Article

The International Journal of
Robotics Research
2016, Vol. 35(12) 1522–1543

DOI: 10.1177/0278364916653815
ijr.sagepub.com

Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios

Bernd Henze, Máximo A. Roa and Christian Ott

Abstract

This work presents a new control approach to multi-contact balancing for torque-controlled humanoid robots. The controller includes a non-strict task hierarchy, which allows the robot to use a subset of its end effectors for balancing while the remaining ones can be used for interacting with the environment. The controller creates a passive and compliant behavior for regulating the center of mass (CoM) location, hip orientation and the poses of each end effector assigned to the interaction task. This is achieved by applying a suitable wrench (force and torque) at each one of the end effectors used for interaction. The contact wrenches at the balancing end effectors are chosen such that the sum of the balancing and interaction wrenches produce the desired wrench at the CoM. The algorithm requires the solution of an optimization problem, which distributes the CoM wrench to the end effectors taking into account constraints for unilaterality, friction and position of the center of pressure. Furthermore, the feedback controller is combined with a feedforward control in order to improve performance while tracking a predefined trajectory, leading to a control structure similar to a PD+ control. The controller is evaluated in several experiments with the humanoid robot TORO.

Keywords

Multi-contact balancing, whole-body control, torque control, humanoid robot

1. Introduction

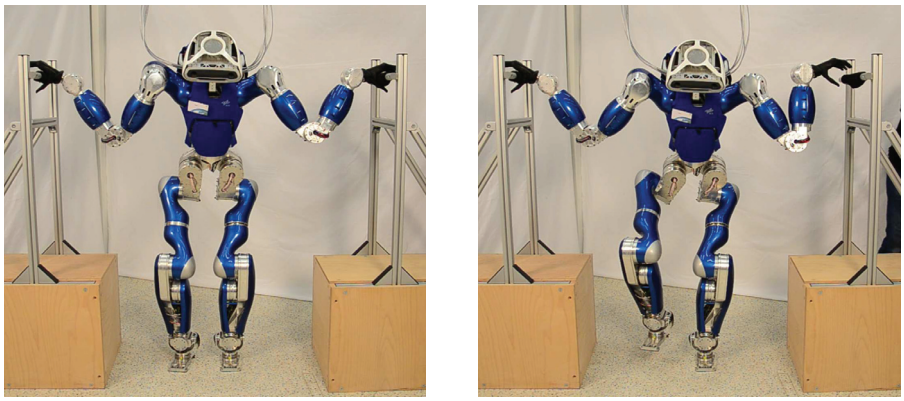
Humanoid robots are in general conceived to provide a natural interaction with humans in everyday tasks and environments. Typical tasks that could be covered by a humanoid robot involve repetitive motions that might be too physically demanding for humans, or exploration of areas which are difficult or risky for a person, such as a disaster scenario or the surface of another planet. To obtain a proper performance in such situations, the robot would require highly robust and agile ways of locomotion, including climbing stairs or industrial ladders, moving through confined spaces or traversing terrain with obstacles or debris.

The basic behavior a humanoid robot should have is the ability to move and balance (compliantly) in the presence of unknown external perturbations. Sensitive compliance and impedance control can be obtained via joint torque sensing and control (Ott et al., 2008). Torque sensing has been explicitly used in the hydraulic humanoid robot CB (Cheng et al., 2007) and implicitly via serial elastic actuators in Pratt and Krupp (2008). At DLR, the full-body humanoid robot TORO (see Figure 1) was developed based on the torque-controlled drive units of the DLR-KUKA Light-Weight Robot, which can be controlled both in position or torque mode (Ott et al., 2010).

The balance of a humanoid robot, i.e. the control of the linear and angular momentum of the system, is achieved through the application of suitable contact forces to the ground using the finite support area of the feet (Macchietto et al., 2009). However, challenging scenarios might require that the robot uses all of its extremities to get additional support in the environment, thus creating multi-contact scenarios. In general, the contact is established with the end effectors (hands and feet), but other parts of the robot such as knees or elbows could potentially be employed. To generate locomotion, some of the end effectors need to guarantee the balance while others can be lifted and repositioned as required. Other applications might require that the hands perform some desired manipulation task, for instance opening a door, lifting an object or closing a valve, while the robot is standing still. In any case, the forces coming from the end effectors used in such additional task must be considered to guarantee the overall stability of the robot.

German Aerospace Center (DLR), Institute of Robotics and Mechatronics, Germany

Corresponding author:
Bernd Henze, German Aerospace Center (DLR), Institute of Robotics and Mechatronics, Münchener Str. 20, 82234 Weßling, Germany.
Email: bernd.henze@dlr.de



(a) Four contacts: Both hands and both feet.

(b) Two contacts: Right hand and left foot.

Fig. 1. Torque-controlled humanoid robot TORO balancing with multiple contacts.

A straightforward way to cope with multi-contact situations is to consider separately the upper and lower body of the robot. In general, the upper body is operated in impedance mode to perform the manipulation task, while the lower body is employed for the balancing task. This allows the generation of modular control algorithms for the two tasks. Wrenches originating from the manipulation task with the upper body must be taken into account by the balancing control. This is achieved for instance in Ibanez et al. (2012) using a zero moment point (ZMP)-based balancer for the lower body and a position-based controller for manipulation with the upper body. The balancer uses preview control in combination with a disturbance input that accounts for the wrenches caused by the manipulation task, which enables the robot to perform whole-body motions and manipulations. A similar solution is presented in Ott et al. (2013) for kinesthetic teaching in combination with interaction-aware balancing. Here, the external wrenches resulting from the interaction are estimated with a momentum-based observer and handed over to a ZMP-based balancer, which takes them into account via a disturbance input. Those disturbances are integrated into a preview control, so that the robot is able to react to them in advance.

More versatile approaches are obtained when using a whole-body control approach. To consider all the degrees of freedom (DoFs) of the robot, inverse kinematics can be used to find the joint commands that generate a desired trajectory for the center of mass (CoM). For instance, a dynamic balance force controller was proposed to determine full-body joint torques based on the desired motion of the CoM, combined with some desired virtual task forces (Stephens and Atkeson, 2010). This approach controls the motion of the CoM and the angular momentum of the robot by computing suitable contact forces via a quadratic optimization problem. The mapping of the contact forces to the joint torques is solved considering the multi-body dynamics of the system. In addition to the force distribution, the control of internal forces during multi-contact interaction tasks was studied in Sentis et al. (2010) or Righetti et al. (2013). based on the concept of a virtual-linkage model, which provides a representation of the internal forces and resultant forces at the CoM with respect to reaction forces on the supporting surfaces. They also propose a prioritization of different tasks using the framework of operational space control. Using inverse dynamics, an optimal solution to the balancing problem can be obtained, optimizing the position of the center of pressure (CoP) in each foot (Wensing et al., 2013), or minimizing the contact wrenches at each end effector (Righetti et al., 2013). Mistry et al. (2010) proposed an inverse-dynamics-based controller, which used an inverse dynamics computation for a feed-forward control action in combination with a PD controller on the actuated joint angles. In this work, it was assumed that the external forces act only at the end-effectors in rigid contact, and an orthogonal decomposition of the constraint Jacobian was used in order to avoid the need of measuring these external forces.

Passivity theory has proven a powerful tool to design robust controllers for robots that physically interact with the environment (Arimoto, 2000; Stramigioli, 2001). Passivity-based impedance and compliance controllers based on joint torque sensing have been traditionally applied to manipulation tasks (Albu-Schäffer et al., 2007b; Ott et al., 2008). The application of such framework to whole-body biped balancing control was first proposed in Hyon et al. (2007). This controller provides gravity compensation, making the robot compliant and thus facilitating physical interactions and adaptation to unknown external forces. This approach computes a ground reaction force able to compensate perturbations on the robot position, and directly transforms the desired force to joint torques. It is able to cope with an arbitrary number of interaction points with the environment, but does not require force measurement at such points and does not use inverse kinematics or inverse dynamics. The controller was tested both in simulation and on a real humanoid (Cheng et al., 2007), and has been extended to compensate for yaw perturbations and to provide adaptability to unknown rough terrain (Hyon, 2009). The ideas from that controller were later extended to obtain a controller

for the robot posture, tested in the DLR biped robot (Ott et al., 2011), which actively exploits the observation that the problems of grasping an object and balancing a robot are fundamentally similar. The main objective is to keep the position and orientation of the robot despite external perturbations. When a perturbation changes the posture of the frame attached to the CoM, the wrench required to recover the original position is computed, and then it is distributed among predefined contact points to create the desired balancing forces. The work of Henze et al. (2014a) represents another extension of Ott et al. (2011), in which the approach for torque-based balancing was combined with model predictive control (MPC). A similar approach was followed by Audren et al. (2014). The latter two references focused on predictions using simplified models, but were evaluated only by means of simulations.

This paper presents a passivity-based balancing control for torque-controlled humanoid robots. It extends our previous work in Ott et al. (2011) by:

- a feedforward control to improve the tracking performance;
- the capability of exploiting multiple contact points;
- a non-strict task hierarchy allowing interactions with the environment while balancing.

The task hierarchy enables the robot to use some of its end effectors for interacting with the environment (e.g. for manipulation tasks) while the remaining end effectors are used for maintaining the balance by generating suitable contact wrenches. More precisely, the parametrization employed in the task hierarchy is not only restricted to assigning whole end effectors to a task, but is also able to assign every single component of each end effector wrench to a different task. The proposed approach does not distinguish between the upper and lower part of the robot. Furthermore, the method allows the controller to be reconfigured online, thus providing a non-static assignment to the balancing and manipulation task. The desired task weights are used in the optimization process, thus allowing a prioritization of the two tasks if both cannot be fulfilled at the same time. The proposed feedforward control is added to the feedback loop in order to improve performance and robustness in the tracking case. The resulting structure of the controller is similar to PD+ control, which allows us to prove stability and passivity of the closed-loop system. The controller has been implemented and tested in different situations with the humanoid robot TORO, thus showing the performance and robustness of the proposed approach. One of the conducted experiments involves TORO balancing on three layers of gym mats, which is, to the best of the authors' knowledge, the first experimental realization of a humanoid robot balancing on a compliant support surface.

In contrast to other methods, the proposed passivity-based balancing controller aims at stabilizing the system by implementing a compliant reaction force for disturbances from the reference trajectory. This is different from common approaches based on inverse dynamics, where the whole closed-loop dynamics is shaped, as e.g. in Sentis et al. (2010) or Righetti et al. (2013). Furthermore, our controller does not require a measurement of the external forces.

The proposed balancing controller utilizes the operational space formulation from Khatib (1987) by expressing the systems dynamics with coordinates that are directly relevant for the task the robot is supposed to fulfill. More precisely, we use the CoM location to formulate the goal of balancing, and the Cartesian end effector pose for interaction with the environment, since interaction tasks are usually expressed in Cartesian coordinates relative to the world. In Williams and Khatib (1993) a framework called virtual linkage is presented and used to describe the internal forces inside an object manipulated by multiple robotic arms. The concept was reused, e.g. by Sentis et al. (2010), for controlling the internal forces of a humanoid robot balancing with multiple contacts. The balancing controller presented in this work reduces the internal forces by reducing the contact wrenches, and thereby does not utilize the concept of virtual links for characterizing internal forces.

The paper is organized as follows. Section 2 summarizes the theoretical background related to the problem of grasping, and develops the dynamical model of the robot. Section 3 describes the whole-body controller, along with discussions on stability, redundancy and singular configurations of the system. Section 4 describes further implementation details, such as the simplification of the optimization problem to obtain a solution in real time, and the estimation of the object pose using the available sensorial information. Section 5 presents the experiments used to test the validity and robustness of the desired control, including the general behavior of the controller and specific tests on the redundancy of the wrench distribution, the tracking performance, and the balancing behavior on compliant surfaces. Section 6 concludes the paper.

2. Modeling

2.1. Relation between balancing and grasping

Our previous work in Ott et al. (2011) presented an approach for posture and balance control capable of dealing with unknown external perturbations by distributing the required balancing forces among predefined contact points at the feet. The approach is strongly based on the observation that the problems of grasping an object and balancing a robot are fundamentally similar, in the sense that both try to achieve a desired wrench $\boldsymbol{F}_o$ (on the object in the grasping case, on the robot in the balancing case) based on the application of suitable forces at the contact points $\boldsymbol{f}_i$ (at the fingertips or at the feet, respectively). Figure 2 illustrates such similarity. By using the same modeling of forces as in grasping, in Ott et al. (2011) the force required for balancing is distributed to the contact points; the final solution

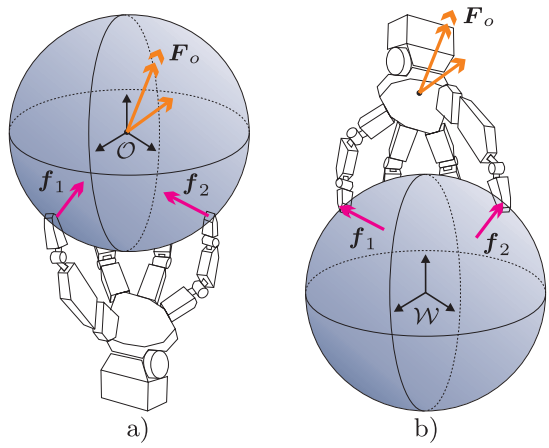


Fig. 2. Force distribution is fundamentally similar in (a) grasping and (b) balancing.

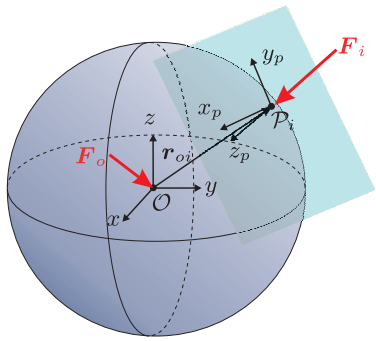


Fig. 3. Object and contact coordinate frames.

is obtained via an optimization problem that includes additional constraints such as friction cones, unilaterality of the contacts, and CoP location inside the supporting feet.

An object is grasped to resist external perturbations or to manipulate it in a dexterous way. In grasp theory, the location of each contact is described by a coordinate frame $\mathcal{P}_i$ that has a relative position $\boldsymbol{r}_{oi}$ and orientation $\boldsymbol{R}_{oi}$ with respect to the object frame $\mathcal{O}$ (see Figure 3). Commonly, the reference frame is located at the CoM of the object. The contact wrenches $\boldsymbol{F}_i = (\boldsymbol{f}_i^{\mathrm{T}}\ \boldsymbol{\tau}_i^{\mathrm{T}})^{\mathrm{T}}$ are usually given in the contact frame $\mathcal{P}_i$, while the object wrench $\boldsymbol{F}_o$ and the vector $\boldsymbol{r}_{oi}$ are specified in the object frame $\mathcal{O}$ (Murray et al., 1994).

In general, a fingertip can only apply forces in certain directions, described by a contact model. The frictional properties at the fingertip are commonly described using Coulomb's friction model, which states that slippage is avoided when $\|\boldsymbol{f}_{\parallel}\| \leq \mu\,|f_{\perp}|$, where $f_{\perp}$ and $\boldsymbol{f}_{\parallel}$ are the normal and tangential components, and μ is the friction coefficient. Therefore, the set of allowed forces at the i^{th} contact point is

$$\mathcal{F}_i = \left\{ \boldsymbol{f}_i \in \mathbb{R}^3 \;\middle|\; \|\boldsymbol{f}_{i,\parallel}\| \leq \mu f_{i,\perp}, f_{i,\perp} \geq 0 \right\} \tag{1}$$

Geometrically, $\mathcal{F}_i$ represents a friction cone with an axis along the surface normal and a semiangle of $\varphi = \operatorname{atan}(\mu)$. In addition to the friction constraints, the forces must also fulfill the positivity restriction ($f_{i,\perp} \geq 0$) due to the unilaterality of the contacts, i.e. the fingers can push but cannot pull the object.

The generalized wrench $\boldsymbol{F}_i$ that can be applied at a contact point is described by $\boldsymbol{F}_i = \boldsymbol{B}_i \boldsymbol{f}_i$, with $\boldsymbol{B}_i$ being the wrench basis that characterizes the contact model. For instance, considering a frictional point contact, the applied wrench is

$$\boldsymbol{F}_i = \begin{pmatrix} \boldsymbol{f}_i \\ \boldsymbol{\tau}_i \end{pmatrix} = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{0} \end{bmatrix} \boldsymbol{f}_i = \boldsymbol{B}_i \boldsymbol{f}_i \tag{2}$$

The total wrench $\boldsymbol{F}_o$ on the object is the sum of the contributions from each one of the Ψ contacts

$$\begin{aligned} \boldsymbol{F}_o &= \sum_{i=1}^{\Psi} \boldsymbol{Ad}_{oi}^{\mathrm{T}} \boldsymbol{F}_i = \sum_{i=1}^{\Psi} \boldsymbol{Ad}_{oi}^{\mathrm{T}} \boldsymbol{B}_i \boldsymbol{f}_i \\ &= \begin{bmatrix} \boldsymbol{G}_1 & \dots & \boldsymbol{G}_\psi \end{bmatrix} \begin{pmatrix} \boldsymbol{f}_1 \\ \vdots \\ \boldsymbol{f}_\psi \end{pmatrix} = \boldsymbol{G}\boldsymbol{f}, \end{aligned} \tag{3}$$

where $\boldsymbol{G}_i = \boldsymbol{Ad}_{oi}^{\mathrm{T}} \boldsymbol{B}_i$ is called the contact map. The transpose of the adjoint matrix

$$\boldsymbol{Ad}_{oi}^{\mathrm{T}} = \begin{bmatrix} \boldsymbol{R}_{oi} & \boldsymbol{0} \\ \hat{\boldsymbol{r}}_{oi}\boldsymbol{R}_{oi} & \boldsymbol{R}_{oi} \end{bmatrix} \tag{4}$$

maps the contact wrench $\boldsymbol{F}_i$ given in the frame $\mathcal{P}_i$ to the object frame $\mathcal{O}$. Here, $\hat{\boldsymbol{r}}_{oi}$ denotes the cross-product matrix

$$\hat{\boldsymbol{r}}_{oi} = \begin{bmatrix} 0 & -p_z & p_y \\ p_z & 0 & -p_x \\ -p_y & p_x & 0 \end{bmatrix} \tag{5}$$

for the vector $\hat{\boldsymbol{r}}_{oi} = (p_x\ p_y\ p_z)^{\mathrm{T}}$ given in $\mathcal{O}$. Stacking all of the $\boldsymbol{f}_i$ into the vector $\boldsymbol{f}$ leads to the grasp matrix $\boldsymbol{G} = [\,\boldsymbol{G}_1\ \dots\ \boldsymbol{G}_\psi\,]$ that maps the contact forces, characterized by the contact model, to the object coordinate frame $\mathcal{O}$.

In the context of balancing, it is convenient to assume a contact model capable of transmitting a full wrench, which corresponds to a wrench basis $\boldsymbol{B}_i = \boldsymbol{I}$. In the following sections, all quantities will be expressed with respect to a world frame $\mathcal{W}$ for simplicity. If $\boldsymbol{F}_i$ and $\boldsymbol{F}_o$ are given in $\mathcal{W}$, then the contact map changes to

$$\tilde{\boldsymbol{G}}_i = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \hat{\boldsymbol{r}}_{oi} & \boldsymbol{I} \end{bmatrix} \tag{6}$$

with $\boldsymbol{r}_{oi}$ expressed in $\mathcal{W}$ as well. The mapping between the contact wrenches $\boldsymbol{F}_i$ and the object wrench $\boldsymbol{F}_o$ changes into

$$\tilde{\boldsymbol{G}} = \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} & \dots & \boldsymbol{I} & \boldsymbol{0} \\ \hat{\boldsymbol{r}}_{o1} & \boldsymbol{I} & \dots & \hat{\boldsymbol{r}}_{o\Psi} & \boldsymbol{I} \end{bmatrix} \tag{7}$$

accordingly, which we will call the 'rotated grasp matrix'.

2.2. Dynamic model

This section describes the dynamic model of the robot that will later be used in Section 3.2 to derive the balancing controller. The controller is intended to handle an arbitrary number of end effectors, which can either be in contact or not in contact with the environment. Since the contact configuration can change over time, we consider a dynamic model with a free floating base, which allows a higher flexibility in the use of the end effectors when compared with a fixed base model (which predefines one end effector as the robot base, thus it must always be fixed to the environment).

The state of the floating base is described by the position $\boldsymbol{x}_b \in \mathbb{R}^3$ and orientation $\boldsymbol{R}_b \in \mathcal{SO}(3)$ of the frame $\mathcal{B}$ relative to the world frame $\mathcal{W}$, with corresponding linear and angular velocities $\dot{\boldsymbol{x}}_b \in \mathbb{R}^3$ and $\boldsymbol{\omega}_b \in \mathbb{R}^3$. Usually the hip (or torso) is chosen as base link, since it is a central body in the structure of a humanoid robot. For the model description, all quantities are expressed relative to the world frame $\mathcal{W}$ unless noted otherwise. The end effectors are defined by frames $\mathcal{T}_i$, with $i = 1 \dots \Psi$ and Ψ being the total number of end effectors. A frame $\mathcal{T}_i$ has a position $\boldsymbol{x}_i \in \mathbb{R}^3$ and orientation $\boldsymbol{R}_i \in \mathcal{SO}(3)$ relative to the world frame, while its corresponding velocities are denoted by $\dot{\boldsymbol{x}}_i \in \mathbb{R}^3$ and $\boldsymbol{\omega}_i \in \mathbb{R}^3$. Using this notation, the dynamics of the robot can be written as

$$\bar{\boldsymbol{M}} \begin{pmatrix} \ddot{\boldsymbol{x}}_b \\ \dot{\boldsymbol{\omega}}_b \\ \ddot{\boldsymbol{q}} \end{pmatrix} + \bar{\boldsymbol{C}} \begin{pmatrix} \dot{\boldsymbol{x}}_b \\ \boldsymbol{\omega}_b \\ \dot{\boldsymbol{q}} \end{pmatrix} + \bar{\boldsymbol{g}} = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{\tau} \end{pmatrix} + \boldsymbol{\tau}_{\text{ext}} \tag{8}$$

with $\bar{\boldsymbol{M}}$ and $\bar{\boldsymbol{C}}$ being the inertia and the Coriolis matrices, respectively. The influence of gravity is considered by the vector $\bar{\boldsymbol{g}}$. The position and velocity of the n joints are given by $\boldsymbol{q} \in \mathbb{R}^n$ and $\dot{\boldsymbol{q}} \in \mathbb{R}^n$, while the joint torques of the actuators are denoted by $\boldsymbol{\tau} \in \mathbb{R}^n$. The generalized external forces are given by $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^{6+n}$.

Remark 1. *If all external forces act at the end effector frames $\mathcal{T}_i$, then $\boldsymbol{\tau}_{ext}$ simplifies to*

$$\boldsymbol{\tau}_{ext} = \sum_{i=1}^{\Psi} \begin{bmatrix} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \hat{\boldsymbol{x}}_{bi} & \boldsymbol{I} \end{bmatrix} \\ \bar{\boldsymbol{J}}_i^{\mathrm{T}} \end{bmatrix} \boldsymbol{F}_i \tag{9}$$

with the Jacobian matrix $\bar{\boldsymbol{J}}_i \in \mathbb{R}^{6\times n}$, the end effector wrench $\boldsymbol{F}_i$, and $\boldsymbol{x}_{bi} = \boldsymbol{x}_i - \boldsymbol{x}_b$ denoting the configuration-dependent lever arm between the end effector frame $\mathcal{T}_i$ and the base frame $\mathcal{B}$.

Since the location of the CoM is crucial for balancing, we will perform a coordinate transformation by replacing $\boldsymbol{x}_b$ with the CoM location, as proposed in Hyon et al. (2007). Furthermore, we want to control the orientation of the base link of the robot. For this, we define another frame $\mathcal{C}$ located directly at the CoM, with the position $\boldsymbol{x}_c$ with respect to $\mathcal{W}$, and with the same orientation $\boldsymbol{R}_c$ as the base link (see Figure 4). The transformation[1] obtained by replacing $\dot{\boldsymbol{x}}_b$ and $\boldsymbol{\omega}_b$ with $\dot{\boldsymbol{x}}_c$ and $\boldsymbol{\omega}_c$ is given by

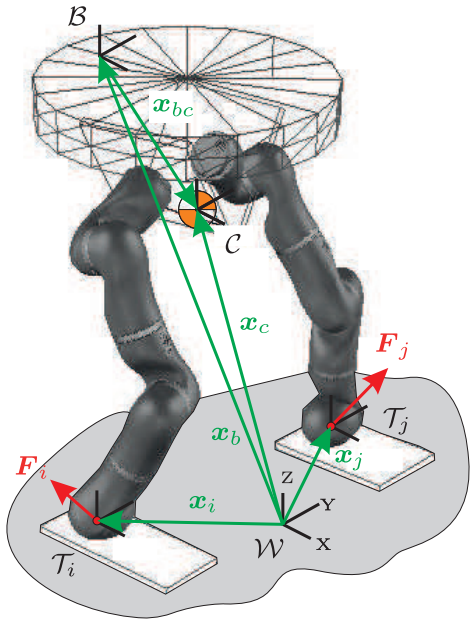


Fig. 4. Floating base model for the robot. The end effector wrenches $\boldsymbol{F}_i$ and $\boldsymbol{F}_j$ are considered in $\boldsymbol{\tau}_{\text{ext}}$.

$$\begin{pmatrix} \dot{\boldsymbol{x}}_c \\ \boldsymbol{\omega}_c \\ \dot{\boldsymbol{q}} \end{pmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{I} & -\hat{\boldsymbol{x}}_{bc} & \boldsymbol{J}_{bc} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}}_{\bar{\boldsymbol{T}}} \begin{pmatrix} \dot{\boldsymbol{x}}_b \\ \boldsymbol{\omega}_b \\ \dot{\boldsymbol{q}} \end{pmatrix}. \tag{10}$$

Here, $\boldsymbol{x}_{bc} = \boldsymbol{x}_c - \boldsymbol{x}_b$ denotes the lever arm between the CoM and the base frame $\mathcal{B}$ expressed in the world frame $\mathcal{W}$. The Jacobian $\boldsymbol{J}_{bc} = \partial \boldsymbol{x}_{bc} / \partial \boldsymbol{q}$ is needed to take into account the relative motion between the frames $\mathcal{B}$ and $\mathcal{C}$. Applying (10) to (8) leads to

$$\underbrace{\begin{bmatrix} m\boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{M}_{11} & \boldsymbol{M}_{12} \\ \boldsymbol{0} & \boldsymbol{M}_{21} & \boldsymbol{M}_{22} \end{bmatrix}}_{\boldsymbol{M}} \begin{pmatrix} \ddot{\boldsymbol{x}}_c \\ \dot{\boldsymbol{\omega}}_c \\ \ddot{\boldsymbol{q}} \end{pmatrix} + \underbrace{\begin{bmatrix} \boldsymbol{C}_1 \\ \boldsymbol{C}_2 \\ \boldsymbol{C}_3 \end{bmatrix}}_{\boldsymbol{C}} \begin{pmatrix} \dot{\boldsymbol{x}}_c \\ \boldsymbol{\omega}_c \\ \dot{\boldsymbol{q}} \end{pmatrix} + \underbrace{\begin{pmatrix} m\boldsymbol{g}_0 \\ \boldsymbol{0} \\ \boldsymbol{0} \end{pmatrix}}_{\boldsymbol{g}} = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{0} \\ \boldsymbol{\tau} \end{pmatrix} + \bar{\boldsymbol{T}}^{-\mathrm{T}} \boldsymbol{\tau}_{\text{ext}}. \tag{11}$$

with the transformed inertia and Coriolis matrix $\boldsymbol{M} = \bar{\boldsymbol{T}}^{-\mathrm{T}} \bar{\boldsymbol{M}} \bar{\boldsymbol{T}}^{-1}$ and $\boldsymbol{C} = \bar{\boldsymbol{T}}^{-\mathrm{T}} \bar{\boldsymbol{C}} \bar{\boldsymbol{T}}^{-1} + \bar{\boldsymbol{T}}^{-\mathrm{T}} \bar{\boldsymbol{M}} \frac{d}{dt} \bar{\boldsymbol{T}}^{-1}$. The scalar m denotes the overall mass of the robot, while $\boldsymbol{g}_0 \in \mathbb{R}^3$ represents the vector of gravitational acceleration. Note that the first line in (11) corresponds to Newton's law describing the CoM motion of the overall system. The second line contains the equations of motion for the hip rotation. The internal motion of the multi-body system is described by the third line of (11).

Remark 2. *If all external forces act at the end effector frames $\mathcal{T}_i$, then $\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{ext}$ simplifies to*

$$\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{ext} = \sum_{i=1}^{\Psi} \begin{bmatrix} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \hat{\boldsymbol{x}}_{ci} & \boldsymbol{I} \end{bmatrix} \\ \boldsymbol{J}_i^{\mathrm{T}} \end{bmatrix} \boldsymbol{F}_i = \begin{bmatrix} \tilde{\boldsymbol{G}} \\ \boldsymbol{J}^{\mathrm{T}} \end{bmatrix} \boldsymbol{F} \tag{12}$$

with $\boldsymbol{x}_{ci} = \boldsymbol{x}_i - \boldsymbol{x}_c$, and a Jacobian matrix $\boldsymbol{J}_i = \bar{\boldsymbol{J}}_i - [\, \boldsymbol{J}_{bc} \;\; \boldsymbol{0} \,] \in \mathbb{R}^{6\times n}$ explicitly computed in Appendix B. According to (7) *we will call $\tilde{\boldsymbol{G}}$ the 'rotated grasp matrix', since it provides a mapping between the set of end effector wrenches $\boldsymbol{F} = (\boldsymbol{F}_1^{\mathrm{T}} \cdots \boldsymbol{F}_\Psi^{\mathrm{T}})^{\mathrm{T}}$ and the total wrench at the CoM. The Jacobian matrices $\boldsymbol{J}_i$ are combined into $\boldsymbol{J} = [\, \boldsymbol{J}_1^{\mathrm{T}} \cdots \boldsymbol{J}_\Psi^{\mathrm{T}} \,]^{\mathrm{T}}$.*

3. Controller design

3.1. Problem statement

The goal of the controller is to achieve compliant balancing for a humanoid robot utilizing multiple contacts, while ensuring the passivity of the overall system. The balancing is achieved by implementing a Cartesian impedance to stabilize the CoM location and the hip orientation relative to the world frame. The necessary contact wrenches are generated with a subset of the end effectors that are in contact with the environment. The remaining end effectors behave according to a Cartesian impedance relative to the world in order to allow an interaction of the robot with the environment. The interaction wrenches must be taken into account by the balancing task.

In preparation for the next section, we will introduce the following notation: the translational and rotational velocity of the CoM frame $\mathcal{C}$ are combined into the vector $\boldsymbol{v}_c = (\dot{\boldsymbol{x}}_c^{\mathrm{T}} \; \boldsymbol{\omega}_c^{\mathrm{T}})^{\mathrm{T}}$, while the translational and rotational velocity of the i^{th} end effector are stacked into $\boldsymbol{v}_i = (\dot{\boldsymbol{x}}_i^{\mathrm{T}} \; \boldsymbol{\omega}_i^{\mathrm{T}})^{\mathrm{T}}$. The end effectors 1 to ψ are actively used for balancing by generating suitable contact wrenches, while the end effectors $\psi+1$ to Ψ are used for interacting with the environment. To clearly distinguish them, the velocities $\boldsymbol{v}_i$ are combined into $\boldsymbol{v}_{\text{bal}} = \left(\boldsymbol{v}_1^{\mathrm{T}} \cdots \boldsymbol{v}_\psi^{\mathrm{T}}\right)^{\mathrm{T}}$ and $\boldsymbol{v}_{\text{int}} = \left(\boldsymbol{v}_{\psi+1}^{\mathrm{T}} \cdots \boldsymbol{v}_\Psi^{\mathrm{T}}\right)^{\mathrm{T}}$, which together form the vector of end effector velocities $\boldsymbol{v} = \left(\boldsymbol{v}_{\text{bal}}^{\mathrm{T}} \; \boldsymbol{v}_{\text{int}}^{\mathrm{T}}\right)^{\mathrm{T}}$. Analogously, the vector of end effector wrenches $\boldsymbol{F}$ is partitioned into $\boldsymbol{F} = \left(\boldsymbol{F}_{\text{bal}}^{\mathrm{T}} \; \boldsymbol{F}_{\text{int}}^{\mathrm{T}}\right)^{\mathrm{T}}$ with $\boldsymbol{F}_{\text{bal}} = \left(\boldsymbol{F}_1^{\mathrm{T}} \cdots \boldsymbol{F}_\psi^{\mathrm{T}}\right)^{\mathrm{T}}$ and $\boldsymbol{F}_{\text{int}} = \left(\boldsymbol{F}_{\psi+1}^{\mathrm{T}} \cdots \boldsymbol{F}_\Psi^{\mathrm{T}}\right)^{\mathrm{T}}$. This notation only introduces a convenient numbering for the end effectors, without any loss of generality.

Furthermore, we will make the following assumptions:

Assumption 1. *The end effectors 1 to ψ are assumed to be in rigid contact with the environment, leading to $\boldsymbol{v}_{bal} = \boldsymbol{0}$.*

Assumption 2. *The Jacobian matrix $\boldsymbol{J}$ is assumed to be square ($n = 6\Psi$) and invertible.*

Assumption 2 will be dropped in Section 3.3, which presents a modification of the controller for dealing with redundant robots and singular configurations. Moreover, Section 5.4 reports a successful experiment in which Assumption 1 is deliberately violated.

3.2. Controller derivation

The balancing controller derived in this section is capable of stabilizing a fixed equilibrium point as well as following a given trajectory. In both cases the controller requires information (signals) on the desired CoM location $\boldsymbol{x}_c^d \in \mathbb{R}^3$, the desired hip orientation $\boldsymbol{R}_c^d \in \mathcal{SO}(3)$, and the desired position $\boldsymbol{x}_i^d \in \mathbb{R}^3$ and orientation $\boldsymbol{R}_i^d \in \mathcal{SO}(3)$ of each end effector. Furthermore, the balancing controller also expects a desired set of end effector wrenches $\boldsymbol{F}^d \in \mathbb{R}^{6\Psi}$. Such information can be generated for example by a planner as shown in Bouyarmane and Kheddar (2011). It is not mandatory for the end effector wrenches to be consistent with the other signals, since the controller can compensate for any discrepancy. In the regulation case, all of the signals required can be specified by hand, while in the tracking case an offline planner can be used for generating the desired trajectory.

The controller consists of two different components (see Figure 5). First, the multi-contact balancing control stabilizes the CoM location, the hip orientation and the Cartesian pose of each end effector. To achieve this, several Cartesian compliances generate a suitable set of wrenches, which are distributed to the end effectors and mapped to the joint torques. The second component is a null space control that stabilizes the redundant motion of the robot by generating suitable torques that are projected into the null space. The torques of both components are added and applied to the robot.

This section presents the multi-contact balancing control in two parts. First, the model in (11) is transformed into task space in order to define the desired behavior of the closed-loop system. Second, the controller is derived by exploiting the structure of the resulting equation for computing the control torques $\boldsymbol{\tau}$.

In order to obtain a model representation offering all quantities used for control, the joint coordinates used in the model (11) are replaced with the Cartesian coordinates of the end effectors. The corresponding transformation $\boldsymbol{T}$ is defined by

$$\begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v} \end{pmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \tilde{\boldsymbol{G}}^{\mathrm{T}} & \boldsymbol{J} \end{bmatrix}}_{\boldsymbol{T}} \begin{pmatrix} \boldsymbol{v}_c \\ \dot{\boldsymbol{q}} \end{pmatrix}. \tag{13}$$

The inverse transformation

$$\begin{pmatrix} \boldsymbol{v}_c \\ \dot{\boldsymbol{q}} \end{pmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ -\boldsymbol{J}^{-1}\tilde{\boldsymbol{G}}^{\mathrm{T}} & \boldsymbol{J}^{-1} \end{bmatrix}}_{\boldsymbol{T}^{-1}} \begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v} \end{pmatrix} \tag{14}$$

is derived from (13) by assuming that the Jacobian matrix $\boldsymbol{J}$ is invertible (see Assumption 2). Considerations on redundant systems and singular configurations will be explained in Section 3.3. Applying (14) to (11) leads to the transformed model

$$\boldsymbol{\Lambda} \begin{pmatrix} \dot{\boldsymbol{v}}_c \\ \dot{\boldsymbol{v}} \end{pmatrix} + \boldsymbol{\mu} \begin{pmatrix} \boldsymbol{v}_c \\ \boldsymbol{v} \end{pmatrix} + \boldsymbol{g} = \begin{bmatrix} -\tilde{\boldsymbol{G}} \\ \boldsymbol{I} \end{bmatrix} \boldsymbol{J}^{-\mathrm{T}}\boldsymbol{\tau} + \boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{\text{ext}} \tag{15}$$

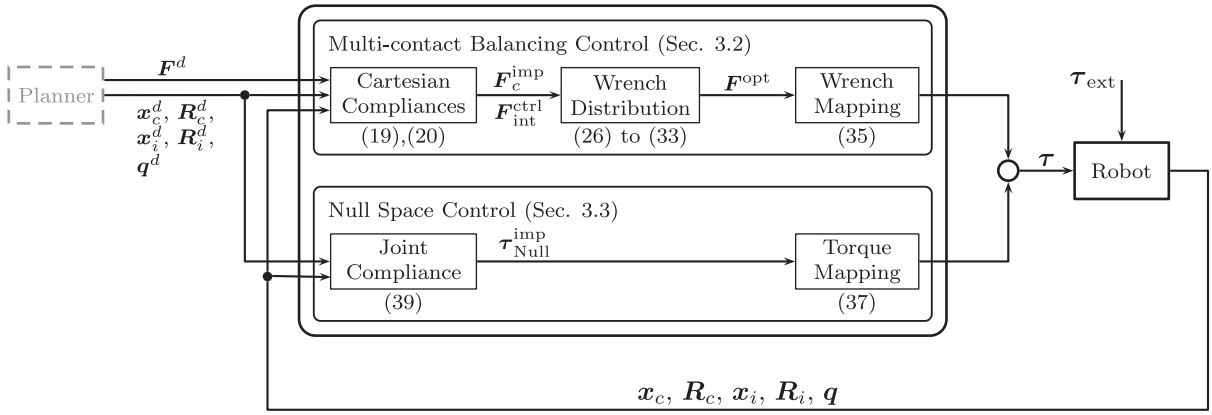


Fig. 5. Overview of the control architecture.

with $\boldsymbol{\Lambda} = \boldsymbol{T}^{-\mathrm{T}}\boldsymbol{M}\boldsymbol{T}^{-1}$ and $\boldsymbol{\mu} = \boldsymbol{T}^{-\mathrm{T}}\boldsymbol{C}\boldsymbol{T}^{-1} + \boldsymbol{T}^{-\mathrm{T}}\boldsymbol{M}\frac{d}{dt}(\boldsymbol{T}^{-1})$.

Remark 3. *If all external forces act at the end effector frames $\mathcal{T}_i$, then $\boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{ext}$ simplifies to*

$$\boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{ext} = \begin{bmatrix}\boldsymbol{0}\\ \boldsymbol{I}\end{bmatrix}\boldsymbol{F}. \tag{16}$$

Note the duality between the structure of the model in configuration space (11) *and in Cartesian coordinates* (15)*. In configuration space, the torque $\boldsymbol{\tau}$ only acts on the last line of* (11) *while the end effector wrenches influence $\boldsymbol{F}$ all three lines of* (11) *(see Remark 2). Utilizing the Cartesian velocities $\boldsymbol{v}$ of the end effectors leads to an opposite structure, with $\boldsymbol{\tau}$ influencing all lines of* (15) *and $\boldsymbol{F}$ acting only on the last line of* (15)*, as shown in* (16)*.*

Let us introduce

$$\begin{aligned}\Delta\boldsymbol{x}_c &= \boldsymbol{x}_c - \boldsymbol{x}_c^d, & \Delta\boldsymbol{x}_i &= \boldsymbol{x}_i - \boldsymbol{x}_i^d,\\ \Delta\boldsymbol{R}_c &= (\boldsymbol{R}_c^d)^{\mathrm{T}}\boldsymbol{R}_c, & \Delta\boldsymbol{R}_i &= (\boldsymbol{R}_i^d)^{\mathrm{T}}\boldsymbol{R}_i,\\ \Delta\boldsymbol{v}_c &= \boldsymbol{v}_c - \boldsymbol{v}_c^d, & \Delta\boldsymbol{v}_i &= \boldsymbol{v}_i - \boldsymbol{v}_i^d\end{aligned} \tag{17}$$

as deviations in velocity, position and orientation from the predefined trajectory. Therefore, we choose the closed-loop system behavior to be

$$\boldsymbol{\Lambda}\begin{pmatrix}\Delta\dot{\boldsymbol{v}}_c\\ \Delta\dot{\boldsymbol{v}}_{\text{bal}}\\ \Delta\dot{\boldsymbol{v}}_{\text{int}}\end{pmatrix} + \boldsymbol{\mu}\begin{pmatrix}\Delta\boldsymbol{v}_c\\ \Delta\boldsymbol{v}_{\text{bal}}\\ \Delta\boldsymbol{v}_{\text{int}}\end{pmatrix} = \boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{\text{ext}} - \begin{pmatrix}\boldsymbol{F}_c^{\text{imp}}\\ \boldsymbol{F}_{\text{bal}}^{\text{ctrl}}\\ \boldsymbol{F}_{\text{int}}^{\text{ctrl}}\end{pmatrix} \tag{18}$$

with

$$\boldsymbol{F}_c^{\text{imp}} = \begin{pmatrix}\boldsymbol{K}_c\Delta\boldsymbol{x}_c + \boldsymbol{D}_c\Delta\dot{\boldsymbol{x}}_c\\ -\boldsymbol{\tau}_r(\boldsymbol{\Sigma}_c, \Delta\boldsymbol{R}_c) + \boldsymbol{B}_c\Delta\boldsymbol{\omega}_c\end{pmatrix} \tag{19}$$

and

$$\boldsymbol{F}_{\text{int}}^{\text{ctrl}} = \underbrace{\begin{pmatrix}\boldsymbol{K}_{\psi+1}\Delta\boldsymbol{x}_{\psi+1} + \boldsymbol{D}_{\psi+1}\Delta\dot{\boldsymbol{x}}_{\psi+1}\\ -\boldsymbol{\tau}_r(\boldsymbol{\Sigma}_{\psi+1}, \Delta\boldsymbol{R}_{\psi+1}) + \boldsymbol{B}_{\psi+1}\Delta\boldsymbol{\omega}_{\psi+1}\\ \vdots\\ \boldsymbol{K}_{\Psi}\Delta\boldsymbol{x}_{\Psi} + \boldsymbol{D}_{\Psi}\Delta\dot{\boldsymbol{x}}_{\Psi}\\ -\boldsymbol{\tau}_r(\boldsymbol{\Sigma}_{\Psi}, \Delta\boldsymbol{R}_{\Psi}) + \boldsymbol{B}_{\Psi}\Delta\boldsymbol{\omega}_{\Psi}\end{pmatrix}}_{\boldsymbol{F}_{\text{int}}^{\text{imp}}} + \boldsymbol{F}_{\text{int}}^{d}. \tag{20}$$

The left-hand side of (18) represents the deviation of the robot from the desired trajectory, while the right-hand side describes a difference in the external wrenches. Here, the influence of the external forces that really act on the system is given by $\boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{\text{ext}}$. The wrenches that the controller intends to generate are represented by $\boldsymbol{F}_c^{\text{imp}}$, and $\boldsymbol{F}^{\text{ctrl}} = \left((\boldsymbol{F}_{\text{bal}}^{\text{ctrl}})^{\mathrm{T}}\ (\boldsymbol{F}_{\text{int}}^{\text{ctrl}})^{\mathrm{T}}\right)^{\mathrm{T}}$.

The translation of the CoM is controlled by a linear spring–damper system, with symmetric and positive definite stiffness and damping matrices $\boldsymbol{K}_c > 0$ and $\boldsymbol{D}_c > 0$. The orientation of the hip is controlled by a torque $\boldsymbol{\tau}_r(\boldsymbol{\Sigma}_c, \Delta\boldsymbol{R}_c)$ emulating a rotational spring as defined by (46) in Appendix C, in combination with a symmetric and positive-definite damping matrix $\boldsymbol{B}_c > 0$.

The end effectors that are actively used for balancing ($i = 1 \ldots \psi$) are assumed to be in rigid contact according to Assumption 1, leading to $\Delta\dot{\boldsymbol{v}}_{\text{bal}} = \Delta\boldsymbol{v}_{\text{bal}} = \boldsymbol{0}$. The contact wrenches that the controller intends to generate are represented by $\boldsymbol{F}_{\text{bal}}^{\text{ctrl}}$ and will be determined later on. The remaining end effectors ($i = \psi + 1, \ldots, \Psi$), which are used for interaction with the environment, are controlled to behave like a virtual impedance for stabilizing the pose of each end effector. The impedance consists of a linear spring $\boldsymbol{K}_i > 0$, a linear damper $\boldsymbol{D}_i > 0$, a rotational spring $\boldsymbol{\tau}_r(\boldsymbol{\Sigma}_i, \Delta\boldsymbol{R}_i)$ and a rotational damper $\boldsymbol{B}_i > 0$. The matrices $\boldsymbol{K}_i$, $\boldsymbol{D}_i$ and $\boldsymbol{B}_i$ are symmetric and positive definite, while $\boldsymbol{\tau}_r(\boldsymbol{\Sigma}_i, \Delta\boldsymbol{R}_i)$ is given by (46) in Appendix C. The vector $\boldsymbol{F}_{\text{int}}^{d}$ denotes a desired wrench that the robot should generate as part of the interaction task. It is provided by the planner shown in Figure 5 as part of $\boldsymbol{F}^d = \left((\boldsymbol{F}_{\text{bal}}^{d})^{\mathrm{T}}\ (\boldsymbol{F}_{\text{int}}^{d})^{\mathrm{T}}\right)^{\mathrm{T}}$. The stability of the closed-loop system given by (18) is discussed in Section 3.4. Note that the structure of (18) is inspired by a PD+ control of a robotic manipulator, which is well known in control theory (Paden and Panja, 1988). The goal of the presented balancing controller is to deliberately admit external disturbances taken into account by $\boldsymbol{\tau}_{\text{ext}}$ and react to them in a compliant way, instead of compensating for them. The feedforward terms in (18), represented by $\left((\dot{\boldsymbol{v}}_c^d)^{\mathrm{T}}\ (\dot{\boldsymbol{v}}^d)^{\mathrm{T}}\right)^{\mathrm{T}}$ and $\left((\boldsymbol{v}_c^d)^{\mathrm{T}}\ (\boldsymbol{v}^d)^{\mathrm{T}}\right)^{\mathrm{T}}$, allow the robot to follow a desired trajectory as it is intended by a PD+ control.

Remark 4. *Let us consider the static case of* (18)*. If all external forces act at the end effector frames $\mathcal{T}_i$, then* (18) *simplifies to*

$$\mathbf{0} = \begin{pmatrix} \mathbf{0} \\ \boldsymbol{F}_{bal} \\ \boldsymbol{F}_{int} \end{pmatrix} - \begin{pmatrix} \boldsymbol{F}_{c}^{imp} \\ \boldsymbol{F}_{bal}^{ctrl} \\ \boldsymbol{F}_{int}^{ctrl} \end{pmatrix} \tag{21}$$

with the end effector wrenches $\boldsymbol{F} = (\boldsymbol{F}_{bal}^{\mathrm{T}}\ \boldsymbol{F}_{int}^{\mathrm{T}})^{\mathrm{T}}$ (see Remark 3). This can be used to determine the static deviations $\Delta\boldsymbol{x}_i$ and $\Delta\boldsymbol{R}_i$ for the interaction end effectors $\psi+1$ to Ψ. The static deviation for the CoM location results in $\Delta\boldsymbol{x}_c = \mathbf{0}$ and in $\Delta\boldsymbol{R}_c = \boldsymbol{I}$ for the hip orientation. Furthermore, $\boldsymbol{F}_{bal}$ is equal to $\boldsymbol{F}_{bal}^{ctrl}$.

By comparing (18) with (15), the required control torque $\boldsymbol{\tau}$ must satisfy

$$\begin{bmatrix} -\tilde{\boldsymbol{G}} \\ \boldsymbol{I} \end{bmatrix} \boldsymbol{J}^{-\mathrm{T}}\boldsymbol{\tau} = \boldsymbol{\Lambda}\begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} + \boldsymbol{\mu}\begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix} + \boldsymbol{g} - \begin{pmatrix} \boldsymbol{F}_c^{\mathrm{imp}} \\ \boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}} \\ \boldsymbol{F}_{\mathrm{int}}^{\mathrm{ctrl}} \end{pmatrix}. \tag{22}$$

Note that $\boldsymbol{\tau}$ and $\boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}}$ are the remaining free variables in (22), which can be determined within two steps exploiting the structure of (22). First, the commanded wrenches $\boldsymbol{F}_c^{\mathrm{imp}}, \boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}}$ and $\boldsymbol{F}_{\mathrm{int}}^{\mathrm{ctrl}}$ are computed representing the external load condition of the robot. Second, the control torque $\boldsymbol{\tau}$ is computed characterizing the internal load inside the robot structure.

To obtain $\boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}}$, equation (22) is partitioned into

$$-\tilde{\boldsymbol{G}}(\boldsymbol{J}^{-\mathrm{T}}\boldsymbol{\tau}) = \boldsymbol{\Lambda}_1\begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} + \boldsymbol{\mu}_1\begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix} + \begin{pmatrix} m\boldsymbol{g}_0 \\ \mathbf{0} \end{pmatrix} - \boldsymbol{F}_c^{\mathrm{imp}} \tag{23}$$

$$(\boldsymbol{J}^{-\mathrm{T}}\boldsymbol{\tau}) = \boldsymbol{\Lambda}_2\begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} + \boldsymbol{\mu}_2\begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix} - \begin{pmatrix} \boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}} \\ \boldsymbol{F}_{\mathrm{int}}^{\mathrm{ctrl}} \end{pmatrix} \tag{24}$$

based on $\boldsymbol{\Lambda} = [\,\boldsymbol{\Lambda}_1^{\mathrm{T}}\ \boldsymbol{\Lambda}_2^{\mathrm{T}}\,]^{\mathrm{T}}$ and $\boldsymbol{\mu} = [\,\boldsymbol{\mu}_1^{\mathrm{T}}\ \boldsymbol{\mu}_2^{\mathrm{T}}\,]^{\mathrm{T}}$ with $\boldsymbol{\Lambda}_1, \boldsymbol{\mu}_1 \in \mathbb{R}^{6\times 6+6\Psi}$ and $\boldsymbol{\Lambda}_2, \boldsymbol{\mu}_2 \in \mathbb{R}^{6\Psi\times 6+6\Psi}$. Eliminating the common variable $\boldsymbol{J}^{-\mathrm{T}}\boldsymbol{\tau}$ yields to

$$\tilde{\boldsymbol{G}}\begin{pmatrix} \boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}} \\ \boldsymbol{F}_{\mathrm{int}}^{\mathrm{ctrl}} \end{pmatrix} = \underbrace{(\boldsymbol{\Lambda}_1 + \tilde{\boldsymbol{G}}\boldsymbol{\Lambda}_2)\begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} + (\boldsymbol{\mu}_1 + \tilde{\boldsymbol{G}}\boldsymbol{\mu}_2)\begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix}}_{\text{feedforward}} + \underbrace{\begin{pmatrix} m\boldsymbol{g}_0 \\ \mathbf{0} \end{pmatrix}}_{\text{gravity compensation}} - \underbrace{\boldsymbol{F}_c^{\mathrm{imp}}}_{\text{feedback}}. \tag{25}$$

The right-hand side of (25) can be interpreted as the overall wrench that the controller needs to generate at the CoM in order to maintain balance. It consists of a feedforward part, a gravity compensation and a feedback term as in PD+ control of a single manipulator (Paden and Panja, 1988). The left-hand side of (25) is given by the end effector wrenches, which must add up to the desired overall wrench at the CoM.

Since the rotated grasp matrix $\tilde{\boldsymbol{G}}$ in (25) has 6 lines and $\boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}}$ a size of 6ψ, $\boldsymbol{F}_{\mathrm{bal}}^{\mathrm{ctrl}}$ cannot be directly obtained if there is more than one end effector used for balancing ($\psi > 1$). In order to resolve this force distribution problem caused by the redundancy in $\tilde{\boldsymbol{G}}$, the following constrained quadratic optimization problem is used:

$$\min_{\boldsymbol{F}^{\mathrm{opt}}} \left(\frac{1}{2}\boldsymbol{\delta}_{\mathrm{bal}}^{\mathrm{T}}\boldsymbol{Q}_{\mathrm{bal}}\boldsymbol{\delta}_{\mathrm{bal}} + \frac{1}{2}\boldsymbol{\delta}_{\mathrm{int}}^{\mathrm{T}}\boldsymbol{Q}_{\mathrm{int}}\boldsymbol{\delta}_{\mathrm{int}} + \frac{1}{2}\boldsymbol{\delta}_c^{\mathrm{T}}\boldsymbol{Q}_c\boldsymbol{\delta}_c \right) \tag{26}$$

with the residua

$$\boldsymbol{\delta}_{\mathrm{bal}} = \boldsymbol{F}_{\mathrm{bal}}^{\mathrm{opt}} - \boldsymbol{F}_{\mathrm{bal}}^{d} \tag{27}$$

$$\boldsymbol{\delta}_{\mathrm{int}} = \boldsymbol{F}_{\mathrm{int}}^{\mathrm{opt}} - \boldsymbol{F}_{\mathrm{int}}^{\mathrm{ctrl}} \tag{28}$$

$$\boldsymbol{\delta}_c = \tilde{\boldsymbol{G}}\begin{pmatrix} \boldsymbol{F}_{\mathrm{bal}}^{\mathrm{opt}} \\ \boldsymbol{F}_{\mathrm{int}}^{\mathrm{opt}} \end{pmatrix} - (\boldsymbol{\Lambda}_1 + \tilde{\boldsymbol{G}}\boldsymbol{\Lambda}_2)\begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} - (\boldsymbol{\mu}_1 + \tilde{\boldsymbol{G}}\boldsymbol{\mu}_2)\begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix} - \begin{pmatrix} m\boldsymbol{g}_0 \\ \mathbf{0} \end{pmatrix} + \boldsymbol{F}_c^{\mathrm{imp}} \tag{29}$$

and with respect to the constraints

$$f_{i,\perp}^{\mathrm{opt}} \geq f_i^{\min} \qquad \forall i = 1 \ldots \psi \tag{30}$$

$$\left\| \boldsymbol{f}_{i,\parallel}^{\mathrm{opt}} \right\| \leq \mu_i \left| f_{i,\perp}^{\mathrm{opt}} \right| \qquad \forall i = 1 \ldots \psi \tag{31}$$

$$\boldsymbol{p}(\boldsymbol{F}_i^{\mathrm{opt}}) \in \mathcal{S}_i \qquad \forall i = 1 \ldots \psi \tag{32}$$

$$\boldsymbol{\tau}^{\max} \geq \left| \boldsymbol{J}^{\mathrm{T}}\left(\boldsymbol{\Lambda}_2\begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} + \boldsymbol{\mu}_2\begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix} - \boldsymbol{F}^{\mathrm{opt}} \right) \right|. \tag{33}$$

The optimization tries to find an optimal set of end effector wrenches $\boldsymbol{F}^{\mathrm{opt}} = \left((\boldsymbol{F}_{\mathrm{bal}}^{\mathrm{opt}})^{\mathrm{T}}\ (\boldsymbol{F}_{\mathrm{int}}^{\mathrm{opt}})^{\mathrm{T}} \right)^{\mathrm{T}}$ based on the choice of the weighting matrices $\boldsymbol{Q}_c$, $\boldsymbol{Q}_{\mathrm{bal}}$ and $\boldsymbol{Q}_{\mathrm{int}} > 0$, which need to be symmetric and positive definite. By choosing the weights $\boldsymbol{Q}_c, \boldsymbol{Q}_{\mathrm{int}} >> \boldsymbol{Q}_{\mathrm{bal}}$, the optimization problem corresponds to a non-strict hierarchy consisting of three different tasks.

The task with the highest priority is given by the inequality constraints (30) to (33). Here, equations (30) to (32) represent the contact model and ensure that Assumption 1 ($\dot{\boldsymbol{v}}_{\mathrm{bal}} = \boldsymbol{v}_{\mathrm{bal}} = \mathbf{0}$) holds. Let $f_{i,\perp}^{\mathrm{opt}}$ be the projection of the contact force $\boldsymbol{f}_i^{\mathrm{opt}}$ onto the normal of the contact surface $\mathcal{S}_i$, while $\boldsymbol{f}_{i,\parallel}^{\mathrm{opt}}$ denotes the projection of $\boldsymbol{f}_i^{\mathrm{opt}}$ into the tangential plane to $\mathcal{S}_i$. Then, the constraint (30) demands that the end effector always pushes with a minimal contact force $f_i^{\min} \geq 0$ against the contact surface, thus preventing the end effector from lifting off. The constraint (31) ensures no slippage by limiting $\boldsymbol{f}_{i,\parallel}^{\mathrm{opt}}$. Friction is characterized by the friction coefficient μ_i. The inequality constraint (32) prevents the end effector from tilting by restricting the location of the CoP $\boldsymbol{p}(\boldsymbol{F}_i^{\mathrm{opt}}) \in \mathbb{R}^3$ at each end effector to be inside its corresponding contact surface $\mathcal{S}_i$. The position of the CoP can be obtained from

$$\boldsymbol{\tau}_{i,\parallel}^{\mathrm{opt}} + \boldsymbol{p}(\boldsymbol{F}_i^{\mathrm{opt}}) \times \boldsymbol{f}_{i,\parallel}^{\mathrm{opt}} = \mathbf{0}. \tag{34}$$

The constraints (30) to (32) can be partially dropped depending on the model of the contact. The inequality constraint (33) is obtained from (24), and limits the joint torques to a maximum value of $\boldsymbol{\tau}^{\max}$.

The task with the medium priority tries to fulfill the two soft constraints in form of minimizing the residua

(28) and (29). The first one implements the impedance given by $\boldsymbol{F}_{\text{int}}^{\text{ctrl}}$ acting on the interaction end effectors. The second one demands that the end effector wrenches sum up to the desired overall wrench at the CoM implementing the impedance $\boldsymbol{F}_{c}^{\text{imp}}$ (see (25)). Thus, we will call this task the 'impedance task' in the remainder of the paper.

The task with the lowest priority is given by the soft constraint (27). Here, $\boldsymbol{F}_{\text{bal}}^{d}$ denotes a desired wrench distribution for the end effectors 1 to ψ, provided by the offline planner shown in Figure 5 as a part of $\boldsymbol{F}^{d} = \left((\boldsymbol{F}_{\text{bal}}^{d})^{\text{T}} \ (\boldsymbol{F}_{\text{int}}^{d})^{\text{T}} \right)^{\text{T}}$. Thus, this task will be called 'wrench distribution task'.

The optimization problem is formulated such that the soft constraints (28) and (29) are automatically relaxed (online) in order to obtain a feasible solution $\boldsymbol{F}^{\text{opt}}$ even if the soft constraints and the inequality constraints (30) to (33) cannot be satisfied at the same time. In that case, the desired wrench at the CoM and at the interaction end effectors cannot be completely generated. Thus, the robot is no longer subjected to the impedances $\boldsymbol{F}_{c}^{\text{imp}}$ and $\boldsymbol{F}_{\text{int}}^{\text{ctrl}}$. However, this does not necessarily mean that the system becomes unstable, because the portion of the wrenches that can still be generated might be enough to stabilize the CoM and the interaction end effectors. Note that one can decide which soft constraint should be relaxed first by choosing the ratio between $\boldsymbol{Q}_{c}$ and $\boldsymbol{Q}_{\text{int}}$, which corresponds to dividing the impedance task into two sub-priority levels.

Remark 5. *It was assumed in Section 3.1 that the end effectors* 1 *to* ψ *are in rigid contact with the environment (Assumption 1). Note that the constraints* (30) *to* (32) *are imposed on the commanded contact wrenches* $\boldsymbol{F}_{bal}^{opt}$ *and not on the contact wrenches that really act on the system. In consequence, it can happen that the real contact wrenches* $\boldsymbol{F}_{bal}$ *violate the constraints* (30) *to* (32)*, leading to an end effector that lifts off, slides or tilts. But this effect can be neglected if the tracking error and the external disturbances are sufficiently small, as seen in* (18)*.*

Note that the addition of feedforward terms to the balancing controller is an extension of the work in Ott et al. (2011). In (18)*, these feedforward terms help to reduce the discrepancy between the commanded and real end effector wrenches, and thereby increase the robustness of the approach in the tracking case. An experimental verification of this statement is presented in Section 5.3 by comparing the balancer performance with and without feedforward control.*

Remark 6. *It is also possible to apply the constraints* (30) *to* (32) *to the real contact wrenches* $\boldsymbol{F}_{bal}$ *instead of the commanded ones* $\boldsymbol{F}_{bal}^{opt}$*. For this, we can exploit the constraint* $\boldsymbol{v}_{bal} = \boldsymbol{0}$ *(Assumption 1) in order to obtain a relation between* $\boldsymbol{F}_{bal}$*, the system state, the control torque* $\boldsymbol{\tau}$ *and the other external wrenches. Since determining the external wrenches is not too practical, we do not apply the constraints on the real contact wrenches.*

After determining the wrench distribution $\boldsymbol{F}^{\text{opt}}$, equation (24) can be used for computing the control torque

$$\boldsymbol{\tau} = \boldsymbol{J}^{\text{T}} \left(\underbrace{\boldsymbol{\Lambda}_2 \begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} + \boldsymbol{\mu}_2 \begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix}}_{\text{feedforward}} - \boldsymbol{F}^{\text{opt}} \right). \tag{35}$$

Here, $\boldsymbol{\Lambda}_2 \begin{pmatrix} \dot{\boldsymbol{v}}_c^d \\ \dot{\boldsymbol{v}}^d \end{pmatrix} + \boldsymbol{\mu}_2 \begin{pmatrix} \boldsymbol{v}_c^d \\ \boldsymbol{v}^d \end{pmatrix}$ represent another part of the feedforward control action.

Remark 7. *It is also possible to formulate the optimization problem in terms of the control torque* $\boldsymbol{\tau}$ *instead of the end effector wrenches* $\boldsymbol{F}^{opt}$*. In that case, equation* (35) *can be used for replacing* $\boldsymbol{F}^{opt}$ *with* $\boldsymbol{\tau}$ *in (26) to (33).*

3.3. Redundant systems and singular configurations

In Section 3.2, $\boldsymbol{J}$ is assumed to be square and invertible (Assumption 2) in order to perform the coordinate transformation (14) leading to $\boldsymbol{\Lambda}$ and $\boldsymbol{\mu}$. If the robot is in a singular configuration then $\boldsymbol{\Lambda}$ and $\boldsymbol{\mu}$ will tend to infinity, while in the case of a redundant robot, $\boldsymbol{\Lambda}$ and $\boldsymbol{\mu}$ are no longer defined. In any case, for dealing with singular configurations and redundant robots, a generalized damped pseudo-inverse can be used instead of $\boldsymbol{J}^{-1}$,

$$\boldsymbol{J}^{\dagger} = (\boldsymbol{J}^{\text{T}}\boldsymbol{J} + \rho^2 \boldsymbol{I})^{-1} \boldsymbol{J}^{\text{T}} \tag{36}$$

with $\rho \in \mathbb{R}$ as damping factor.

Since the controller regulates the configuration of the robot in task coordinates, there might exist a null space within the configuration space of the robot. In order to prevent a drift within this null space, we add a conventional null space controller

$$\boldsymbol{\tau}_{\text{Null}} = \boldsymbol{\tau} + \left(\boldsymbol{I} - \boldsymbol{J}^{\text{T}} (\boldsymbol{J}^{\text{T}})^{\ddagger} \right) \boldsymbol{\tau}_{\text{Null}}^{\text{imp}} \tag{37}$$

with

$$(\boldsymbol{J}^{\text{T}})^{\ddagger} = \left(\boldsymbol{J}\boldsymbol{M}_{22}^{-1}\boldsymbol{J}^{\text{T}} \right)^{-1} \boldsymbol{J}\boldsymbol{M}_{22}^{-1} \tag{38}$$

and

$$\boldsymbol{\tau}_{\text{Null}}^{\text{imp}} = -\boldsymbol{K}_{\text{Null}} \left(\boldsymbol{q} - \boldsymbol{q}_0 \right) - \boldsymbol{D}_{\text{Null}} \dot{\boldsymbol{q}}. \tag{39}$$

The torque $\boldsymbol{\tau}_{\text{Null}}^{\text{imp}}$ is provided by a PD-controller with symmetric and positive definite stiffness and damping matrices $\boldsymbol{K}_{\text{Null}} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{D}_{\text{Null}} \in \mathbb{R}^{n \times n}$. Then, $\boldsymbol{\tau}_{\text{Null}}^{\text{imp}}$ is projected into the null space of a dynamically consistent damped pseudo-inverse $(\boldsymbol{J}^{\text{T}})^{\ddagger}$. The joint position and velocities $\boldsymbol{q}_{\text{ref}}$ and $\dot{\boldsymbol{q}}_{\text{ref}}$ are computed from the trajectory given by $\boldsymbol{x}_c^d$, $\boldsymbol{R}_c^d$, $\boldsymbol{x}_i^d$, $\boldsymbol{R}_i^d$ and $\dot{\boldsymbol{x}}_c^d$, $\boldsymbol{\omega}_c^d$, $\dot{\boldsymbol{x}}_i^d$, $\boldsymbol{\omega}_i^d$.

The choice of the damping in $\boldsymbol{J}^{\dagger}$ is a compromise between robustness against singularities and the accuracy of the solution. From (25) and (35), one can note that this approximation only affects parts of the feedforward control

and not the feedback control. As a consequence, the tracking behavior of the overall controller might deteriorate if the robot is close to a singularity, but the feedback loop will still be capable of stabilizing the robot.

3.4. Characteristics of the controller: stability and passivity

The structure of the closed-loop system (18) is similar to the closed-loop dynamics of a PD+ controller, which is well studied in robot control theory (Paden and Panja, 1988). The advantage of a PD+ controller is that it allows precise tracking of the reference in free motion and results in a passive system for interaction with the environment. Moreover, it does not require a measurement of the external wrenches because it does not shape the inertia of the closed-loop system (Ott, 2008). Knowledge of the inertia matrix and the Coriolis terms are required only for the tracking terms, and not for the feedback part (see (25) and (35)).

Note that the following proof only considers non-redundant robots in non-singular configurations (see Assumption 1 and 2). For a general discussion on redundant robotic manipulators, please refer to the relevant literature (e.g. Ott, 2008). Furthermore, it is assumed that the soft constraints (28) and (29) are exactly satisfied ($\boldsymbol{\delta}_c = \mathbf{0}$ and $\boldsymbol{\delta}_{\text{int}} = \mathbf{0}$).

As a consequence of the PD+ structure of the presented controller, we can conclude the asymptotic stability and passivity of the closed-loop system.

Theorem 1 (Asymptotic stability)**.** *Let the soft constraints* (28) *and* (29) *be exactly satisfied and let Assumption 1 and 2 hold. Then, the closed-loop system* (18) *is asymptotically stable in case of a free motion characterized by* $\boldsymbol{F}_{gen} = (\boldsymbol{T}^*)^{\mathrm{T}}\,\boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{ext} - (\,\mathbf{0}\;(\boldsymbol{F}^d_{int})^{\mathrm{T}}\,)^{\mathrm{T}} = \mathbf{0}$.

Theorem 2 (Passivity)**.** *Let the soft constraints* (28) *and* (29) *be exactly satisfied and let Assumption 1 and 2 hold. Then,* $\boldsymbol{F}_{gen} = (\boldsymbol{T}^*)^{\mathrm{T}}\,\boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{ext} - (\,\mathbf{0}\;(\boldsymbol{F}^d_{int})^{\mathrm{T}}\,)^{\mathrm{T}}$ *and* $(\,\boldsymbol{v}_c^{\mathrm{T}}\;\boldsymbol{v}_{int}^{\mathrm{T}}\,)^{\mathrm{T}}$ *represent a passive port to the closed-loop system* (18) *for the regulation case given by* $\dot{\boldsymbol{v}}^d_c = \boldsymbol{v}^d_c = \mathbf{0}$ *and* $\dot{\boldsymbol{v}}^d = \boldsymbol{v}^d = \mathbf{0}$.

The proof of both aspects requires that Assumption 1 holds, demanding a rigid contact for the end effectors 1 to ψ, or in other words $\dot{\boldsymbol{v}}_{\text{bal}} = \boldsymbol{v}_{\text{bal}} = \mathbf{0}$. In consequence, (18) has only $6 + 6(\Psi - \psi)$ DoFs represented by $\boldsymbol{v}_c$ and $\boldsymbol{v}_{\text{int}}$. In order to reduce (18) to these remaining DoFs, we apply the transformation

$$\begin{pmatrix} \Delta\boldsymbol{v}_c \\ \Delta\boldsymbol{v}_{\text{bal}} \\ \Delta\boldsymbol{v}_{\text{int}} \end{pmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I} \end{bmatrix}}_{\boldsymbol{T}^*} \begin{pmatrix} \Delta\boldsymbol{v}_c \\ \Delta\boldsymbol{v}_{\text{int}} \end{pmatrix} \tag{40}$$

leading to the reduced closed-loop system behavior

$$\boldsymbol{\Lambda}^* \begin{pmatrix} \Delta\dot{\boldsymbol{v}}_c \\ \Delta\dot{\boldsymbol{v}}_{\text{int}} \end{pmatrix} + \boldsymbol{\mu}^* \begin{pmatrix} \Delta\boldsymbol{v}_c \\ \Delta\boldsymbol{v}_{\text{int}} \end{pmatrix} = (\boldsymbol{T}^*)^{\mathrm{T}}\,\boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{\text{ext}} - \begin{pmatrix} \boldsymbol{F}_c^{\text{imp}} \\ \boldsymbol{F}_{\text{int}}^{\text{opt}} \end{pmatrix} \tag{41}$$

$$\boldsymbol{\Lambda}^* \begin{pmatrix} \Delta\dot{\boldsymbol{v}}_c \\ \Delta\dot{\boldsymbol{v}}_{\text{int}} \end{pmatrix} + \boldsymbol{\mu}^* \begin{pmatrix} \Delta\boldsymbol{v}_c \\ \Delta\boldsymbol{v}_{\text{int}} \end{pmatrix} + \begin{pmatrix} \boldsymbol{F}_c^{\text{imp}} \\ \boldsymbol{F}_{\text{int}}^{\text{imp}} \end{pmatrix} = \underbrace{(\boldsymbol{T}^*)^{\mathrm{T}}\,\boldsymbol{T}^{-\mathrm{T}}\bar{\boldsymbol{T}}^{-\mathrm{T}}\boldsymbol{\tau}_{\text{ext}} - \begin{pmatrix} \mathbf{0} \\ \boldsymbol{F}_{\text{int}}^{d} \end{pmatrix}}_{\boldsymbol{F}_{\text{gen}}} \tag{42}$$

with $\boldsymbol{\Lambda}^* = (\boldsymbol{T}^*)^{\mathrm{T}}\,\boldsymbol{\Lambda}\boldsymbol{T}^*$ and $\boldsymbol{\mu}^* = (\boldsymbol{T}^*)^{\mathrm{T}}\,\boldsymbol{\mu}\boldsymbol{T}^*$. The right-hand side of (42) can be considered as a generalized wrench $\boldsymbol{F}_{\text{gen}}$. To show asymptotic stability, let us consider the case of a free motion with $\boldsymbol{F}_{\text{gen}} = \mathbf{0}$ leading to

$$\boldsymbol{\Lambda}^* \begin{pmatrix} \Delta\dot{\boldsymbol{v}}_c \\ \Delta\dot{\boldsymbol{v}}_{\text{int}} \end{pmatrix} + \boldsymbol{\mu}^* \begin{pmatrix} \Delta\boldsymbol{v}_c \\ \Delta\boldsymbol{v}_{\text{int}} \end{pmatrix} + \begin{pmatrix} \boldsymbol{F}_c^{\text{imp}} \\ \boldsymbol{F}_{\text{int}}^{\text{imp}} \end{pmatrix} = \mathbf{0}. \tag{43}$$

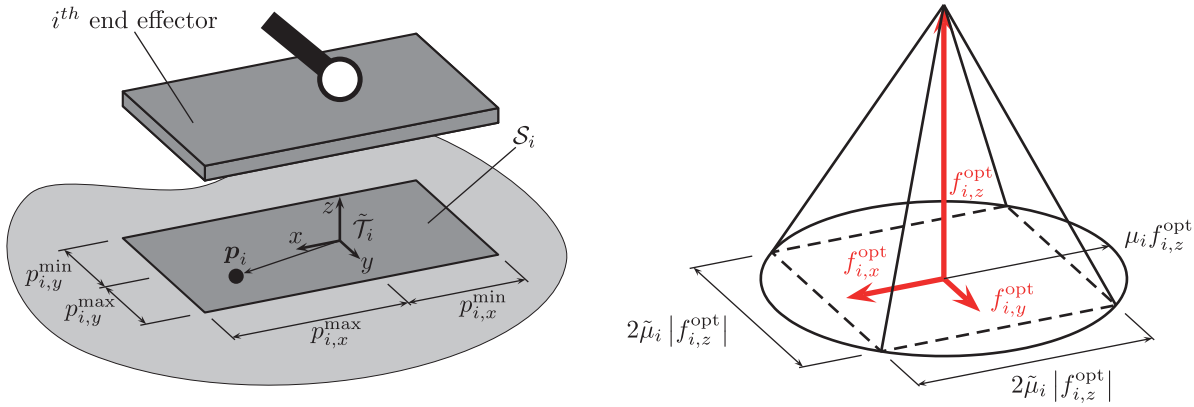
Since $\boldsymbol{\Lambda}^*$ and $\boldsymbol{\mu}^*$ depend on $\boldsymbol{x}_c$, $\boldsymbol{R}_c$, $\boldsymbol{q}$ and $\dot{\boldsymbol{x}}_c$, $\boldsymbol{\omega}_c$, $\dot{\boldsymbol{q}}$, the system (42) is time-varying. Asymptotic stability of (42) can be shown following the procedure presented in Paden and Panja (1988), where a time-varying Lyapunov function is used to prove asymptotic stability for a PD+ controller. The passivity of the reduced closed-loop system (42) can be tested following Ott (2008). The consideration of the regulation case ($\dot{\boldsymbol{v}}^d_c = \boldsymbol{v}^d_c = \mathbf{0}$ and $\dot{\boldsymbol{v}}^d = \boldsymbol{v}^d = \mathbf{0}$) leads to the conclusion that $\boldsymbol{F}_{\text{gen}}$ and $\left(\,\boldsymbol{v}_c^{\mathrm{T}}\;\boldsymbol{v}_{\text{int}}^{\mathrm{T}}\,\right)^{\mathrm{T}}$ represent a passive port to the reduced system (42).

Note that the above analysis applies to the desired closed-loop dynamics (18), i.e. assuming that the optimization problem allows a feasible solution in which the soft constraints (28) and (29) are exactly fulfilled. A stability analysis of the case where the soft constraints are not exactly fulfilled would be considerably more cumbersome; in this case, the constraints will have an influence on the quality of the task fulfillment according to their corresponding priorities.

4. Implementation

4.1. Simplification of the optimization problem

This section presents a formulation of the optimization problem stated in (26) to (33), which is less complex and therefore easier to implement on a real-time control system. First of all, we choose a simple geometric shape, a rectangle, to describe the contact surfaces $\mathcal{S}_i$, as illustrated in Figure 6a. For simplicity, the friction cone used in (31) is approximated by a four-sided pyramid (see Figure 6b), characterized by $\tilde{\mu}_i$. Furthermore, all quantities concerning $\mathcal{S}_i$ are expressed in a local end effector frame $\tilde{\mathcal{T}}_i$ instead of $\mathcal{T}_i$. The frame $\tilde{\mathcal{T}}_i$ is defined such that the z-axis is normal



(a) CoP for a contact surface.

(b) Convex pyramid inscribed in the friction cone.

Fig. 6. Simplifications for reducing the complexity of the optimization problem.

to $\mathcal{S}_i$ pointing away from the environment, while x and y coincide with the axes of symmetry of the rectangle.

In contrast to the previous section, the TCP impedance from (20) is activated for all end effectors, including those that are not actively involved in balancing. From an empirical point of view, this has proven to increase accuracy during the transition phase if an end effector is lifted and then repositioned again in contact with the environment. If an end effector is used for balancing, the influence of the TCP impedance is negligibly low anyway, since $\Delta \boldsymbol{x}_i \approx \boldsymbol{0}$ and $\Delta \boldsymbol{R}_i \approx \boldsymbol{0}$.

With the formulation presented in (26) to (33) the size of the optimization problem is constant at 6Ψ variables. Thus, it can be easily reconfigured for different multi-contact balancing conditions by choosing the parameters accordingly, without requiring a change in the structure of the optimization problem. This allows to change the task assignment of the end effectors online, for example for lifting and moving one end effector from one location to another. Furthermore, the parametrization can be changed continuously to avoid a hard jump of commanded forces or torques. This is beneficial in transient situations where one end effector is intentionally breaking or establishing contact, which requires this end effector to be moved from one level in the task hierarchy to another. So far we have only considered assigning one end effector to a certain task, but the formulation also allows assigning one task to each single DoF in the Cartesian space of the end effector. Another side effect of using (29) as a soft constraint instead of a hard constraint is that it reduces the computation time for iteratively solving the constrained quadratic optimization problem.[2]

4.2. Estimation of the state of the base frame

The performance of the balancing controller is directly linked to the quality of tracking the base frame $\mathcal{B}$ of the humanoid robot. The location $\boldsymbol{x}_c$ of the CoM can be computed from the position of the base frame $\boldsymbol{x}_b$, the kinematics and the inertia parameters of the robot. This section presents a method for estimating the state of the base frame by utilizing only internal sensors, namely the onboard inertial measurement unit (IMU) and the kinematic information of the robot. In contrast to Stelzer et al. (2012) and Bloesch et al. (2013) who proposed Kalman filters for state estimation, our method aims at a direct (lag-free) estimation of the base frame and its velocity from contact information, kinematics, and IMU measurements.

Let us assume that all end effectors ($i = 1 \ldots \Psi$) are in rigid contact with the environment (Assumption 1), leading to $\dot{\boldsymbol{v}} = \boldsymbol{v} = \boldsymbol{0}$, and that the poses of all end effectors given by $\boldsymbol{x}_i$ and $\boldsymbol{R}_i$ are well known with respect to the world frame $\mathcal{W}$. Then, each end effector can be used to compute exactly one estimation for the state of the base frame $\mathcal{B}$ evaluating $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ and the relevant $\boldsymbol{x}_i$ and $\boldsymbol{R}_i$. Each estimation consists of a value for $\boldsymbol{x}_b$, $\dot{\boldsymbol{x}}_b$, $\boldsymbol{R}_b$ and $\boldsymbol{\omega}_b$ representing the whole state of the base frame. An onboard IMU attached to the base of the robot provides another estimation for $\boldsymbol{R}_b$ and $\boldsymbol{\omega}_b$. In order to reduce the influence of uncertainties and noise in the estimation, the information coming from different data sources (end effectors and IMU) is averaged according to the scheme sketched in Figure 7. The method consists of four steps that sequentially compute $\boldsymbol{R}_b$, $\boldsymbol{\omega}_b$, $\boldsymbol{x}_b$ and $\dot{\boldsymbol{x}}_b$, in this order.

The computation of $\boldsymbol{R}_b$ consists of two steps. First, the estimates provided by the end effectors are averaged based on Appendix D. Here, $\alpha_i \in [0, 1]$, with $\sum_{i=1}^{\Psi} \alpha_i = 1$ denoting the weights that determine the influence of each end effector on the resulting mean orientation. The result is then combined with the base orientation estimated by the IMU. Since the IMU is operated in a rather noisy environment in terms of electromagnetic interferences, it can show a drift about the vertical axis of the world frame (when the Earth's magnetic field cannot be properly evaluated). In order to remove the drift, the estimation provided by the IMU and the average provided by the end effectors are both decomposed into two consecutive rotations.[3] The first, $\boldsymbol{R}_{\text{ver}}$, is a rotation about the vertical axis of the world frame $\mathcal{W}$.

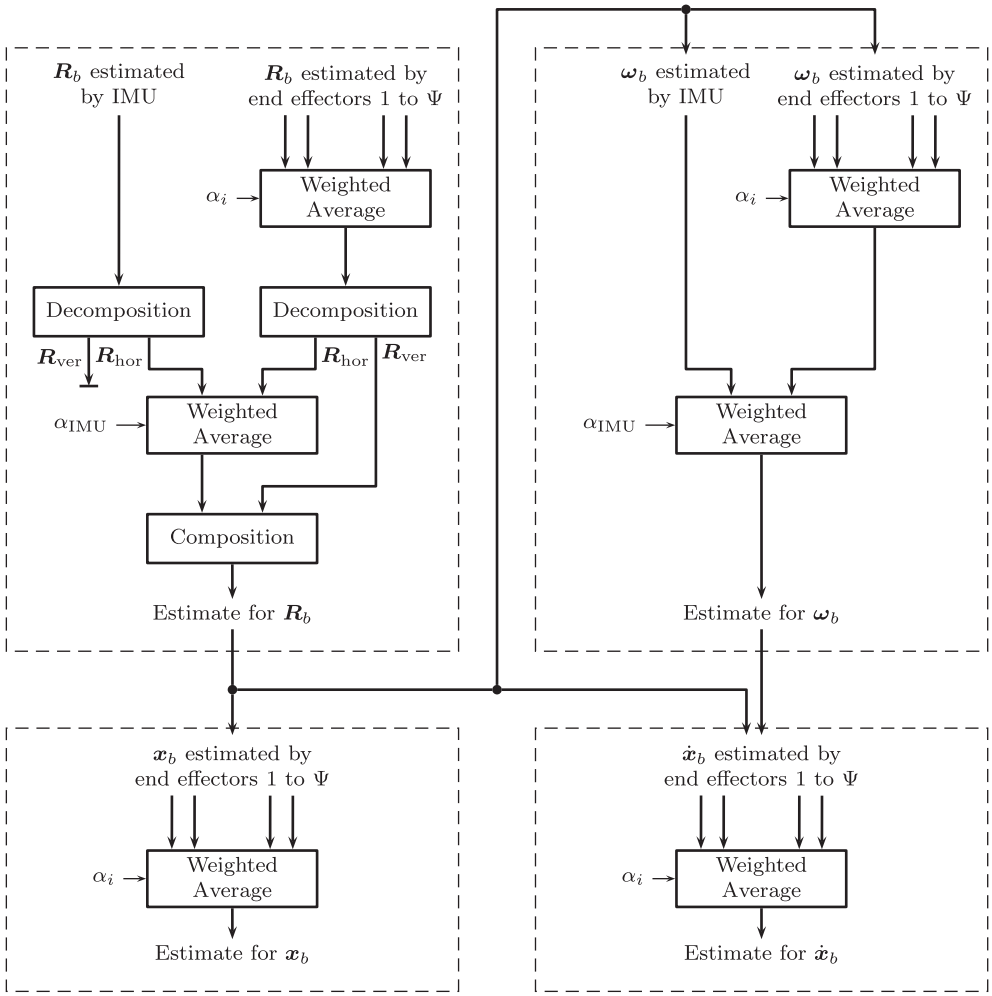


Fig. 7. Method used for estimating the state of the base frame.

The second, $\boldsymbol{R}_{\text{hor}}$, is a rotation about an axis lying in the horizontal plane of $\mathcal{W}$. The horizontal rotations are averaged based on the weight $\alpha_{\text{IMU}} \in [0, 1]$ that determines the importance given to the IMU data. Then, the horizontal rotation provided by the IMU and the one provided by the kinematics are averaged according to Appendix D. The result is composed with the vertical estimate provided by the kinematics, thus providing an estimated overall orientation $\boldsymbol{R}_b$ of the base frame. The vertical rotation provided by the IMU is not processed any further since it is affected by the drift.

The computation of $\boldsymbol{\omega}_b$ consists of two analogous stages. First, the estimates provided by the end effectors are averaged based on the weights α_i, which can be used to specify the influence of each end effector. Afterwards, the result is averaged with the estimation of $\boldsymbol{\omega}_b$ provided by the IMU using the weights α_{IMU} and $1 - \alpha_{\text{IMU}}$.

The computation of $\boldsymbol{x}_b$ and $\dot{\boldsymbol{x}}_b$ is much simpler, as the only relevant source of information is the kinematics. For both computations, the estimates are averaged based on the weights α_i.

Note that the computation must occur in the order $\boldsymbol{R}_b$, $\boldsymbol{\omega}_b$, $\boldsymbol{x}_b$ and $\dot{\boldsymbol{x}}_b$, as shown in Figure 7, because $\boldsymbol{R}_b$ is necessary for estimating $\boldsymbol{\omega}_b$, $\boldsymbol{x}_b$, and $\dot{\boldsymbol{x}}_b$. Also, the angular velocity $\boldsymbol{\omega}_b$ is required for estimating $\dot{\boldsymbol{x}}_b$.

By adapting the weights α_i and α_{IMU}, the algorithm can consider the reliability of each type of sensor information. If the algorithm should only use the information about the orientation provided by the IMU, then $\alpha_{\text{IMU}} = 1$. This corresponds to the assumption of point contacts at the TCPs. Note that this configuration is used for the experiments in Section 5 for dealing with a compliant support surface. The opposite situation is $\alpha_{\text{IMU}} = 0$, which causes the algorithm to rely exclusively on the kinematic information. This corresponds to the assumption of non-rotating contacts at the end effectors. In addition to this, if one end effector is not in contact with the environment, then the corresponding weight must be set to $\alpha_i = 0$.

Note that the algorithm presented above requires $\boldsymbol{x}_i$ and $\boldsymbol{R}_i$ to be known. During the initialization of the controller, one can choose the world frame arbitrarily fixed to the environment, thus leading to an initial $\boldsymbol{x}_i$ and $\boldsymbol{R}_i$. In case of

repositioning an end effector, $\boldsymbol{x}_i$ and $\boldsymbol{R}_i$ must be updated before the end effector is used again for estimating the base frame. To achieve this, the pose of the end effectors relative to the world can be measured by using the estimation of the base frame $\mathcal{B}$ provided by the other end effectors and the IMU. Naturally, the corresponding weight must be set to $\alpha_i = 0$ until the update is completed.

5. Experimental results

The evaluation of the proposed controller was done through four experiments with the humanoid robot TORO, developed by DLR (German Aerospace Center). In the current version, TORO has 27 DoFs (not counting the hands), a height of 1.74 m and a weight of 76.4 kg (Englsberger et al., 2014; Henze et al., 2014b). The 25 joints located in the legs, arms and hip are based on the technology of the DLR-KUKA LBR (Lightweight robot arm), and can be operated in both position and torque control mode (Albu-Schäffer et al., 2007a). In addition, there are two DoFs in the neck of the robot, implemented with servo motors. In terms of sensing, in addition to the position and torque sensors for the joints based on the LBR, the robot is equipped with force-torque sensors at the feet and an IMU measuring the orientation and angular velocity of the hip.

The experiments use the legs and arms to establish contact with the environment while operating the corresponding joints in torque controlled mode. For convenience, the two neck joints are locked all the time, as they are not necessary for conducting the experiments. The proposed controller is implemented in Matlab/Simulink, using qpOASES (Ferreau et al., 2008) for solving the constraint quadratic optimization problem given by (26) to (33). The controller is computed at a rate of 1 kHz, while the low-level joint controller is executed at a speed of 3 kHz (Albu-Schäffer et al., 2007a). Videos of the experiments presented in this section can be found in Extension 1. The parameters used in the controller are given in Tables 1 and 2. The values for stiffness and damping are chosen such that a human can easily apply perturbation forces to the robot. The weighting parameters are chosen such that the robot mainly generates forces with the end effectors, reducing the contact torques. Experience shows that there is no need for damping along the vertical axis of $\boldsymbol{D}_c$, as the legs of the robot are rather stretched in the standard configuration (see Figure 10a), leading to a relatively high influence of the joint friction on the vertical motion of the CoM. The contact model is the same throughout all of the experiments, assuming unilaterality, friction and a CoP constraint as described in Sections 3.2 and 4.1. The weights $\boldsymbol{Q}_i$ are chosen such that the robot generates mainly forces with the balancing end effectors, reducing the contact torques. In order to make the experiments more challenging, only the translational z-component of the local hand frames $\mathcal{T}_{\text{HandR}}$ and $\mathcal{T}_{\text{HandL}}$ (see Figure 6b) are used for balancing, while the other DoFs of the hands are assigned to the impedance task.

The algorithm for estimating the base frame (see Section 4.2) is parametrized as given in Table 2, such that it uses the information on the feet location in order to estimate the position of the base frame. The rotation of the base about the vertical axis of the world frame ($\boldsymbol{R}_{\text{ver}}$) is also provided by the feet. The rotation about an axis in the horizontal plane of the world frame ($\boldsymbol{R}_{\text{hor}}$) is instead measured by the onboard IMU. This particular parametrization is chosen because of the experiment presented in Section 5.4, where the robot balances on a compliant support surface in form of three layers of gym mats. Due to the softness of the mats, the feet move constantly (see Extension 1) and thus cannot provide valid information on the orientation of the world frame. As a consequence, the algorithm for estimating the base frame is parametrized such that it uses the IMU as much as possible for determining the orientation of the base. The kinematic information of the arms is not used ($\alpha_{\text{HandR}} = \alpha_{\text{HandL}} = 0$) since the contacts established by the hands are not firm enough to be evaluated. For convenience, the same parametrization is used throughout all of the experiments. In order to evaluate the accuracy of the estimation, we measured the position and orientation of the base with an external tracking system and compared it with the estimated data. Experiments[4] have shown that the error concerning position and orientation is less than 0.01 m and 0.25 ° respectively for a representative motion and the particular parametrization given by Table 2. As a consequence, the base frame estimation is considered accurate enough to be used in the following experiments. Note that in the remainder of this paper, all quantities are given with respect to the world frame $\mathcal{W}$, except for the contact wrenches that are given in their local end effector frame $\tilde{\mathcal{T}}_i$, as defined in Section 4.1. The world frame $\mathcal{W}$ is located in the middle between both feet, as shown e.g. in Figure 8a.

5.1. Redundancy in the wrench distribution

The first experiment shows how the robot can exploit the redundancy of the wrench distribution with its balancing end effectors. The setup can be seen in Figures 8a and 8b, where the robot uses the feet and the right hand for balancing, while the left hand is assigned to the impedance task. The robot is brought into two different configurations by moving the CoM to the left and to the right. As seen in Figure 8c, the CoM position shows a static control error of 0.01 m and 0.03 m respectively due to joint friction, but it can be counteracted by adding an integral control to the CoM compliance. In both configurations, the CoM creates a torque about the x-axis of the world frame $\mathcal{W}$, which must be counteracted by the balancing end effectors. For this, the robot can use the contact torques $\tau_{\text{FootR,x}}$ and $\tau_{\text{FootL,x}}$, the contact force[5] $\tau_{\text{HandR,z}}$ and the ratio between $f_{\text{FootR,z}}$ and $f_{\text{FootL,z}}$.

Let us first consider the pose where the CoM is shifted to the left. In this configuration, the robot cannot use the right hand, as this would require the robot to pull with the hand,

Table 1. Parameters for the multi-contact balancing controller.

Parameter	CoM	
$\boldsymbol{K}_c$	diag(1500 1500 3000) N/m	
$\boldsymbol{D}_c$	diag(169 102 0) Ns/m	
$\boldsymbol{L}_c$	diag(200 100 100) Nm/rad	
$\boldsymbol{B}_c$	diag(15 17 10) Nms/rad	
$\boldsymbol{Q}_c$	diag(10^6 10^6 10^6 10^6 10^6 10^6)	
Parameter	FootR, FootL	HandR, HandL
$\boldsymbol{K}_i$	diag(600 600 600) N/m	diag(600, 600, 600) N/m
$\boldsymbol{D}_i$	diag(30 30 30) Ns/m	diag(10, 10, 10) Ns/m
$\boldsymbol{L}_i$	diag(15 15 15) Nm/rad	diag(10, 10, 10) Nm/rad
$\boldsymbol{B}_i$	diag(0.5 0.5 0.5) Nms/rad	diag(0.1, 0.1, 0.1) Nms/rad
$\boldsymbol{Q}_i$ (wrench distribution task)	diag(10^{-3} 10^{-3} 10^{-3} 1 1 1)	diag($\frac{1}{40}$ $\frac{1}{40}$ $\frac{1}{40}$, 1, 1, 1)
$\boldsymbol{Q}_i$ (impedance task)	diag(10^3 10^3 10^3 10^3 10^3 10^3)	diag(10^3 10^3 10^3 10^3 10^3 10^3)
$f_i^{\min}$	50 N	20 N
$\tilde{\mu}_i$	0.4	0.4
$p_{\mathrm{i,x}}^{\min}$	−0.07 m	−0.03 m
$p_{\mathrm{i,x}}^{\max}$	0.13 m	0.03 m
$p_{\mathrm{i,y}}^{\min}$	−0.045 m	−0.03 m
$p_{\mathrm{i,y}}^{\max}$	0.045 m	0.03 m

violating the unilaterality of the contact (the hand is not closed around the bar). Thus, the controller only commands the minimum contact force $f_{\mathrm{HandR,z}}^{\mathrm{opt}} = f_{\mathrm{HandR}}^{\min} = 20\,\mathrm{N}$, as seen in Figure 8c. In order to generate the necessary torque about the x-axis of $\mathcal{W}$, the controller shifts the weight of the robot almost completely to the left leg $f_{\mathrm{FootL,z}}^{\mathrm{opt}} = 673\,\mathrm{N}$. The contact force in the right leg $f_{\mathrm{FootR,z}}^{\mathrm{opt}} = 80\,\mathrm{N}$ is close to its minimum $f_{\mathrm{FootR}}^{\min} = 50\,\mathrm{N}$. Thus, the controller also utilizes the torque of the right foot by shifting the CoP to $p_{\mathrm{FootR,y}}^{\mathrm{opt}} = 0.032\,\mathrm{m}$. The CoP of the left leg stays rather in the center of the foot.

If the CoM is shifted to the right, the robot can utilize the right hand by commanding a contact force of 65 N, in contrast to the previous pose. Due to the high lever arm of the hand (1.42 m), the contact force in the hand is almost enough to counteract the torque about the x-axis of the world frame induced by the displacement of the CoM. In consequence, the ratio between $f_{\mathrm{FootR,z}}$ and $f_{\mathrm{FootL,z}}$ is equal and the CoPs are both in the middle of their respective foot. This is remarkable, as the CoM shift to the right is much higher (0.17 m) than it was to the left (0.10 m).

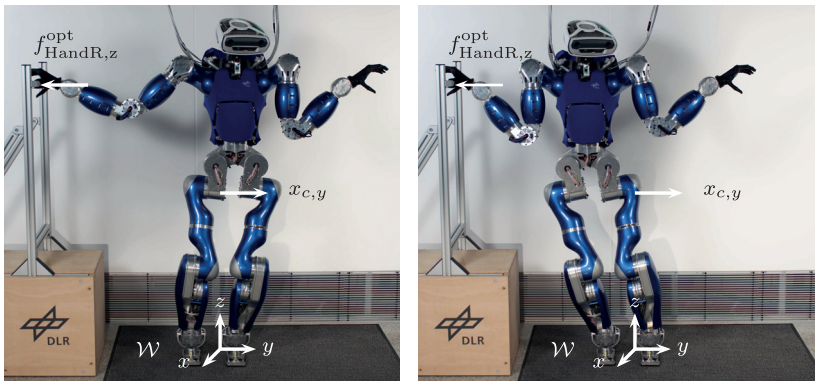
Summarizing the findings of this experiment, there are more DoFs in the wrench distribution than necessary for generating the overall wrench at the CoM. The controller chooses one out of the infinite numbers of solutions based on the cost function (26). In case some inequality constraints are activated, the corresponding DoFs can only be used up to a certain extent. In order to generate the overall wrench on the CoM, the controller redistributes the contact wrenches to the remaining DoFs of the optimization problem.

Table 2. Parameters for estimating the base frame.

Parameter	IMU	
α_{IMU}	1	
Parameter	FootR, FootL	HandR, HandL
α_i (wrench distribution task)	[0, 1]	0
α_i (impedance task)	0	0

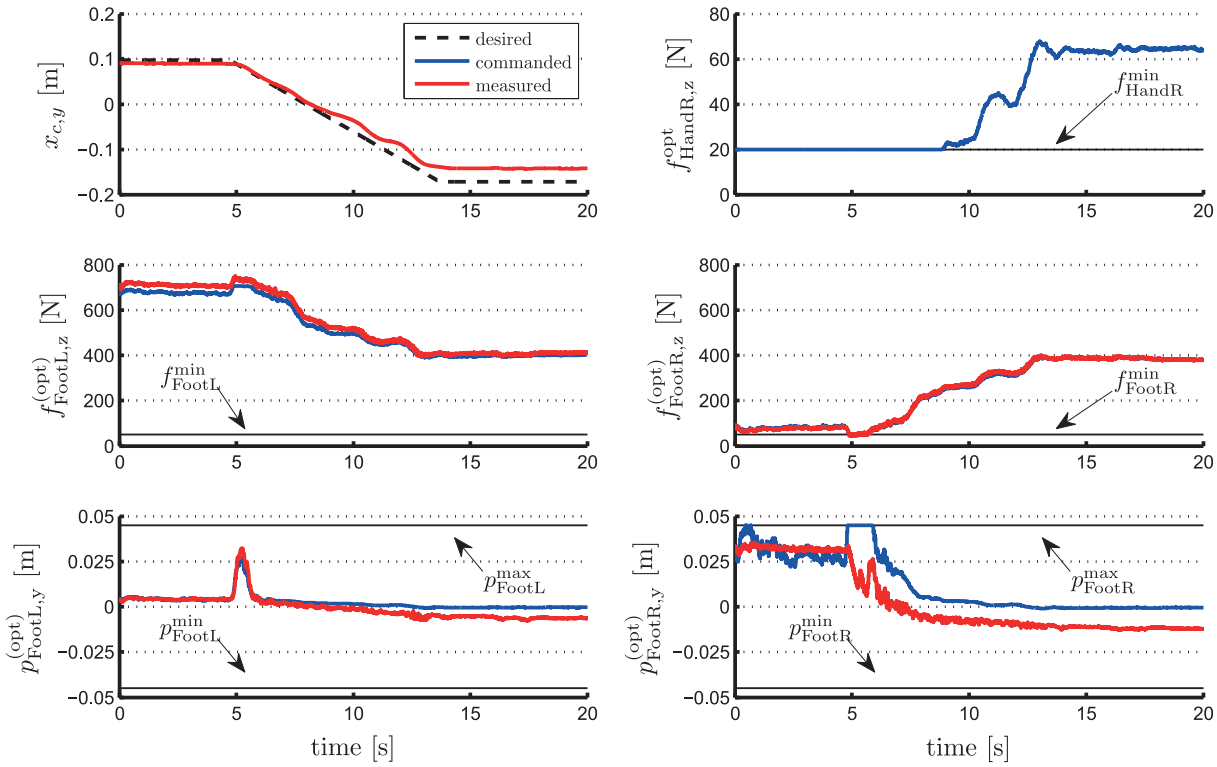
5.2. *Balancing while performing an interaction task*

The second experiment shows the capability of the robot to maintain balance while performing an interaction task. For this, the robot is placed in front of a table, as shown in Figure 9a, from which it lifts up a box of bottles with a weight of 12.2 kg, corresponding to 16% of the robot mass. In this particular setup, the hands are assigned to the impedance task and are thereby subjected to the Cartesian end effector compliance given by (20). In order to lift the box, the desired position for both hands ($\boldsymbol{x}_{\mathrm{HandR/L}}^{d}$) is slowly moved by 0.10 m along the vertical axis of the world frame. In order to counteract the weight of the box, the vertical components of $\boldsymbol{F}_{\mathrm{HandR}}^{d}$ and $\boldsymbol{F}_{\mathrm{HandL}}^{d}$ are ramped simultaneously from 0 N to −62 N. As seen in Figure 9b, lifting the box has almost no effect on the CoM location: the control error increases only by 0.007 m. The error in the hip orientation about the y-axis of the world frame $\mathcal{W}$ is increased by only 1.0 °. This observation can be explained by (25): $\boldsymbol{F}_{\mathrm{bal}}^{\mathrm{opt}}$ is always chosen such that the overall wrench on the CoM is equal to the right-hand side of (25), even in the presence of



(a) CoM shifted to the left.

(b) CoM shifted to the right.



(c) CoM position and contact forces.

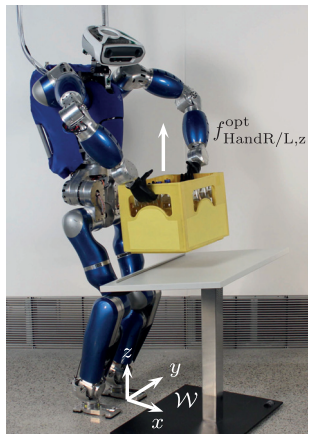
Fig. 8. Exploiting the redundancy in the wrench distribution. Black dashed: desired values. Blue solid: signals commanded by the controller. Red solid: measured signals.

an interaction wrench $\boldsymbol{F}^{\text{opt}}_{\text{int}}$. In other words, the feet counteract the wrenches generated by the compliance of the hands. The same conclusion can be drawn from Remark 4, studying the static case of all external wrenches acting at the end effectors as in this experiment. From (21), $\boldsymbol{F}^{\text{imp}}_{c} = \boldsymbol{0}$ must hold for the static case, leading to $\Delta\boldsymbol{x}_c = \boldsymbol{0}$ and $\Delta\boldsymbol{R}_c = \boldsymbol{I}$. The fact that the legs counteract the interaction wrenches can also bee seen in Figure 9b: the vertical foot forces are increased by 62 N and 48 N, which in sum corresponds roughly to the additional weight of the box. Furthermore, the box induces a torque about the y-axis of the world frame $\mathcal{W}$, which the robot counteracts by shifting the CoP of the left and right foot by 0.048 m and 0.041 m to the front.

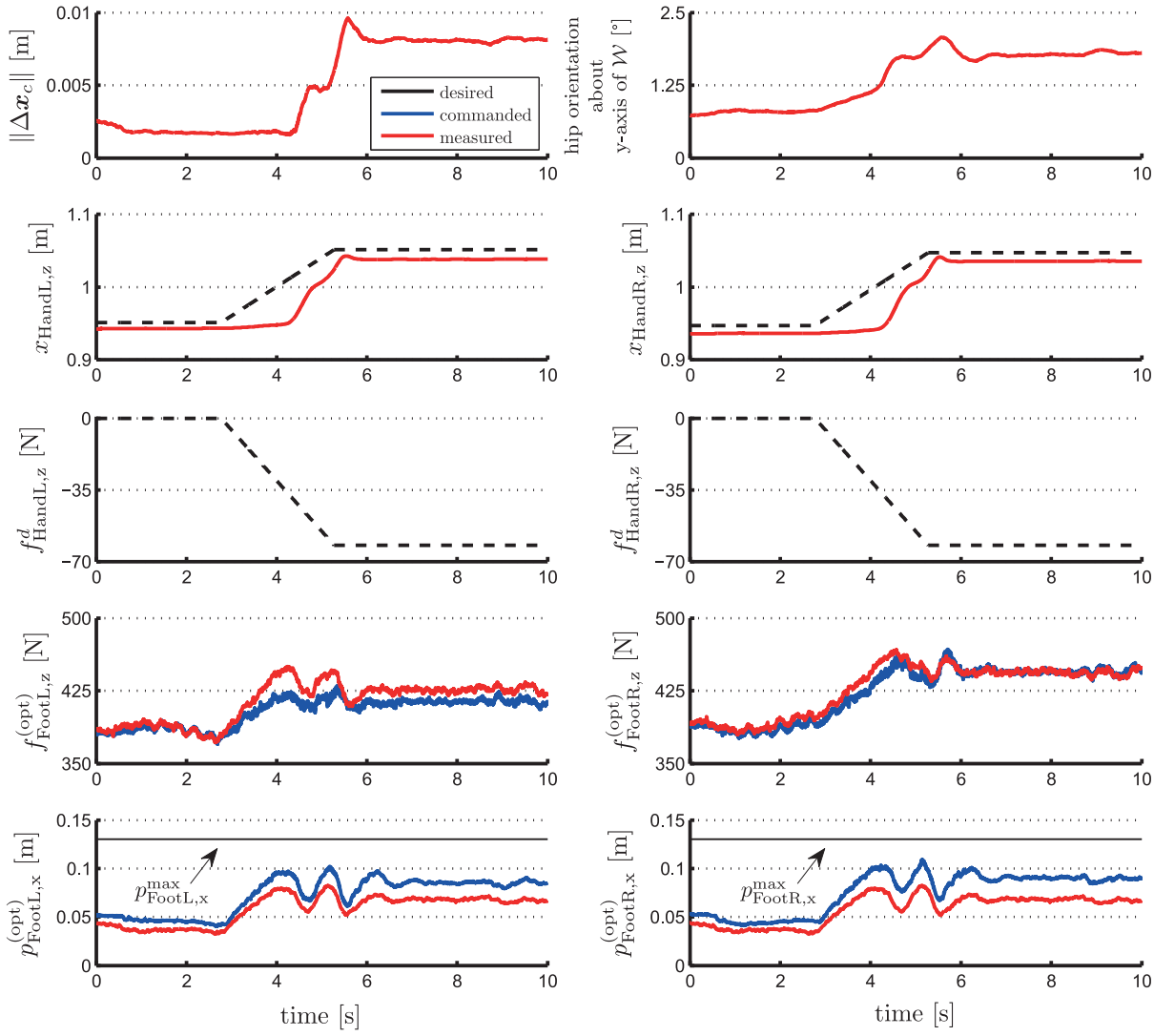
In summary, this experiment shows the importance of taking the interaction wrenches into account for balancing in order to prevent a disturbance of the CoM.

5.3. Tracking performance

The third experiment evaluates the performance of the proposed controller in the tracking case. For this, TORO stands with both feet in contact with the ground, as shown in Figure 10a; the legs are used for balancing while the arms are operated in impedance mode. The desired CoM position moves along the y-axis of the world frame $\mathcal{W}$ according to a sinusoidal signal with a frequency of 1/3 Hz. The amplitude



(a) Setup of the experiment.



(b) CoM position and end effector forces.

Fig. 9. Lifting a box with bottles from a table. Black dashed: desired values. Blue solid: signals commanded by the controller. Red solid: measured signals.

of the signal is linearly increased from 0 m to 0.06 m during the first 5 s, as shown in Figure 10. Afterwards, the amplitude is held constant for 10 s and then linearly decreased to 0 m within 5 s. Figure 10b also presents the result of the experiment; the CoM position shows an overshoot of up to 0.025 m if the feedforward control in (25) and (35) is disabled by setting $\dot{\boldsymbol{\nu}}^d = \boldsymbol{\nu}^d = \mathbf{0}$. Activating the feedforward control ($\dot{\boldsymbol{\nu}}^d \in \mathbb{R}^{6+n}$, $\boldsymbol{\nu}^d \in \mathbb{R}^{6+n}$) reduces the tracking error significantly to less than 0.007 m. In addition, the required contact wrenches are reduced as well, leading to a smaller excitation of the CoP of the right and the left foot (0.018 m for deactivated feedforward control, 0.008 m for activated feedforward control). The amplitude of the vertical contact forces is reduced by up to 90 N. In any case, both controllers are able to maintain balance.

The amplitude for the desired CoM position was increased in a second run from 0.06 m to 0.08 m, as shown in Figure 10c. Due to the stronger excitation, the controller is only capable of maintaining the balance if the feedforward control is activated. Without the feedforward term, the robot is no longer capable of shifting its weight from one leg to the other because the right leg hits the lower limit $f^{\min}_{\text{FootR}} = 50\,\text{N}$ at 5.1 s. In addition, the CoP for the right and the left foot hit the lower limit of $p^{\min}_{\text{FootR,y}} = p^{\min}_{\text{FootL,y}} = -0.045\,\text{m}$ at 5.7 s and 5.1 s, respectively. As a consequence, the optimization problem (26) to (33) becomes infeasible, meaning that the controller is no longer capable of generating the required overall wrench at the CoM and at the interaction end effectors. The robot fails at 5.8 s, when both feet tilt about their right edge, as can be seen in Extension 1.

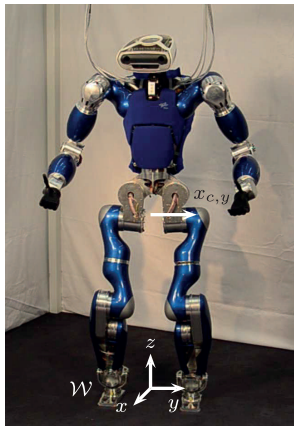
5.4. Balancing on compliant support

The fourth experiment tests the robustness of the proposed control approach against uncertainties in the contact surface. To generate a soft, compliant support, the robot is placed on three layers of gym mats, as shown in Figure 11a. The gym mats have a size of 200 cm × 200 cm × 8 cm, and are made of a compound foam with a density of 20 kg/m^3. After the controller is initialized, the robot is kicked from the left side; the perturbation created on the CoM position is shown in Figure 11b and 11c. The robot counteracts the disturbance in the frontal plane mainly by shifting its weight from one leg to the other, instead of changing the CoP of the feet ($p^{\text{opt}}_{\text{FootR,y}}$ and $p^{\text{opt}}_{\text{FootL,y}}$). This can be explained by the values chosen for the weights $\boldsymbol{Q}_i$ (see Table 1), which penalize the foot torques more than the foot forces. For balancing in the sagittal plane, the controller does not have any other option than using the foot torques, which leads to a high excitation of both CoPs ($p^{\text{opt}}_{\text{FootR,x}}$ and $p^{\text{opt}}_{\text{FootL,x}}$) covering the whole size of the feet. The strength of the kick brings the robot to the limit of stable balancing: at 0.5 s, the vertical force of the left foot ($f^{\text{opt}}_{\text{FootL,z}}$) almost hits the limit of $f^{\min}_{\text{FootL}} = 50\,\text{N}$. The corresponding commanded CoP $\boldsymbol{p}^{\text{opt}}_{\text{FootL}}$ moves to the right front corner of the foot represented by $p^{\text{opt}}_{\text{FootL,x}} = p^{\max}_{\text{FootL,x}} = 0.13\,\text{m}$ and $p^{\text{opt}}_{\text{FootL,y}} = p^{\min}_{\text{FootL,y}} = -0.045\,\text{m}$. Later, the commanded CoP of the right foot hits the front limit ($p^{\text{opt}}_{\text{FootR,x}} = p^{\min}_{\text{FootR,x}} = 0.13\,\text{m}$) at 1.3 s. The difficulty in balancing on a compliant support lies in the unmodeled dynamics of the contacts: the support surface must be deformed first before the desired contact wrench can be generated. As shown in Figure 11c, the vertical contact forces show a deviation between the commanded and the measured value of up to 83 N for the right and 111 N for the left leg. There is also a strong deviation between the measured and the commanded CoP positions. According to the sensor readings, the CoP of the right foot is even outside the contact area at 1.5 s, which appears to be impossible at first glance. However, at this instance of time the commanded and the measured force $f^{\text{opt}}_{\text{FootR,z}}$ and $f_{\text{FootR,z}}$ are rather small and the computation of the CoP is thereby ill-conditioned. In addition, note that the force-torque sensor for measuring the contact wrench is not located at the sole of the foot but at the ankle. During the whole experiment, the feet are constantly moving, as can be seen in Extension 1. In particular, the right foot has a high acceleration at $t = 1.5\,\text{s}$, which might cause the inertia of the right foot to influence the measurement.

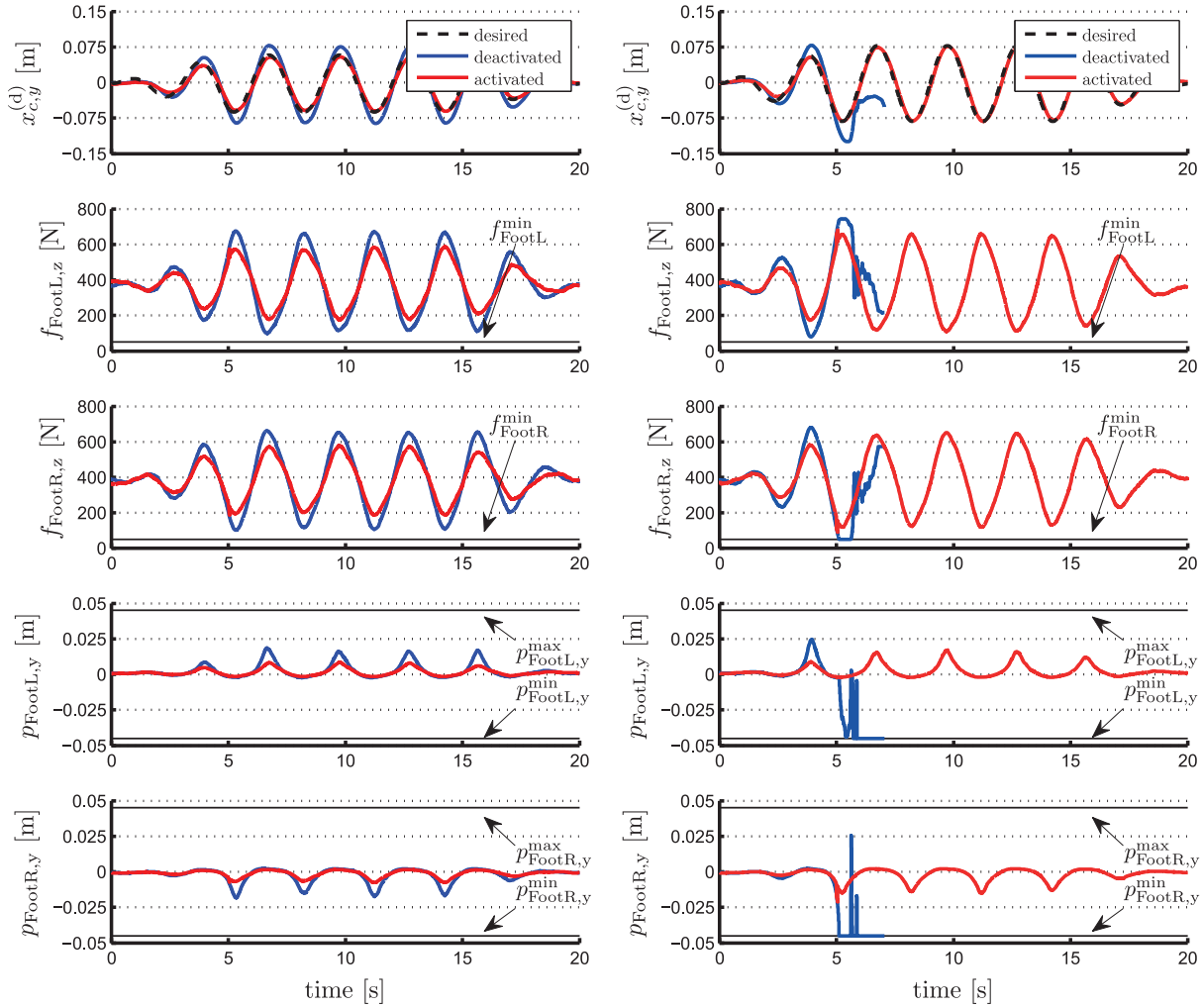
One of the keys for achieving a robust behavior is the use of the estimation of the base frame described in Section 4.2. Due to the compliant support, the feet must move continuously in order to adapt to the support surface (as shown in Extension 1). By using only the IMU ($\alpha_{\text{IMU}} = 1$) for estimating the orientation of the of the base $\boldsymbol{R}_b$, the latter is not affected by the motion of the feet. The influence of the motion on the position of the base $\boldsymbol{x}_b$ is reduced by averaging the translational information provided by the kinematics of the right and the left leg ($\alpha_{\text{FootR}} = \alpha_{\text{FootL}} = 1$, $\alpha_{\text{HandR}} = \alpha_{\text{HandL}} = 0$). Figure 11 shows the position of the CoM that was computed online by the controller based on the estimation of the base frame. The estimated CoM location is also used for control, which demonstrates the robustness of the presented approach with respect to unmodeled contact dynamics.

6. Discussion and conclusion

In this work we have developed a balancing controller that allows torque-controlled humanoid robots to be operated in multi-contact scenarios. The control algorithm optimizes the end effector wrenches according to a non-strict task hierarchy. The task with the highest priority ensures a compliant behavior of a subset of the end effectors to allow the robot to interact with its environment, as e.g. for manipulating an object. The second task creates a compliant behavior of the CoM location and the hip orientation by generating proper contact wrenches with the remaining end effectors. The task with the lowest priority level solves the wrench distribution problem while trying to match the balancing end effector wrenches with a default wrench distribution. The proposed optimization problem takes into account several constraints concerning unilaterality, friction and the



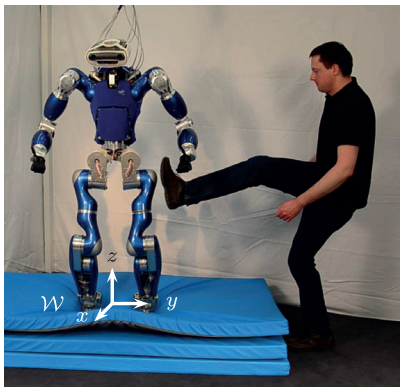
(a) Setup of the experiment.



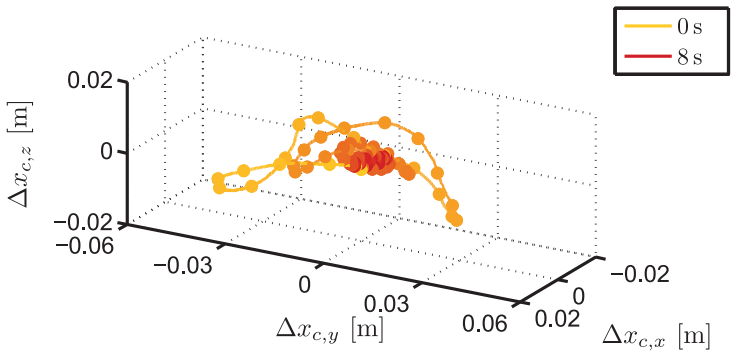
(b) Behavior for a maximum amplitude of 0.06m.

(c) Behavior for a maximum amplitude of 0.08m.

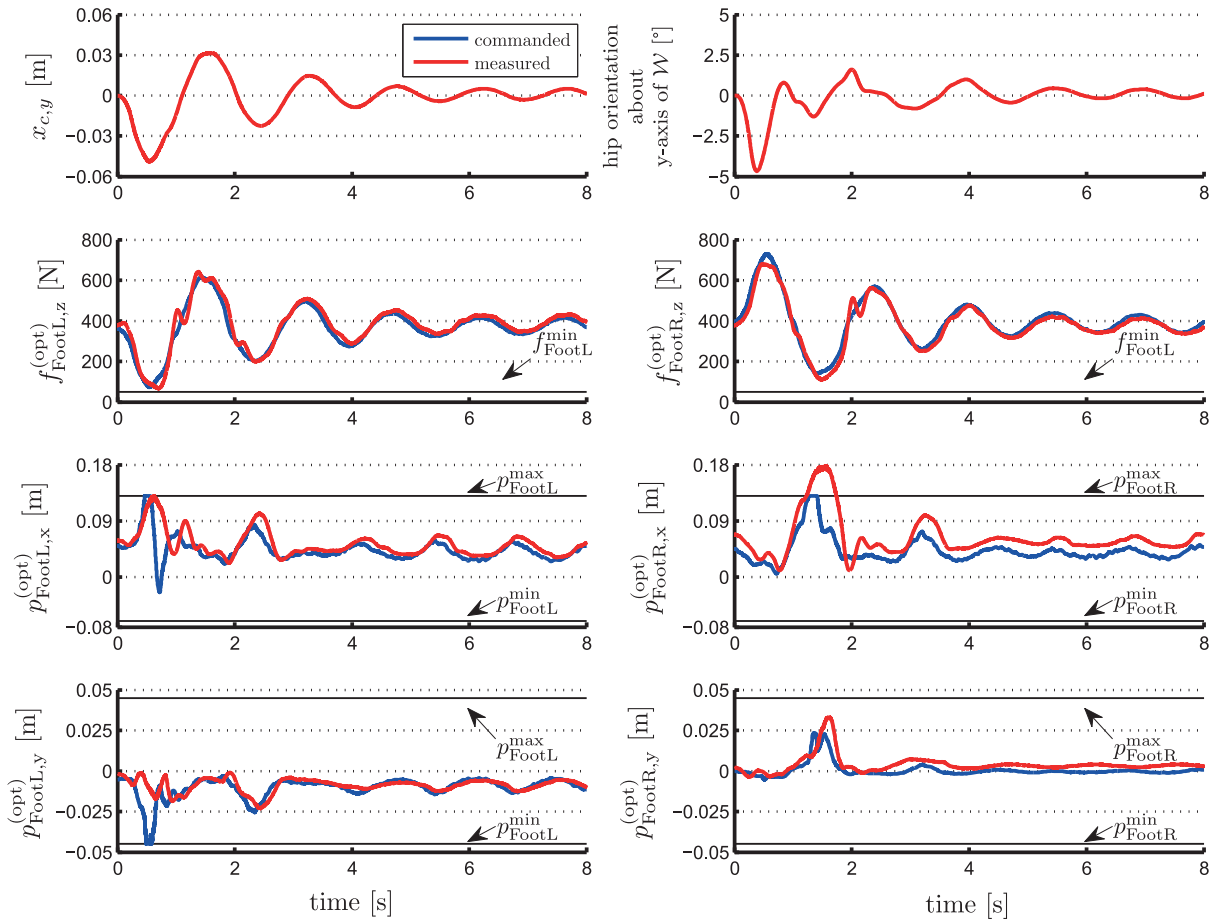
Fig. 10. Tracking of a desired CoM trajectory, corresponding to a sinusoidal signal. Black dashed: desired trajectory. Blue solid: feedforward controller deactivated. Red solid: feedforward controller activated.



(a) Setup of the experiment.



(b) Trajectory of the CoM: Two consecutive dots represent positions that are measured every 0.1 s. The color fades from yellow into red over time.



(c) CoM and CoP position, and deviation in the contact forces.

Fig. 11. Balancing on compliant support: perturbation in the CoM position as a result of a kick from the left side. Blue solid: signals commanded by the controller. Red solid: measured signals.

CoP location for the end effectors that are used for maintaining the balance. The presented approach extends the work of Ott et al. (2011) by (a) adding a feedforward action to improve the tracking performance, (b) exploiting multiple contact points and (c) implementing a task hierarchy allowing an interaction with the environment while balancing. The proposed feedforward control action is added to the feedback loop resulting in a control structure similar to PD+ control, which allows us to proof the stability and passivity of the closed-loop system. The passivity ensures a robust behavior of the robot with respect to the environment, although the approach is model-based and therefore potentially prone to modeling errors. In addition to this, the constraints concerning unilaterality, friction and the CoP

location are applied to the commanded contact wrenches, which means that the real contact wrenches can potentially violate the constraints. However, applying the constraints to the real contact wrenches would require the exact knowledge of the external disturbances.

Several experiments have been conducted with the humanoid robot TORO in order to show the features and performance of the proposed control approach. The controller shows a good performance both in the regulation and tracking cases. In addition, the controller shows a high robustness, allowing TORO to balance on a compliant support surface (in this case, on three layers of gym mats). This robustness is possible thanks to an algorithm for estimating the state of the base frame, which has also been presented in this work. To the best of our knowledge, this is the first experimental realization of a humanoid robot balancing on a compliant support surface.

In the future, we plan to combine this approach for multi-contact balancing with a concept of prioritized multi-task control involving null space projection techniques in order to extend the presented non-strict hierarchy of tasks.

Acknowledgements

The authors would like to thank Alexander Werner, from DLR, for his great help when conducting the experiments with TORO. Also, great support was received from Robert Burger and Florian Schmidt on the real-time software aspects related to the robot TORO.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was partially supported by the Helmholtz Association (grant number VH-NG-808) and by the European Commission (grant number H2020-ICT-645097, COMANOID).

Notes

1. A similar transformation is used in Ott et al. (2011) and Henze et al. (2014a), where $\dot{\boldsymbol{x}}_c$ and $\boldsymbol{\omega}_c$ are expressed in frame $\mathcal{B}$ instead of $\mathcal{W}$. In Garofalo et al. (2015) the linear and angular momentum are used instead of $\dot{\boldsymbol{x}}$ and $\boldsymbol{\omega}$, thus leading to a decoupled structure of the inertia matrix.
2. The authors were able to reduce the computation time by 30% using qpOASES (Ferreau et al., 2008) for solving the optimization problem on the real-time computer of TORO (Intel(R) Core(TM) i7-4700EQ processor with four cores running at 2.4 GHz). The benchmark was computed on one core without the 'hot-start' functionality of qpOASES.
3. This decomposition has been previously used in Fritsch et al. (2013) for attitude estimation in quadrotors.
4. The experiments were performed with an active tracking system, K610 by Nikon Metrology NV, with markers on the torso of TORO.
5. Note that the contact wrenches are given in their local end effector frame $\tilde{\mathcal{T}}_i$ as defined in Section 4.1 (see the introduction to Section 5).
6. The method presented in Gramkow (2001) requires that the dot products of $\boldsymbol{\eta}_1'$ with all the other unit quaternions involved in (47) are positive, which can be ensured by (a) exploiting the fact that $\boldsymbol{\eta}_i'$ and $-\boldsymbol{\eta}_i'$ represent the same rotation and by (b) assuming that the orientations are less than 180 ° apart from each other. The latter assumption holds in our case as we average several estimates of the orientation $\boldsymbol{R}_b$, which are probably close to each other.

References

Albu-Schäffer A, Haddadin S, Ott C, Stemmer A, Wimboeck T and Hirzinger G (2007a) The DLR lightweight robot - design and control concepts in human environments. *Industrial Robot: An International Journal* 34(5): 376–385.

Albu-Schäffer A, Ott C and Hirzinger G (2007b) A unified passivity-based control framework for position, torque and impedance control of flexible joint robots. *The International Journal of Robotics Research* 26(1): 23–39.

Arimoto S (2000) Passivity-based control. In: *IEEE International Conference on Robotics and Automation*, pp. 227–232.

Audren H, Vaillant J, Kheddar A, Escande A, Kaneko K and Yoshida E (2014) Model preview control in multi-contact motion – application to a humanoid robot. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4030–4035.

Bloesch M, Gehring C, Fankhauser P, Hutter M, Hoepflinger M and Siegwart R (2013) State estimation for legged robots on unstable and slippery terrain. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 6058–6064.

Bouyarmane K and Kheddar A (2011) Using a multi-objective controller to synthesize simulated humanoid robot motion with changing contact configurations. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4414–4419.

Cheng G, Hyon SH, Morimoto J, et al. (2007) CB: A humanoid research platform for exploring neuroscience. *Advanced Robotics* 21(10): 1097–1114.

Englsberger J, Werner A, Ott C, et al. (2014) Overview of the torque-controlled humanoid robot TORO. In: *IEEE-RAS International Conference on Humanoid Robots*, Madrid, Spain, pp. 916–923.

Ferreau HJ, Bock HG and Diehl M (2008) An online active set strategy to overcome the limitations of explicit MPC. *The International Journal of Robust and Nonlinear Control* 18(8): 816–830.

Fritsch O, Henze B and Lohmann B (2013) Fast and saturating thrust direction control for a quadrotor helicopter. *at - Automatisierungstechnik* 61(3): 172–182.

Garofalo G, Henze B, Englsberger J and Ott C (2015) On the inertially decoupled structure of the floating base robot dynamics. In: *International Conference on Mathematical Modelling*, pp. 322–327.

Gramkow C (2001) On averaging rotations. *The International Journal of Computer Vision* 42(1–2): 7–16.

Henze B, Ott C and Roa MA (2014a) Posture and balance control for humanoid robots in multi-contact scenarios based on model predictive control. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3253–3258.

Henze B, Werner A, Roa MA, Garofalo G, Englsberger J and Ott C (2014b) Control applications of TORO - a torque controlled humanoid robot. In: *IEEE-RAS International Conference on Humanoid Robots*, pp. 841–841.

Hyon SH (2009) Compliant terrain adaptation for biped humanoids without measuring ground surface and contact forces. *IEEE Transactions on Robotics* 25(1): 171–178.

Hyon SH, Hale JG and Cheng G (2007) Full-body compliant human–humanoid interaction: Balancing in the presence of unknown external forces. *IEEE Transactions on Robotics* 23(5): 884–898.

Ibanez A, Bidaud P and Padois V (2012) Unified preview control for humanoid postural stability and upper-limb interaction adaptation. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1801–1808.

Khatib O (1987) A unified approach for motion and force control of robot manipulators: The operational space formulation. *Journal of Robotics and Automation* 3(1): 43–53.

Macchietto A, Zordan V and Shelton CR (2009) Momentum control for balance. *ACM Transactions on Graphics* 28(3): 80.

Mistry M, Buchli J and Schaal S (2010) Inverse dynamics control of floating base systems using orthogonal decomposition. In: *IEEE International Conference on Robotics and Automation*, pp. 3406–3412.

Murray RM, Li Z and Sastry SS (1994) *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL: CRC Press.

Ott C (2008) *Cartesian Impedance Control of Redundant and Flexible-Joint Robots* (*Springer Tracts in Advanced Robotics*, vol. 49). New York: Springer Publishing Company.

Ott C, Albu-Schäffer A, Kugi A and Hirzinger G (2008) On the passivity based impedance control of flexible joint robots. *IEEE Transactions on Robotics* 24(2): 416–429.

Ott C, Baumgärtner C, Mayr J, et al. (2010) Development of a biped robot with torque controlled joints. In: *IEEE-RAS International Conference on Humanoid Robots*, pp. 167–173.

Ott C, Henze B and Lee D (2013) Kinesthetic teaching of humanoid motion based on whole-body compliance control with interaction-aware balancing. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4615–4621.

Ott C, Roa MA and Hirzinger G (2011) Posture and balance control for biped robots based on contact force optimization. In: *IEEE-RAS International Conference on Humanoid Robots*, pp. 26–33.

Paden B and Panja R (1988) Globally asymptotically stable 'PD+' controller for robot manipulators. *The International Journal of Control* 47(6): 1697–1712.

Pratt J and Krupp B (2008) Design of a bipedal walking robot. In: *Proceedings of the International Society for Optics and Photonics*, vol. 6962.

Righetti L, Buchli J, Mistry M, Kalakrishnan M and Schaal S (2013) Optimal distribution of contact forces with inverse-dynamics control. *The International Journal of Robotics Research* 32(3): 280–298.

Sentis L, Park J and Khatib O (2010) Compliant control of multicontact and center-of-mass behaviors in humanoid robots. *IEEE Transactions on Robotics* 26(3): 483–501.

Stelzer A, Hirschmueller H and Goerner M (2012) Stereo-vision-based navigation of a six-legged walking robot in unknown rough terrain. *The International Journal of Robotics Research* 31(4): 381–402.

Stephens BJ and Atkeson CG (2010) Dynamic balance force control for compliant humanoid robots. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1248–1255.

Stramigioli S (2001) *Modeling and IPC Control of Interactive Mechanical Systems: A Coordinate-free Approach* (*Lecture Notes in Control and Information Sciences*, vol. 266). Berlin: Springer-Verlag.

Wensing PM, Hammam GB, Dariush B and Orin DE (2013) Optimizing foot centers of pressure through force distribution in a humanoid robot. *The International Journal of Humanoid Robotics* 10(03): 1–21.

Williams D and Khatib O (1993) The virtual linkage: a model for internal forces in multi-grasp manipulation. In: *International Conference on Robotics and Automation*, vol. 1, pp. 1025–1030.

Zhang S and Fasse ED (2000) Spatial compliance modeling using a quaternion-based potential function method. *Multibody System Dynamics* 4(1): 75–101.

Appendix A: Index to multimedia extensions

Archives of IJRR multimedia extensions published prior to 2014 can be found at http://www.ijrr.org. After 2014 all videos are available on the IJRR YouTube channel at http://www.youtube.com/user/ijrrmultimedia.

Table of Multimedia Extensions

Extension	Type	Description
1	Video	Nine experiments with the humanoid robot TORO, four of them are discussed in detail in Section 5.

Appendix B: Transformation of the Jacobian matrix

The transformed Jacobian matrix in (12) can be obtained from

$$\begin{aligned}
&\begin{bmatrix}\begin{bmatrix}\boldsymbol{I} & -\hat{\boldsymbol{x}}_{bi}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix} & \bar{\boldsymbol{J}}_i\end{bmatrix}\bar{\boldsymbol{T}}^{-1}\\
&\quad=\begin{bmatrix}\begin{bmatrix}\boldsymbol{I} & \hat{\boldsymbol{x}}_{ib}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix} & \bar{\boldsymbol{J}}_i\end{bmatrix}\begin{bmatrix}\begin{bmatrix}\boldsymbol{I} & \hat{\boldsymbol{x}}_{bc}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix} & \begin{bmatrix}-\boldsymbol{J}_{bc}\\ \boldsymbol{0}\end{bmatrix}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\\
&\quad=\begin{bmatrix}\begin{bmatrix}\boldsymbol{I} & \hat{\boldsymbol{x}}_{ib}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{I} & \hat{\boldsymbol{x}}_{bc}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix} & \bar{\boldsymbol{J}}_i-\begin{bmatrix}\boldsymbol{I} & \hat{\boldsymbol{x}}_{ib}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{J}_{bc}\\ \boldsymbol{0}\end{bmatrix}\end{bmatrix}\\
&\quad=\begin{bmatrix}\begin{bmatrix}\boldsymbol{I} & \hat{\boldsymbol{x}}_{ic}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix} & \bar{\boldsymbol{J}}_i-\begin{bmatrix}\boldsymbol{J}_{bc}\\ \boldsymbol{0}\end{bmatrix}\end{bmatrix}=\begin{bmatrix}\begin{bmatrix}\boldsymbol{I} & -\hat{\boldsymbol{x}}_{ci}\\ \boldsymbol{0} & \boldsymbol{I}\end{bmatrix} & \boldsymbol{J}_i\end{bmatrix}
\end{aligned}\tag{44}$$

Appendix C: Rotational spring

To deal with deviations in orientation, we introduce $\boldsymbol{\tau}_r(\boldsymbol{\Sigma}, \boldsymbol{R}) \in \mathbb{R}^3$ in Section 3.2 as the torque of a general torsional spring, in the sense presented in Zhang and Fasse (2000) and Ott et al. (2011). Let $\mathcal{A}'$ be a general frame whose orientation with respect to a general frame $\mathcal{A}$ is given by the rotation matrix $\boldsymbol{R}$. Furthermore, let $\underline{\boldsymbol{\eta}} \in \mathbb{R}^3$ and $\eta \in \mathbb{R}$ be the vector and the scalar part of a unit quaternion $\boldsymbol{\eta} \in \mathbb{R}^4$ representing the rotation $\boldsymbol{R}$. The characteristic of the spring is given by the potential

$$V = 2\underline{\boldsymbol{\eta}}^{\mathrm{T}} \boldsymbol{\Sigma} \underline{\boldsymbol{\eta}} \tag{45}$$

where $\boldsymbol{\Sigma} \in \mathcal{SO}(3)$ is a symmetric and positive-definite stiffness matrix. The torque generated by the spring (expressed in frame $\mathcal{A}$) can be derived from (45) as

$$\boldsymbol{\tau}_r(\boldsymbol{\Sigma}, \boldsymbol{R}) = -2\boldsymbol{R}\left(\eta \boldsymbol{\Sigma} \underline{\boldsymbol{\eta}} + \underline{\boldsymbol{\eta}} \times \boldsymbol{\Sigma} \underline{\boldsymbol{\eta}}\right). \tag{46}$$

Appendix D: Averaging of rotation matrices

Let $\boldsymbol{\eta}'_i$ be the unit quaternion representation of a set of general rotations $\boldsymbol{R}'_i \in \mathcal{SO}(3)$ with $i' = 1 \ldots n'$. The weighted average of the rotations can be computed according to

$$\boldsymbol{\eta}' = \frac{\sum_{i'=1}^{n'} \alpha'_i \boldsymbol{\eta}'_i}{\left\| \sum_{i'=1}^{n'} \alpha'_i \boldsymbol{\eta}'_i \right\|} \tag{47}$$

based on the method[6] presented in Gramkow (2001). The weighting coefficients are represented by $\alpha'_i \in [0; 1]$ satisfying $\sum_{i'=1}^{n'} \alpha'_i = 1$.