, \lambda_{c,y}, \lambda_{c,z}\}$ with three rotational and translational components along x , y , and z axes, respectively. The subscript c is the force index. Matrix $\mathbf{H}(\mathbf{q})$ in (1) can be decomposed into four sub-matrices [28]:

$$\mathbf{H}(\mathbf{q}) = \begin{bmatrix} \mathbf{I}_{fb} & \mathbf{F} \\ \mathbf{F}^T & \mathbf{H}_r \end{bmatrix} \quad (2)$$

where $\mathbf{I}_{fb} \in \mathbb{R}^{6 \times 6}$ represents the aggregated links' spatial inertia projected onto the floating-base, $\mathbf{F} \in \mathbb{R}^{6 \times n}$ is a matrix whose columns store the spatial forces required at the floating-base to provide unit acceleration for the each joint variable, and $\mathbf{H}_r \in \mathbb{R}^{n \times n}$ is the joint space inertia matrix. The inertia matrix $\mathbf{H}(\mathbf{q})$ can be obtained using the composite rigid body algorithm (CRBA), $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is obtained analytically using a modified recursive Newton-Euler (NE) method¹, and $\mathbf{G}(\mathbf{q})$ is computed using the recursive NE algorithm [28]. The contact Jacobian matrix $\mathbf{J}_c(\mathbf{q})$ can be described as follows:

$$\mathbf{J}_c(\mathbf{q}) = \begin{bmatrix} {}^0\mathbf{X}_c^* \\ \mathbf{J}_{c,r}^T \end{bmatrix}^T \quad (3)$$

In (3), $\mathbf{J}_{c,r}$ is the Jacobian matrix defined for the contact frame using actuated joint variables. ${}^0\mathbf{X}_c^*$ represents the spatial transformation matrix in force space defined from the contact frame to the base frame $\{0\}$. In case of multiple contact points ${}^0\mathbf{X}_c^*$ can be expressed as $({}^0\mathbf{X}_c^* = [{}^0\mathbf{X}_1^* \ {}^0\mathbf{X}_2^* \ \dots \ {}^0\mathbf{X}_k^*])$. The floating base portion of the robot dynamics can be extracted if both sides of (1) is multiplied by a complementary matrix $\bar{\mathbf{S}} = [\mathbf{I}_{6 \times 6} \ \mathbf{0}_{6 \times n}]$:

$$[\mathbf{I}_{fb} \ \mathbf{F}]\ddot{\mathbf{q}} + \mathbf{C}_b(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_b(\mathbf{q}) = {}^0\mathbf{X}_c^* \boldsymbol{\lambda} \quad (4)$$

In (4), $\mathbf{C}_b(\mathbf{q}, \dot{\mathbf{q}}) = \bar{\mathbf{S}}\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{G}_b(\mathbf{q}) = \bar{\mathbf{S}}\mathbf{G}(\mathbf{q})$.

1) Contact Force Computation: Using (4), the joint variables can be linearly mapped to the contact force, allowing us to compute the contact forces that are necessary to estimate the ZMP via multibody dynamics:

$$\lambda_{c,i} = \mathbf{s}_{c,i} {}^c\mathbf{X}_0^* ([\mathbf{I}_{fb} \ \mathbf{F}]\ddot{\mathbf{q}} + \mathbf{C}_b(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_b(\mathbf{q})), \quad (5)$$

where $\mathbf{s}_{c,i} \in \mathbb{R}^{1 \times 6k}$ is force component selection vector. The transformation matrix ${}^c\mathbf{X}_0^*$ depends on the contact phase. During the left single support phase ${}^c\mathbf{X}_0^* = {}^{lf}\mathbf{X}_0^*$. Likewise, ${}^c\mathbf{X}_0^* = {}^{rf}\mathbf{X}_0^*$ for the right single support phase. For the double support phase, ${}^c\mathbf{X}_0^* = [{}^0\mathbf{X}_{rf}^* \ {}^0\mathbf{X}_{lf}^*]^\dagger$, where $\dagger$ stands for a pseudo inverse.

¹<https://github.com/AhmedFahmySoliman/RobustExoControl>.

2) Centroidal Momentum Matrix (CMM): As described in [15], the centroidal momentum matrix, $\mathbf{A}_g$, constitutes a linear mapping between joint velocities and the centroidal momentum of the system,

$$\mathbf{h}_g = \mathbf{A}_g \dot{\mathbf{q}}, \quad (6)$$

where $\mathbf{h}_g = [\mathbf{k}^T \ \mathbf{l}^T]^T$ is the centroidal momentum vector comprising the angular, $\mathbf{k}$, and translational, $\mathbf{l}$, components. In this connection, the CMM is yielded by multiplying the inertia matrix of (4) with the transformation matrix between the CoM and floating-base:

$$\mathbf{A}_g = {}^{com}\mathbf{X}_0^* [\mathbf{I}_{fb} \ \mathbf{F}] \quad (7)$$

Note that the external wrench projected onto the CoM is equal to the rate of change of centroidal momentum: $\dot{\mathbf{h}}_g = [\dot{\mathbf{k}}^T \ \dot{\mathbf{l}}^T]^T$. The connection between $\dot{\mathbf{h}}_g$ and joint variables is obtained as follows.

$$\dot{\mathbf{h}}_g = \dot{\mathbf{A}}_g \dot{\mathbf{q}} + \mathbf{A}_g \ddot{\mathbf{q}}. \quad (8)$$

The time derivative of CMM ($\dot{\mathbf{A}}_g$) is obtained using the following expression:

$$\dot{\mathbf{A}}_g = {}^{com}\mathbf{X}_0^* \mathbf{C}_b(\mathbf{q}, \dot{\mathbf{q}}). \quad (9)$$

III. LOCOMOTION CONTROL SYNTHESIS

The general control block diagram is displayed in Fig. 3. This figure illustrates the configuration of 3 distinct controllers, namely, the proposed controller ZIF (ZMP Impedance Feedback, red blocks), CMC (Centroidal Momentum Control, blue blocks), and BAC (Basic Admittance Control, green blocks). The common ground in applying the controller is the proposed PST (Prioritized Stack of Tasks) optimization block, which can be configured in centroidal momentum mode or CoM trajectory mode.

A. PST Optimization Block

Due to the underactuated nature of the robot, it is not possible to fully realize Cartesian task inputs. Therefore, a PST-based optimization algorithm was proposed to prioritize tasks in the form of constrained least squares QP (Quadratic Programming) [29]. This algorithm prioritizes the desired tasks in different levels. For instance, the tasks with lower priority, e.g., swing foot orientation, are generated without tampering with the tasks with higher priority, e.g., the center of mass motion. In order to realize the prioritization, a sequence of QP is solved such that the prior constraints are imposed to the lower prior constraints in the form of equality constraints.

The proposed PST optimization block can be configured in centroidal momentum mode or in CoM trajectory mode. To this end, the general form of a constrained least square optimization is given as below,

$$\boldsymbol{\chi}^* = \arg \min_{\forall \boldsymbol{\chi} \in \mathbb{R}^{2(n+6)}} \|\mathbf{C}_{ls} \boldsymbol{\chi} - \mathbf{d}_{ls}\|^2 \quad (10)$$

$$s/t \ \mathbf{A}_{eq} \boldsymbol{\chi} = \mathbf{b}_{eq} \quad (11)$$

$$\mathbf{A}_{ineq} \boldsymbol{\chi} \leq \mathbf{b}_{ineq}, \quad (12)$$

where χ is the optimization variable represents the joint velocities and accelerations $[\dot{\mathbf{q}}^T \ddot{\mathbf{q}}^T]^T$. The joint displacements is obtained per sample as $\mathbf{q}_j = \mathbf{q}_{j-1} + \dot{\mathbf{q}}_j t_s + 0.5 \ddot{\mathbf{q}}_j t_s^2$, where t_s is the sampling period, and j is the sampling index. The vector $\mathbf{d}_{ls}$ includes the command variables. Equations (11)-(12) represents the equality and inequality constraints. The matrices $\mathbf{C}_{ls}$, $\mathbf{A}_{eq}$, $\mathbf{d}_{ls}$, and $\mathbf{b}_{eq}$ are formulated as block matrices:

$$\mathbf{C}_{ls} = \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \dot{\mathbf{C}} & \mathbf{C} \end{bmatrix}, \mathbf{A}_{eq} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \dot{\mathbf{A}} & \mathbf{A} \end{bmatrix}, \mathbf{d}_{ls} = \begin{bmatrix} \mathbf{d} \\ \dot{\mathbf{d}} \end{bmatrix}, \mathbf{b}_{eq} = \begin{bmatrix} \mathbf{b} \\ \dot{\mathbf{b}} \end{bmatrix} \quad (13)$$

In (13), the submatrices, and subvectors $(\mathbf{C}, \mathbf{A}, \mathbf{d}, \mathbf{b})$ are assigned based on the configuration: i) CoM trajectory mode, ii) centroidal momentum mode.

1) *CoM Trajectory-based PST Optimization*: In this mode, CoM Jacobian ($\mathbf{J}_{com}$), torso orientation Jacobian ($\mathbf{J}_{torso}$), and feet position and orientation Jacobian ($\mathbf{J}_T$) matrices are stacked into the matrix $\mathbf{J}$ such that $\mathbf{J} = [\mathbf{J}_T^T \mathbf{J}_{torso}^T \mathbf{J}_{com}^T]^T$. In this mode, the submatrices $\mathbf{C}$ and $\mathbf{A}$ are defined as $\mathbf{C} = \mathbf{J}_p$ and $\mathbf{A} = \mathbf{J}_{p-1}$ in (13), where the subscripts $(p, p-1)$ are the lower and higher priority level indicators. Task space command variables ($^{cmd}\mathbf{p}$) can be expressed as below:

$$^{cmd}\mathbf{p} = [^{cmd}\mathbf{p}_T^T \ ^{cmd}\mathbf{p}_{torso}^T \ ^{cmd}\mathbf{p}_{com}^T]^T \quad (14)$$

in which the task space commands $^{cmd}\mathbf{p}_{com}$, $^{cmd}\mathbf{p}_{torso}$, and $^{cmd}\mathbf{p}_T$ stand for CoM position, torso orientation, foot position and orientation, respectively. The submatrices $\mathbf{d}$ and $\mathbf{b}_{eq}$ are defined as $\mathbf{d} = ^{cmd}\dot{\mathbf{p}}_p$ and $\mathbf{b}_{eq} = ^{cmd}\dot{\mathbf{p}}_{p-1}$.

2) *Centroidal Momentum-based PST Optimization*: In this mode, $\mathbf{J}_{com}$ and $\mathbf{J}_{torso}$ are replaced with the centroidal momentum matrix $\mathbf{A}_g$ defined in (7). Together with the feet position and orientation Jacobian $\mathbf{J}_T$, $\mathbf{A}_g$ is stacked in the matrix Φ such that $\Phi = [\mathbf{J}_T^T \mathbf{A}_g^T]^T$. Task space command variables ($^{cmd}\mathbf{p}$) in this mode can be expressed as in the following:

$$^{cmd}\dot{\mathbf{p}} = [^{cmd}\dot{\mathbf{p}}_T^T \ ^{cmd}\dot{\mathbf{h}}_g^T]^T \quad (15)$$

where task space commands $^{cmd}\dot{\mathbf{p}}_T$ and $^{cmd}\dot{\mathbf{h}}_g$ stand for feet spatial velocity, and centroidal momentum, respectively. Accordingly, the submatrices in (13) for this framework are defined as $\mathbf{C} = \Phi_p$, $\mathbf{A} = \Phi_{p-1}$, $\mathbf{d} = ^{cmd}\dot{\mathbf{p}}_p$, and $\mathbf{b}_{eq} = ^{cmd}\dot{\mathbf{p}}_{p-1}$.

B. Contact Force Constraints

The contact forces are imposed to the proposed PST optimization block to define feasible contact force boundaries via the linear inequality constraint (12). To extract a linear relation between contact forces components and joint variables, the decomposed floating base dynamics given in (4) is used.

1) *Unilateral Constraint*: The purpose of this constraint is to allow only repulsive forces. This condition ensures that the vertical component of the contact force meets the following condition:

$$\lambda_{c,z} \geq 0. \quad (16)$$

2) *Friction Constraint*: Coulomb friction model is used to constraint the tangential contact forces $(\lambda_{cx}, \lambda_{cy})$ relative to λ_{cz} to avoid support foot slippage. The friction pyramid is used to approximate the friction cone used to describe Coulomb friction model:

$$|\lambda_{c,x}|, |\lambda_{c,y}| \leq \mu \lambda_{c,z}. \quad (17)$$

where friction coefficient is denoted by μ .

3) *Center of Pressure Constraint*: To ensure dynamic balance, the horizontal contact moments $(m_{c,x}, m_{c,y})$ are constrained relative to λ_{cz} by considering the support polygon. The feasible region for the horizontal contact moments are defined by the inequalities below:

$$|m_{c,x}| < \frac{w}{2} \lambda_{c,z}, \quad |m_{c,y}| < \frac{L}{2} \lambda_{c,z}, \quad (18)$$

where w and L represent the lateral and longitudinal dimensions of the support polygon, respectively.

C. ZMP Impedance Feedback (ZIF) Controller

The ZMP feedback is provided by means of a virtual spring damper system, i.e., ZMP impedance; see Fig. 4. In classical ZMP feedback methods, the controller is based on simple pendulum model and it usually enforces the actual ZMP to follow the reference ZMP that is arbitrarily defined within the support polygon [30], [31]. In the proposed method, two spring-damper couples are defined between the support polygon boundaries and the measured ZMP, whose natural length resides within the polygon. In doing so, the measured ZMP is always pushed back into the support polygon by the force λ_{sx} that is defined as below:

$$\lambda_{sx} = b_x \dot{x}_{zmp} + k_x x_{zmp} \quad (19)$$

where (k_x, b_x) are spring-damper coefficients; see Fig. 4. Using (19) and a scaling factor of K_x , the force λ_{sx} can be approximately mapped to multibody x-axis ZMP which is the ratio between $m_{c,y}$ and $\lambda_{c,z}$:

$$\frac{m_{c,y}}{\lambda_{c,z}} = \frac{(b_x \dot{x}_{zmp} + k_x x_{zmp})}{K_x} \quad (20)$$

Recall that the contact moments and forces $(m_{c,y}, \lambda_{c,z})$ can be yielded via the floating dynamics eqs. (4)-(5). With this in mind, (20) can be processed to compute the base acceleration that corresponds to x-axis ZMP correction action ($\ddot{\mathbf{p}}_{zif,x}$):

$$\ddot{\mathbf{p}}_{zif,x} = -^{com}\mathbf{X}_0(\xi \mathbf{I}_{fb})^\dagger \xi (\mathbf{F}\ddot{\mathbf{q}}_r + \mathbf{C}\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}\mathbf{b}(\mathbf{q})) \quad (21)$$

$$\xi = (\mathbf{s}_{my} + \frac{(b_x \dot{x}_{zmp} + k_x x_{zmp})}{K_x} \mathbf{s}_z)^c \mathbf{X}_0^*. \quad (22)$$

where, $\ddot{\mathbf{q}}_r$ is the actuated joint acceleration vector, and the term ξ is used for simplification. A similar approach is carried out for the y-axis ZMP correction action ($\ddot{\mathbf{p}}_{zif,y}$). Using (21) and (22), the command signals are calculated as follows:

$$\begin{aligned} [^{cmd}\ddot{\mathbf{p}}_{torso}^T \ ^{cmd}\ddot{\mathbf{p}}_{com}^T] &= [^{ref}\ddot{\mathbf{p}}_{com}^T \ \mathbf{0}_{1 \times 3}] + \ddot{\mathbf{p}}_{zif,x}^T \mathbf{K}_{1,x} + \ddot{\mathbf{p}}_{zif,y}^T \mathbf{K}_{1,y} \\ [^{cmd}\ddot{\mathbf{p}}_{torso}^T \ ^{cmd}\ddot{\mathbf{p}}_{com}^T] &= [^{ref}\ddot{\mathbf{p}}_{com}^T \ \mathbf{0}_{1 \times 3}] + \ddot{\mathbf{p}}_{zif,x}^T \mathbf{K}_{2,x} + \ddot{\mathbf{p}}_{zif,y}^T \mathbf{K}_{2,y} \end{aligned} \quad (23)$$

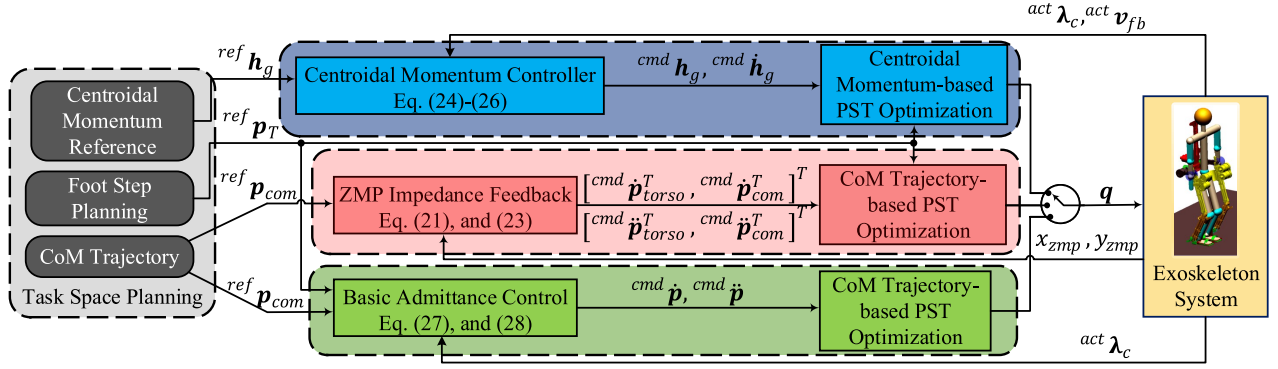


Fig. 3: Block diagram of the controller implementation. All the controllers, including our ZIF method, use the proposed PST optimization block to handle underactuation in a dynamically feasible manner. The variable $ref \mathbf{p}_T$ denotes the foot trajectory reference

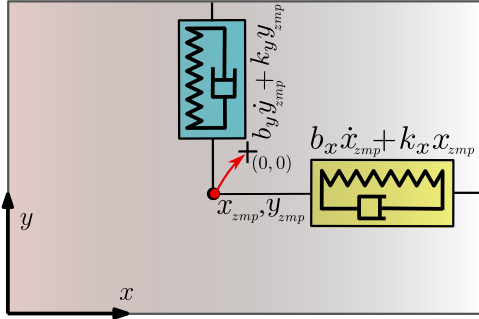


Fig. 4: ZMP impedance is described by means of a 2-DoF virtual spring-damper system.

In (23), $(\mathbf{K}_{1,x}, \mathbf{K}_{1,y}, \mathbf{K}_{2,x}, \mathbf{K}_{2,y})$ are positive definite diagonal gain matrices. To obtain $(\dot{\mathbf{p}}_{zif,x}, \dot{\mathbf{p}}_{zif,y})$, $(\ddot{\mathbf{p}}_{zif,x}, \ddot{\mathbf{p}}_{zif,y})$ are integrated numerically. Designing the command signal via (21)-(23), the ZMP correction signal could be mapped to the CoM position and torso orientation by exploiting the multibody dynamics of the system.

D. Centroidal Momentum Control (CMC)

To implement centroidal momentum feedback [15], the command signal to the PST optimization block expressed in (15) is designed as below:

$$\begin{aligned} cmd \mathbf{h}_g &= ref \mathbf{h}_g + \mathbf{M}_{p,1} error \mathbf{h}_g + \mathbf{M}_d error \dot{\mathbf{h}}_g \\ cmd \dot{\mathbf{h}}_g &= ref \dot{\mathbf{h}}_g + \mathbf{M}_{p,2} error \dot{\mathbf{h}}_g \end{aligned} \quad (24)$$

where $(\mathbf{M}_{p,1}, \mathbf{M}_{p,2}, \mathbf{M}_d)$ are positive semidefinite diagonal gain matrices. The reference and error values of momentum signals are symbolized with $ref \mathbf{h}_g$ and $error \mathbf{h}_g$. The centroidal momentum error is obtained by projecting the spatial floating base velocity error onto the CoM.

$$error \mathbf{h}_g = {}^{com} \mathbf{X}_0^* \mathbf{I}_{fb} (ref \mathbf{v}_{fb} - act \mathbf{v}_{fb}). \quad (25)$$

To avoid numeric differentiation for $error \dot{\mathbf{h}}_g$, contact force error is projected onto the CoM, in which $ref \lambda_{com}$ is extracted from the desired CoM trajectory.

$$error \dot{\mathbf{h}}_g = ref \lambda_{com} - {}^{com} \mathbf{X}_c^* act \lambda_c. \quad (26)$$

E. Basic Admittance Controller (BAC)

In this control method [1], contact force references are computed using the target CoM trajectory. Contact force errors are obtained as follows: $\lambda_{error} = ref \lambda_c - act \lambda_c$. Following this step, an admittance block is employed to generate the Cartesian velocity value that correspond to force error:

$$\dot{\mathbf{p}}_{AC} = \mathcal{L}^{-1} \left(\mathbf{G} \left(\frac{s}{sb_i + k_i} \right) \lambda_{error} \right), \quad (27)$$

In (27), $\mathcal{L}^{-1}(\bullet)$ is inverse of Laplace operator, and $\mathbf{G}(\bullet) \in \mathbb{R}^{6k \times 6k}$ is a diagonal matrix containing admittance blocks for each axis. The parameters (b_i, k_i) are the damping and stiffness coefficients concerning the i^{th} axis; $i = x, y, z$ for translational force components, and $i = mx, my, mz$ for roll, pitch, and yaw moment components. Yaw axis moment is omitted. The eq. (27) should be considered for each contact point. The variable s represents the Laplace operator. The command signal in (14) can be configured for this controller as in the following:

$$\begin{aligned} cmd \dot{\mathbf{p}} &= ref \dot{\mathbf{p}} + \dot{\mathbf{p}}_{AC} \\ cmd \ddot{\mathbf{p}} &= ref \ddot{\mathbf{p}} + \mathbf{K}_p \lambda_{error} \end{aligned} \quad (28)$$

where $(ref \dot{\mathbf{p}}, ref \ddot{\mathbf{p}})$ denote reference velocity and acceleration in Cartesian space. $\mathbf{K}_p$ is a diagonal matrix that stores proportional force feedback gains.

IV. SIMULATION EXPERIMENTS

In order to check the feasibility of the proposed locomotion control framework, a series of dynamic walking simulations were conducted using the 3D dynamic simulator MSC ADAMS. These simulations were executed using a PC that possesses an Intel (R) Xeon (R) W-2155 processor (3.3 GHz) with a 32.0 GB RAM, running on a 64-bit operating system. The 40 kg robot model was coupled with 12 distinct anthropomorphic user models (66 ~ 102 kg and 170 ~ 190 cm); see section II.

To present a comparative view and to examine individual contributions of the identified controllers, 3 controllers were employed: i) Basic Admittance Control (BAC) that embodies contact force feedback [1], ii) Centroidal Momentum Control (CMC) [15], iii) the proposed ZMP Impedance Feedback

control (ZIF). The controller gains were heuristically tuned in accordance with the non-excitation of unmodeled dynamics and they are tabulated in Table I for CMC, ZIF, and BAC, respectively.

Furthermore, two different scenarios were simulated to investigate the robustness characteristics of the controllers: 1) Dynamic walking (SC#1). 2) Perturbed dynamic walking (SC#2). In both simulation scenarios, the walking parameters were kept the same; the single and double support periods were 0.4 and 0.2 s, the maximum foot clearance was 4 cm. The forward velocity was 0.3 m/s for SC#1 and 0.2 m/s for SC#2, and thus, the stride lengths were adjusted accordingly: 38.5 cm and 25.7 cm, respectively. The controllers' performances were evaluated based on ZMP variation in x and y directions (x_{zmp}, y_{zmp}) and pitch axis torso angular velocity (ω_y). Each controller was simulated for both scenarios and repeated 12 times for each anthropomorphic subject.

	Parameter	Value
CMC	$\mathbf{M}_{p,1}$	$diag(0, 0, 0, 5.1, 7.2, 1) 10^{-2}$
	$\mathbf{M}_{p,2}$	$diag(8, 6, 0, 30, 6, 10) 10^{-3}$
	$\mathbf{M}_d$	$diag(0, 0, 0, 0.17, 2, 1)$
ZIF	$[K_x, k_x, K_y, k_y]$	$(1, 8, 7, 4) N/m$
	$[b_x, b_y]$	$(0.1, 0.02) N.s/m$
	$[\mathbf{K}_{1,x}, \mathbf{K}_{1,y}]$	$(0.51_{6 \times 6}, 0.221_{6 \times 6})$
	$[\mathbf{K}_{2,x}, \mathbf{K}_{2,y}]$	$(0.451_{6 \times 6}, 0.21_{6 \times 6})$
BAC	$[k_x, k_y, k_z, k_{mx}, k_{my}]$	$(3, 50, 70, 1500, 22000) N/m$
	$[b_x, b_y, b_z, b_{mx}, b_{my}]$	$(2, 70, 180, 2000, 26000) N.s/m$
	$\mathbf{K}_p$	$diag(17, 10, 7, 5, 0.25) 10^{-5}$

TABLE I: Gain values for each controller

A. Scenario 1 (SC#1): Dynamic Walking

The results concerning this scenario are displayed in Fig. 5. Pink, green, and blue regions indicate the corresponding variations for ZIF, BAC, and CMC, respectively. Processing the data collected from 12 anthropomorphic subjects, mean $\pm$ SD regions concerning $x_{zmp}, y_{zmp}, \omega_y$ variations were displayed. Mean variations were denoted via solid lines.

As the result, all controllers provided a dynamically balanced locomotion behavior. Therefore, we argue that they are feasible options when coupled with the proposed PST optimization block for the exoskeleton-aided walking support task. Among these controllers, the ZMP variations were comparatively well limited for the case of ZIF. In particular, peak-to-peak mean ZMP variations were observed to be farther from the support polygon boundaries generally. When it comes to the suppression of torso angle, the CMC was outperformed by the other controllers. This may imply that the CMC is more sensitive to parameter uncertainties.

B. Scenario 2 (SC#2): Perturbed Dynamic Walking

The results concerning this scenario are displayed in Fig. 6. In this scenario, an external impulsive force of 200 N was applied at the hip of the system, first along the x

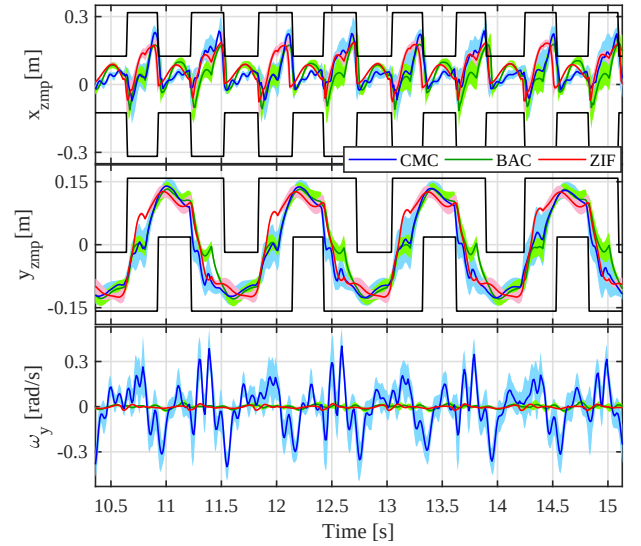


Fig. 5: x -axis and y -axis ZMP variations with respect to the local frame, and pitch axis torso angular velocity for the case of disturbance-free walking, scenario 1.

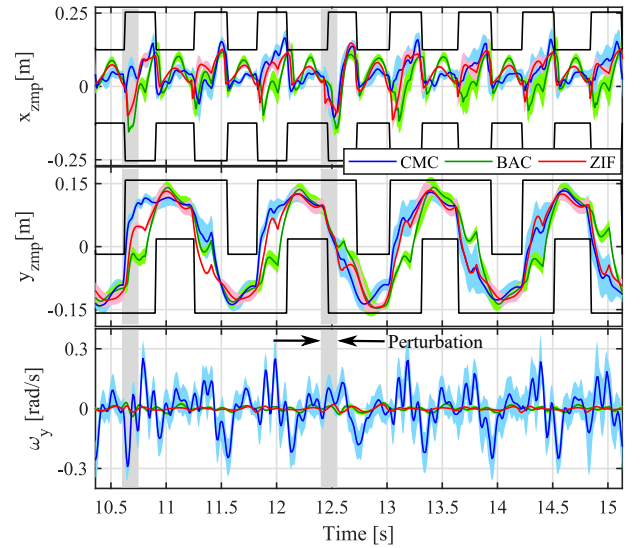


Fig. 6: x -axis and y -axis ZMP variations with respect to the local frame, and pitch axis torso angular velocity for the case of perturbed walking, scenario 2. The gray regions represent the perturbation instants.

direction when $t = 10.6$ s, then along the y direction when $t = 12.4$ s. These periods were indicated via gray shaded regions in the figure. Similarly, pink, green, and blue regions indicate the corresponding variations for ZIF, BAC, and CMC, respectively. Processing the data collected from 12 anthropomorphic subjects, mean $\pm$ SD regions concerning $x_{zmp}, y_{zmp}, \omega_y$ variations were displayed. Mean variations were denoted via solid lines.

We observe that all 3 controllers enabled the human-robot coupled system to exhibit dynamically balanced walking behavior, despite applied perturbations. The measured ZMP variations were always within the support polygon. Differing

Scenario	Controllers		$\Delta\mu = \bar{C}_I - \bar{C}_{II}$	σ_Δ^2	$pValue$	$tValue$ $t(11)$	95% CI	
	C_I	C_{II}					Lower bound	Upper bound
SC#1	BAC	CMC	-0.2782	0.0015	$p < 0.001$	30.0829	-0.3043	-0.2521
	BAC	ZIF	0.0183	0.0005	$p = 0.006$	3.4383	0.0033	0.0333
	CMC	ZIF	0.2965	0.0017	$p < 0.001$	30.6207	0.2692	0.3238
SC#2	BAC	CMC	-0.2016	0.0021	$p < 0.001$	18.6009	-0.2321	-0.1710
	BAC	ZIF	0.0208	0.0002	$p < 0.001$	6.8967	0.0123	0.0293
	CMC	ZIF	0.2223	0.0018	$p < 0.001$	22.3806	0.1943	0.2504

TABLE II: Statistical significance among the controller performances.

from the results of SC#1, the standard deviation regions are comparatively enlarged for BAC and CMC controllers, particularly after the perturbations were applied. Qualitatively speaking, the proposed controller ZIF outperformed the other controllers in terms of ZMP variations. Moreover, pitch axis torso angular velocity was not contained well when we used CMC, suggesting a similar result we obtained in the case of SC#1.

C. Statistical Analysis

To determine the most favorable controller option, we statistically analyzed the results presented in previous subsections. To define metrics, we measured the peak-to-peak values of x-axis ZMP, y-axis ZMP, and pitch axis torso angular velocity for each step. Taking the average over these peak values makes a representative metric for the analysis. We argue that the peak value of ZMP in each step is quite definitive regarding the balance condition and it may vary in each step. Therefore, an average of such peak values gives us a clearer view of the performance assessment. The aforementioned metrics are denoted as x_{zmp}^{pp} , y_{zmp}^{pp} , and ω_y^{pp} concerning x-axis ZMP, y-axis ZMP, and pitch axis torso angular velocity, respectively.

Analysis of variance was performed using Factorial Repeated Measures ANOVA to test the significant difference between controllers concerning both walking scenarios. We used the MATLAB function *ranova*(●), at the significance level of 0.05. There was a significant difference between controllers in SC#1, $F(2,22) = 798.1819, p < 0.001$, and SC#2, $F(2,22) = 401.8652, p < 0.001$.

Afterward, a post-hoc test was conducted using Bonferroni with adjusted significance level of 0.0167 per test (0.05/3). MATLAB function *multcompare*(●) was used and the results were displayed in Table II. The analysis revealed a significant reduction in average peak-to-peak values when ZIF ($M = 0.2113, SD = 0.0152$) was used, compared to BAC ($M = 0.2295, SD = 0.0306$), and CMC ($M = 0.5078, SD = 0.0584$) for the case of SC#1. When it comes to SC#2, the peak-to-peak averages are significantly less when ZIF ($M = 0.1855, SD = 0.0179$) was used, compared to BAC ($M = 0.2062, SD = 0.0159$), and CMC ($M = 0.4078, SD = 0.0635$). The $tValue$ and $pValue$ parameters for each pairwise comparison are reported in Table II, columns 6 and 7.

Columns 2 and 3 indicate the first and second controllers being compared, which are represented by C_I and C_{II} respectively. The fourth and fifth columns show the mean difference,

$\Delta\mu$, and mean squared error, σ_Δ^2 , of the two controllers in comparison. The last two columns display the 95% confidence interval (CI) of the two controllers' mean difference.

The results indicate that the ZIF controller is statistically more favorable when compared to CMC and BAC, concerning both scenarios. Furthermore, Fig. 7 illustrates the variation between the three controllers based on the averaged values of the three metrics, namely, $x_{zmp}^{pp}, y_{zmp}^{pp}, \omega_y^{pp}$ both for SC#1 and SC#2. Similarly, the controller ZIF provided less variation when compared to CMC and BAC, indicating its robustness against parameter uncertainties and perturbations. Necessary data and scripts can be obtained from <https://codeocean.com/capsule/2800487/>.

V. CONCLUDING REMARKS

In this paper, we presented a locomotion control methodology concerning the robot-aided walking task for an underactuated bipedal exoskeleton with an arbitrary human user. Differing from the case of classical bipeds where each robot leg has at least 6 active joints, it is quite difficult to satisfy several constraints to ensure dynamic walking. To remedy this issue, we proposed two contributions: i) an optimization algorithm that employs a prioritized stack of tasks to satisfy these multiple constraints hierarchically to handle underactuation, 2) a feedback controller, named ZMP Impedance Feedback, which utilizes multibody dynamics to satisfy ZMP constraints by means of a virtual spring-damper couple. The proposed optimization algorithm allows the synthesis of other feedback controllers, e.g., admittance control or centroidal momentum control.

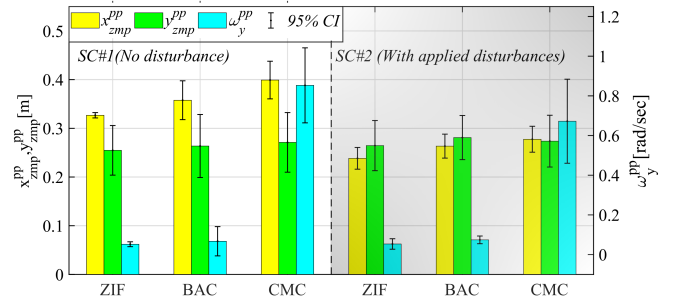


Fig. 7: Values of x_{zmp}^{pp} , y_{zmp}^{pp} , and ω_y^{pp} for 12 anthropomorphic subjects.

The proposed optimization algorithm was thoroughly tested with 3 distinct feedback controllers in a realistic simulation environment in which a 40 kg underactuated exoskeleton model and 12 anthropomorphic (66 ~ 102 kg) subjects were utilized. As the result, the human-robot combined systems exhibited dynamically balanced walking behavior robustly, despite parameter uncertainties and applied perturbations. This result adequately confirmed the validity of the proposed optimization algorithm in handling the underactuation problem. Among the feedback controllers, the proposed ZMP Impedance controller outperformed other controllers with statistical significance.

One of the major drawbacks of this method is related to its computational complexity. In particular, the inequality constraints provided in (12) cripples the real-time applicability of this work. Therefore, we are working toward the computationally efficient implementation of the proposed optimization algorithm such that it can be implemented to our underactuated lower-body exoskeleton Co-Ex, built *in-house* [10].

REFERENCES

- [1] B. Ugurlu, H. Oshima, E. Sariyildiz, T. Narikiyo, and J. Babic, "Active compliance control reduces upper body effort in exoskeleton-supported walking," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 2, pp. 144–153, Apr. 2020.
- [2] R. J. Farris, H. A. Quintero, and M. Goldfarb, "Preliminary evaluation of a powered lower limb orthosis to aid walking in paraplegic individuals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 6, pp. 652–659, Dec. 2011.
- [3] A. J. Young and D. P. Ferris, "State of the art and future directions for lower limb robotic exoskeletons," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 2, pp. 171–182, Feb. 2017.
- [4] S. Wang, L. Wang, C. Meijneke, E. van Asseldonk, T. Hoellinger, G. Cheron, Y. Ivanenko, V. L. Scaleia, F. Sylos-Labini, M. Molinari, F. Tamburella, I. Pisotta, F. Thorsteinsson, M. Ilzkovitz, J. Gancet, Y. Nevatia, R. Hauffe, F. Zanow, and H. van der Kooij, "Design and control of the MINDWALKER exoskeleton," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 23, no. 2, pp. 277–286, Mar. 2015.
- [5] G. Barbareschi, R. Richards, M. Thornton, T. Carlson, and C. Holway, "Statically vs dynamically balanced gait: Analysis of a robotic exoskeleton compared with a human," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, Aug. 2015, pp. 6728–6731.
- [6] C. Meijneke, G. van Oort, V. Sluiter, E. van Asseldonk, N. L. Tagliamonte, F. Tamburella, I. Pisotta, M. Masciullo, M. Arquilla, M. Molinari, A. R. Wu, F. Dzeladini, A. J. Ijspeert, and H. van der Kooij, "Symbitron exoskeleton: Design, control, and evaluation of a modular exoskeleton for incomplete and complete spinal cord injured individuals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, pp. 1–1, 2021.
- [7] B. Ugurlu, C. Doppmann, M. Hamaya, P. Forni, T. Teramae, T. Noda, and J. Morimoto, "Variable ankle stiffness improves balance control: Experiments on a bipedal exoskeleton," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 1, pp. 79–87, Feb. 2016.
- [8] O. Harib, A. Hereid, A. Agrawal, T. Gurriet, S. Finet, G. Boeris, A. Duburcq, M. E. Mungai, M. Masselin, A. D. Ames, K. Sreenath, and J. W. Grizzle, "Feedback control of an exoskeleton for paraplegics: Toward robustly stable, hands-free dynamic walking," *IEEE Control Systems Magazine*, vol. 38, no. 6, pp. 61–87, Dec. 2018.
- [9] A. F. Soliman, P. Sendur, and B. Ugurlu, "3-d dynamic walking trajectory generation for a bipedal exoskeleton with underactuated legs: A proof of concept," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*. IEEE, Jun. 2019, pp. 599–604.
- [10] M. C. Yildirim, A. T. Kansizoglu, S. Emre, M. Derman, S. Coruk, A. F. Soliman, P. Sendur, and B. Ugurlu, "Co-ex: A torque-controllable lower body exoskeleton for dependable human-robot co-existence," in *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*. IEEE, Jun. 2019, pp. 605–610.
- [11] M. Nikkiah, H. Ashrafiuon, and F. Fahimi, "Robust control of underactuated bipeds using sliding modes," *Robotica*, vol. 25, no. 3, pp. 367–374, May 2007.
- [12] C. Chevallereau, J. Grizzle, and C.-L. Shih, "Asymptotically stable walking of a five-link underactuated 3-d bipedal robot," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 37–50, Feb. 2009.
- [13] K. A. Hamed and J. W. Grizzle, "Event-based stabilization of periodic orbits for underactuated 3-d bipedal robots with left-right symmetry," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 365–381, Apr. 2014.
- [14] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *2014 IEEE-RAS International Conference on Humanoid Robots*. IEEE, Nov. 2014, pp. 295–302.
- [15] P. M. Wensing and D. E. Orin, "Improved computation of the humanoid centroidal dynamics and application for whole-body control," *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1550039, Mar. 2016.
- [16] M. Mistry, J. Nakanishi, G. Cheng, and S. Schaal, "Inverse kinematics with floating base and constraints for full body humanoid robot control," in *Humanoids 2008 - 8th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2008, pp. 22–27.
- [17] P. M. Wensing and D. E. Orin, "Generation of dynamic humanoid behaviors through task-space control with conic optimization," in *2013 IEEE International Conference on Robotics and Automation*. IEEE, May 2013, pp. 3103–3109.
- [18] L. Righetti, J. Buchli, M. Mistry, and S. Schaal, "Control of legged robots with optimal distribution of contact forces," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*. IEEE, Oct. 2011, pp. 318–324.
- [19] B. Siciliano and J.-J. Slotine, "A general framework for managing multiple tasks in highly redundant robotic systems," in *Fifth International Conference on Advanced Robotics Robots in Unstructured Environments*. IEEE, 1991.
- [20] M. Mistry, J. Nakanishi, and S. Schaal, "Task space control with prioritization for balance and locomotion," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2007, pp. 331–338.
- [21] A. Herzog, N. Rotella, S. Mason, F. Grimmering, S. Schaal, and L. Righetti, "Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid," *Autonomous Robots*, vol. 40, no. 3, pp. 473–491, Aug. 2015.
- [22] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1006–1028, May 2014.
- [23] B. Henze, A. Dietrich, and C. Ott, "An approach to combine balancing with hierarchical whole-body control for legged humanoid robots," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 700–707, Jul. 2016.
- [24] W.-H. Ho, T.-Y. Shiang, C.-C. Lee, and S.-Y. Cheng, "Body segment parameters of young chinese men determined with magnetic resonance imaging," *Medicine & Science in Sports & Exercise*, vol. 45, no. 9, pp. 1759–1766, Sep. 2013.
- [25] P. de Leva, "Adjustments to zatsiorsky-seluyanov's segment inertia parameters," *Journal of Biomechanics*, vol. 29, no. 9, pp. 1223–1230, Sep. 1996.
- [26] R. Dumas, L. Chêze, and J.-P. Verriest, "Adjustments to McConville et al. and young et al. body segment inertial parameters," *Journal of Biomechanics*, vol. 40, no. 3, pp. 543–553, Jan. 2007.
- [27] M. Mistry, J. Buchli, and S. Schaal, "Inverse dynamics control of floating base systems using orthogonal decomposition," in *2010 IEEE International Conference on Robotics and Automation*. IEEE, May 2010, pp. 2406–2412.
- [28] R. Featherstone, *Rigid Body Dynamics Algorithms*. Springer US, 2008.
- [29] S. Boyd and L. Vandenberghe, *Convex Optimization*. USA: Cambridge University Press, 2004.
- [30] N. Nazir, H. Izu, S. Nakaura, and M. Sampei, "An analysis of zmp control problem of humanoid robot with compliances in sole of the foot," *IFAC Proceedings Volumes*, vol. 38, no. 1, pp. 337–342, 2005.
- [31] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*. IEEE, 2003.