2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)
Beijing, China, November 6-9, 2018

Balancing Control through Post-Optimization of Contact Forces

Arturo Laurenzi[1,2,*], Enrico Mingo Hoffman[1,*], Matteo Parigi Polverini[1] and Nikos G. Tsagarakis[1]

Abstract— **In this work we present a novel method to address the balancing problem for torque controlled legged robots through post-optimization of contact forces. The main concept consists in treating a legged robot as a fully actuated fixed-base system in order to compute the desired joint torques according to a fixed-based torque controller. The under-actuated component of the obtained torques is then mapped into contact forces through an optimal distribution problem. Besides extending previous work to the floating-base case, the proposed method has the notable advantage of avoiding the specification of a desired momentum of rotation, in addition to a reduced number of decision variables compared to full-inverse dynamics methods. The effectiveness of our approach has been validated in simulation using two different humanoid platforms: the *CENTAURO* and the *COMAN*+ robots, both recently developed at Istituto Italiano di Tecnologia (IIT). Preliminary experimental results on *COMAN*+ are also presented.**

I. INTRODUCTION

Balancing represents a crucial requirement for legged robots expected to cope with a variety of unstructured terrains and environments. Despite a deep knowledge on the dynamics governing the balancing of articulated bodies, balancing is still considered a challenging control problem, especially when dealing with torque controlled humanoids and legged robots in general. In this respect, the existing control approaches are generally classified in two categories.

The first category achieves balancing through a two-stage methodology. An optimal contact force distribution problem is first solved with respect to the robot centroidal dynamics. This phase will be hereafter referred to as *pre-optimization* of contact forces. The computed contact forces are then mapped to joint torques under quasi-static assumptions, see Hyon et al. [1], [2] and Ott et al. [3], [4], [5], or through inverse dynamics, as in [6], [7] and [8].

In opposition to the first category, the humanoid balancing problem can be alternatively addressed in a single-stage fashion by entirely exploiting the full-body inverse dynamics. Several inverse dynamics controllers, e.g. [9], [10], [11], compute joint torques by modeling contacts as rigid constraints and projecting the dynamics into a constraint free space. In this way an explicit solution of the contact force distribution problem is not required, although optimality of the problem is not guaranteed. Nevertheless, Righetti et al. [12], [13] showed that is possible to design inverse dynamics controllers and operational space controllers that are optimal with respect to any combination of linear and quadratic cost in the contact forces and in the torque commands. On the other hand, methods based on a hierarchical Quadratic Programming (QP) formulation of the full-body inverse dynamics, see [14], [15], [16], and [17], [18], explicitly consider contact forces as variables for the resulting optimization problem.

It is worth pointing out that, despite a clear advantage in terms of required computation time of the first category of methods over the second, the *pre-optimization* of contact forces raises a major concern about the role of the momentum of rotation in balance control. It is known in fact that the kinetic momentum of rotation is not directly related to the actual orientation of an articulated system [19], [20]. As a consequence, controlling the momentum of rotation for a balancing task may end up in a body rotation which is incompatible with the task itself.

In a recent work by the authors [21] a prioritized Cartesian impedance controller for redundant fixed-base robots has been proposed, based on a hierarchical QP formulation. Extending this approach to floating-base legged robots to subsequently address the humanoid balancing problem is the main focus of the present paper. The proposed balancing controller belongs to the first category of methods, i.e. it adopts a two-stage technique. In this respect, aiming to overcome the main limitation inherent in the *pre-optimization* of contact forces, i.e. the control of the momentum of rotation, we propose to treat a legged robot as a fully actuated fixed-base system in order to compute the desired joint torques according to [21]. Only at this point, the under-actuated component of the obtained torques can be mapped into contact forces through an optimal distribution problem, hereafter referred to as *post-optimization* of contact forces. This way the tricky specification of a desired momentum of rotation is circumvented. In addition to the aforementioned advantage over *pre-optimization* methods and the reduced number of decision variables compared to single-stage methods, the add-on nature of the proposed control approach entails no modification of the original algorithm in [21]. It is also worth noticing that our post-optimization yields a systematic way to adapt a generic fixed-base controller to an underatuated system.

The paper is organized as follows. The proposed torque control approach to humanoid balancing is introduced and discussed in Sec. II. Simulation results performed on both *CENTAURO* and *COMAN*+ humanoid platforms are presented in Sec. III-A, together with preliminary experimental results. Finally, concluding remarks and future work directions can be found in Sec. IV.

[1] Advanced Robotics Department (ADVR), Istituto Italiano di Tecnologia, Genova, Italy {arturo.laurenzi, enrico.mingo, matteo.parigi, nikolaos.tsagarakis}@iit.it
[2] DIBRIS, Universitá di Genova, Italy
[*] These authors have contributed equally to this work.

II. Mathematical formulation

This section introduces the mathematical formulation of our prioritized force controller. In Section II-A we introduce the notation, and we give some brief background on our previous work [21]. Then, in Section II-B we present the mathematical modeling for the floating-base case; we highlight the major differences that forbid direct use of our previous work. Finally, in Section II-C we present the main contribution of this paper, which is the extension to the floating-base case.

A. The fixed-base case

Let us consider a generic task $\boldsymbol{x} \in \mathbb{R}^m$ expressed in terms of a function $\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{q})$; the corresponding task velocity is obtained by differentiation:

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}, \tag{1}$$

where $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{m\times n}$ is the task Jacobian matrix. The dependence on $\boldsymbol{q}$ will be omitted in the following for brevity. We are interested in finding joint torques such that our task shows a desired behaviour with respect to reference signals, and external perturbations as well. To this aim, let us introduce the robot dynamics in contact with the environment:

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{\tau} + \boldsymbol{J}^T\boldsymbol{F}_x; \tag{2}$$

in (2), $\boldsymbol{B} \in \mathbb{R}^{n\times n}$ represents the joint-space inertia matrix, $\boldsymbol{h} \in \mathbb{R}^n$ is the vector of *bias torques* which are needed to produce zero joint acceleration, and $\boldsymbol{\tau} \in \mathbb{R}^n$ is the vector of joint torques. External wrenches $\boldsymbol{F}_x \in \mathbb{R}^m$ applied to our task are also considered. By further differentiation of (1) and substituting (2) we obtain the task dynamics as follows [22]:

$$\boldsymbol{\Lambda}(\boldsymbol{q})\ddot{\boldsymbol{x}} + \boldsymbol{\mu}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{F}_{\tau|x} + \boldsymbol{F}_x \tag{3}$$

Where:

- $\boldsymbol{\Lambda} \in \mathbb{R}^{m\times m}$ is the task-space inertia matrix; it is given by
$$\boldsymbol{\Lambda} = \left(\boldsymbol{J}\boldsymbol{B}^{-1}\boldsymbol{J}^T\right)^{-1}; \tag{4}$$
- $\boldsymbol{\mu} \in \mathbb{R}^m$ is the bias force vector (i.e. the force required to produce zero task acceleration);
- $\boldsymbol{F}_\tau \in \mathbb{R}^m$ is the task space force that is produced by the joint torques; its expression is the following:
$$\boldsymbol{F}_{\tau|x} = \bar{\boldsymbol{J}}^T\boldsymbol{\tau}, \tag{5}$$
where $\bar{\boldsymbol{J}} \in \mathbb{R}^{n\times m}$ is known as the *dynamically-consistent pseudo-inverse* of $\boldsymbol{J}$, which has the following expression:
$$\bar{\boldsymbol{J}} = \boldsymbol{B}^{-1}\boldsymbol{J}^T\boldsymbol{\Lambda}. \tag{6}$$

From (3) it can be seen that an arbitrary task behaviour can be achieved through the term $\boldsymbol{F}_{\tau|x}$, which is controlled through proper choice of the joint torques according to (5). Moreover, for a redundant robot the matrix $\bar{\boldsymbol{J}}^T$ has a non-empty null space, enabling a prioritized multiple task formulation. Finally, the prioritized formulation is solved as cascade of QPs of the following form:

$$\begin{aligned} \min_{\boldsymbol{\tau}_i} \; & \|\boldsymbol{A}_i\,\boldsymbol{\tau}_i - \boldsymbol{b}_i\|^2 + \epsilon\|\boldsymbol{\tau}_i\|^2 \\ \text{s.t. } & \boldsymbol{b}_\text{l} \le \boldsymbol{D}\,\boldsymbol{\tau}_i \le \boldsymbol{b}_\text{u} \\ & \boldsymbol{u}_\text{l} \le \boldsymbol{\tau}_i \le \boldsymbol{u}_\text{u} \\ & \boldsymbol{A}_{i-1}\boldsymbol{\tau}^*_{i-1} = \boldsymbol{A}_{i-1}\boldsymbol{\tau}_i \\ & \vdots \\ & \boldsymbol{A}_1\,\boldsymbol{\tau}^*_1 = \boldsymbol{A}_1\,\boldsymbol{\tau}_i \end{aligned}\quad, \tag{7}$$

for values of i ranging from $i = 1$ to the number of priority levels. Hierarchical force control is obtained by choosing $\boldsymbol{A}_i = \boldsymbol{J}_i\boldsymbol{B}^{-1}$ and $\boldsymbol{b}_i = \boldsymbol{\Lambda}_i^{-1}\boldsymbol{F}_i$; on top of it, Cartesian impedance control is achieved by selecting virtual spring-dampers as desired forces.

By using this formulation, it was possible to implement mixed a stiff/compliant behaviour with priority enforcement on the bi-manual upper body of our *CENTAURO* and *CO-MAN+* robots.

B. The floating-base case

The main focus of the present work is the extension of the algorithm that was presented in the previous section to the floating-base case. From a modeling point of view, a legged robot shows two structural differences with respect to a fixed-base one:

- *under-actuation*: floating-base robots can be described in terms of n degrees of freedom (one for each joint) plus six additional coordinates describing the pose of some robot link w.r.t. to an inertial world frame. Such a link is usually called *floating-base*. These additional DoFs are usually modelled by introducing a virtual six-dof chain of passive (unactuated) joints, which is known as *virtual chain*.
- *Contact forces*: legged robots must always interact with the environment in order to be controlled in their full $(6+n)$-dimensional coordinate space. Indeed, from the so-called centroidal dynamics equation, we know that the global motion of the robot is entirely given by the contact forces.

The virtual chain formulation allows to easily extend the fixed-base dynamics (2) to the floating-base case. We just augment the generalized coordinate vector with six virtual joints as in the following equation[1]:

$$\boldsymbol{q} = \begin{bmatrix}\boldsymbol{q}_u \\ \boldsymbol{q}_a\end{bmatrix}, \tag{8}$$

where $\boldsymbol{q} \in \mathbb{R}^{6+n}$ is obtained by stacking the configuration vector of virtual joints $\boldsymbol{q}_u \in \mathbb{R}^6$ with the one corresponding to the n actuated joints $\boldsymbol{q}_a \in \mathbb{R}^n$. Then, the floating-base dynamics equation is given by

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q},\dot{\boldsymbol{q}}) = \boldsymbol{S}\boldsymbol{\tau} + \boldsymbol{J}_\text{C}^T\boldsymbol{F}_\text{C} + \boldsymbol{J}^T\boldsymbol{F}_x; \tag{9}$$

[1] We use the subscripts "u" for *unactuated*, and "a" for *actuated*, respectively.

compared to (2), the joint torques vector is pre-multiplied by a matrix $\boldsymbol{S} \in \mathbb{R}^{(n+6)\times n}$ that maps actuation torques into torques for the full floating-base robot:

$$\boldsymbol{S} = \begin{bmatrix} \boldsymbol{0}_{6\times n} \\ \boldsymbol{I}_{n\times n} \end{bmatrix}. \tag{10}$$

Finally, contact forces are taken into account by introducing the Jacobian of all support links $\boldsymbol{J}_{\mathrm{C}} \in \mathbb{R}^{k\times(n+6)}$ and the corresponding overall contact wrench $\boldsymbol{F}_{\mathrm{C}} \in \mathbb{R}^{k}$, with k equal to the contact constraint dimension (e.g. $k = 12$ for a humanoid in double support).

By repeating the same steps as for the fixed-base case, the task dynamics is obtained as follows:

$$\boldsymbol{\Lambda}(\boldsymbol{q})\ddot{\boldsymbol{x}} + \boldsymbol{\mu}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{F}_{\tau|x} + \boldsymbol{F}_{\mathrm{C}|x} + \boldsymbol{F}_{x}. \tag{11}$$

It can be seen that the contribution from actuated joint torques is changed slightly w.r.t. the fixed-base case, and it has the following expression:

$$\boldsymbol{F}_{\tau|x} = \left(\bar{\boldsymbol{J}}^{T}\boldsymbol{S}\right)\boldsymbol{\tau}; \tag{12}$$

however, this does not affect the mathematical formulation of our prioritized controller. The most important change is given by the coupling between the task dynamics and the contact wrenches, through the term

$$\boldsymbol{F}_{\mathrm{C}|x} = \bar{\boldsymbol{J}}^{T}\boldsymbol{J}_{\mathrm{C}}^{T}\boldsymbol{F}_{C} = \boldsymbol{\Lambda}\left[\boldsymbol{J}\boldsymbol{B}^{-1}\boldsymbol{J}_{\mathrm{C}}^{T}\right]\boldsymbol{F}_{C}; \tag{13}$$

on this regard, it is worth noticing that coupling matrix between square brackets in (13) is non-zero for any task $\boldsymbol{x}$ that is specified w.r.t. the world frame: indeed, all tasks are coupled through the virtual chain connecting the floating-base to the world frame itself.

Such a coupling hinders the direct application of our algorithm [21]; in order to tackle the problem, the authors see three possible methods, as explained below:

- *pre-optimization of contact forces*. This approach has been successfully used in [3] for balancing control applied to the lower-body of the *TORO* robot. From the robot centroidal dynamics, the authors compute contact forces that achieve the desired center-of-mass and angular momentum behavior. More specifically, the angular momentum is used to perform orientation control of the robot base link.
  Once that such forces have been obtained, they can be made to disappear from the dynamics formulation by, for instance, redefining the bias torque vector as
  $$\hat{\boldsymbol{h}} = \boldsymbol{h} - \boldsymbol{J}_{\mathrm{C}}^{T}\boldsymbol{F}_{\mathrm{C}}. \tag{14}$$
  However, it is the authors' belief that such an idea contains a pitfall, namely that the centroidal dynamics of the robot is constrained to the value obtained during the pre-optimization phase (see remark in Section III-A.2). Consequently, the center-of-mass behaviour is always a first-priority task for the resulting controller, and the same applies to the angular momentum. This is undesirable for a prioritized controller; moreover, it is not clear which reference should be assigned to the angular momentum, given its non-holonomy as stated in [19].
- *Joint optimization of joint torque and contact forces*. This approach is conceptually similar to the one of [18], where the optimization is carried out over contact forces and joint accelerations. Full control over the task hierarchy and system momentum is retained at the cost of an increased number of optimization variables.
- *Post-optimization of contact forces*. A third option, which is the main contribution of the present work, consists in treating the floating-base robot as a fixed-base one, by mathematically replacing the sum of under-actuated joint torques and contact torques with an equivalent completely-actuated torque vector. Then, a post-optimization phase is set up in order to map back the obtained virtual joint torques and forces to equivalent contact wrenches. At the best of the authors' knowledge, this approach has not been explored before. A full description of such method is the subject of the following section.

C. Post-optimization of contact forces

Starting from (9), let us define an *equivalent fully-actuated torque vector* $\bar{\boldsymbol{\tau}}$

$$\bar{\boldsymbol{\tau}} = \boldsymbol{S}\boldsymbol{\tau} + \boldsymbol{J}_{\mathrm{C}}^{T}\boldsymbol{F}_{\mathrm{C}}; \tag{15}$$

with such a definition, the floating-base dynamics formally resembles the fixed-base one:

$$\boldsymbol{B}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{h}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \bar{\boldsymbol{\tau}} + \boldsymbol{J}^{T}\boldsymbol{F}_{x}; \tag{16}$$

consequently, our algorithm of [21] can be applied without any modification, yielding some optimized value $\bar{\boldsymbol{\tau}}^{*}$ for the fully-actuated torque vector (15) that permits to achieve the desired hierarchical motion and interaction.

Our problem is then to recover the contact force information, by taking into account the under-actuated nature of the system. Essentially, this amounts to solving the system of equations (15) for $\boldsymbol{\tau}$ and $\boldsymbol{F}_{C}$. On this regard, notice that the number of equations is $N_{\mathrm{eq}} = n + 6$, while the number of variables is $N_{\mathrm{var}} = n + k$; this means that in an unconstrained, full-rank case the problem admits $\infty^{N_{\mathrm{r}}}$ many solutions, with $N_{\mathrm{r}} = k - 6$.

We can reduce the number of unknowns by exploiting the structure of the actuation matrix (10). By focusing on the first six rows of (15) we obtain

$$\bar{\boldsymbol{\tau}}_{u} = \boldsymbol{J}_{\mathrm{C},u}^{T}\boldsymbol{F}_{\mathrm{C}}, \tag{17}$$

where the subscript "u" indicates the sub-matrix corresponding to the unactuated virtual joints. Equation (17) is torque-independent, and contains only the contact forces as variables. Once that these have been determined, joint torques can be recovered by looking at the bottom n rows of (15) and solving for $\boldsymbol{\tau}$:

$$\boldsymbol{\tau} = \bar{\boldsymbol{\tau}}_{a} - \boldsymbol{J}_{\mathrm{C},a}^{T}\boldsymbol{F}_{\mathrm{C}}. \tag{18}$$

The transpose of the unactuated part of the contact Jacobian acts as a *grasp matrix* $\boldsymbol{G} \in \mathbb{R}^{6\times k}$:

$$\boldsymbol{G} = \boldsymbol{J}_{\mathrm{C},u}^{T}; \tag{19}$$

hence, we can draw inspiration from [3] and obtain a least-squares solution for $\boldsymbol{F}_\text{C}$:

$$\boldsymbol{F}_\text{C}^* = \boldsymbol{G}^\dagger \,\bar{\boldsymbol{\tau}}_u, \tag{20}$$

where the dagger symbol $\dagger$ denotes the pseudo-inverse. However, we can also exploit a QP formulation in order to enforce inequality constraints, as for instance friction cones, as follows:

$$\begin{aligned} &\min_{\boldsymbol{F}_\text{C}} \; \|\boldsymbol{G}\boldsymbol{F}_\text{C} - \bar{\boldsymbol{\tau}}_u\|^2 \\ &\text{s.t. } \boldsymbol{b}_\text{l} \leq \boldsymbol{D}\,\boldsymbol{F}_\text{C} \leq \boldsymbol{b}_\text{u} \\ &\qquad \boldsymbol{u}_\text{l} \leq \boldsymbol{F}_\text{C} \leq \boldsymbol{u}_\text{u}. \end{aligned} \tag{21}$$

Torque constraints can be introduced as well by considering the dependency on the contact forces as given by (18).

D. Discussion

The main advantage of the proposed post-optimization formulation with respect to the pre-optimization is clearly given by the proper handling of priorities between tasks, since neither the Center of Mass (CoM) motion nor the robot angular momentum need to be set *a-priori*. Moreover, the user is relieved from specifying a target angular momentum to the robot, which seems to be problematic as discussed in Section II-B. With our approach, target values for the centroidal dynamics are obtained from the fixed-base solution $\bar{\boldsymbol{\tau}}^*$, and *only then* the corresponding contact wrenches are optimized.

The main drawback is that the first optimization stage may give back a solution that is not feasible under force constraints (e.g. friction cones), which means that the objective value of (21) will be greater than zero. This could lead to loss of performance and, in the worst case, instability of the closed loop system. However, this situation can eventually be used to detect the need to perform a *recovery* action, e.g. a step.

On the other hand, a joint torque-force optimization strategy could perform better in such a case, since constraints on forces are taken into account for motion/force control as well. The price to pay is an increased number of decision variables.

As a final consideration, our post-optimization approach gives us the possibility of adapting our fixed-base formulation to the floating-base case without any need for modification, since the under-actuation is dealt with at a separate stage that is *completely decoupled* from force/motion control. As a matter of fact, the proposed formulation permits to adapt *any* fixed-base torque controller to the floating base case, while fully retaining its behavior.

III. Simulations and Preliminary Experiments

In order to validate the proposed extension of our controller [21] to the floating-base case, we set up two simulation scenarios in Gazebo involving two different legged platforms: *CENTAURO* [23], a 39 degrees-of-freedom (DoF) hybrid wheeled-legged quadruped equipped with a humanoid upper-body and *COMAN+*, which is a 28 DoF, 1.70 meters tall humanoid robot. Both robots are fully torque-controlled by feeding back the measured link-side joint torques. Moreover, our control architecture *XBotCore* [24] allows for transparent switch between simulation and real hardware, and hard real-time control on the actual robot. We conclude the section with a brief overview of some preliminary experimental results on the *COMAN+* robot.

A. Gazebo simulations

The controller's first stage is formalized in the following way:

$$\begin{pmatrix} \left(\sum_i \;{}^{\text{World}}\mathcal{T}_{\text{Foot}_i}\right) / \\ {}^{\text{World}}\mathcal{T}_{\text{Waist}} / \\ \left({}^{\text{World}}\mathcal{T}_{\text{LHand}} + {}^{\text{World}}\mathcal{T}_{\text{RHand}}\right) / \\ \mathcal{T}_{\text{Posture}} \end{pmatrix} << \left(\mathcal{C}_{\substack{\text{Joint Torque}\\ \text{Limits}}}\right), \tag{22}$$

where the symbol ${}^{\text{A}}\mathcal{T}_{\text{B}}$ denotes a Cartesian impedance task of the frame B relative to the frame A; such a task is implemented by commanding simple virtual wrenches as in the following expression:

$$\boldsymbol{F}_d = \mathbf{K}_d\left(\boldsymbol{x}_d - \boldsymbol{x}\right) + \mathbf{D}_d\left(\dot{\mathbf{x}}_d - \dot{\boldsymbol{x}}\right), \tag{23}$$

where $\mathbf{x}$, $\mathbf{x}_d$, $\dot{\mathbf{x}}$ and $\dot{\mathbf{x}}_d$ represent the actual and desired Cartesian poses and twists, respectively, whereas $\mathbf{K}_d$ and $\mathbf{D}_d$ denote the desired Cartesian impedance. These are combined by means of the operators "+" and "/", which are used to set aggregation and null-space relations, respectively. Finally, the symbol "$<<$" denotes insertion of constraints into the problem.

Recall from Section II-C that the first stage of our method computes a fully actuated torque vector, that is then mapped to an under-actuated torque vector via post-optimization of contact forces. We implement the post-optimization stage as in (21), considering linearized friction cones as inequality constraints:

$$|\boldsymbol{F}_t| \leq \frac{\sqrt{2}\mu}{2}\boldsymbol{F}_n, \quad \boldsymbol{F}_n \geq 0; \tag{24}$$

where $\boldsymbol{F}_t$ and $\boldsymbol{F}_n$ are the tangential and normal components of contact forces, respectively, and $\mu = 0.3$ is the considered friction coefficient. As a final observation, for all our simulation experiments the state of the floating-base link is directly taken from the simulator.

1) CENTAURO balancing under external disturbances: The first simulation scenario consists on a balancing task for the *CENTAURO* robot under the disturbance of external forces applied on the robot waist. Screen shots from the performed simulations are reported in Fig. 1. The desired Cartesian impedance for the waist position task has been set equal to: $\mathbf{K}_d = \begin{bmatrix}500 & 500 & 500\end{bmatrix} \frac{N}{m}$ and $\mathbf{D}_d = \begin{bmatrix}200 & 200 & 200\end{bmatrix} \frac{Ns}{m}$.

A constant force of 200 N is first applied downward along the z-direction for 2 s. The obtained contact forces along the z-direction are shown in Fig. 2(a), while the corresponding waist position error is shown in Fig. 2(b). As expected, the maximum value of the actual waist position error (blue solid

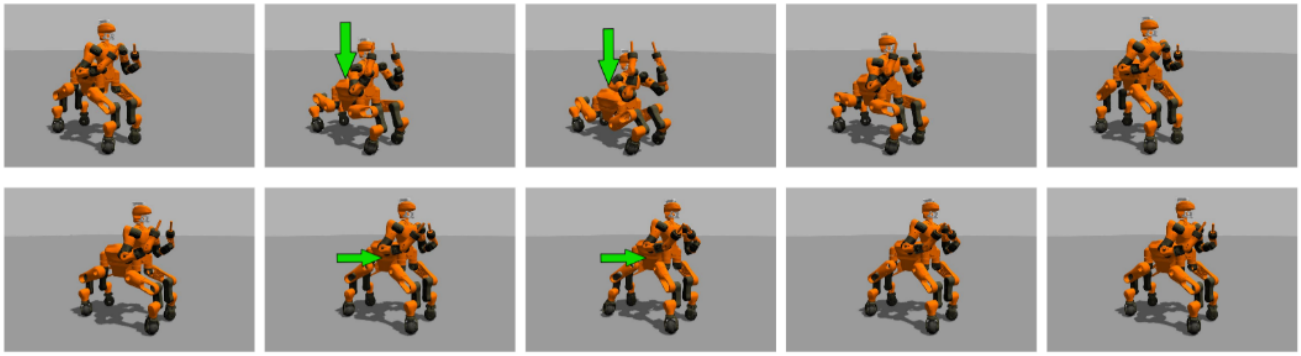
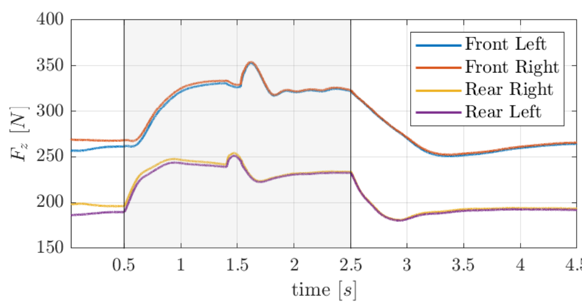


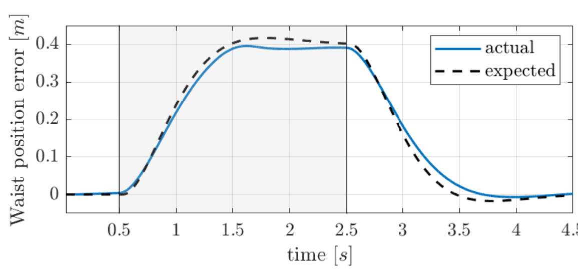
Fig. 1. Screen shots from *CENTAURO* simulations in Gazebo. In the upper plots a constant force of 200 N is applied downward on the robot waist, while in the lower plots a constant force of 90 N is applied sideways.

line), approximately 0.4 m, is consistent with the expected value (black solid line) obtained through (23).

To further highlight the impact of friction cones, a second simulation involves the application of a constant force of 90 N sideways along the y-direction, see Fig. 3. Note that, according to (24), the x and y components of the contact force on the rear right leg (yellow lines) are driven to zero at once with the z-component.



(a) z-component of contact forces.



(b) Actual (blue solid line) vs. expected (black dashed line) waist position error along the pushing direction, i.e. the z-direction.

Fig. 2. Time histories from *CENTAURO* simulation: an external constant force of 200 N is applied downward (z-direction) on the robot waist for 2 s (shaded area).

2) COMAN+ picking a box: In the second simulation scenario we consider the *COMAN+* robot picking a two kilogram box from the knee level. However, the box weight is not known to the controller. The task has to be performed using whole-body motions to reach the box and pick it up.

Notice that, in order to be able to reach the box, the

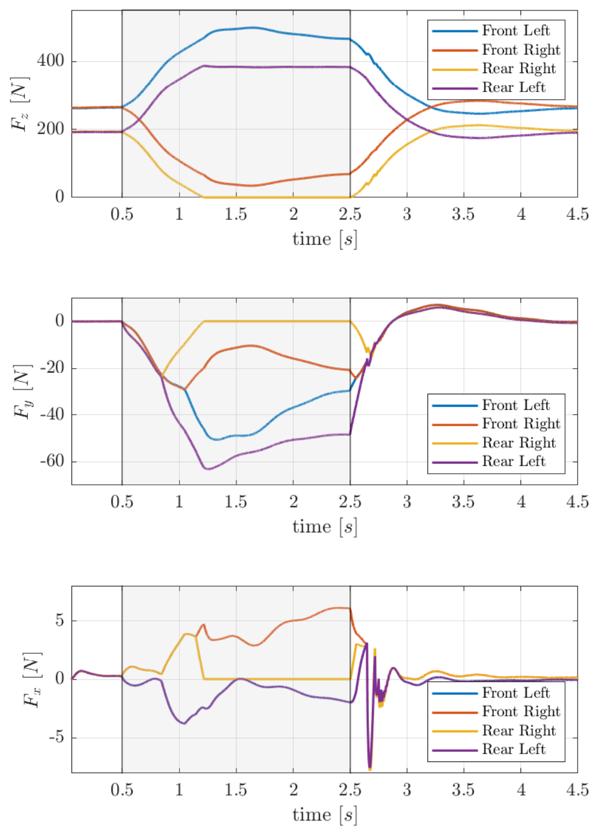
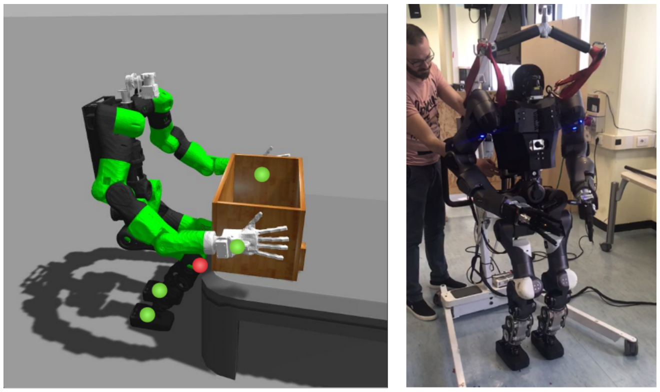


Fig. 3. Contact forces' time history from *CENTAURO* simulation: an external constant force of 90 N is applied sideways (y-direction) on the robot waist for 2 s (shaded area).

z-translation task is removed from the waist control. As a side-note, the orientation is managed using a quaternion formulation as in [25]. The outcome of the simulation experiment can be seen in the accompanying video[2].

Remark: Notice that the obtained whole-body motion would be impossible under a pre-optimization framework, in which the CoM trajectory is decided beforehand. In this

[2]The attached video is also available at `https://youtu.be/p8fwwV_zZa8`



a. b.

Fig. 4. On the left, *COMAN+* in simulation. During the squat motion needed to pick up the box, the knees of the robot (red dot) enter in contact with the table generating unwanted forces which perturb the CoM. Green dots represent intentional contacts. On the right preliminary experiments with the real *COMAN+* platform

case, the only solution is to explicitly decrease the CoM height in order for the hands to reach the box.

B. Preliminary Experiments on the COMAN+ platform

In this preliminary experiment we test the presented controller, with the real humanoid robot *COMAN+*. As the only difference compared to our simulations, in this case we need to estimate the state of our base-link in terms of pose and twist w.r.t. an inertial world frame. In order to do so, we simply fuse kinematic information from the lower-body with IMU measurements, assuming fixed contacts.

After some controller tuning, we managed to have the robot balance on two feet, even when subject to small external perturbations (see Figure 4*b*). Despite the encouraging results, we experienced instabilities and strong vibrations, in a very similar way as described in [26]; indeed, implementation of full-torque controllers on legged robots appears to be challenging, and it needs further investigation from the authors in order to improve the robustness of the controller and successfully deploy it to real systems.

IV. Conclusion and Future Works

In this work, a novel method to to formulate task-space inverse dynamics of floating-base robots has been developed as a follow up of the work in [21]. The method consists in a two step optimization: during a first optimization stage, a fully-actuated torque vector that realizes the desired tasks is found using [21]; such a vector contains non-zero torques at the virtual joints, and is not directly applicable to the robot. Hence, during a second optimization stage this wrench is mapped back to the available contacts. The proposed method, in comparison to [3], has the main advantage of avoiding to constrain the robot centroidal dynamics.

The method has been tested first in simulation using two floating base robots with substantially different kinematic structure: the *CENTAURO* and the *COMAN+* robot. The first is a wheeled, quadruped robot with humanoid torso, the second a bipedal humanoid robot. Furthermore, preliminary results are reported in the real *COMAN+* robot which is subject to small perturbations.

Future works will address first the implementation on the real hardware which, as stated previously, at the moment presents some stability problems, very similar to the one in [26]. Second we would like to explore a single-stage torque-force optimization, and compare it to [18]. Finally, it is the authors' belief that this method can be extended to other problem that can be treated by introducing virtual kinematic chains, for example when contact forces to pick a box has to be computed.

Acknowledgment

The research leading to these results has received funding from the European Unions Horizon 2020 research and innovation program under grant agreements No. 644839 (CENTAURO) and No. 644727 (CogIMon).

References

[1] S.-H. Hyon, J. G. Hale, G. Cheng, *et al.*, "Full-body compliant human-humanoid interaction: Balancing in the presence of unknown external forces," *IEEE Trans. Robotics*, vol. 23, no. 5, pp. 884–898, 2007.

[2] S.-H. Hyon, "Compliant terrain adaptation for biped humanoids without measuring ground surface and contact forces," *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 171–178, 2009.

[3] C. Ott, M. A. Roa, and G. Hirzinger, "Posture and balance control for biped robots based on contact force optimization," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*. IEEE, 2011, pp. 26–33.

[4] B. Henze, A. Dietrich, and C. Ott, "An approach to combine balancing with hierarchical whole-body control for legged humanoid robots," *IEEE Robotics and Automation Letters*, vol. 1, no. 2, pp. 700–707, 2016.

[5] B. Henze, A. Dietrich, M. A. Roa, and C. Ott, "Multi-contact balancing of humanoid robots in confined spaces: Utilizing knee contacts," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 697–704.

[6] S.-H. Lee and A. Goswami, "Ground reaction force control at each foot: A momentum-based humanoid balance controller for non-level and non-stationary ground," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 3157–3162.

[7] ——, "A momentum-based balance controller for humanoid robots on non-level and non-stationary ground," *Autonomous Robots*, vol. 33, no. 4, pp. 399–414, 2012.

[8] B. J. Stephens and C. G. Atkeson, "Dynamic balance force control for compliant humanoid robots," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 1248–1255.

[9] O. Khatib, L. Sentis, J. Park, and J. Warren, "Whole-body dynamic behavior and control of human-like robots," *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 29–43, 2004.

[10] L. Sentis and O. Khatib, "Synthesis of whole-body behaviors through hierarchical control of behavioral primitives," *International Journal of Humanoid Robotics*, vol. 2, no. 04, pp. 505–518, 2005.

[11] L. Sentis, J. Park, and O. Khatib, "Compliant control of multicontact and center-of-mass behaviors in humanoid robots," *IEEE Transactions on robotics*, vol. 26, no. 3, pp. 483–501, 2010.

[12] L. Righetti, J. Buchli, M. Mistry, and S. Schaal, "Inverse dynamics control of floating-base robots with external constraints: A unified view," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1085–1090.

[13] L. Righetti, J. Buchli, M. Mistry, M. Kalakrishnan, and S. Schaal, "Optimal distribution of contact forces with inverse-dynamics control," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 280–298, 2013.

[14] N. Mansard, "A dedicated solver for fast operational-space inverse dynamics," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4943–4949.

[15] L. Saab, O. E. Ramos, F. Keith, N. Mansard, P. Soueres, and J.-Y. Fourquet, "Dynamic whole-body motion generation under rigid contacts and other unilateral constraints," *IEEE Transactions on Robotics*, vol. 29, no. 2, pp. 346–362, 2013.

[16] A. Escande, N. Mansard, and P.-B. Wieber, "Hierarchical quadratic programming: Fast online humanoid-robot motion generation," *The International Journal of Robotics Research*, vol. 33, no. 7, pp. 1006–1028, 2014.

[17] A. Herzog, L. Righetti, F. Grimminger, P. Pastor, and S. Schaal, "Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 981–988.

[18] A. Herzog, N. Rotella, S. Mason, F. Grimminger, S. Schaal, and L. Righetti, "Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid," *Autonomous Robots*, vol. 40, no. 3, pp. 473–491, 2016.

[19] P.-B. Wieber, "Holonomy and nonholonomy in the dynamics of articulated motion," in *Fast motions in biomechanics and robotics*. Springer, 2006, pp. 411–425.

[20] P.-B. Wieber, R. Tedrake, and S. Kuindersma, "Modeling and control of legged robots," in *Springer handbook of robotics*. Springer, 2016, pp. 1203–1234.

[21] E. Mingo Hoffman, A. Laurenzi, N. G. Tsagarakis, and D. G. Caldwell, "Multi-priority cartesian impedance control based on quadratic programming optimization," in *IEEE International Conference on Robotics and Automation, ICRA*, 2018.

[22] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: modelling, planning and control*. Springer Science & Business Media, 2010.

[23] L. Baccelliere, N. Kashiri, L. Muratore, A. Laurenzi, M. Kamedula, A. Margan, S. Cordasco, J. Malzahn, and N. G. Tsagarakis, "Development of a human size and strength compliant bi-manual platform for realistic heavy manipulation tasks," in *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*. IEEE, 2017, pp. 5594–5601.

[24] L. Muratore, A. Laurenzi, E. M. Hoffman, A. Rocchi, D. G. Caldwell, and N. G. Tsagarakis, "Xbotcore: A real-time cross-robot software platform," in *Robotic Computing (IRC), IEEE International Conference on*. IEEE, 2017, pp. 77–80.

[25] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational space control: A theoretical and empirical comparison," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 737–757, 2008.

[26] J. Englsberger, G. Mesesan, A. Werner, and C. Ott, "Torque-based dynamic walking - A long way from simulation to experiment," in *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, Jan. 2018.