Passive Whole-Body Control for Quadruped Robots: Experimental Validation over Challenging Terrain

Preprint · March 2019

DOI: 10.1109/LRA.2019.2908502

4 authors:

Shamel Fahmi
Istituto Italiano di Tecnologia
8 PUBLICATIONS **88** CITATIONS
SEE PROFILE

Carlos Mastalli
The University of Edinburgh
36 PUBLICATIONS **709** CITATIONS
SEE PROFILE

Michele Focchi
Istituto Italiano di Tecnologia
66 PUBLICATIONS **2,274** CITATIONS
SEE PROFILE

Claudio Semini
Istituto Italiano di Tecnologia
140 PUBLICATIONS **3,554** CITATIONS
SEE PROFILE

Passive Whole-body Control for Quadruped Robots: Experimental Validation over Challenging Terrain

Shamel Fahmi, Carlos Mastalli, Michele Focchi and Claudio Semini

Submitted: October 15, 2018. Accepted: March 11, 2019.

To be published in:

Robotics and Automation Letters (RA-L), 2019.

To cite this paper:

S. Fahmi, C. Mastalli, M. Focchi and C. Semini, "Passive Whole-body Control for Quadruped Robots: Experimental Validation over Challenging Terrain", *IEEE Robotics and Automation Letters*, 2019.

For this and other publications from the Dynamic Legged Systems lab (DLS):
`https://dls.iit.it/dls-publications`

Passive Whole-body Control for Quadruped Robots: Experimental Validation over Challenging Terrain

Shamel Fahmi[1†], Carlos Mastalli[1,2†], Michele Focchi[1] and Claudio Semini[1]

***Abstract*—We present experimental results using a passive whole-body control approach for quadruped robots that achieves *dynamic* locomotion while compliantly balancing the robot's trunk. We formulate the motion tracking as a Quadratic Program (QP) that takes into account the full robot rigid body dynamics, the actuation limits, the joint limits and the contact interaction. We analyze the controller's robustness against inaccurate friction coefficient estimates and unstable footholds, as well as its capability to redistribute the load as a consequence of enforcing actuation limits. Additionally, we present practical implementation details gained from the experience with the real platform. Extensive experimental trials on the 90 kg Hydraulically actuated Quadruped (HyQ) robot validate the capabilities of this controller under various terrain conditions and gaits. The proposed approach is superior for accurate execution of highly dynamic motions with respect to the current state of the art.**

***Index Terms*—whole-body control, quadrupedal locomotion, optimization, passivity, active impedance**

I. Introduction

ACHIEVING dynamic locomotion requires reasoning about the robot's dynamics, actuation limits and interaction with the environment while traversing challenging terrain (such as rough or sloped terrain). Optimization-based techniques can be exploited to attain these objectives in locomotion planning and control of legged robots. For instance, one approach is to use non-linear Model Predictive Control (MPC) while taking into consideration the full dynamics of the robot. Yet, it is often challenging to meet real-time requirements because the solver can get stuck in local minima, unless proper warm-starting is used [1]. Thus, current research often relies on low dimensional models or constraint relaxation approaches to meet such requirements (e.g. [2]). Other approaches rely on decoupling the motion planning from the motion control [3, 4, 5]. Along this line, an optimization-based motion planner could rely on low dimensional models to compute Center of Mass (CoM) trajectories and footholds while a locomotion controller tracks these trajectories.

Many recent contributions in locomotion control have been proposed in the literature that were successfully tested on bipeds and quadrupeds (e.g. [6, 7, 8, 9, 5, 10]). Some of them are based on quasi-static assumptions or lower dimensional models [11, 12, 13]. This often limits the dynamic locomotion capabilities of the robot [6]. Consequently, another approach, that is preferable for dynamic motion, is based on Whole-Body Control (WBC). WBC facilitates such decoupling between the motion planning and control in such a way that it is easy to accomplish multiple tasks while respecting the robot's behavior [9]. These tasks might include motion tasks for the robot's end effectors (legs and feet) [8, 9], but also could be utilized for contacts anywhere on the robot's body [14] or for a cooperative manipulation task between robots [15]. WBC casts the locomotion controller as an *optimization* problem, in which, by incorporating the full dynamics of the legged robot, all of its Degrees of Freedom (DoFs) are exploited in order to spread the desired motion tasks globally to all the joints. This allows us to reason about multiple tasks and solve them in an optimization fashion while respecting the full system dynamics and the actuation and interaction constraints. WBC relies on the fact that robot dynamics and constraints could be formulated, at each loop, as linear constraints with a convex cost function (i.e., a QP) [2]. This allows us to solve the optimization problem in real-time.

Passivity theory is proven to guarantee a certain degree of robustness during interaction with the environment [16]. For that reason, such tool is commonly exploited in the design of locomotion controllers to ensure a passive contact interaction. Passivity based WBC in humanoids was introduced first by [17] to effectively balance the robot when experiencing contacts. By providing compliant tracking and gravity compensation, the humanoid was able to adapt to unknown disturbances. The same approach was further extended first by [12] and later by [8]. The former extended [17] to posture control, while the latter analyzed the passivity of a humanoid robot in multi-contact scenarios (by exploiting the similarity with PD+ control [18]).

In our previous work [13], the locomotion controller was designed for *quasi-static* motions using only the robot's centroidal dynamics. Under that assumption, we noticed that during dynamic motions, the effect of the leg dynamics no longer negligible; and thus, it becomes necessary to abandon the quasi-static assumption to achieve good tracking. Second, since the robot is constantly interacting with the environment (especially during walking and running), it is crucial to ensure a compliant and passive interaction. For these reasons, in this paper, we improve our previous work [13] by implementing a passivity based WBC that incorporates the full robot dynamics and interacts compliantly with the environment, while satisfy-

Manuscript received: October, 15, 2018; Revised February, 10, 2019; Accepted 11, March, 2019.

This paper was recommended for publication by Editor Kyu-Jin Cho upon evaluation of the Associate Editor and Reviewers' comments. This paper was also presented at the RoboSoft 2019 conference.

†These authors contributed equally to this work.

[1]Dynamic Legged Systems Lab, Istituto Italiano di Tecnologia (IIT), Genova, Italy. firstname.lastname@iit.it

[2]Gepetto Team, LAAS-CNRS, Toulouse, France. carlos.mastalli@laas.fr.

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

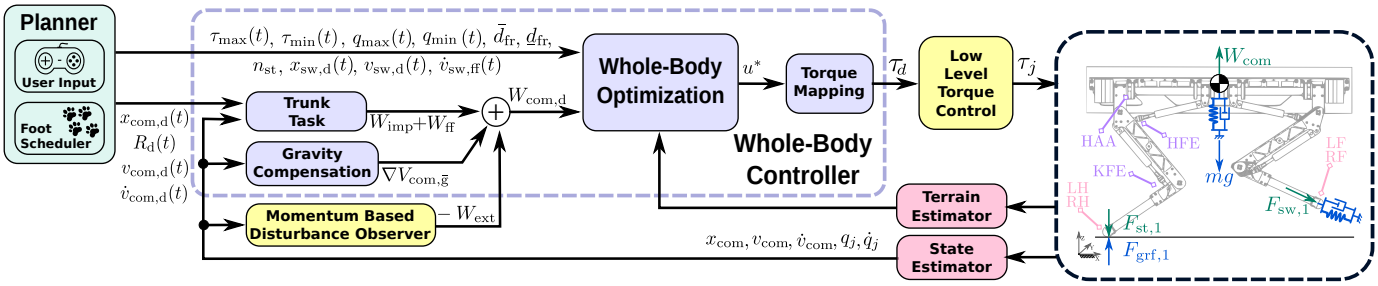

Fig. 1: Overview of the whole-body controller as part of our locomotion framework. The dashed black box (cartouche) presents an overview of the robot's joints, feet and generated wrenches and forces. LH, LF, RH and RF are Left-Hind, Left-Front, Right-Hind and Righ-Front legs, respectively. HAA, HFE and KFE are Hip Abduction/Adduction, Hip Flexion/Extension and Knee Flexion/Extension, respectively.

ing the kinematic and torque limits. Our WBC implementation is capable of achieving *faster* dynamic motions than our previous work. We also integrate terrain mapping and state estimation on-board and present some practical implementation details gained from the experience with the real platform.

Contributions: In this paper, we mainly present *experimental* contributions in which we demonstrate the effectiveness of the controller both in simulation and experiments on HyQ. Compared to previous work on passivity-based WBC [8, 12], in which experiments were conducted on the robot while standing (not walking or running), we tested our controller on HyQ during crawling and trotting. Similar to the recent successful work of [5] and [19] in quadrupedal locomotion over rough terrain, we used similar terrain templates to present experiments of our passive WBC on HyQ using multiple gaits over slopes and rough terrain of different heights.

The rest of this paper is structured as follows: In Section II we present the detailed formulation and design of our WBC followed by its passivity analysis in Section III. Section IV presents further crucial implementation details. Finally we present our simulation and experimental results in Section V followed by our conclusions in Section VI.

II. Whole-Body Controller (WBC)

In this section we present and formulate our WBC. Figure 1 depicts the main components of our locomotion framework. Given high-level user velocity commands, the planner generates a reference motion online [20] or offline [4], and provides it to the WBC. Such references include the desired trajectories for CoM, trunk orientation and swing legs. The *state estimator* supplies the controller with an estimate of the actual state of the robot, by fusing leg odometry, inertial sensing, visual odometry and LIDAR while, the *terrain estimator*, provides an estimate of the terrain inclination (i.e. surface normal). Finally, there is a momentum-based observer that estimates external disturbances [20] and a lower-level torque controller.

The goal of the designed WBC is to keep the quadruped robot balanced (during running, walking or standing) while interacting passively with the environment. The motion tasks of a quadruped robot can be categorized into a *trunk task* and a *swing task*. The trunk task regulates the position of the CoM and the orientation of the trunk[1] and is achieved by implementing a Cartesian-based impedance controller with a feed-forward term[2]. The swing task regulates the swing foot trajectory in order to place it in the desired location while achieving enough clearance from the terrain. Similar to the trunk task, the swing task is achieved by implementing a Cartesian-based impedance controller with a feed-forward term. The WBC realizes these tasks by computing the optimal generalized accelerations and contact forces [6] via QP and mapping them to the desired joint torques while taking into account the full dynamics of the robot, the properties of the terrain (*friction constraints*), the unilaterality of the contacts (e.g. the legs can only push and not pull) (*unilateral constraints*), and the actuator's *torque/kinematic limits*. The desired torques, will be sent to the lower-level (torque) controller.

A. Robot Model

For a legged robot with n DoFs and c feet, the forward kinematics of each foot is defined by n_a coordinates[3]. The total dimension of the feet operational space is $n_f = n_a c$. This can be separated into stance ($n_{\text{st}} = n_a c_{\text{st}}$) and swing feet ($n_{\text{sw}} = n_a c_{\text{sw}}$). Since we are interested in regulating the position of the CoM, we formulate the dynamics in terms of the CoM, using its velocity rather than the base velocity[4] [12]. Assuming that all the external forces are exerted on the *stance feet*, we write the equation of motion that describes the full dynamics of the robot as:

$$\underbrace{\begin{bmatrix} M_{\text{com}} & 0_{6\times n} \\ 0_{n\times 6} & \bar{M}_j \end{bmatrix}}_{M(q)} \underbrace{\begin{bmatrix} \dot{v}_{\text{com}} \\ \ddot{q}_j \end{bmatrix}}_{\ddot{q}} + \underbrace{\begin{bmatrix} h_{com} \\ \bar{h}_j \end{bmatrix}}_{h} = \begin{bmatrix} 0_{6\times n} \\ \tau_j \end{bmatrix} + \underbrace{\begin{bmatrix} J^T_{\text{st,com}} \\ J^T_{\text{st,j}} \end{bmatrix}}_{J_{\text{st}}(q)^T} F_{\text{grf}} \tag{1}$$

where the first 6 rows represent the (un-actuated) floating base part and the remaining n rows represent the actuated part. $q \in SE(3) \times \mathbb{R}^n$ represents the pose of the whole floating-base system while $\dot{q} = \begin{bmatrix} v^T_{\text{com}} & \dot{q}^T_j \end{bmatrix}^T \in \mathbb{R}^{6+n}$ and $\ddot{q} = [\dot{v}^T_{\text{com}} \ \ddot{q}^T_j]^T \in \mathbb{R}^{6+n}$ are the vectors of generalized velocities and accelerations, respectively. $v_{\text{com}} = [\dot{x}^T_{\text{com}} \ \omega^T_b]^T \in \mathbb{R}^6$ and $\dot{v}_{\text{com}} = [\ddot{x}^T_{\text{com}} \ \dot{\omega}^T_b]^T \in \mathbb{R}^6$ are the spatial velocity and acceleration of the floating-base expressed at the CoM. $M(q) \in \mathbb{R}^{(6+n)\times(6+n)}$ is the inertia matrix, where $M_{\text{com}}(q) \in \mathbb{R}^{6\times 6}$ is the composite rigid body inertia matrix of the robot expressed at the CoM. $h \in \mathbb{R}^{6+n}$ is the force vector that

[1]Since HyQ is not equipped with arms, it suffices for us to control the trunk orientation instead of the whole robot angular momentum.

[2]This is similar to a PD+ controller [18].

[3]Without the loss of generality, we consider a quadruped robot with $n = 12$ DoFs with point feet, where $c = 4$ and $n_a = 3$.

[4]In this coordinate system, the inertia matrix is block diagonal [12]. For the detailed implementation of the dynamics using the base velocity, see [17].

accounts for Coriolis, centrifugal, and gravitational forces[5]. $\tau \in \mathbb{R}^n$ are the actuated joint torques while $F_{\rm grf} \in \mathbb{R}^{n_{\rm st}}$ is the vector of Ground Reaction Forces (GRFs) (contact forces). In this context, the floating base Jacobian $J \in \mathbb{R}^{n_{\rm f}\times(6+n)}$ is separated into swing Jacobian $J_{\rm sw} \in \mathbb{R}^{n_{\rm sw}\times(6+n)}$ and stance Jacobian $J_{\rm st} \in \mathbb{R}^{n_{\rm st}\times(6+n)}$ which could be further expanded into $J_{\rm st,com} \in \mathbb{R}^{n_{\rm st}\times 6}$, $J_{\rm st,j} \in \mathbb{R}^{n_{\rm st}\times n}$, $J_{\rm sw,com} \in \mathbb{R}^{n_{\rm sw}\times 6}$ and $J_{\rm sw,j} \in \mathbb{R}^{n_{\rm sw}\times n}$. The operator $\bar{[\cdot]}$ denotes the matrices/vectors recomputed after the coordinate transform to the CoM [17]. Following the sign convention in Fig. 1, recalling the first 6 rows in (1), and by defining the gravito-inertial *CoM wrench* as $W_{\rm com} = M_{\rm com}\dot{v}_{\rm com} + h_{com} \in \mathbb{R}^6$, we can write the *floating-base dynamics* as:

$$W_{\rm com} = J_{\rm st,com}^T F_{\rm grf} \tag{2}$$

such that $J_{\rm st,com}^T$ maps $F_{\rm grf}$ to the CoM wrench space.

The feet velocities $v = [v_{\rm st}^T\ v_{\rm sw}^T]^T \in \mathbb{R}^{n_{\rm f}}$ could be separated into stance $v_{\rm st} \in \mathbb{R}^{n_{\rm st}}$ and swing $v_{\rm sw} \in \mathbb{R}^{n_{\rm sw}}$ feet velocities. The mapping between v and the generalized velocities $\dot{q}$ is:

$$v = J\dot{q} \tag{3a}$$

$$v = \begin{bmatrix} J_{\rm com} & J_j \end{bmatrix} \begin{bmatrix} v_{\rm com} \\ \dot{q}_j \end{bmatrix} = J_{\rm com} v_{\rm com} + J_j \dot{q}_j \tag{3b}$$

such that $J_{\rm com} \in \mathbb{R}^{n_{\rm f}\times 6}$ and $J_j \in \mathbb{R}^{n_{\rm f}\times n}$. Similar to the feet velocities, we split the feet force vector $F = [F_{\rm st}^T\ F_{\rm sw}^T]^T \in \mathbb{R}^{n_{\rm f}}$, into $F_{\rm st} \in \mathbb{R}^{n_{\rm st}}$ and $F_{\rm sw} \in \mathbb{R}^{n_{\rm sw}}$.

Assumption 1: The robot is walking over rigid terrain in which the stance feet do not move (i.e., $v_{\rm st} = \dot{v}_{\rm st} = 0$).

B. Trunk and Swing Leg Control Tasks

To compliantly achieve a desired motion of the *trunk*, we define the desired wrench at the CoM $W_{\rm com,d}$ using the following: 1) a Cartesian impedance at the CoM $W_{\rm imp}$ that is represented by a stiffness term ($\nabla V_{\rm com,K} = K_{\rm com}\Delta x_{\rm com}$) with positive definite stiffness matrix $K_{\rm com} \in R^{6\times 6}$ and a damping term ($D_{\rm com}\Delta v_{\rm com}$) with positive definite damping matrix $D_{\rm com} \in R^{6\times 6}$, 2) a virtual gravitational potential gradient to render gravity compensation ($\nabla V_{\rm com,\bar{g}} = mg$)[6], 3) a feedforward term to improve tracking ($W_{\rm ff} = M_{\rm com}\dot{v}_{\rm com,d}$) and a compensation term for external disturbances $-W_{\rm ext}$ [20]:

$$W_{\rm com,d} = W_{\rm imp} + \nabla V_{\rm com,\bar{g}} + W_{\rm ff} - W_{\rm ext} \tag{4a}$$

$$W_{\rm imp} = \nabla V_{\rm com,K} + D_{\rm com}\Delta v_{\rm com} \tag{4b}$$

such that $\Delta x_{\rm com} = x_{\rm com,d} - x_{\rm com}$, $\Delta v_{\rm com} = v_{\rm com,d} - v_{\rm com}$ are the tracking errors $\in \mathbb{R}^6$ of the position and velocity, respectively.

Similarly, the tracking of the swing task is obtained by the virtual force $F_{\rm sw,d} \in \mathbb{R}^{n_{\rm sw}}$. This is generated by 1) a Cartesian impedance at the swing foot that is represented by a stiffness term ($\nabla V_{\rm sw} = K_{\rm sw}\Delta x_{\rm sw}$) with positive definite stiffness matrix $K_{\rm sw} \in R^{n_{\rm sw}\times n_{\rm sw}}$ and a damping term ($D_{\rm sw}\Delta v_{\rm sw}$) with positive definite damping matrix $D_{\rm sw} \in R^{n_{\rm sw}\times n_{\rm sw}}$, and 2) a feedforward term to improve tracking ($F_{\rm sw,ff} = M_{\rm sw}\dot{v}_{\rm sw,ff}$):

$$F_{\rm sw,d} = \nabla V_{\rm sw} + D_{\rm sw}\Delta v_{\rm sw} + F_{\rm sw,ff} \tag{5}$$

such that $\Delta x_{\rm sw} = x_{\rm sw,d} - x_{\rm sw}$ and $\Delta v_{\rm sw} = v_{\rm sw,d} - v_{\rm sw}$ are the tracking errors of the swing feet positions and velocities respectively. Alternatively, it is possible to write this task at the acceleration level, with the difference that the gains $\rm K_{sw}$ and $\rm D_{sw}$ have no physical meaning:

$$\dot{v}_{\rm sw,d} = \dot{v}_{\rm sw,ff} + \mathrm{K_{sw}}\Delta x_{\rm sw} + \mathrm{D_{sw}}\Delta v_{\rm sw} \tag{6}$$

C. Optimization

To fulfill the motion tasks in Section II-B and to distribute the load on the stance feet, while respecting the mentioned constraints, we formulate the QP:

$$\min_{u=[\ddot{q}^T\ F_{\rm grf}^T]^T} \|W_{\rm com} - W_{\rm com,d}\|_Q^2 + \|u\|_R^2 \tag{7a}$$

$$\text{s.t.} \qquad Au = b \tag{7b}$$

$$\underline{d} < Cu < \bar{d} \tag{7c}$$

such that our decision variables $u = [\ddot{q}^T\ F_{\rm grf}^T]^T \in \mathbb{R}^{6+n+n_{\rm st}}$ are the generalized accelerations $\ddot{q}$ and the contact forces $F_{\rm grf}$. The cost function (7a) is designed to minimize the *trunk task* and to regularize the solution. The equality constraints (7b) encode dynamic consistency, stance constraints and swing tasks. The inequality constraints (7c) encode friction constraints, joint kinematic and torque limits. All constraints are stacked in the matrix $A^T = \begin{bmatrix} A_p^T & A_{\rm st}^T & A_{\rm sw}^T \end{bmatrix}$ and $C^T = \begin{bmatrix} C_{\rm fr}^T & C_j^T & C_\tau^T \end{bmatrix}$ and detailed in the following sections.

1) Cost: The first term of the cost in (7a) represents the *tracking* error between the actual $W_{\rm com}$ and the desired $W_{\rm com,d}$ CoM wrenches from (2) and (4a) respectively. Since $W_{\rm com}$ is not a decision variable, we compute it from the contact forces (see (2)) and re-write $\|W_{\rm com} - W_{\rm com,d}\|_Q^2$ in the form of $\|Gu - g_0\|_Q^2$ with:

$$G = \begin{bmatrix} 0_{6\times(6+n)} & J_{\rm st,com}^T \end{bmatrix}, \quad g_0 = W_{\rm com,d} \tag{8}$$

2) Physical consistency: To enforce physical consistency between $F_{\rm grf}$ and $\ddot{q}$, we impose the dynamics of the unactuated part of the robot (the trunk dynamics in (2)) as an equality constraint:

$$A_p = \begin{bmatrix} M_{\rm com} & 0_{6\times n} & -J_{\rm st,com}^T \end{bmatrix}, \quad b_p = -h_{com} \tag{9}$$

3) Stance condition: We can encode the stance feet constraints by re-writing them at the acceleration level in order to be compatible with the decision variables. Since $v_{\rm st} = J_{\rm st}\dot{q}$, differentiating once in time, yields to $\dot{v}_{\rm st} = J_{\rm st}\ddot{q} + \dot{J}_{\rm st}\dot{q}$. Recalling Assumption 1 yields $J_{\rm st}\ddot{q} + \dot{J}_{\rm st}\dot{q} = 0$ which is encoded as:

$$A_{\rm st} = \begin{bmatrix} J_{\rm st} & 0_{n_{\rm st}\times n_{\rm st}} \end{bmatrix}, \quad b_{\rm st} = -\dot{J}_{\rm st}\dot{q} \tag{10}$$

such that $\dot{J}_{\rm st}$ is the time derivative of $J_{\rm st}$. For numerical precision, we compute the product $\dot{J}_{\rm st}\dot{q}$ using spatial algebra.

[5]Note that $h_{com} = -mg + v_{com} \times^* M_{com} v_{com}$ according to the spatial algebra notation, where m is the total robot mass.

[6]$\nabla V_{[.]}$ denotes the gradient of a potential function $V_{[.]}$. For more information regarding the Cartesian stiffness and gravitational potentials, see [8].

4) Swing task: Similar to Section II-C3, we can encode the swing task directly as an *equality constraint*, i.e. by enforcing the swing feet to follow a desired swing acceleration $\dot{v}_{\mathrm{sw}}(q) = \dot{v}_{\mathrm{sw,d}} \in \mathbb{R}^{n_{\mathrm{sw}}}$ yielding:

$$J_{\mathrm{sw}}\ddot{q} + \dot{J}_{\mathrm{sw}}\dot{q} = \dot{v}_{\mathrm{sw,d}} \tag{11}$$

that in matrix form becomes[7]:

$$A_{\mathrm{sw}} = \begin{bmatrix} J_{\mathrm{sw}} & 0_{n_{\mathrm{sw}} \times n_{\mathrm{st}}} \end{bmatrix}, \quad b_{\mathrm{sw}} = \dot{v}_{\mathrm{sw,d}} - \dot{J}_{\mathrm{sw}}\dot{q} \tag{12}$$

where we computed $\dot{v}_{\mathrm{sw,d}}$ as in (6). Note that this implementation is analogous to the trunk task in Section II-B. The difference is that this implementation is at the acceleration level while the other is at the force level. However, without any loss of generality, the formulation (5) could also be used. In Section IV-B we incorporate slacks in the optimization to allow temporary violation of the swing tasks (e.g. useful when the kinematic limits are reached).

5) Friction cone constraints: To avoid slippage and obtain a smooth loading/unloading of the legs, we incorporate friction/unilaterality constraints. For that, we ensure that the contact forces lie inside the friction cones and their normal components stay within some user-defined values (i.e. maximum and minimum force magnitudes). We approximate the friction cones with square pyramids to express them with linear constraints. The fact that the ground contacts are unilateral, can be naturally encoded by setting an "almost-zero" lower bound on the normal component, while the upper bound allows us to regulate the amount of "loading" for each leg. We define the friction inequality constraints as:

$$\underline{d}_{\mathrm{fr}} < C_{\mathrm{fr}} u < \bar{d}_{\mathrm{fr}}, \quad C_{\mathrm{fr}} = \begin{bmatrix} 0_{p \times (6+n)} & F_{\mathrm{fr}} \end{bmatrix} \tag{13}$$

with:

$$F_{\mathrm{fr}} = \begin{bmatrix} F_0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & F_c \end{bmatrix}, \quad \underline{d}_{\mathrm{fr}} = \begin{bmatrix} \underline{f}_0 \\ \vdots \\ \underline{f}_c \end{bmatrix}, \quad \bar{d}_{\mathrm{fr}} = \begin{bmatrix} \bar{f}_0 \\ \vdots \\ \bar{f}_c \end{bmatrix} \tag{14}$$

where $F_{\mathrm{fr}} \in \mathbb{R}^{p \times n_{\mathrm{st}}}$ is a block diagonal matrix that encodes the friction cone boundaries and select the normals, for each stance leg and $\underline{d}_{\mathrm{fr}}, \bar{d}_{\mathrm{fr}} \in \mathbb{R}^p$ are the lower/upper bounds respectively. For the detailed implementation of the friction constraints refer to [13].

6) Torque limits: We notice that the torques be obtained from the decision variables since they can be expressed as a bi-linear function of $\ddot{q}_j$ and F_{grf}. Therefore, the constraint on the joint torques (i.e., the actuation limits $\tau_{\min} < \tau_j < \tau_{\max}$) can be encoded by exploiting the actuated part of the dynamics (1):

$$\begin{aligned} &\underline{d}_\tau < C_\tau u < \bar{d}_\tau, \quad C_\tau = \begin{bmatrix} 0_{n\times 6} & \bar{M}_j & -J_{\mathrm{st,j}}^T \end{bmatrix} \\ &\underline{d}_\tau = -\bar{h}_j + \tau_{\min}(q_j), \quad \bar{d}_\tau = -\bar{h}_j + \tau_{\max}(q_j) \end{aligned} \tag{15}$$

where $\tau_{\min}(q_j), \tau_{\max}(q_j) \in \mathbb{R}^n$ are the lower/upper bounds on the torques. In the case of our quadruped robot, these bounds must be recomputed at each control loop because they depend on the joint positions. This is due to the presence of linkages on the sagittal joints (HFE and KFE), that set a joint-dependent profile on the maximum torque (non-linear in the joint range).

7) Joint kinematic limits: We enforce joint kinematic constraints as function of the joint accelerations (i.e. $\ddot{q}_{j_{\min}} < \ddot{q}_j < \ddot{q}_{j_{\max}}$). We select them via the matrix C_j:

$$\underline{d}_j < C_j u < \bar{d}_j, \quad C_j = \begin{bmatrix} 0_{n\times 6} & I_{n\times n} & 0_{n\times n_{\mathrm{st}}} \end{bmatrix} \tag{16}$$

$$\underline{d}_j = \ddot{q}_{j_{\min}}(q_j), \qquad \bar{d}_j = \ddot{q}_{j_{\max}}(q_j) \tag{17}$$

such that $\ddot{q}_{j_{\min}}(q_j)$ and $\ddot{q}_{j_{\max}}(q_j)$ are the upper/lower bounds on accelerations. These bounds should be recomputed at each control loop. They are set in order to make the joint to reach the end-stop at a zero velocity in a time interval $\Delta t = 10dt$, where dt is the loop duration. For instance, if the joint is at a distance $q_{j_{\max}} - q_j$ from the end-stop with a velocity $\dot{q}_j$, the deceleration to cover this distance in a time interval Δt, and approach the end-stop with zero velocity, will be:

$$\ddot{q}_{j_{\min,\max}} = -\frac{2}{\Delta t^2}(q_{j_{\min,\max}} - q_j - \Delta t\, \dot{q}_j) \tag{18}$$

D. Torque computation

We map the optimal solution $u^* = \begin{bmatrix} \ddot{q}^* & F^*_{\mathrm{grf}} \end{bmatrix}$ obtained by solving (7), into desired joint torques $\tau_d^* \in \mathbb{R}^n$ using the actuated part of the dynamics equation of the robot as:

$$\tau_d^* = \bar{M}_j \ddot{q}_j^* + \bar{h}_j - J_{\mathrm{st,j}}^T F^*_{\mathrm{grf}} \tag{19}$$

III. Passivity Analysis

The overall system consists of the WBC, the robot and the environment. This system is said to be *passive* if all these components, and their interconnections are passive [16]. If the robot and the environment are passive, and the controller is proven to be passive, then the overall system is passive [21]. A system (with input u and output y) is said to be passive if there exists a storage function S that is bounded from below and its derivative $\dot{S}$ is less than or equal to its supply rate ($s = y^T u$). In this context, we define the total energy stored in the controller to be the candidate storage function for the controller $S = V$. The rest of this section is devoted to analyze the passivity of the overall system.

Assumption 2: A feasible solution exists for the QP in (7) in which the motion tasks are achieved. Moreover, we do not consider the feed-forward terms in (4a) and (5) leaving this to future developments.

We start by defining the velocity error at the joints and at the stance feet to be $\Delta\dot{q}_j = \dot{q}_{j,d} - \dot{q}_j$, and $\Delta v_{\mathrm{st}} = v_{\mathrm{st,d}} - v_{\mathrm{st}}$, respectively. We also define the desired feet forces $F_{\mathrm{d}} = [F_{\mathrm{st,d}}^T \; F_{\mathrm{sw,d}}^T]^T$ such that, by following the sign convention in Fig. 1, the mapping between F_{d} and $W_{\mathrm{com,d}}$ is expressed as[8]:

$$W_{\mathrm{com,d}} = -J_{\mathrm{com}}^T F_{\mathrm{d}}, \tag{20}$$

while mapping between F_d and τ_j is expressed as[9]

$$\tau \;\; = \;\; J_j^T F_d. \tag{21}$$

[7]Alternatively, it is possible to write the swing task at the joint space rather than in the operational space by changing the matrix $A_{\mathrm{sw}}, b_{\mathrm{sw}}$.

[8]Since we are analyzing the passivity of the controller, we are interested in the forces exerted by the robot on the environment rather than the forces exerted by the environment on the robot. Hence the mapping in (20) is negative.

[9]Assuming a perfect low level torque control tracking (i.e., $\tau_{\mathrm{d}} = \tau$).

By defining $\nabla V_{\text{com}} = \nabla V_{\text{com,K}} + \nabla V_{\text{com,}\bar{\text{g}}}$ and recalling (20), we rewrite (4a) under Assumption 2 as:

$$\begin{aligned} \nabla V_{\text{com}} &= W_{\text{com,d}} - D_{\text{com}} \Delta v_{\text{com}} && \text{(22a)} \\ &= -J_{\text{com}}^T F_{\text{d}} - D_{\text{com}} \Delta v_{\text{com}}. && \text{(22b)} \end{aligned}$$

A. Analysis

The overall energy in the whole-body controller is the one stored in the virtual impedance at the CoM and the potential energy due to gravity compensation (V_{com}), and the energy stored in the virtual impedances at the swing feet (V_{sw}) [10]:

$$V = V_{\text{com}} + V_{\text{sw}}. \tag{23}$$

The time derivatives are[11]:

$$\dot{V} = \dot{V}_{\text{com}} + \dot{V}_{\text{sw}} = \Delta v_{\text{com}}^T \nabla V_{\text{com}} + \Delta v_{\text{sw}}^T \nabla V_{\text{sw}}. \tag{24}$$

Recalling (5) and (22b), (24) yields:

$$\begin{aligned} \dot{V} &= \Delta v_{\text{com}}^T (-J_{\text{com}}^T F_{\text{d}} - D_{\text{com}} \Delta v_{\text{com}}) + \\ &\quad \Delta v_{\text{sw}}^T (F_{\text{sw,d}} - D_{\text{sw}} \Delta v_{\text{sw}})). \end{aligned} \tag{25}$$

We regroup $\dot{V}$ in terms of the non-damping terms $\dot{V}_1$ and damping terms $\dot{V}_2$ yielding:

$$\begin{aligned} \dot{V}_1 &= -\Delta v_{\text{com}}^T J_{\text{com}}^T F_{\text{d}} + \Delta v_{\text{sw}}^T F_{\text{sw,d}} && \text{(26a)} \\ \dot{V}_2 &= -\Delta v_{\text{com}}^T D_{\text{com}} \Delta v_{\text{com}} - \Delta v_{\text{sw}}^T D_{\text{sw}} \Delta v_{\text{sw}}. && \text{(26b)} \end{aligned}$$

We rewrite (3b) in terms of Δv_{com}, Δv and $\Delta \dot{q}_{\text{j}}$ as:

$$\begin{aligned} \Delta v &= J_{\text{com}} \Delta v_{\text{com}} + J_{\text{j}} \Delta \dot{q}_{\text{j}} && \text{(27a)} \\ \Delta v_{\text{com}}^T J_{\text{com}}^T &= -\Delta \dot{q}_{\text{j}}^T J_{\text{j}}^T + \Delta v^T. && \text{(27b)} \end{aligned}$$

Plugging, (27b) in (26a) yields[12]:

$$\begin{aligned} \dot{V}_1 &= \Delta \dot{q}_{\text{j}}^T J_{\text{j}}^T F_{\text{d}} - \Delta v^T F_{\text{d}} + \Delta v_{\text{sw}}^T F_{\text{sw,d}} && \text{(28a)} \\ &= \Delta \dot{q}_{\text{j}}^T J_{\text{j}}^T F_{\text{d}} - \Delta v_{\text{st}}^T F_{\text{st,d}}. && \text{(28b)} \end{aligned}$$

Plugging (21) and into (28b) yields:

$$\dot{V}_1 = \Delta \dot{q}_{\text{j}}^T \tau - \Delta v_{\text{st}}^T F_{\text{st,d}}. \tag{29}$$

Under Assumption 1, (29) yields:

$$\dot{V}_1 = \Delta \dot{q}_{\text{j}}^T \tau. \tag{30}$$

Thus, $\dot{V}$ could be rewritten as:

$$\dot{V} = \Delta \dot{q}_{\text{j}}^T \tau - \Delta v_{\text{com}}^T D_{\text{com}} \Delta v_{\text{com}} - \Delta v_{\text{sw}}^T D_{\text{sw}} \Delta v_{\text{sw}}. \tag{31}$$

[10]In this analysis we use the formulation (5).

[11]The time derivative of an arbitrary storage function $\dot{V}(\Delta x(t))$ that is a function of $\Delta x(t)$ could be written as $\dot{V} = \frac{d}{dt} \Delta x^T(t) \cdot \frac{\partial}{\partial \Delta x(t)} V$ that is written for short as $\dot{V} = \Delta v^T \nabla V$.

[12]From the definition of v and F_d, we get $v^T F_d = v_{\text{st}}^T F_{\text{st,d}} + v_{\text{sw}}^T F_{\text{sw,d}}$.

B. Proof

Under Assumption 2, the designed WBC is an impedance control with gravity compensation that, similar to a PD+ [18], defines a map of $(\dot{q}_{\text{j}} - \dot{q}_{\text{j,d}}) \mapsto -\tau$[13]. This controller is passive if V is bounded from below and $\dot{V} \leq \Delta \dot{q}_{\text{j}}^T \tau$. Since V consists of positive definite potentials that resemble Cartesian stiffnesses at the CoM and the swing leg(s), and under the assumption the gravitational potential is bounded from below (see [22]), V is also bounded from below. Additionally, recalling (31) proves that the controller is indeed passive; thus, the overall system is passive.

IV. Implementation Details

This section describes some pragmatic details that we found crucial in the implementation on the real platform.

A. Stance task

Uncertainties in estimating the terrain's normal direction and friction coefficient could result in slippage. This can lead to considerable motion of the stance feet with possible loss of stability. To avoid this, a joint impedance feedback loop could be run in parallel to the WBC, at the price of losing the capability of optimizing the GRFs. A cleaner solution is to incorporate, in the optimization, Cartesian impedances specifically designed to keep the relative distance among the stance feet constant (we denote it *stance task*).

The stance task has an influence *only* when there is an anomalous motion in the stance feet, retaining the possibility to freely optimize for GRFs in normal situations. This can be achieved by re-formulating the stance condition in (10) as a desired stance feet acceleration $\dot{v}_{\text{st,d}}$ as:

$$\dot{v}_{\text{st,d}} = K_{\text{st}}(\hat{x}_{\text{st}} - x_{\text{st}}) - D_{\text{st}} v_{\text{st}}. \tag{32}$$

This term is added to b_{st} (in (10)) as $b_{st} = -\dot{J}_{st}\dot{q} + \ddot{x}_{f_{st}}^d$ where $\hat{x}_{\text{st}}$ is a sample of the foot position at the touchdown (expressed in the world frame).

B. Constraint Softening

Adding slack variables to an optimization problem is commonly done to avoid infeasible solutions, by allowing a certain degree of constraint violation. Infeasibility can occur when hard constraints are conflicting with each other, which can be the case in our QP. Hence, some of the equalities/inequalities in (7) should be relaxed if they are in conflict.

We decided not to introduce slacks in the torque constraints (Section II-C6) or the dynamics (Section II-C2) keeping them as hard constraints, since torque constraints and physical consistency should never be violated. On the other hand it is important to allow a certain level of relaxation for the swing tasks in Section II-C4 that could be violated if the joint kinematic limits are reached[14].

To relax the constraints of the swing task, first, we augment the decision variables $\tilde{u} = [u^T \ \epsilon^T]^T \in \mathbb{R}^{6+n+k+n_{sw}}$ with the vector of slack variables $\epsilon \in \mathbb{R}^{n_{sw}}$ where we introduce a slack

[13]Note that $\dot{q}_{\text{j}} - \dot{q}_{\text{j,d}} = -\Delta \dot{q}_{\text{j}}$. Thus, the controller with the map $(\dot{q}_{\text{j}} - \dot{q}_{\text{j,d}}) \mapsto -\tau$ has a supply rate of $\Delta \dot{q}_{\text{j}}^T \tau$.

[14]Using slacks in friction constraints did not result in significant improvements.

variable for each direction of the swing tasks. Then, we replace the equality constraint $A_{sw}u + b_{sw} = 0$ of the swing tasks, by two inequality constraints:

$$-\epsilon \leq A_{sw}\tilde{u} + b_{sw} \leq \epsilon$$
$$\epsilon \geq 0. \tag{33}$$

The first inequality in (33) restricts the solution to a bounded region around the original constraint while the second one ensures that the slack variables remain non-negative. To make sure that there is a constraint violation only when the constraints are conflicting, we minimize the norm of the slack vector adding a regularization term $\alpha\|\epsilon\|^2$ to (7a) with a high weight α.

To reduce the computational complexity, we could have introduced a single slack for each swing task (rather than one for each direction). However, this could create coupling errors in the tracking. For instance, since the Hip Flexion-Extension (HFE) joint (see Fig. 1) mainly acts in the XZ plane, if it reaches its joint limit, only that plane should be affected leaving the Y direction unaffected. A single slack couples the three directions causing tracking errors also in the Y direction. Conversely, using multiple slacks, only the directions the HFE acts upon, will be affected.

V. Results

In this section we validate the capabilities of the controller under various terrain conditions and locomotion gaits. The WBC and torque control loops run in real-time threads at 250 Hz and 1 kHz, respectively. We set the gains for the swing tasks to $K_{sw} = \text{diag}(2000, 2000, 2000)$ and $D_{sw} = \text{diag}(20, 20, 20)$, while for the trunk task we set $K_x = \text{diag}(2000, 2000, 2000)$ $D_x = \text{diag}(400, 400, 400)$ and $K_\theta = \text{diag}(1000, 1000, 1000)$ $D_\theta = \text{diag}(200, 200, 200)$. These values proved to be working in both simulation and real experiments. The results are collected in the accompanying video[15]. Additionally, in experimental trials, we also included a low gain joint-space attractor (PD controller) for the swing task, since imprecise torque tracking of the knee joints (due to the low inertia) produce control instabilities in an operational space implementation (e.g. the one in Section II-C4).

A. Constraint Softening through Slack Variables

In Fig. 2 we artificially incremented the lower limit of the LH-HFE joint. When the limit is hit, the bound on the joint acceleration (18) produces a desired torque command that stops its motion. This "naturally" clamps the actual joint position to the limit (bottom-left plot) and influences the foot tracking mainly along the X and Z directions.

Computational time: the solution of the QP takes between 90-110 μs on a Intel i5 machine without the slacks variables. After augmenting the problem with the slack variables and its constraints, it increases 30 % on average (120-150 μs). However it still remains suitable for real-time implementation (250 Hz).

[15]https://youtu.be/Lg3V_juoE1w

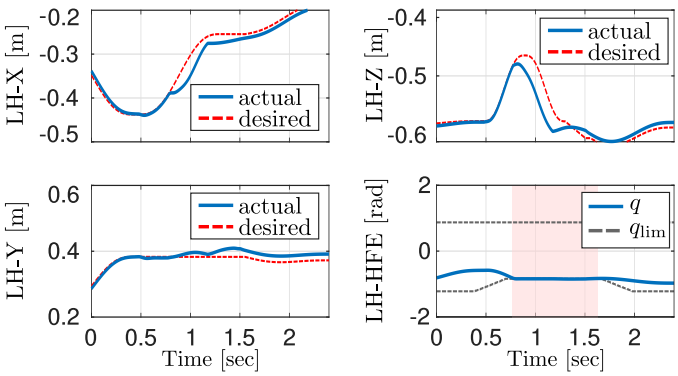

Fig. 2: Simulation. Effect of (kinematic limits) slacks variables on foot tracking. The upper-left/right and bottom-left plots show the tracking of the desired foot position (LH leg) in X, Y and Z, respectively. Bottom-right plot depicts the joint limits (black line) and the actual position (blue line) of the HFE joint. The red shaded area underlines that when the slack increases, the HFE joint is properly clamped.

B. Friction Constraints and Bounded Slippage

We evaluated in simulation the controller performance against inaccurate friction coefficient estimates μ, which define incorrectly the friction cone constraints in the WBC. In the accompanying video, we show an example where the robot crawls at 0.11 m/s on a slippery floor ($\mu = 0.4$) while we set the friction coefficient to $\mu = 1.0$ in the WBC, to emulate an estimation error. Simulation results support the fact that foot slippage remains bounded by the action of the *stance task* (Section IV-A). If we gradually correct μ the slippage events completely disappear; allowing an increase of forward velocity up to 0.16 m/s.

C. Torque Limits and Load Redistribution

We analyzed in simulation the effect of adding an artificial torque limit, in our WBC. This helps us to derive controllers that are robust against joint damages. Figure 3 shows a reduction of the torque limits down to 26 Nm in the Left-Front (LF)-KFE joint and the load redistribution among the other joints (HAA and HFE) of the LF leg. Indeed, while the KFE joint torque is clamped, the HFE is loaded more (lower plot). This load redistribution did not affect the trunk motion and it demonstrates how the controller exploits the torque redundancy by finding a new load distribution. We carried out also intensive experimental validation in various challenging terrains. Slopes increase the probability of reaching torque limits because of the more demanding kinematic configurations. Indeed, in Fig. 4, the robot reached three times the torque limits (red shaded areas). Crossing this terrain would not be possible without enforcing the torque limits as hard constraints.

D. Different Torque Regularization Schemes

By setting different regularizations in (7a) [13], we can either choose to maximize the robustness to uncertainties in the friction parameters (e.g. GRFs closer to the friction cone normals) or to minimize the joint torques[16]. In the latter

[16]Setting the weighting matrix $R_{kk} = J_{st}S^T R_\tau S J_{st}^T$ where: R_{kk} is the sub-block of R that regularizes for GRFs variables in (7) and S selects the actuation joints.

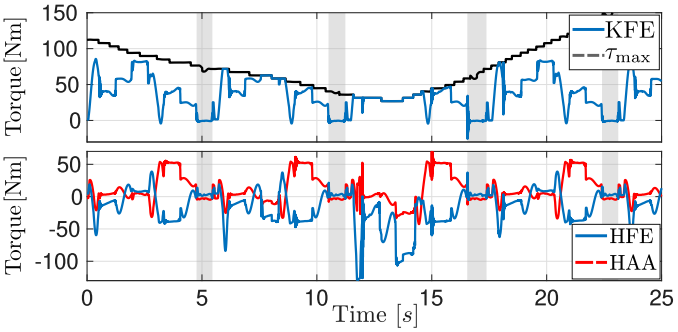

Fig. 3: Simulation. Effect of introducing an artificial torque limit on the KFE joint of the LF leg during a typical crawl. The shaded area represents the swing phase of the leg while the unshaded part is the stance phase. We reduced the torque limit down to 26 Nm (black line, upper plot), and as a consequence, the HFE torque is increased by the controller (bottom plot).

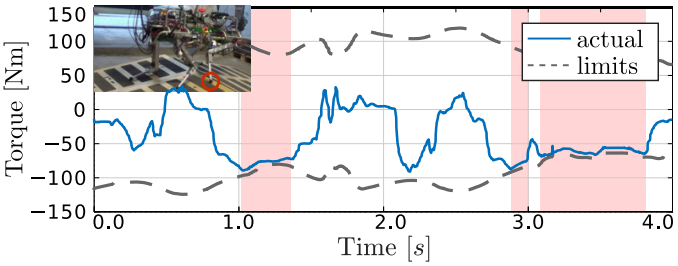

Fig. 4: Real experiments. Reaching the torque limits on the RF-HFE joint while climbing up and down two ramps. The HyQ robot reached three times its torque limits (red shaded areas). The real torque (blue line) is tracking the desired one (not shown) computed from our whole-body controller while satisfying the joint limits (black line). The torque limits are time-varying due to the joint mechanism.

case, for instance, we could encourage the controller to use a particular joint by increasing its corresponding weight. If we gradually increase the weight of the Knee Flexion-Extension (KFE) joints (see accompanying video), the effect of torque regularization becomes visible because the GRFs are no longer vertical. Indeed the GRFs start to point toward the knee axis in order to reduce its torque command.

E. Comparison with Previous Controller (Quasi-Static)

We compare our whole-body controller (dynamic) against a centroidal-based controller (quasi-static) [13]. As metric we use the l^2-norm for the linear e_x and angular e_θ tracking errors of the trunk task. If we increase linearly the forward speed from 0.04 m/s to 0.15 m/s, the tracking error is reduced approximately by 50 % in comparison to the quasi-static controller (Fig. 5). This is due to the fact that our WBC computes both joint accelerations and contact forces, which allows a proper mapping of torque commands (inverse dynamics). Indeed this results in better accuracy in the execution of more dynamic motions.

F. Disturbance Rejection against Unstable Foothold

We encoded compliance tracking of the CoM task through a *virtual* impedance. Friction cone constraints help to instantaneously keep the robot's balance whenever a tracking error happens due to, for instance, an unstable foothold. Furthermore

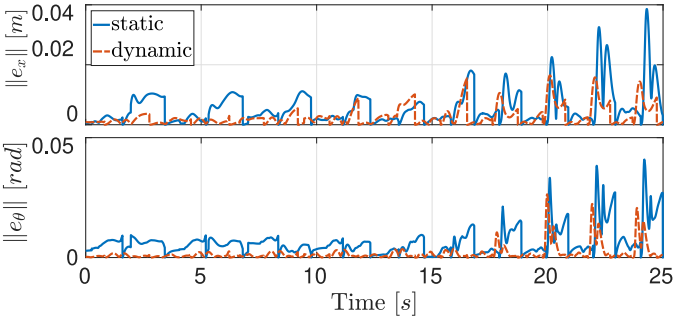

Fig. 5: Simulation. Comparison of tracking errors for the trunk task of a quasi-static controller [13] against our whole-body controller (dynamic). l^2-norms of linear and angular errors are shown in the top and bottom figures. Note that the errors are reset to zero at each step due to re-planning [20].

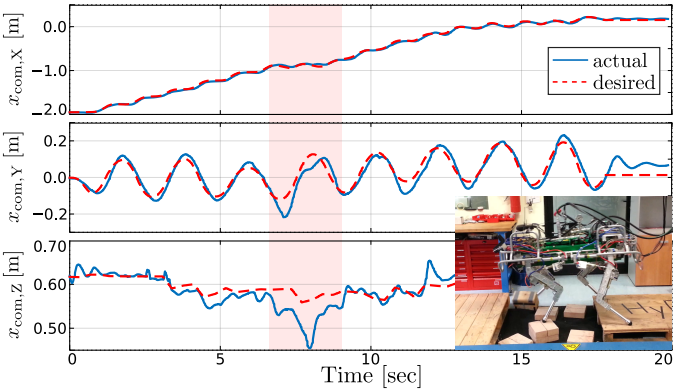

Fig. 6: Real experiments. Disturbance rejection against unstable foothold that occurs when a stepping-stone rolled under the RF leg at $t = 6.5$ s. The controller lost tracking of the CoM height, however, the friction cone constraints keep instantaneously the robot's balance. Indeed a good tracking of the horizontal motion of the CoM is obtained. Note that the red shaded area depicts the moments of the disturbance rejection.

joint constraints (positions and torques) guarantee feasibility of the computed torque commands. Figure 6 shows how the controller compliantly tracks the desired CoM trajectory during an unstable footstep (a rolling stepping-stone) that occurs at $t = 6.5$ s (experiments results from [23]). This creates tracking errors on the CoM height, yet, good tracking performance is kept for the horizontal CoM motion, due to the friction cone constraints that maintained the robot's balance along the entire locomotion.

G. Locomotion over Slopes

These experiments have been performed with *online* terrain mapping [20]. Both the terrain mapping and the whole-body controller make use of a drift-free state estimation algorithm to obtain the body state. The friction cone constraints of the controller are described given the real terrain normals provided by an onboard mapping algorithm[17]. The friction coefficient has been conservatively set to 0.7 for all the experiments. Figure 7 shows different snapshots of various challenging terrain used to evaluate our controller. The centroidal trajectory, gait and footholds are computed simultaneously as described in [4].

[17]The controller action can be greatly improved by setting the real terrain normal (under each foot) rather than using a default value for all the feet.

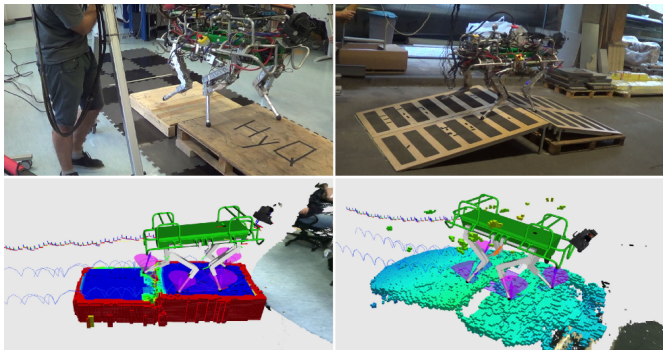

Fig. 7: Snapshots of experimental trials to evaluate our whole-body control and the *online* terrain mapping. Left column: crossing a 22 cm gap with a 7 cm step. Right column: traversing two ramps with a 15 cm gap between them.

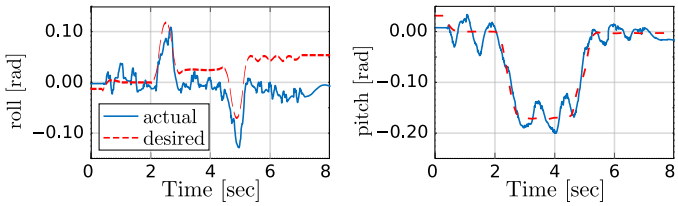

Fig. 8: Real experiments. Roll and pitch tracking performance while climbing up a ramp with a trotting gait. Although there is under-actuation the controller can still track roll and pitch motions.

H. Tracking Performance with Different Gaits

The quadrupedal trotting gait is difficult to control because the robot uses only two legs at the time to achieve the tracking of the desired CoM motion and of the trunk orientation. Figure 8 depicts the roll and pitch tracking for climbing up a ramp during a trotting gait. Although a trot is an under-actuated gait, our controller can still track the desired orientation. Moreover, in these cases, the orientation error is always below 0.2 rad.

VI. Conclusion

This paper presented an experimental validation of our passive WBC. Compared to our previous work [13], the presented WBC enables higher dynamic motions thanks to the use of the full dynamics of the robot. Although similar controllers have been proposed in the literature (e.g. [6, 8, 5]), we validated our locomotion controller on HyQ over a wide range of challenging terrain (slopes, gaps, stairs, etc.), using different gaits (crawl and trot). Additionally, we have analyzed the controller capabilities against 1) inaccurate friction coefficient estimation, 2) unstable footholds, 3) changes in the regularization scheme and 4) the load redistribution under restrictive torque limits. Extensive experimental results validated the controller performance together with the *online* terrain mapping and the state estimation. Moreover, we demonstrated experimentally the superiority of our WBC compared to a quasi-static control scheme [13].

References

[1] T. Erez, K. Lowrey, Y. Tassa, V. Kumar, S. Kolev, and E. Todorov, "An integrated system for real-time Model Predictive Control of humanoid robots," in *IEEE/RAS International Conference on Humanoid Robots*, 2013.

[2] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 2589–2594.

[3] F. Farshidian, E. Jelavic, A. Satapathy, M. Giftthaler, and J. Buchli, "Real-time motion planning of legged robots: A model predictive control approach," in *Humanoid Robotics (Humanoids), 2017 IEEE-RAS 17th International Conference on*. IEEE, 2017, pp. 577–584.

[4] B. Aceituno-Cabezas, C. Mastalli, H. Dai, M. Focchi, A. Radulescu, D. G. Caldwell, J. Cappelletto, J. C. Grieco, G. Fernández-López, and C. Semini, "Simultaneous contact, gait, and motion planning for robust multilegged locomotion via mixed-integer convex optimization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2531–2538, 2018.

[5] C. D. Bellicoso, F. Jenelten, C. Gehring, and M. Hutter, "Dynamic locomotion through online nonlinear motion optimization for quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2261–2268, 2018.

[6] 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, Mar 2016.

[7] T. Koolen, S. Bertrand, G. Thomas, T. de Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot atlas," *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1650007, 2016.

[8] B. Henze, M. A. Roa, and C. Ott, "Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios," *The International Journal of Robotics Research*, vol. 35, no. 12, pp. 1522–1543, 2016.

[9] F. Farshidian, E. Jelavi, A. W. Winkler, and J. Buchli, "Robust whole-body motion control of legged robots," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.

[10] D. Kim, J. Lee, J. Ahn, O. Campbell, H. Hwang, and L. Sentis, "Computationally-robust and efficient prioritized whole-body controller with contact constraints," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 1–8.

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

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

[13] M. Focchi, A. Del Prete, I. Havoutis, R. Featherstone, D. G. Caldwell, and C. Semini, "High-slope terrain locomotion for torque-controlled quadruped robots," *Autonomous Robots*, vol. 41, no. 1, pp. 259–272, 2017.

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

[15] K. Bouyarmane, K. Chappellet, J. Vaillant, and A. Kheddar, "Quadratic programming for multirobot and task-space force control," *IEEE Transactions on Robotics*, 2018.

[16] S. Stramigioli, "Energy-aware robotics," in *Mathematical Control Theory I*. Springer, 2015, pp. 37–50.

[17] 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.

[18] R. Ortega, J. A. L. Perez, P. J. Nicklasson, and H. J. Sira-Ramirez, *Passivity-based control of Euler-Lagrange systems: mechanical, electrical and electromechanical applications*. Springer Science & Business Media, 2013.

[19] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, "Dynamic locomotion in the mit cheetah 3 through convex model-predictive control," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[20] M. Focchi, R. Orsolino, M. Camurri, V. Barasuol, C. Mastalli, D. G. Caldwell, and C. Semini, "Heuristic planning for rough terrain locomotion in presence of external disturbances and variable perception quality," *Springer Track in Advanced Robotics series*, no. 9, 2019.

[21] A. J. van der Schaft, *L2-gain and passivity techniques in nonlinear control*. Springer, vol. 2.

[22] C. Ott, A. Albu-Schaffer, A. Kugi, and G. Hirzinger, "On the passivity-based impedance control of flexible joint robots," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 416–429, 2008.

[23] C. Mastalli, M. Focchi, I. Havoutis, A. Radulescu, S. Calinon, J. Buchli, D. G. Caldwell, and C. Semini, "Trajectory and foothold optimization using low-dimensional models for rough terrain locomotion," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2017.