Whole-Body Control and Estimation of Humanoid Robots with Link Flexibility

Giulio Romualdi, Nahuel Alejandro Villa, Stefano Dafarra, Daniele Pucci, Olivier Stasse

Whole-Body Control and Estimation of Humanoid Robots with Link Flexibility

Giulio Romualdi[1], Nahuel A. Villa[2], Stefano Dafarra[1], Daniele Pucci[1,3] and Olivier Stasse[2,4]

***Abstract*— This article presents a whole-body controller for humanoid robots affected by concentrated link flexibility. We characterize the link flexibility by introducing passive joints at the concentration of deflections, which separate the flexible links into two or more rigid bodies. In this way, we extend our robot model to take link deflections into account as underactuated extra degrees of freedom, allowing us to design a whole-body controller capable to anticipate deformations. Since in a real scenario, the deflection is not directly measurable, we present an observer aiming at estimating the flexible joint state, namely position, velocity, and torque, only considering the measured contact force and the state of actuated joint. We validate the overall approach in simulations with the humanoid robot TALOS, whose hip is mechanically flexible due to a localized mechanical weakness. Furthermore, the paper compares the proposed whole-body control strategy with state-of-the-art approaches. Finally, we analyze the performance of the estimator in the case of different values of hip elasticity.**

I. Introduction

The standard state-of-the-art controllers for bipedal locomotion are designed on several layers [1], [2], [3], with the higher layers, closer to the user, producing instructions for the lower layers which work closer to the robot. In the lowest level, the motor drivers work at high frequency to guarantee either certain torque or acceleration on the robot joints, according to the whole-body controller instructions. In turn, the whole-body controller uses a model of the whole-body dynamics to compute the needed joint actuation for the robot to follow higher level instructions while keeping feasible contact forces on the environment.

The whole-body control layer typically assumes that all robot links are rigid bodies, which models correctly the whole-body motion in most cases [4], [5], [6], but becomes problematic with even small link deflections. This problem is common in heavy robot with mechanical weakness in their structure, such as the robot Talos ($\approx$ 100 kg), whose hips bend while walking. At the modeling level, several approaches have been proposed to describe multi-link flexible robots [7]. The *lumped-parameter* approach separates flexible links into two or more rigid bodies by defining fictitious passive elastic joints [8], [9], [10], [11]. Using this approach, we extend the robot model to incorporate the flexible degrees of freedom as passive joints. Still, it is important to mention that the identification of the correct passive joint placement as well as the elastic parameters is not straightforward and has been the object of several studies [10], [12]. Alternatively, the finite element method (FEM) can also be used to model the flexibility of links [13], [14]. However, while providing enhanced modeling capabilities, FEM methods are computationally greedy, which generally forbids their use in time-critical applications.

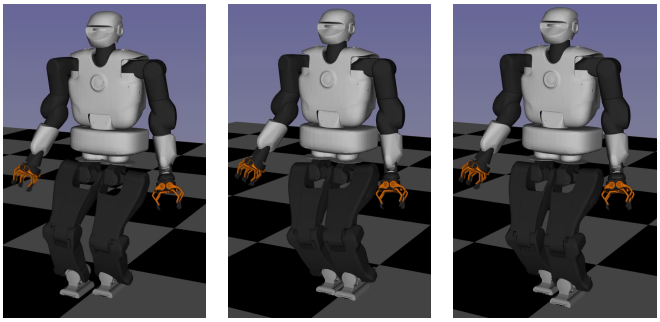


Fig. 1: TALOS walking with the proposed controller.

From the control perspective, link flexibility is often treated with passive elastic joints [9], [10], [11], [15]. To address the elasticity of the robot link, the authors of [9], [10], [11], [15] locally compensate for the effect of deflections by modifying the measured position and velocity of some actuated joints considered in the whole-body controller.

This paper contributes to the design of a control architecture for humanoid robots affected by link elasticity. More precisely, the main contributions follow. i) A whole-body controller allowing for the locomotion of humanoid robots with flexible links. Our controller incorporates a model of flexible links following an approach similar to [9], but differently from the state of the art, our design does not perform any local compensation of deflections. ii) As link deflections are not directly measurable, we propose an observer that uses only the measured contact forces and information from actuated joints to estimate the position, velocity and torques of the elastic joints. iii) We validate our approach on the simulated torque-controlled humanoid robot TALOS [16], and we show how it performs different from the standard rigid-body-based controller.

The paper is organized as follows: Sec. II introduces our notation and recalls some concepts of floating base system, Sec. III details the whole-body controller, Sec. IV discusses the design of our elastic-joint observer, Section V presents the simulation results on the TALOS humanoid robot, and finally, Sec. VI concludes the paper.

[1] Artificial and Mechanical Intelligence, Italian Institute of Technology, Genoa, Italy, (e-mail: name.surname@iit.it)

[2] Gepetto Team, LAAS-CNRS, Université de Toulouse, Toulouse, France. (e-mail: name.surname@laas.fr)

[3]Machine Learning and Optimisation, University of Manchester, Manchester, UK

[4] Artificial and Natural Intelligence Toulouse Institute, Toulouse, France

This work was supported by the European Union's Horizon 2020 through the SoftManBot project under Agreement 869855.

II. Background

A. Notation

- $I_{m\times n}$ and $0_{m\times n}$ denote the $m \times n$ identity and zero matrices. When $m = n$, we use simply I_n, 0_n.
- $\mathcal{I} = (p_\mathcal{I}, [\mathcal{I}])$ denotes the inertial frame. $p_\mathcal{I}$ is the origin and $[\mathcal{I}]$ its orientation.
- given $\mathcal{A} = (p_\mathcal{A}, [\mathcal{A}])$ and $\mathcal{B} = (p_\mathcal{B}, [\mathcal{B}])$, then $\mathcal{B}[\mathcal{A}] = (p_\mathcal{B}, [\mathcal{A}])$ has its origin in $p_\mathcal{B}$ and is oriented as $\mathcal{A}$.
- Given ${}^\mathcal{A}p_\mathcal{C} \in \mathbb{R}^3$ and ${}^\mathcal{B}p_\mathcal{C} \in \mathbb{R}^3$, representations of $p_\mathcal{C}$ in $\mathcal{A}$ and $\mathcal{B}$ respectively ${}^\mathcal{A}p_\mathcal{C} = {}^\mathcal{A}R_\mathcal{B}\,{}^\mathcal{B}p_\mathcal{C} + {}^\mathcal{A}p_\mathcal{B} = {}^\mathcal{A}H_\mathcal{B}\,[\,{}^\mathcal{B}p_\mathcal{C}^\top \;\; 1\,]^\top$. ${}^\mathcal{A}H_\mathcal{B} \in \mathrm{SE}(3)$ is the homogeneous transformation and ${}^\mathcal{A}R_\mathcal{B} \in \mathrm{SO}(3)$ is the rotation matrix.
- the *hat operator* is $\cdot^\wedge : \mathbb{R}^3 \to \mathfrak{so}(3)$, where $\mathfrak{so}(3)$ is the set of skew-symmetric matrices and $x^\wedge y = x \times y$.
- the *vee operator* is $\cdot^\vee : \mathfrak{so}(3) \to \mathbb{R}^3$.
- given a rotation matrix, *log operator* returns the associated element of $\mathfrak{so}(3)$ $\log : \mathrm{SO}(3) \to \mathfrak{so}(3)$.
- ${}^\mathcal{A}v_{\mathcal{C},\mathcal{B}} \in \mathbb{R}^3$ denotes the relative linear velocity of the frame $\mathcal{B}$ with respect to $\mathcal{C}$, expressed in $\mathcal{A}$.
- ${}^\mathcal{A}\omega_{\mathcal{C},\mathcal{B}} \in \mathbb{R}^3$ denotes the relative angular velocity of the frame $\mathcal{B}$ with respect to $\mathcal{C}$, expressed in $\mathcal{A}$.
- the velocity of a frame $\mathcal{B}$ w.r.t. the frame $\mathcal{A}$ is uniquely identified by the twist ${}^\mathcal{A}\mathrm{v}_{\mathcal{A},\mathcal{B}}^\top = [\,{}^\mathcal{A}v_{\mathcal{A},\mathcal{B}}^\top \; {}^\mathcal{A}\omega_{\mathcal{A},\mathcal{B}}^\top\,]$.
- $\times^*$ is the dual spatial cross product on $\mathbb{R}^6$ such that ${}^\mathcal{A}\mathrm{v}_{\mathcal{A},\mathcal{B}}\times^* = \begin{bmatrix} {}^\mathcal{A}\omega_{\mathcal{A},\mathcal{B}}^\wedge & 0_3 \\ {}^\mathcal{A}v_{\mathcal{A},\mathcal{B}}^\wedge & {}^\mathcal{A}\omega_{\mathcal{A},\mathcal{B}}^\wedge \end{bmatrix}$.
- we drop the suffix $\mathcal{I}$ to identify the velocity of a frame with respect to $\mathcal{I}$, i.e., ${}^\mathcal{A}v_{\mathcal{I},\mathcal{B}} = {}^\mathcal{A}v_\mathcal{B}$, ${}^\mathcal{A}\omega_{\mathcal{I},\mathcal{B}} = {}^\mathcal{A}\omega_\mathcal{B}$.

B. Humanoid Robot Model

A humanoid robot is a floating base multi-body system made up of $n+1$ links joined by n joints each with one degree of freedom. The robot configuration is fully described by the triplet $q = ({}^\mathcal{I}p_\mathcal{B}, {}^\mathcal{I}R_\mathcal{B}, s) \in \mathbb{R}^3 \times \mathrm{SO}(3) \times \mathbb{R}^n$ where $\mathcal{B}$ is the base frame and s the joint positions. The velocity of the floating system is represented by the triplet $\nu = ({}^\mathcal{I}v_\mathcal{B}, {}^\mathcal{I}\omega_\mathcal{B}, \dot{s}) \in \mathbb{R}^3 \times \mathfrak{so}(3) \times \mathbb{R}^n \cong \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^n$.

The position and orientation of any frame $\mathcal{A}$, attached to some link, is identified by ${}^\mathcal{I}H_\mathcal{A} \in \mathrm{SE}(3)$, and its twist $\mathrm{v}_\mathcal{A} = J_\mathcal{A}(q)\nu$ is linearly related to the triplet ν by the Jacobian matrix $J_\mathcal{A}(q)$.

Each link of the robot follows the rigid-body dynamics described by the *Euler-Poincaré Equations* [17]

$$\mathbb{M}_L\,{}^L\dot{\mathrm{v}}_L + {}^L\mathrm{v}_L \times^* \mathbb{M}_L\,{}^L\mathrm{v}_L = \mathbb{M}_L \begin{bmatrix} {}^\mathcal{I}R_L^\top g \\ 0_{3\times 1} \end{bmatrix} + \sum_{k=1}^{n_f} {}_L\mathrm{f}_k, \quad (1)$$

where L stands for link frame, $\mathbb{M}_L \in \mathbb{R}^{6\times 6}$ is the constant inertia tensor, ${}^L\mathrm{v}_L \in \mathbb{R}^6$ is the link spatial velocity, $\sum_{k=1}^{n_f} {}_L\mathrm{f}_k \in \mathbb{R}^6$ are the n_f wrenches exerted on the link. The *proper sensor acceleration* is given as [18, Sec. 2.4.4]

$$\alpha_L^g := {}^L X_{L[\mathcal{I}]}\,{}^{L[\mathcal{I}]}\dot{\mathrm{v}}_L - \begin{bmatrix} {}^\mathcal{I}R_L^\top g \\ 0_{3\times 1} \end{bmatrix}. \quad (2)$$

${}^L X_{L[\mathcal{I}]}$ is the adjoint matrix, α_L^g is the acceleration obtained by an inertial measurement unit aligned to the link. Combining (1) with (2), we obtain [18, Eq. (2.80)]

$$\phi_L = \sum_{k=1}^{n_f} {}_L\mathrm{f}_k, \quad (3a)$$

$$\phi_L = \mathbb{M}_L\alpha_L^g + \begin{bmatrix} 0_{3\times 1} \\ {}_L\omega_L \end{bmatrix} \times^* \mathbb{M}_L \begin{bmatrix} 0_{3\times 1} \\ {}_L\omega_L \end{bmatrix}. \quad (3b)$$

The complete robot dynamics can be described as [19]

$$M(q)\dot{\nu} + \eta(q,\nu) = B\tau + \sum_k J_{\mathcal{C}_k}^\top(q)\mathrm{f}_k. \quad (4)$$

$M(q) \in \mathbb{R}^{(n+6)\times(n+6)}$ is the mass matrix, $\eta(q,\nu) \in \mathbb{R}^{n+6}$ contains the Coriolis, centrifugal and gravitational effects. $\tau \in \mathbb{R}^n$ are the joint torques, acting only on joints as indicated by the selection matrix $B \in \mathbb{R}^{(n+6)\times n}$. We express each contact wrench $\mathrm{f}_k \in \mathbb{R}^6$ in the frame $\mathcal{C}_k[\mathcal{I}]$ located at the contact point and aligned to the inertial frame, $J_{\mathcal{C}_k}(q)$ is its associated Jacobian.

Given a frame $\bar{G} = (x_\mathrm{CoM}, [\mathcal{I}])$ placed at the robot's center of mass (CoM) and aligned to the inertial frame, and applying a given change of coordinates as in [20], the dynamics (4) can be expressed separating the first 6 rows, referring to the centroidal momentum of the robot ${}_{\bar{G}}h$ [21], from the last n rows referring to the actuated joints.

The rate of change of the centroidal momentum of the robot can be expressed by means of the external contact wrenches acting on the system [17], [18]

$${}_{\bar{G}}\dot{h} = m\bar{g} + \sum_k {}_{\bar{G}}X^{\mathcal{C}_k[\mathcal{I}]}\mathrm{f}_k, \quad (5)$$

where $m\bar{g}^\top = \begin{bmatrix} 0 & 0 & -mg & 0 & 0 & 0 \end{bmatrix} \in \mathbb{R}^6$ is the spatial robot weight and the adjoint matrix ${}_{\bar{G}}X^{\mathcal{C}_i[\mathcal{I}]} \in \mathbb{R}^{6\times 6}$ changes the coordinates of a 6D force from $\mathcal{C}_k[\mathcal{I}]$ to $\bar{G}$.

C. Model of the link flexibility

Let us consider that some robot links are affected by punctual mechanical weakness allowing the apparition of concentrated deformations. We model the extra degrees of freedom introduced by this flexible behavior by splitting each flexible body into two or more rigid links connected by passive virtual joints [11], [10]. So, the i-th virtual joint exerts a torque τ_i that depends on the deflection s_i and its velocity $\dot{s}_i$ i.e. $\gamma_i : \mathbb{R}\times\mathbb{R} \to \mathbb{R}$ as

$$\tau_i = \gamma_i(s_i, \dot{s}_i). \quad (6)$$

By numerically differentiating the position of the elastic joint s_i over a sampling period $\mathrm{d}\,t$, we discretize (6) as

$$\tau_i[k] = \gamma_i(s_i[k], s_i[k-1]), \quad (7)$$

where $k \in \mathbb{Z}$ is the discretized time.

Following [10], [11], we assume that γ_i models the elastic behavior as

$$\gamma_i(s_i, \dot{s}_i) = -k_i s_i - d_i \dot{s}_i, \quad (8)$$

where k_i and d_i are, respectively, the stiffness and damping of the i-th elastic joint. Discretizing the joint velocity with

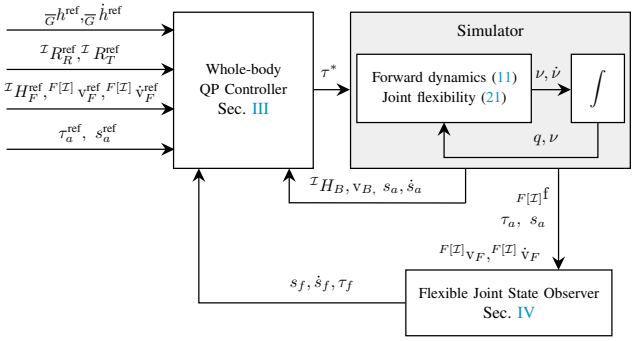


Fig. 2: Controller architecture.

the forward Euler method, we approximate the elastic joint state, namely its position and velocity, by

$$s_i[k] = \frac{d_i s_i[k-1] - \tau_i[k]\,\mathrm{d}\,t}{k_i\,\mathrm{d}\,t + d_i}, \tag{9a}$$

$$\dot{s}_i[k] = \frac{s_i[k] - s_i[k-1]}{\mathrm{d}\,t}. \tag{9b}$$

D. Modeling of a floating base system with elastic joints

Let us redefine the robot configuration as:

$$q = ({}^{\mathcal{I}}p_{\mathcal{B}}, {}^{\mathcal{I}}R_{\mathcal{B}}, s^a, s^f), \quad \nu = \left({}^{\mathcal{I}}v_{\mathcal{B}}, {}^{\mathcal{I}}\omega_{\mathcal{B}}, \dot{s}^a, \dot{s}^f\right) \tag{10}$$

with n_a actuated joints s^a and n_f elastic joints s^f, so that the robot posture has $n = n_a + n_f$ degrees of freedom, and the robot dynamics (4) becomes:

$$M_b\dot{\nu} + \eta_b = \sum_k J_{\mathcal{C}_k b}^\top \mathrm{f}_k\,, \tag{11a}$$

$$M_s\dot{\nu} + \eta_s = \begin{bmatrix} I_{n_a} \\ 0_{n_f \times n_a} \end{bmatrix}\tau^a + \begin{bmatrix} 0_{n_a \times n_f} \\ I_{n_f} \end{bmatrix}\tau^f + \sum_k J_{\mathcal{C}_k s}^\top \mathrm{f}_k. \tag{11b}$$

where we have separated the base dynamics (subscript $_b$ corresponding to the 6 first rows) from the postural dynamics (subscript $_s$ corresponding to the n final rows).

III. Whole-body Controller

The whole-body controller guarantees the tracking of desired kinematic quantities while ensuring the feasibility of the contact forces. The proposed controller computes the desired actuated joint torques τ^a, considering the robot dynamics (11) and the elastic joint model (8). The control objective is met by framing the controller as a constrained optimization problem in which low-priority tasks are embedded in the cost function, and high-priority tasks are handled as constraints. In the following equations, the superscript $^{\text{ref}}$ denotes the quantities provided by a high-level planner that assumes the robot made of rigid links only. Fig 2 presents the input and output quantities of the Whole-body controller.

A. Costs and Constraints

This section introduces the list of low and high priority tasks considered in the optimal control problem.

1) Centroidal momentum task: To ask for a desired centroidal momentum trajectory, we introduce the task as

$$\Psi_h = {}_{\bar{G}}\dot{h}^* - A_c\mathrm{f} - m\bar{g}. \tag{12}$$

where $A_c = \begin{bmatrix} {}_{\bar{G}}X^{\mathcal{C}_1[\mathcal{I}]} & \cdots & {}_{\bar{G}}X^{\mathcal{C}_i[\mathcal{I}]} & \cdots \end{bmatrix}$ gathers the adjoint matrix for each contact, and the desired value ${}_{\bar{G}}\dot{h}^*$ is chosen from a feedback law to guarantee asymptotic convergence to the desired trajectory

$$\begin{aligned} {}_{\bar{G}}\dot{h}^* = &\begin{bmatrix} m\ddot{x}_{\text{CoM}}^{\text{ref}} \\ {}_{\bar{G}}\dot{h}^{\omega^{\text{ref}}} \end{bmatrix} + \begin{bmatrix} mk_{\text{CoM}}^p \\ 0_3 \end{bmatrix}\left(x_{\text{CoM}}^{\text{ref}} - x_{\text{CoM}}\right) \\ &+ \begin{bmatrix} mk_{\text{CoM}}^d & 0_3 \\ 0_3 & k_{h_\omega} \end{bmatrix}\begin{bmatrix} \dot{x}_{\text{CoM}}^{\text{ref}} - \dot{x}_{\text{CoM}} \\ {}_{\bar{G}}h_\omega^{\text{ref}} - {}_{\bar{G}}h_\omega \end{bmatrix} \end{aligned} \tag{13}$$

with positive definite gains k_{CoM}^p, k_{CoM}^d and k_{h_ω}.

2) Cartesian task: While walking, we often require some link frames to have a specific position and orientation with respect to the inertial frame ${}^{\mathcal{I}}H_L^{\text{ref}} = ({}^{\mathcal{I}}p_L^{\text{ref}}, {}^{\mathcal{I}}R_L^{\text{ref}}) \in \mathbb{R}^3 \times \mathrm{SO}(3)$, by specifying the following task

$$\Psi_{L_{\mathrm{SE}(3)}} = \begin{bmatrix} \dot{v}_L^{*\top} & {}^{\mathcal{I}}\dot{\omega}_L^{*\top} \end{bmatrix}^\top - J_L\dot{\nu} - \dot{J}_L\nu, \tag{14}$$

where $\dot{v}_L^* = \dot{v}_L^{\text{ref}} + k_{L_p}^d(v_L^{\text{ref}} - v_L) + k_{L_p}^p(p_L^{\text{ref}} - p_L)$ and the angular acceleration is set to guarantee almost global stability and convergence of ${}^{\mathcal{I}}R_L$ to ${}^{\mathcal{I}}R_L^{\text{ref}}$ [22]:

$$\begin{aligned} {}^{\mathcal{I}}\dot{\omega}_L^* = {}&{}^{\mathcal{I}}\dot{\omega}_L^{\text{ref}} + k_{L_\omega}^d({}^{\mathcal{I}}\omega_L^{\text{ref}} - {}^{\mathcal{I}}\omega_L) \\ &+ k_{L_\omega}^p\left[\log\left({}^{\mathcal{I}}R_L^{\text{ref}}\,{}^{\mathcal{I}}R_L^\top\right)\right]^\vee. \end{aligned} \tag{15}$$

$k_{L_\omega}^p$ and $k_{L_\omega}^d$ are positive definite matrices. From (14), we introduce the rotational tasks for L, denoted as $\Psi_{L_{\mathrm{SO}(3)}}$

$$\Psi_{L_{\mathrm{SO}(3)}} = \begin{bmatrix} 0_3 & I_3 \end{bmatrix}\Psi_{L_{\mathrm{SE}(3)}}. \tag{16}$$

3) Floating base dynamics task: The whole-body control considers the base (11a) and joint dynamics (11b) as a set of high-priority tasks. We define the base dynamics task as

$$\Psi_{\text{dyn}_b} = \eta_b + M_b\dot{\nu} - \sum_k J_{\mathcal{C}_k b}^\top(q)\mathrm{f}_k\,. \tag{17}$$

If the robot has under-actuated elastic joints, the joint dynamics should consider the measured (or estimated) joint torques acting on the system as

$$\Psi_{\text{dyn}_s} = \eta_s + M_s(q)\dot{\nu} - \begin{bmatrix} I_{n_a} \\ 0_{n_f \times n_a} \end{bmatrix}\tau^a - \begin{bmatrix} 0_{n_a \times n_f} \\ I_{n_f} \end{bmatrix}\tau^f - J_{\mathcal{C}_s}^\top \mathrm{f}. \tag{18}$$

Notice that the last n_f rows of (18) represent the underactuated flexible behavior of the robot.

4) Joint position regularization task: To prevent the controller from computing solutions that generate a huge variation in joint acceleration, we introduce a joint regularization task for both the actuated and elastic joints, as

$$\Psi_{s_a} = \ddot{s}_a^* - \begin{bmatrix} 0_{n_a \times 6} & I_{n_a} & 0_{n_a \times n_f} \end{bmatrix}\dot{\nu} \tag{19a}$$

$$\Psi_{s_f} = \ddot{s}_f^* - \begin{bmatrix} 0_{n_f \times 6} & 0_{n_f \times n_a} & I_{n_f} \end{bmatrix}\dot{\nu}, \tag{19b}$$

with $\ddot{s}_a^*$ equal to

$$\ddot{s}_a^* = \ddot{s}_a^{\text{ref}} + k_{s_a}^d(\dot{s}_a^{\text{ref}} - \dot{s}_a) + k_{s_a}^p(s_a^{\text{ref}} - s_a). \tag{20}$$

Where s_a^{ref} is the desired joint trajectory. $k_{s_a}^d$ and $k_{s_a}^p$ are two positive-defined diagonal matrices. On the other hand, we ask for $\ddot{s}_f^*$ equal to

$$\ddot{s}_f^* = -k_{s_f}^d \dot{s}_f - k_{s_f}^p s_f. \tag{21}$$

$k_{s_f}^d$ and $k_{s_f}^p$ are defined positive matrices. Thanks to (21) the controller tries to stabilize the flexible joint position to zero.

5) Joint torque regularization task: To prevent the controller from providing solutions with large actuated joint torques, we introduce the following task

$$\Psi_\tau = \tau_a^{\text{ref}} - \tau_a. \tag{22}$$

6) Feasible contact force task: The feasibility of the contact 6D force ${}_{\mathcal{C}_k[\mathcal{I}]}\mathrm{f}_k$ is guaranteed by the set of inequalities

$$\Phi_{\mathrm{f}_k} = \Gamma_{\mathcal{C}_k[\mathcal{I}]}\, {}_{\mathcal{C}_k[\mathcal{I}]}\mathrm{f}_k - b, \qquad \Phi_{\mathrm{f}_k} \preceq 0, \tag{23}$$

where $\Gamma_{\mathcal{C}_k[\mathcal{I}]}$ depends on the robot state q as in [23], and the inequalities $\preceq$ hold elementwise.

B. Quadratic programming formulation

We achieve the control objective by designing a constrained quadratic programming (QP) problem of the form

$$\begin{aligned}
\underset{\dot{\nu},\, \tau_a,\, \mathrm{f}}{\text{minimize}} \; & \|\Psi_{T_{\mathrm{SO}(3)}}\|^2_{\Lambda_T} + \|\Psi_{R_{\mathrm{SO}(3)}}\|^2_{\Lambda_R} + \|\Psi_s\|^2_{\Lambda_s} + \|\Psi_\tau\|^2_{\Lambda_\tau} \\
\text{subj. to} \; & \Psi_{\circ_{\mathrm{SE}(3)}} = 0 && \circ = \{L, R\} \\
& \Psi_h = 0 \\
& \Psi_{\mathrm{dyn}_\circ} = 0 && \circ = \{s, \nu\} \\
& \Phi_{\mathrm{f}_\circ} \preceq 0 && \circ = \{L, R\}.
\end{aligned} \tag{24}$$

Here, contact forces f_j, the base acceleration ${}^{B[\mathcal{I}]}\dot{\mathrm{v}}_B$, the actuated joint acceleration $\ddot{s}_a$, and the actuated joint torques τ_a are considered as optimization variables.

In (24), the tracking of the left $\Psi_{L_{\mathrm{SE}(3)}}$ and right $\Psi_{R_{\mathrm{SE}(3)}}$ feet pose is treated as high-priority $\mathrm{SE}(3)$ tasks (14). The Centroidal momentum tracking is considered as a high-priority task (12). We also consider the base (17) and joints dynamics (18) as high-priority tasks. To prevent the controller from asking for a high motion of the upper body, we introduce two $\mathrm{SO}(3)$ tasks, one associated with the chest and the other with the waist orientations, respectively, denoted $\Psi_{T_{\mathrm{SO}(3)}}$ and $\Psi_{R_{\mathrm{SO}(3)}}$. In both cases, we ask the z coordinates of the link frames to remain parallel to the gravity vector. The postural conditions of the actuated and elastic joints (19) are considered as low priority tasks. We regularize the desired actuated joint torques as a low priority task Ψ_τ (22). Finally, to guarantee feasible contact forces for the feet, we add the task (23), denoted respectively as Φ_{f_L} and Φ_{f_R}.

IV. Elastic Joint State Observer

Solving the optimal control problem (24) requires a prior-knowledge of the position, velocity and torque of all elastic joints. These values are fully known in simulation. However, in the real scenario, they must be estimated. We present an algorithm that computes the state of elastic joints considering the contact wrenches acting on the robot feet and the actuated joint state. Let us assume:

- the link flexibility is located on the robot leg;
- the robot leg is described by a kinematic chain where the first node represents the robot base;
- all the flexible links are modeled with m_f elastic joints described by (8) and (9);
- each elastic joint is connected to its child joint with a fake link having zero mass and inertia;
- there exists m_a actuated links and joints between the latest elastic joint and the robot foot.

Let us consider a single leg; we denote the waist of the robot by L_0, all virtual links and elastic joints by L_i^f and s_i^f with $1 \leq i \leq m_f$; and all actual robot links and actuated joints by L_i^a and s_i^a with $m_f+1 \leq i \leq m_a+m_f$, respectively. We also name the frame associated with each link as the link itself. We assume that each link is subject to only two wrenches, exerted by the parent and child links. With an abuse of notation, we consider the environment as the child of the foot $L^a_{m_a+m_f}$. For the dynamics of the links (3), we denote by ϕ_i the wrenches exerted on the actual robot links and on the fictitious flexible links.

A. Forward kinematics

Given the angular velocity and the proper sensor acceleration of the foot $L^a_{m_a+m_f}$, we can compute ϕ_i with $m_f+1 \leq i < m_a+m_f$, recursively. Given the link L_i^a, its body angular velocity writes as

$${}^i\omega_i = {}^iR_{i+1}\, {}^{i+1}\omega_{i+1} + {}^i\omega_{i+1,i}, \;\; m_f+1 \leq i < m_a+m_f \tag{25}$$

where ${}^{i+1}\omega_{i+1}$ is the angular velocity of the child link. ${}^i\omega_{i+1,i}$ is the velocity of L_i^a with respect to L_{i+1}^a written in L_i^a. Similarly, we recall that $\alpha^g_{L_i^a}$ can be computed recursively by considering the child joint position s_{i+1}^a velocity $\dot{s}_{i+1}^a$ and acceleration $\ddot{s}_{i+1}^a$, and $\alpha^g_{L_{i+1}^a}$ as in [18].

We want to underline that if an IMU is mounted on the robot feet, $\phi_{m_a+m_f}$ can be derived from the sensor readouts. Otherwise, we propose two possible solutions: i) In the case of low swing foot velocity and acceleration, we suggest considering ${}^{m_a+m_f}\omega_{m_a+m_f}$ and $\alpha^g_{L^a_{m_a}}$ equal to zero. ii) Assuming that an IMU is mounted on the robot waist. We suggest setting $\alpha^g_{L^a_{m_a}}$ and ${}^{m_a+m_f}\omega_{m_a+m_f}$ at time t equal to the one predicted by the forward kinematics at the previous step $t - \mathrm{d}\,t$. To guarantee the convergence of the algorithm, the choice of $\mathrm{d}\,t$ becomes crucial. Here, we suggest setting $\mathrm{d}\,t$ small enough to capture the evolution of the system dynamics.

B. Inverse dynamics propagation

Assuming that $\phi_{L_i^a}$ has been computed as in Sec. IV-A and considering (13) with $n_f = 2$, the dynamics of L_i^a holds

$$\phi_i = {}_i\mathrm{f}_{\lambda(i),i} + {}_i\mathrm{f}_{i+1,i}, \quad m_f + 1 \leq i < m_a + m_f, \tag{26}$$

where $\lambda(i)$ gives the parent link of L_i^a, ${}_i\mathrm{f}_{\lambda(i),i}$ is the spatial force exerted by the parent link $\lambda(i)$ to L_i^a whose coordinates are expressed in L_i^a. ${}_i\mathrm{f}_{i+1,i}$ is the spatial force exerted by L_{i+1}^a to L_i^a expressed in L_i^a. We now reorganize (26) to reveal its recursive structure:

$$\begin{aligned} {}_i\mathrm{f}_{\lambda(i),i} &= \phi_i - {}_i\mathrm{f}_{i+1,i} && \text{(27a)} \\ &= \phi_i + {}_iX^{i+1}\,{}_{i+1}\mathrm{f}_{i,i+1} && \text{(27b)} \end{aligned}$$

We notice that by projecting (27) into the *joint motion axis* ${}^i\mathbf{s} \in \mathbb{R}^6$ [17], [18], the torque acting on the joint s_i^a writes as $\tau_i^a = {}^i\mathbf{s}^\top\,{}_i\mathrm{f}_{\lambda(i),i} \in \mathbb{R}$. If the system is equipped with joint torque sensors, τ_i^a can be directly measured. Using this information, we attempt to improve the estimation of the 6D force ${}_i\mathrm{f}_{\lambda(i),i}$ as follows:

$$\begin{aligned} {}_i\mathrm{f}^c_{\lambda(i),i} = &\left[(1-\beta)\tau_i^{a_\mathrm{m}} + \beta\,{}^i\mathbf{s}^\top\,{}_i\mathrm{f}_{\lambda(i),i}\right]{}^i\mathbf{s} \\ &+ \left(I_6 - {}^i\mathbf{s}\,{}^i\mathbf{s}^\top\right){}_i\mathrm{f}_{\lambda(i),i}. \end{aligned} \tag{28}$$

Here, ${}_i\mathrm{f}^c_{\lambda(i),i}$ denotes the corrected 6D force and $\tau_i^{a_\mathrm{m}}$ is the measured joint torque. We choose the parameter $\beta \in [0, 1]$ according to our trust among measurement and calculations. At its extremes, $\beta = 0$ considers only the measured torques, and $\beta = 1$ just takes the computed torques.

C. Elastic joint state estimation

Propagating the inverse dynamics in Sec. IV-B, we recursively compute the wrench acting on the joint s_1^a. Assuming a neglected mass and inertia for the flexible link $L_{m_f}^f$, i.e., $\mathbb{M}_{L_{m_f}^f} \approx 0_{6\times6}$, we write the flexible link dynamics as

$$\begin{aligned} {}_{m_f}\mathrm{f}_{m_f-1,m_f} &= -{}_{m_f}\mathrm{f}_{m_f+1,m_f} \\ &= {}_{m_f}X^{m_f+1}\,{}_{m_f+1}\mathrm{f}_{m_f,m_f+1}, \end{aligned} \tag{29}$$

where ${}_{m_f}X^{m_f+1}$ depends on the position of the joint $s^a_{m_f+1}$ obtained in Sec. IV-B.

Given (29), we can compute the elastic joint torque $\tau^f_{m_f}$ by projecting ${}_{m_f}\mathrm{f}_{m_f-1,m_f}$ into the joint motion axis ${}^{m_f}\mathbf{s}$

$$\tau^f_{m_f} = {}^{m_f}\mathbf{s}^\top\,{}_{m_f}\mathrm{f}_{m_f-1,m_f}. \tag{30}$$

Combining (30) with the discretized elastic joint position (9a) we estimate the elastic joint position $s^f_{m_f}$. Applying the same approach, we can estimate the elastic joint torque τ^f_i and position s^f_i, with $1 \leq i \leq m_f - 1$

At each time step, applying the procedure of Sec. IV-A, IV-B, Eq. (29) and (30), we estimate the elastic joint state. The result is finally considered by the whole-body controller to compute the desired actuated joint torques.

V. Results

In this section, we test the control strategy presented in Sec III on a simulated version of the TALOS humanoid robot [16] – see Fig. 1. TALOS's hip flexibility has a significant impact on its leg control and, as a result, its balance and locomotion [6]. Since the hip flexibility is due to the vertical linkage following previous work [10], we model the flexibility along the pitch and roll axis for each leg. As a result, we have $n_f = 4$ while $n_a = 32$. The architecture takes (on average) less than $1\,\mathrm{ms}$ to evaluate its output. The OSQP [24] library is used to solve the optimization problem. The simulations are obtained by integrating the robot dynamics and the joint elasticity as in Fig. 2.

To validate the performance of the proposed architecture, we present two main experiments. First, the proposed strategy is compared with a state-of-the-art task-based inverse dynamics algorithm that considers rigid links [25] in the case of different stiffness parameters k. From now on, the proposed control approach is called *TSID-Flex* and the state-of-the-art controller *TSID-Rigid*. Second, we analyze the performance of the TSID-Flex in the case of different stiffness. In both scenarios, the desired robot CoM, footsteps, and actuated joint trajectories are computed offline[1]. The robot walks 1 meter forward with a step length of $20\,\mathrm{cm}$, starting with the right foot. The first and last steps are $10\,\mathrm{cm}$ long. The double support lasts $0.2\,\mathrm{s}$, while the single support lasts $1.2\,\mathrm{s}$.

A. Comparison between TSID-Flex and TSID-Rigid

In Table I, we summarize the results of the control strategies for different stiffness parameters k. Labels *success* and *failure* mean that the associated controller is either able or not to ensure the robot's balance while walking. In all the experiments presented in this section, the damping parameter is arbitrarily set to $b = 2\sqrt{k}$.

To compare the two controllers, we decided to perform two main experiments. In the former, we choose a set of stiffness parameters such that both whole-body controllers guarantee the balance while walking. In the latter, we decrease the value of the stiffness parameter. Namely:

- **Experiment 1** $k = 1 \times 10^4\,\mathrm{N\,m\,rad^{-1}}$;
- **Experiment 2** $k = 3 \times 10^3\,\mathrm{N\,m\,rad^{-1}}$.

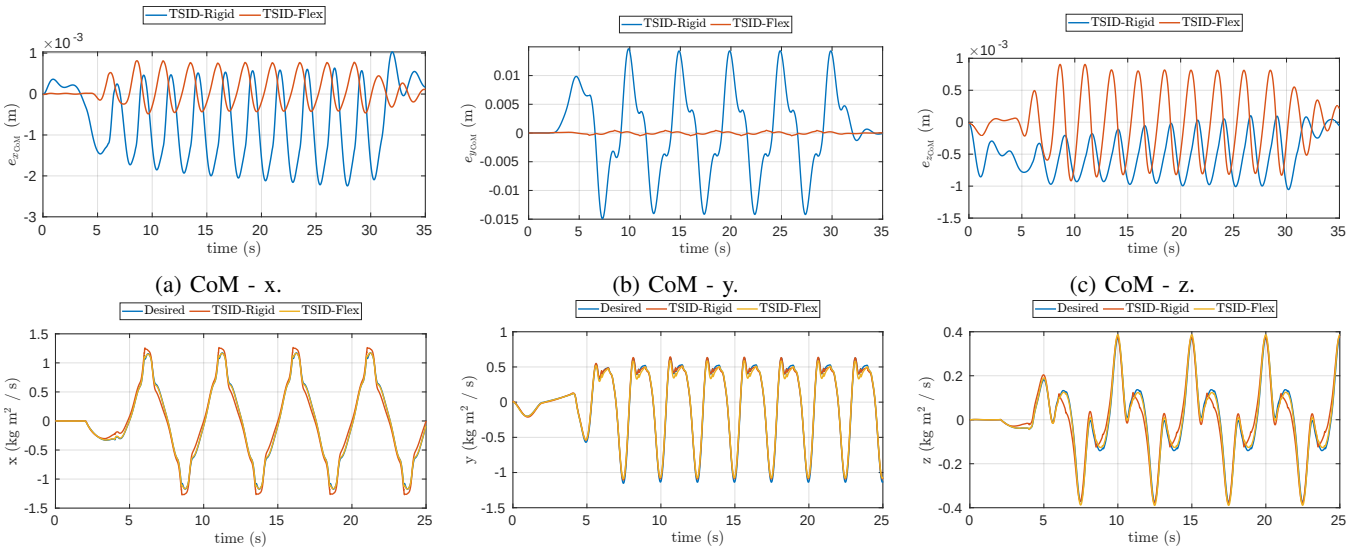
1) Experiment 1: Fig. 3 shows the CoM tracking performance obtained with TSID-Flex and TSID-Rigid in terms of tracking error. The TSID-Flex controller seems to show good tracking performance, and the CoM error is kept below $2\,\mathrm{mm}$. On the other hand, the TSID-Rigid induces a higher error on the CoM tracking. Similar considerations hold also for the tracking of the centroidal angular momentum.

In fact, TSID-Flex ensures a smaller angular momentum error with respect to the TSID-Rigid. One reason for this behavior is that the TSID-Rigid assumes the robot composed only by rigid link, hence all the joints are considered to be actuated. This assumption is generally valid in the case of stiff k, but it does not hold if k decreases. Fig. 5 presents

[1]The whole-body reference trajectories are provided by: github.com/loco-3d/multicontact-api/tree/v2.1.0

TABLE I: Controller outcome in the case of different stiffness parameter k.

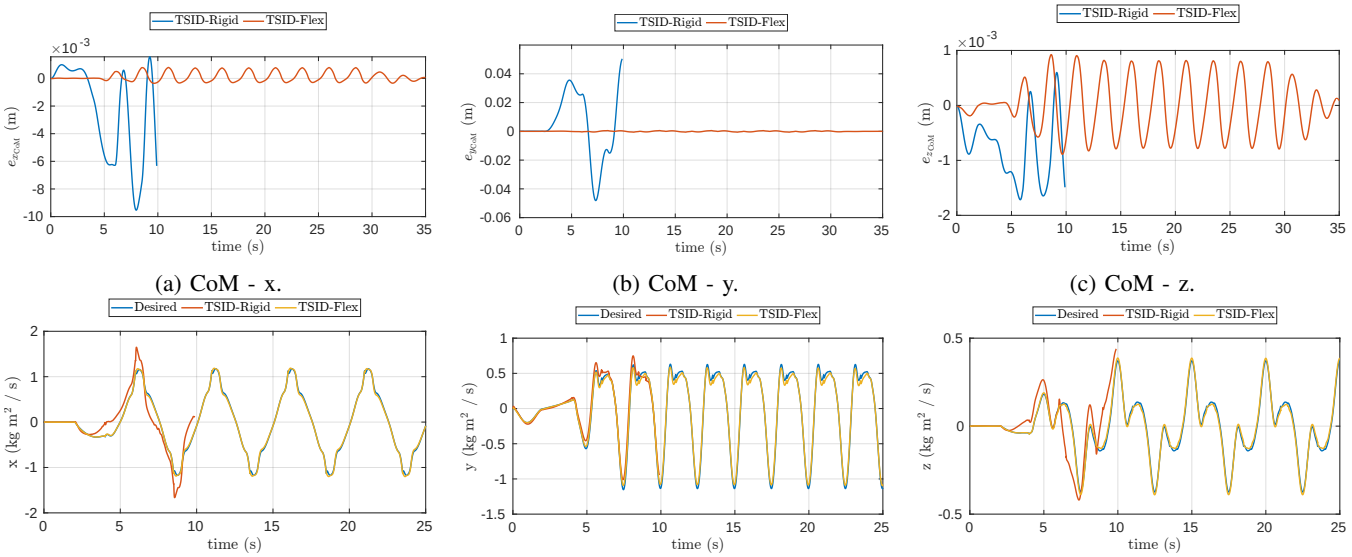
Control	$10\,\mathrm{kN\,m\,rad^{-1}}$	$5\,\mathrm{kN\,m\,rad^{-1}}$	$3\,\mathrm{kN\,m\,rad^{-1}}$	$2\,\mathrm{kN\,m\,rad^{-1}}$	$1\,\mathrm{kN\,m\,rad^{-1}}$
TSID-Rigid	success	success	failure	failure	failure
TSID-Flex	success	success	success	success	success



(a) CoM - x. (b) CoM - y. (c) CoM - z.

(d) Angular Momentum - x. (e) Angular Momentum - y. (f) Angular Momentum - z.

Fig. 3: Centroidal momentum tracking: comparison between TSID-Rigid and TSID-Flex – k = $1 \times 10^4\,\mathrm{N\,m\,rad^{-1}}$



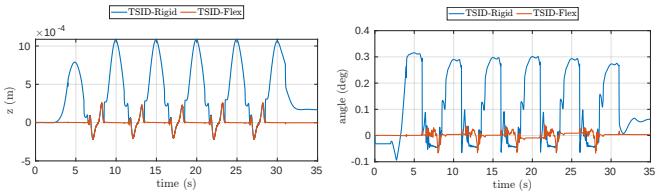
(a) CoM - x. (b) CoM - y. (c) CoM - z.

(d) Angular Momentum - x. (e) Angular Momentum - y. (f) Angular Momentum - z.

Fig. 4: Centroidal momentum tracking: comparison between TSID-Rigid and TSID-Flex – k = $3 \times 10^3\,\mathrm{N\,m\,rad^{-1}}$.

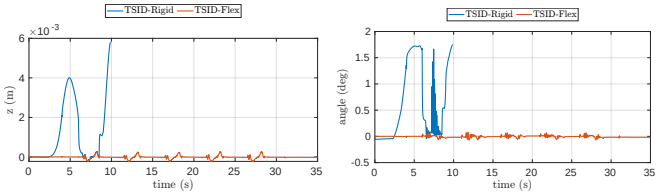
the left foot trajectory error when the whole-body controller is TSID-Flex or TSID-Rigid. The angular error is given by the angle of the axis-angle representation between the foot orientation and the desired orientation [26]. Since TSID-Rigid does not consider link deformations, the controller assumes a wrong foot orientation when the robot is on single support – for $2.5\,\mathrm{s} \leq t \leq 5.5\,\mathrm{s}$ in Fig. 5.

2) Experiment 2: The centroidal quantity tracking problem discussed in Experiment 1 worsens at lower values of the stiffness parameter k. Fig. 4 shows the CoM tracking performance of the two controllers. In Fig. 4 we also present the tracking of the angular momentum. The TSID-Flex is still capable of ensuring good performance. On the other hand, the TSID-Rigid does not consider the elastic joint state. As a consequence, this leads to a non-negligible error on the robot CoM. In order to keep the balance, the TSID-Rigid controller requires high variations of the robot's foot orientation and at $t \approx 10\,\mathrm{s}$ the robot falls – Fig. 6.



(a) Foot tracking position error. (b) Foot tracking angular error.

Fig. 5: Foot tracking: TSID-Rigid and TSID-Flex comparison – $k = 1 \times 10^4\,\mathrm{N\,m\,rad^{-1}}$.



(a) Foot tracking position error. (b) Foot tracking angular error.

Fig. 6: Foot tracking: TSID-Rigid and TSID-Flex comparison – $k = 3 \times 10^3\,\mathrm{N\,m\,rad^{-1}}$.

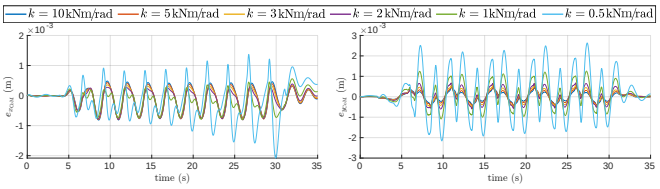
B. *Performance of the TSID-Flex in the case of different k*

We now benchmark the performance of the TSID-Flex controller for different stiffness parameters k, namely $k = 10\,\mathrm{kN\,m\,rad^{-1}}$, $5\,\mathrm{kN\,m\,rad^{-1}}$, $3\,\mathrm{kN\,m\,rad^{-1}}$, $2\,\mathrm{kN\,m\,rad^{-1}}$, $1\,\mathrm{kN\,m\,rad^{-1}}$, and $0.5\,\mathrm{kN\,m\,rad^{-1}}$.

Fig. 7 shows the CoM tracking error for different values of k. The controller ensure acceptable performance for all values of k, i.e., the error is always below $3\,\mathrm{mm}$. However, the lower the stiffness, the higher the tracking error. This increase in tracking error is caused by the elastic joint state estimation error. Fig. 8 presents the estimation error of an elastic joint. Similar considerations hold also for the other three joints. When the robot switches from single support to double support, the estimated torque associated with the elastic joint has a spike – at $t \approx 8.5\,\mathrm{s}$, $11\,\mathrm{s}$ and $13.5\,\mathrm{s}$ in Fig. 8a. As a consequence, the error propagates in the estimation of the elastic joint position and velocity – see Fig. 8b and 8c. This behavior is caused by the discontinuity of the contact force. To mitigate this effect, we may perform a smother transition between single and double support.

VI. Conclusions and Future Work

This paper contributes to the design of a whole-body QP controller for a humanoid robot affected by link flexibility. We model the flexibility by introducing equivalent passive joints that simulate the motion caused by the viscoelastic link deformation. We then considered the passive joints position and velocity as state of the floating base system dynamics. Thanks to this choice, unlike other state-of-the-art approaches, we develop a whole-body controller that explicitly considers link flexibility in the stabilization problem. The paper also details the design of an estimator that aims at computing the elastic joint state in real-time. The proposed approach is validated in a simulated version of the TALOS humanoid robot, where its hip flexibility has a significant impact while



(a) CoM x error. (b) CoM y error.

Fig. 7: CoM Tracking performance in the case of different k.

performing locomotion tasks. Moreover, the architecture is then compared with a state-of-the-art whole-body controller that considers all links rigid. In future work, we plan to mitigate the contact forces discontinuity by performing a smoother transition between contiguous support phases. We also plan to make a comparison with other state-of-the-art controllers that consider the flexibility of the robot link [10]. Another interesting research direction is to consider the link flexibility in the generation of reference centroidal trajectories [27], [28] and whole-body-motion [29], [30].

References

[1] S. Feng, X. Xinjilefu, C. G. Atkeson, and J. Kim, "Robust dynamic walking using online foot step optimization," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 2016-November. IEEE, 10 2016, pp. 5373–5378. [Online]. Available: http://ieeexplore.ieee.org/document/7759790/

[2] J. Englsberger, T. Koolen, S. Bertrand, J. Pratt, C. Ott, and A. Albu-Schaffer, "Trajectory generation for continuous leg forces during double support and heel-to-toe shift based on divergent component of motion," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 9 2014, pp. 4022–4029. [Online]. Available: http://ieeexplore.ieee.org/document/6943128/

[3] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a Momentum-Based Control Framework and Application to the Humanoid Robot Atlas," *International Journal of Humanoid Robotics*, 2016.

[4] G. Romualdi, S. Dafarra, Y. Hu, and D. Pucci, "A Benchmarking of DCM Based Architectures for Position and Velocity Controlled Walking of Humanoid Robots," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*. IEEE, 11 2018, pp. 1–9. [Online]. Available: https://ieeexplore.ieee.org/document/8625025/

[5] M. A. Hopkins, D. W. Hong, and A. Leonessa, "Compliant locomotion using whole-body control and Divergent Component of Motion tracking," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5 2015, pp. 5726–5733. [Online]. Available: http://ieeexplore.ieee.org/document/7140001/

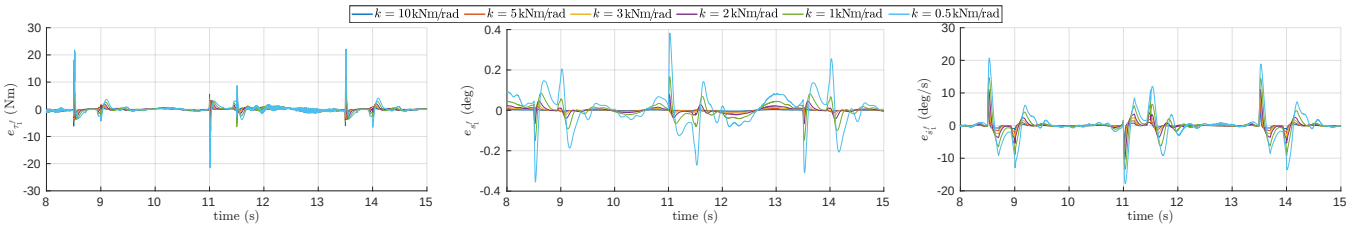
[6] N. Ramuzat, G. Buondonno, S. Boria, and O. Stasse, "Comparison of Position and Torque Whole-Body Control Schemes on the Humanoid Robot TALOS," in *2021 20th International Conference on Advanced Robotics (ICAR)*. IEEE, 12 2021, pp. 785–792.

[7] M. Sayahkarajy, Z. Mohamed, and A. A. Mohd Faudzi, "Review of modelling and control of flexible-link manipulators:," *http://dx.doi.org/10.1177/0959651816642099*, vol. 230, no. 8, pp. 861–873, 4 2016. [Online]. Available: https://journals.sagepub.com/doi/10.1177/0959651816642099

[8] T. Yoshikawa and K. Hosoda, "Modeling of flexible manipulators using virtual rigid links and passive joints," pp. 967–972, 12 2002.

[9] S. Kajita, K. Yokoi, M. Saigo, and K. Tanie, "Balancing a humanoid robot using backdrive concerned torque control and direct angular momentum feedback," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 4, pp. 3376–3382, 2001.

[10] N. A. Villa, P. Fernbach, M. Naveau, G. Saurel, E. Dantec, N. Mansard, and O. Stasse, "Torque Controlled Locomotion of a Biped Robot with Link Flexibility," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids) [Submitted]*. IEEE, 2022.



(a) Elastic joint torque error. (b) Elastic joint position error. (c) Elastic joint velocity error.

Fig. 8: Elastic joint state estimation error.

[11] S. Nakaoka, S. Hattori, F. Kanehiro, S. Kajita, and H. Hirukawa, "Constraint-based dynamics simulator for humanoid robots with shock absorbing mechanisms," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 10 2007, pp. 3641–3647.

[12] M. Benallegue and F. Lamiraux, "Estimation and stabilization of humanoid flexibility deformation using only inertial measurement units and contact information," *International Journal of Humanoid Robotics*, vol. 12, no. 03, p. 1550025, 2015.

[13] S. S. Ge, T. H. Lee, and G. Zhu, "A Nonlinear Feedback Controller for a Single-Link Flexible Manipulator Based on a Finite Element Model," *Journal of Robotic Systems*, vol. 14, no. 3, pp. 165–178, 1997.

[14] D. Subedi, I. Tyapin, and G. Hovland, "Dynamic Modeling of Planar Multi-Link Flexible Manipulators," *Robotics 2021, Vol. 10, Page 70*, vol. 10, no. 2, p. 70, 5 2021. [Online]. Available: https://www.mdpi.com/2218-6581/10/2/70

[15] R. J. Theodore and A. Ghosal, "Robust control of multilink flexible manipulators," *Mechanism and Machine Theory*, vol. 38, no. 4, pp. 367–377, 4 2003.

[16] O. Stasse, T. Flayols, R. Budhiraja, K. Giraud-Esclasse, J. Carpentier, J. Mirabel, A. Del Prete, P. Soueres, N. Mansard, F. Lamiraux, J. P. Laumond, L. Marchionni, H. Tome, and F. Ferro, "TALOS: A new humanoid research platform targeted for industrial applications," *IEEE-RAS International Conference on Humanoid Robots*, pp. 689–695, 12 2017.

[17] R. Featherstone, *Rigid Body Dynamics Algorithms*. Springer, Boston, MA, 2014.

[18] S. Traversaro, "Modelling, Estimation and Identification of Humanoid Robots Dynamics," Ph.D. dissertation, 2017.

[19] P.-B. Wieber, R. Tedrake, and S. Kuindersma, "Modelling and control of legged robots," in *Springer Handbook of Robotics*. Springer, 2016, pp. 1203–1234.

[20] G. Nava, F. Romano, F. Nori, and D. Pucci, "Stability Analysis and Design of Momentum-based Controllers for Humanoid Robots," *Intelligent Robots and Systems (IROS) 2016. IEEE International Conference on*, 2016.

[21] D. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonomous Robots*, 2013.

[22] K. M. Lynch and F. C. Park, *Modern Robotics: Mechanics, Planning, and Control*, 1st ed. 40 W. 20 St. New York, NYUnited States: Cambridge University Press, 2017.

[23] S. Caron, Q. C. Pham, and Y. Nakamura, "Stability of surface contacts for humanoid robots: Closed-form formulae of the Contact Wrench Cone for rectangular support areas," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 5107–5112, 6 2015.

[24] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "OSQP: An Operator Splitting Solver for Quadratic Programs," *2018 UKACC 12th International Conference on Control, CONTROL 2018*, p. 339, 10 2018.

[25] G. Romualdi, S. Dafarra, Y. Hu, P. Ramadoss, F. J. A. Chavez, S. Traversaro, and D. Pucci, "A Benchmarking of DCM-Based Architectures for Position, Velocity and Torque-Controlled Humanoid Robots," *International Journal of Humanoid Robotics*, vol. 17, no. 01, p. 1950034, 2 2020. [Online]. Available: https://www.worldscientific.com/doi/abs/10.1142/S0219843619500348

[26] D. Q. Huynh, "Metrics for 3D rotations: Comparison and analysis," *Journal of Mathematical Imaging and Vision*, 2009.

[27] G. Romualdi, S. Dafarra, G. L'Erario, I. Sorrentino, S. Traversaro, and D. Pucci, "Online Non-linear Centroidal MPC for Humanoid Robot Locomotion with Step Adjustment," in *2022 IEEE International Conference on Robotics and Automation (ICRA)*, 3 2022.

[28] M. Murooka, M. Morisawa, and F. Kanehiro, "Centroidal Trajectory Generation and Stabilization Based on Preview Control for Humanoid Multi-Contact Motion," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 8225–8232, 7 2022. [Online]. Available: https://ieeexplore.ieee.org/document/9807369/

[29] S. Dafarra, G. Romualdi, and D. Pucci, "Dynamic complementarity conditions and whole-body trajectory optimization for humanoid robot locomotion," *IEEE Transactions on Robotics*, pp. 1–20, 2022.

[30] E. Dantec, R. Budhiraja, A. Roig, T. Lembono, G. Saurel, O. Stasse, P. Fernbach, S. Tonneau, S. Vijayakumar, S. Calinon, M. Taix, and N. Mansard, "Whole Body Model Predictive Control with a Memory of Motion: Experiments on a Torque-Controlled Talos," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 5 2021, pp. 8202–8208.