Open access • Proceedings Article • DOI:10.1109/HUMANOIDS.2016.7803330

Perception-less terrain adaptation through whole body control and hierarchical optimization — Source link

C. Dario Bellicoso, Christian Gehring, Jemin Hwangbo, Peter Fankhauser ...+1 more authors

Institutions: ETH Zurich

Published on: 01 Nov 2016 - IEEE-RAS International Conference on Humanoid Robots

Topics: Humanoid robot and Motion planning

Perception-less Terrain Adaptation through Whole Body Control and Hierarchical Optimization

Perception-less Terrain Adaptation through Whole Body Control and Hierarchical Optimization

C. Dario Bellicoso, Christian Gehring, Jemin Hwangbo, Péter Fankhauser, Marco Hutter

***Abstract*— This paper presents a control approach based on a whole body control framework combined with hierarchical optimization. Locomotion is formulated as multiple tasks (e.g. maintaining balance or tracking a desired motion of one of the limbs) which are solved in a prioritized way using QP solvers. It is shown how complex locomotion behaviors can purely emerge from robot-specific inequality tasks (i.e. torque or reaching limits) together with the optimization of balance and system manipulability. Without any specific motion planning, this prioritized task optimization leads to a natural adaption of the robot to the terrain while walking and hence enables blind locomotion over rough grounds. The presented framework is implemented and successfully tested on ANYmal, a torque controllable quadrupedal robot. It enables the machine to walk while accounting for slippage and torque limitation constraints, and even step down from an unperceived 14 cm obstacle. Thereby, ANYmal exploits the maximum reach of the limbs and automatically adapts the body posture and height.**

I. Introduction

Motion planning and control of legged robots is coupled with several challenges. First, legged robots are typically high-dimensional systems with partially redundant degrees of freedom. Second, interaction with the environment is achieved through multiple contact points which impose changing contact constraints and interaction forces. Finally, the environment itself is generally unknown and terrain perception can be limited or absent altogether. To lower the design complexity of motion and control algorithms, locomotion is often described as a set of simpler tasks such as body posture control to keep balance, limb motion to move, or contact force constraints to prevent slippage. A theoretical framework which accommodates this kind of decomposition is the Whole Body Control (WBC) framework coupled with Hierarchical Optimization (HO). Generally speaking, any implementation that generates control signals for all the actuated joints of a robotic system may be called a WBC; however, literature has focused on WBC which implement, at least, floating base inverse dynamics [16]. Tasks can be either solved at the same priority through weighted average or can be prioritized one w.r.t. the other [8].

The first implementations of HO were dealing exclusively with equality constraint tasks (e.g. motion tracking objectives) [20]. Inequality constraints, such as torque or joint limits and friction cone bounds, were implemented by means of potential fields [18], [17]. The first proper attempt at explicitly taking into account inequality constraints has been shown only in recent years [13]. The set of tasks were solved as a cascade of Quadratic Programming (QP) problems constrained by both equalities and inequalities. The solution of each QP, together with its constraints, would shape the space in which to search for the next solution. However, computational requirements were very high, and could become unfeasible to implement on a real time system. Successive work has shown how this issue can be tackled (e.g. [7]), by combining explicit inequality constraint handling with the search of solutions in the null space of higher priority tasks. This method yields QP problems which get smaller as equality constraints are added to the cascade of QPs.

Fig. 1. ANYmal is a fully torque-controllable quadrupedal robot, actuated by custom developed series elastic actuators which provide very precise torque control.

In the present paper, we show how such a control approach can be exploited to create adaptive behaviors for locomotion over challenging terrain. Instead of integrating perception and extending motion planning as a function of the environment, we present clever control tasks of different priorities to exploit the large degree of mobility of legged systems and automatically adapt to (uneven and unknown) ground.

Hence, our contribution is twofold: first, we show a concrete implementation of WBC using HO coupled with the full system dynamics and motion planning for a walking behavior on a real system; second, we also show how inequality constraint tasks in the operational space can be used to achieve reactive and perception-less adaptation to the terrain. The proposed control framework was successfully tested on the torque controllable quadrupedal robot ANYmal (Fig. 1).

This work was supported in part by the Swiss National Science Foundation (SNF) through the National Centre of Competence in Research Robotics.

All authors are with the Robotic Systems Lab, ETH Zurich, Switzerland, bellicoso@mavt.ethz.ch

II. Model formulation

In general, a walking robot can be modeled as a system with a free-floating base B to which legs and arms are attached. The motion of the entire system can be described w.r.t. a fixed inertial frame I. We write the position vector of the base frame w.r.t. the inertial frame as ${}_I\mathbf{r}_{IB} \in \mathbb{R}^3$ and use unit quaternions to parametrize the orientation of the main body. The limb joint generalized positions are stacked in the vector $\mathbf{q}_j \in \mathbb{R}^{n_j}$. We write the generalized positions vector $\mathbf{q}$ and the generalized velocities vector $\mathbf{u}$ as

$$\mathbf{q} = \begin{bmatrix} {}_I\mathbf{r}_{IB} \\ \mathbf{q}_{IB} \\ \mathbf{q}_j \end{bmatrix} \in SE(3) \times \mathbb{R}^{n_j}, \quad \mathbf{u} = \begin{bmatrix} {}_I\mathbf{v}_{IB} \\ {}_B\boldsymbol{\omega}_{IB} \\ \dot{\mathbf{q}}_j \end{bmatrix} \in \mathbb{R}^{n_u}, \tag{1}$$

where $n_u = 6 + n_j$, ${}_B\boldsymbol{\omega}_{IB}$ is the angular velocity of the base w.r.t. the inertial frame expressed in B frame, and $\mathbf{q}_{IB} \in SO(3)$ is the unit quaternion that projects the components of a vector expressed in B frame to those of the same vector expressed in I frame. The equations of motion of legged systems can be written as [19]

$$\mathbf{M}(\mathbf{q})\dot{\mathbf{u}} + \mathbf{h}(\mathbf{q}, \mathbf{u}) = \mathbf{S}^T\boldsymbol{\tau} + \mathbf{J}_s^T\boldsymbol{\lambda}, \tag{2}$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n_u \times n_u}$ is the mass matrix and $\mathbf{h}(\mathbf{q}, \mathbf{u}) \in \mathbb{R}^{n_u}$ is the vector of Coriolis, centrifugal and gravity terms. The selection matrix $\mathbf{S} = \begin{bmatrix} \mathbf{0}_{n_\tau \times (n_u - n_\tau)} & \mathbb{I}_{n_\tau \times n_\tau} \end{bmatrix}$ selects which DoFs are actuated. If all limb joints are actuated, then $n_\tau = n_j$. The vector of constraint forces $\boldsymbol{\lambda}$ is mapped to the joint space torques through the support Jacobian $\mathbf{J}_s \in \mathbb{R}^{3n_c \times n_u}$, which is obtained by stacking the constraint Jacobians as $\mathbf{J}_s = \begin{bmatrix} \mathbf{J}_{C_1}^T & \cdots & \mathbf{J}_{C_{n_c}}^T \end{bmatrix}^T$, with n_c the number of limbs in contact. Motion at the support contact points must be constrained. If the feet are modeled as point contacts, then each contact constraint introduces three equations ${}_I\mathbf{r}_{IC}(t) = const$, which can be differentiated twice to yield

$$\begin{aligned} {}_I\dot{\mathbf{r}}_{IC} &= \mathbf{J}_C\mathbf{u} = \mathbf{0} \\ {}_I\ddot{\mathbf{r}}_{IC} &= \mathbf{J}_C\dot{\mathbf{u}} + \dot{\mathbf{J}}_C\mathbf{u} = \mathbf{0}. \end{aligned} \tag{3}$$

A. Support-consistent dynamics

As shown in [16], if the motion of the whole body remains in the null space of the contact constraints described in (3), then the equations of motion (2) can be projected to a reduced space which still fully describes the system dynamics. Assuming that $\mathbf{J}_s^T$ has full column rank $\rho(\mathbf{J}_s^T) = 3n_c$, we can employ the *QR decomposition* to decompose the support Jacobian

$$\mathbf{J}_s^T = \mathbf{Q}\begin{bmatrix} \mathbf{R} \\ \mathbf{0}_{(n_u - 3n_c) \times 3n_c} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_c & \mathbf{Q}_u \end{bmatrix}\begin{bmatrix} \mathbf{R} \\ \mathbf{0}_{(n_u - 3n_c) \times 3n_c} \end{bmatrix}, \tag{4}$$

where $\mathbf{R} \in \mathbb{R}^{3n_c \times 3n_c}$, $\mathbf{Q}_c = \mathbf{Q}\mathbf{S}_c \in \mathbb{R}^{n_u \times 3n_c}$ and $\mathbf{Q}_u = \mathbf{Q}\mathbf{S}_u \in \mathbb{R}^{n_u \times (n_u - 3n_c)}$, with

$$\mathbf{S}_c = \begin{bmatrix} \mathbb{I}_{3n_c \times 3n_c} \\ \mathbf{0}_{(n_u - 3n_c) \times 3n_c} \end{bmatrix}, \quad \mathbf{S}_u = \begin{bmatrix} \mathbf{0}_{3n_c \times (n_u - 3n_c)} \\ \mathbb{I}_{(n_u - 3n_c) \times (n_u - 3n_c)} \end{bmatrix}. \tag{5}$$

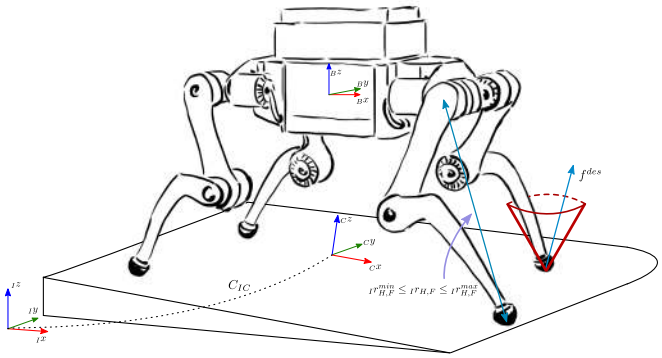

Fig. 2. Solving the locomotion problem requires to solve several sub-tasks at the same time. These typically include solving for the equations of motion, limiting the resulting contact forces to lie within the friction cone, track a desired motion.

By left multiplying eq.(2) by $\mathbf{Q}^T$, we can project the equations of motion to the contact constrained and unconstrained spaces

$$\mathbf{Q}_c^T[\mathbf{M}(\mathbf{q})\dot{\mathbf{u}} + \mathbf{h}(\mathbf{q}, \mathbf{u})] = \mathbf{Q}_c^T\mathbf{S}^T\boldsymbol{\tau} + \mathbf{R}\boldsymbol{\lambda} \tag{6a}$$

$$\mathbf{Q}_u^T[\mathbf{M}(\mathbf{q})\dot{\mathbf{u}} + \mathbf{h}(\mathbf{q}, \mathbf{u})] = \mathbf{Q}_u^T\mathbf{S}^T\boldsymbol{\tau}, \tag{6b}$$

As long as the motion of the system is in the null space of the contact constraints (3), eq.(6b) describes the dynamics of the entire system. Note that the constraint forces $\boldsymbol{\lambda}$ do not appear in eq.(6b). These can be reconstructed from eq.(6a) by writing

$$\boldsymbol{\lambda} = \mathbf{R}^{-1}\mathbf{Q}_C^T\left[\mathbf{M}(\mathbf{q})\dot{\mathbf{u}} + \mathbf{h}(\mathbf{q}, \mathbf{u}) - \mathbf{S}^T\boldsymbol{\tau}\right]. \tag{7}$$

III. Hierarchical Optimization

A task $\mathbf{T}$ can be defined as a set of linear equality and/or inequality constraints on the solution vector $\mathbf{x} \in \mathbb{R}^n$:

$$\mathbf{T} : \begin{cases} \mathbf{A}\mathbf{x} - \mathbf{b} = \mathbf{w} \\ \mathbf{D}\mathbf{x} - \mathbf{f} \leq \mathbf{v} \end{cases}, \tag{8}$$

where $\mathbf{w}$ and $\mathbf{v}$ are slack variables to be minimized. A set of tasks $\mathbf{T}_1, \ldots, \mathbf{T}_p$ can be either solved at the same time, optionally weighted against each other, or in a strict prioritized order. Solving p tasks yields an optimal solution $\mathbf{x}^*$. As shown in [7], to ensure strict prioritization the next solution $\mathbf{x}_{p+1}$ can be found in the null space of all higher priority equality constraints $\mathbf{Z}_p = \mathcal{N}\left(\underline{\mathbf{A}}_p\right)$, with $\underline{\mathbf{A}}_p = \begin{bmatrix} \mathbf{A}_1^T & \ldots & \mathbf{A}_p^T \end{bmatrix}^T$, of the higher priority constraints, yielding $\mathbf{x} = \mathbf{x}^* + \mathbf{Z}_p\mathbf{z}_{p+1}$, where $\mathbf{z}_{p+1}$ is a vector which lives in the row space of $\mathbf{Z}_p$. Solving for a new task $\mathbf{T}_{p+1}$ means computing $\mathbf{z}_{p+1}^*$ and $\mathbf{v}_{p+1}^*$ from the following QP

problem:

$$\begin{aligned}\min_{\mathbf{z}_{p+1},\mathbf{v}_{p+1}} \quad & \frac{1}{2}\|\mathbf{A}_{p+1}(\mathbf{x}^* + \mathbf{Z}_p\mathbf{z}_{p+1}) - \mathbf{b}_{p+1}\|^2 + \frac{1}{2}\|\mathbf{v}_{p+1}\|^2 \\ \text{s. t.} \quad & \mathbf{D}_{p+1}(\mathbf{x}^* + \mathbf{Z}_p\mathbf{z}_{p+1}) - \mathbf{f}_{p+1} \leq \mathbf{v}_{p+1} \\ & \mathbf{D}_p(\mathbf{x}^* + \mathbf{Z}_p\mathbf{z}_{p+1}) - \mathbf{f}_p \leq \mathbf{v}_p^* \\ & \vdots \\ & \mathbf{D}_1(\mathbf{x}^* + \mathbf{Z}_p\mathbf{z}_{p+1}) - \mathbf{f}_1 \leq \mathbf{v}_1^* \\ & \mathbf{v}_{p+1} \geq \mathbf{0}.\end{aligned} \tag{9}$$

Writing $\boldsymbol{\xi}_p^T = \begin{bmatrix}\mathbf{z}_p^T & \mathbf{v}_p^T\end{bmatrix}$, problem eq.(9) can be rewritten as

$$\begin{aligned}\min_{\boldsymbol{\xi}_{p+1}} \quad & \frac{1}{2}\boldsymbol{\xi}_{p+1}^T\mathbf{H}_{p+1}\boldsymbol{\xi}_{p+1} + \mathbf{c}_{p+1}^T\boldsymbol{\xi}_{p+1} \\ \text{s. t.} \quad & \tilde{\mathbf{D}}_{p+1}\boldsymbol{\xi}_{p+1} \leq \tilde{\mathbf{f}}_{p+1},\end{aligned} \tag{10}$$

where

$$\begin{aligned}\mathbf{H}_{p+1} &= \begin{bmatrix}\mathbf{Z}_p^T\mathbf{A}_{p+1}^T\mathbf{A}_{p+1}\mathbf{Z}_p & \mathbf{0} \\ \mathbf{0} & \mathbb{I}\end{bmatrix} \\ \mathbf{c}_{p+1} &= \begin{bmatrix}\mathbf{Z}_p^T\mathbf{A}_{p+1}^T(\mathbf{A}_{p+1}\mathbf{x}^* - \mathbf{b}_{p+1}) \\ \mathbf{0}\end{bmatrix} \\ \tilde{\mathbf{D}}_{p+1} &= \begin{bmatrix}\mathbf{D}_{p+1}\mathbf{Z}_p & -\mathbb{I} \\ \mathbf{D}_p\mathbf{Z}_p & \mathbf{0} \\ \vdots & \\ \mathbf{D}_1\mathbf{Z}_p & \mathbf{0} \\ \mathbf{0} & -\mathbb{I}\end{bmatrix}, \tilde{\mathbf{f}}_{p+1} = \begin{bmatrix}\mathbf{f}_{p+1} - \mathbf{D}_{p+1}\mathbf{x}^* \\ \mathbf{f}_p - \mathbf{D}_p\mathbf{x}^* + \mathbf{v}_p^* \\ \vdots \\ \mathbf{f}_1 - \mathbf{D}_1\mathbf{x}^* + \mathbf{v}_1^* \\ \mathbf{0}\end{bmatrix}.\end{aligned} \tag{11}$$

The computation of the null space basis $\mathbf{Z}_p$ of the stack of equality constraints $\underline{\mathbf{A}}$ can be implemented employing the *QR* or the *SVD* decomposition. To speed up the computation time, we use the former decomposition, which has a complexity that grows with the cube of the rows of its argument. For this reason, the computation of the null space of $\underline{\mathbf{A}}$ becomes slower as more equalities are added to the stack of tasks. A more efficient approach is then to use an iterative approach. Given two full row rank matrices $\mathbf{A}_1$ and $\mathbf{A}_2$, the null space basis of $\underline{\mathbf{A}} = \begin{bmatrix}\mathbf{A}_1^T & \mathbf{A}_2^T\end{bmatrix}^T$, $\mathbf{Z}_2$ can be computed as

$$\mathbf{Z}_2 = \mathcal{N}(\mathbf{A}_1)\,\mathcal{N}(\mathbf{A}_2\mathcal{N}(\mathbf{A}_1)) = \mathbf{Z}_1\mathcal{N}(\mathbf{A}_2\mathbf{Z}_1) \tag{12}$$

It can be easily shown that eq.(12) is a null space basis for both $\mathbf{A}_1$ and $\mathbf{A}_2$:

$$\mathbf{A}_1\mathbf{Z}_2 = \mathbf{A}_1\mathbf{Z}_1\mathcal{N}(\mathbf{A}_2\mathbf{Z}_1) = \mathbf{0}\cdot\mathcal{N}(\mathbf{A}_2\mathbf{Z}_1) = \mathbf{0} \tag{13a}$$

$$\mathbf{A}_2\mathbf{Z}_2 = \mathbf{A}_2\mathbf{Z}_1\mathcal{N}(\mathbf{A}_2\mathbf{Z}_1) = \mathbf{0}. \tag{13b}$$

IV. Whole body control as task formulation

We formulate the WBC problem as a quadratic optimization problem composed of linear equality and inequality tasks. To this end, all (dynamic) constraints and objectives need to be brought into the form of (8). This optimization problem is then solved using the hierarchical optimization algorithm to determine the desired actuator torques $\boldsymbol{\tau}_d$.

A. Equality tasks

Dynamic consistency: The equations of motion of a mechanical system set a relationship between the generalized accelerations $\dot{\mathbf{u}}$, the constraint forces $\boldsymbol{\lambda}$ and the actuation torques $\boldsymbol{\tau}$. As shown in section II-A, as long as the motion is in the null space of the contact constraints, the contact forces can be expressed as a linear combination of torques and resulting motion. As a consequence, we search for a solution $\mathbf{x}$ in the space of motion and torques, resulting in

$$\mathbf{x}_d = \begin{bmatrix}\dot{\mathbf{u}}_d^T & \boldsymbol{\tau}_d^T\end{bmatrix}^T, \tag{14}$$

with $\dot{\mathbf{u}}_d \in \mathbb{R}^n$ and $\boldsymbol{\tau}_d \in \mathbb{R}^r$. Using this optimization variable, the equations of motion (6b) can be written as

$$\mathbf{Q}_u^T\begin{bmatrix}-\mathbf{M}(\mathbf{q}) & \mathbf{S}^T\end{bmatrix}\mathbf{x}_d = \mathbf{Q}_u^T\mathbf{h}(\mathbf{q},\mathbf{u}). \tag{15}$$

Contact motion constraints: Since legs in contact with the ground are not allowed to move, the following task is required to ensure contact consistency:

$$\begin{bmatrix}{}_I\mathbf{J}_{F_i} & \mathbf{0}_{3\times n_\tau}\end{bmatrix}\mathbf{x}_d = -{}_I\dot{\mathbf{J}}_{F_i}\mathbf{u}. \tag{16}$$

Task space motion tracking: Similar to the motion constraints formulated for the feet in contact, we can also define motion tasks for other points of interest such as e.g. the system center of gravity. However, in contrast to the contacts, we want to define these tasks often not with respect to the inertial frame. In our previous works ([4], [5]) we have shown how to estimate the terrain orientation by fitting a plane through the most recent stance location of each foot. The roll and pitch of this plane, together with the yaw of the main body, define the orientation of the so-called *control frame* C, in which we define our desired body and feet motions. The mapping to operational space velocities in the inertial frame is defined by ${}_C\mathbf{v} = \mathbf{C}_{CI}\cdot{}_I\mathbf{v} = \mathbf{C}_{CI}\cdot{}_I\mathbf{J}\mathbf{u}$, where $\mathbf{C}_{CI}$ is the rotation matrix that projects the components of a vector from I frame to C frame. The control frame is time-varying, hence using the well known relationship $\dot{\mathbf{C}}_{CI} = \mathcal{S}({}_C\boldsymbol{\omega}_{CI})\mathbf{C}_{CI}$, where $\mathcal{S}(\cdot)$ is the skew-matrix operator defined as $\mathcal{S}(\mathbf{a})\mathbf{b} = \mathbf{a}\times\mathbf{b}$, we can compute the mapping between accelerations in I and C frame as ${}_C\dot{\mathbf{v}} = {}_C\mathbf{J}\dot{\mathbf{u}} + {}_C\dot{\mathbf{J}}\mathbf{u}$. The tasks for different motion objectives can now be formulated. Given a motion plan ${}_C\mathbf{r}_{x,y}^{des}$, ${}_C\dot{\mathbf{r}}_{x,y}^{des}$ and ${}_C\ddot{\mathbf{r}}_{x,y}^{des}$ for the x and y components of the center of mass, tracking in the Operational Space can be achieved by writing

$$\begin{aligned}\begin{bmatrix}{}_C\mathbf{J}_{B_{x,y}} & \mathbf{0}_{2\times n_\tau}\end{bmatrix}\mathbf{x}_d = {}_C\ddot{\mathbf{r}}_{x,y}^{des} + \mathbf{K}_D\cdot{}_C\dot{\tilde{\mathbf{r}}}_{x,y} \\ + \mathbf{K}_P\cdot{}_C\tilde{\mathbf{r}}_{x,y} - {}_C\dot{\mathbf{J}}_{B_{x,y}}\mathbf{u},\end{aligned} \tag{17}$$

where $\mathbf{K}_P, \mathbf{K}_D \in \mathbb{R}^{2\times 2}$ are diagonal positive definite matrices which define proportional and derivative gains, ${}_C\dot{\tilde{\mathbf{r}}}_{x,y}$ is the velocity error ${}_C\dot{\mathbf{r}}_{x,y}^{des} - {}_C\dot{\mathbf{r}}_{x,y}$, and ${}_C\tilde{\mathbf{r}}_{x,y}$ is the position error ${}_C\mathbf{r}_{x,y}^{des} - {}_C\mathbf{r}_{x,y}$.

The same approach is taken for the main body orientation given the desired motion plan $\mathbf{q}_{CB}^{des}$ and ${}_C\boldsymbol{\omega}_{CB}{}^{des}$. The orientation error is defined by the *box minus* operator [2] which yields $\tilde{\boldsymbol{\varphi}} = \mathbf{q}_{CB}^{des}\boxminus\mathbf{q}_{CB} \in \mathbb{R}^3$, while the velocity error

is defined as ${}_C\tilde{\boldsymbol{\omega}}_{CB} = {}_C\boldsymbol{\omega}^{des}_{CB} - {}_C\boldsymbol{\omega}_{CB}$. Hence, the tracking task can be written as

$$\begin{bmatrix} {}_C\mathbf{J}_{B_{x,y}} & \mathbf{0}_{3\times n_\tau} \end{bmatrix} \mathbf{x}_d = \mathbf{K}_D \cdot {}_C\tilde{\boldsymbol{\omega}}_{CB} + \mathbf{K}_P \cdot \tilde{\boldsymbol{\varphi}} - {}_C\dot{\mathbf{J}}_B\mathbf{u}. \tag{18}$$

Finally, for foot motions we define fifth order Hermite splines which provide desired position, velocity and acceleration in the control frame C. We define the $i-th$ foot velocity error as ${}_C\dot{\tilde{\mathbf{r}}}_{F_i} = {}_C\dot{\mathbf{r}}^{des}_{F_i} - {}_C\dot{\mathbf{r}}_{F_i}$ and the position error as ${}_C\tilde{\mathbf{r}}_{F_i} = {}_C\mathbf{r}^{des}_{F_i} - {}_C\mathbf{r}_{F_i}$.

$$\begin{aligned} \begin{bmatrix} {}_C\mathbf{J}_{F_i} & \mathbf{0}_{3\times n_\tau} \end{bmatrix} \mathbf{x}_d = {}_C\ddot{\mathbf{r}}^{des}_{F_i} + \mathbf{K}_D \cdot {}_C\dot{\tilde{\mathbf{r}}}_{F_i} \\ + \mathbf{K}_P \cdot {}_C\tilde{\mathbf{r}}_{F_i} - {}_C\dot{\mathbf{J}}_{F_i}\mathbf{u}. \end{aligned} \tag{19}$$

Energy optimization: Similar to the motion optimization objectives, we can also introduce tasks for contact forces or torques. These tasks are mostly taken at lowest priority. Optimization of joint torque is typically used in order to improve energetic efficiency [10]:

$$\begin{bmatrix} \mathbf{0}_{n_\tau\times n_u} & \mathbb{I}_{n_\tau\times n_\tau} \end{bmatrix} \mathbf{x}_d = 0. \tag{20}$$

Slippage reduction: In order to minimize contact forces, which typically reduces the risk of slippage in case of model uncertainty, we can formulate the following:

$$\mathbf{R}^{-1}\mathbf{Q}_c^T \begin{bmatrix} -\mathbf{M}(\mathbf{q}) & \mathbf{S}^T \end{bmatrix} \mathbf{x}_d = \mathbf{R}^{-1}\mathbf{Q}_c^T\mathbf{h}(\mathbf{q},\mathbf{u}). \tag{21}$$

Manipulability optimization: Another very interesting optimization criterion is manipulability of the system in order to stay away from non-singular postures. Rather than setting a constant desired main body height w.r.t. the estimated terrain, we can use this DOF to avoid the knee singularities of all legs. To achieve this, we use the definition of manipulability measure as described in [19]:

$$m(\mathbf{q}) = \sqrt{\det\left({}_I\mathbf{J}_{BF}(q) \cdot {}_I\mathbf{J}^T_{BF}(q)\right)}, \tag{22}$$

where ${}_I\mathbf{J}_{B,F}(q)$ is the 3×3 translational Jacobian that maps leg joint velocities to the operational space linear velocities of a foot. We set the desired height motion as a weighted sum of the gradients of the manipulability measures of all legs w.r.t. to the knee flexion-extension joints $q_{k_{KFE}}$ and damp it as a function of the main body height velocity:

$$\begin{aligned} {}_I\ddot{z}^{des}_B &= k_0 \sum_{k=0}^{n_{legs}} w_k \frac{\partial m_k(\mathbf{q})}{\partial q_{k_{KFE}}} - k_{dI}\dot{z}_B \\ &= k_0 \sum_{k=0}^{n_{legs}} w_k m_k(\mathbf{q}) \operatorname{Tr}\{\frac{\partial {}_I\mathbf{J}_{k_{BF}}}{\partial q_{k_{KFE}}} \cdot {}_I\mathbf{J}^\dagger_{k_{BF}}\} - k_{dI}\dot{z}_B, \end{aligned} \tag{23}$$

where ${}_I\mathbf{J}^\dagger_{k_{BF}}$ is the pseudo-inverse of ${}_I\mathbf{J}_{k_{BF}}$, $w_k \geq 0, w_k \in \mathbb{R}$ is the weighting relative to leg k, and k_0 is a positive real weighting factor. We compute the Hessian matrix $\partial {}_I\mathbf{J}_{k_{BF}}/\partial q_{k_{KFE}}$ using the algorithms described in [11]. The task can then be written as

$$\begin{bmatrix} \mathbf{S}_z & \mathbf{0}_{1\times n_\tau} \end{bmatrix} \mathbf{x}_d = {}_I\ddot{z}^{des}_B, \tag{24}$$

where $\mathbf{S}_z \in \mathbb{R}^{1\times n_u}$ is the row vector which selects the entry in $\mathbf{x}_d$ associated to the main body height motion.

B. Inequality Tasks

In order to set motion constraints $\mathbf{r}^{min}(t) \leq \mathbf{r}(t) \leq \mathbf{r}^{max}(t)$ in the configuration space, we can approximate $\mathbf{r}(t)$ with its Taylor expansion around the current time instant $\bar{t}$ truncated at the second derivative:

$$\mathbf{r}(t) \simeq \mathbf{r}(\bar{t}) + \dot{\mathbf{r}}(\bar{t})\delta t + \frac{1}{2}\ddot{\mathbf{r}}(\bar{t})\delta t^2, \tag{25}$$

with $\delta t = t - \bar{t}$. The inequality constraints

$$\mathbf{r}^{min} \leq \mathbf{r}(\bar{t}) + \dot{\mathbf{r}}(\bar{t})\delta t + \frac{1}{2}\ddot{\mathbf{r}}(\bar{t})\delta t^2 \leq \mathbf{r}^{max} \tag{26}$$

can be then rewritten as

$$\begin{aligned} \frac{\delta t^2}{2}\mathbf{J}\dot{\mathbf{u}} &\leq (\mathbf{r}^{max} - \mathbf{r}(\bar{t})) - \delta t\left(\mathbf{J} + \frac{\delta t}{2}\dot{\mathbf{J}}\right)\mathbf{u} \\ -\frac{\delta t^2}{2}\mathbf{J}\dot{\mathbf{u}} &\leq (\mathbf{r}(\bar{t}) - \mathbf{r}^{min}) + \delta t\left(\mathbf{J} + \frac{\delta t}{2}\dot{\mathbf{J}}\right)\mathbf{u}. \end{aligned} \tag{27}$$

The time interval δt can be interpreted as a degree of freedom in the design of the control system. By setting it as $\delta t = k_t t_s$ with t_s the control period, the positive parameter $k_t \in \mathbb{R}$ can be tuned to set the number of control loops necessary for the inequality constraints in eq.(27) to be deactivated, which determines how soft the constraint will be perceived at the configuration level.

Torque limits: Using this approach, we can formulate the following joint torque limitation task to ensure that the solution complies with the hardware limitations defined by minimum τ^{min} and maximum τ^{max} admissible torques.

$$\begin{bmatrix} \mathbf{0}_{n_\tau\times n_u} & \mathbb{I}_{n_\tau\times n_\tau} \\ \mathbf{0}_{n_\tau\times n_u} & -\mathbb{I}_{n_\tau\times n_\tau} \end{bmatrix} \mathbf{x}_d \leq \begin{pmatrix} \tau^{max}\mathbb{1}_{n_\tau\times 1} \\ -\tau^{min}\mathbb{1}_{n_\tau\times 1} \end{pmatrix}, \tag{28}$$

where $\mathbb{1}_{n_\tau\times 1} = \begin{bmatrix} 1 & \cdots & 1 \end{bmatrix}^T \in \mathbb{R}^{n_\tau}$.

Contact force limits: To avoid slipping, the resulting contact forces $\boldsymbol{\lambda} \in \mathbb{R}^{3n_c}$ must be constrained to lie within the friction cones. To obtain linear constraints, we approximate the friction cone with a pyramid. Since we approximate the local terrain with a plane whose orientation defines the control frame, all the friction pyramids will be aligned with it. Hence, given the friction coefficient μ, the heading ${}_I\hat{\mathbf{h}}$, lateral ${}_I\hat{\mathbf{l}}$ and normal ${}_I\hat{\mathbf{n}}$ directions of the control frame expressed in the inertial frame, the friction constraints on the contact force ${}_I\mathbf{f}_i$ for the $i-th$ support leg can be written as

$$\begin{aligned} ({}_I\hat{\mathbf{h}} - {}_I\hat{\mathbf{n}}\mu)^T {}_I\mathbf{f}_i &\leq \mathbf{0} \\ -({}_I\hat{\mathbf{h}} + {}_I\hat{\mathbf{n}}\mu)^T {}_I\mathbf{f}_i &\leq \mathbf{0} \\ ({}_I\hat{\mathbf{l}} - {}_I\hat{\mathbf{n}}\mu)^T {}_I\mathbf{f}_i &\leq \mathbf{0} \\ -({}_I\hat{\mathbf{l}} + {}_I\hat{\mathbf{n}}\mu)^T {}_I\mathbf{f}_i &\leq \mathbf{0}. \end{aligned} \tag{29}$$

To avoid jumps in the actuation signals when a leg is switching between support and swing mode, we modulate the normal component of $\mathbf{f}_i$ according to the motion plan by gradually limiting it to zero before lift-off, and gradually ramping up from zero on touch-down. To achieve this, we add two additional constraints:

$$f_i^{min} \leq {}_I\hat{\mathbf{n}}^T {}_I\mathbf{f}_i \leq f_i^{max}, \tag{30}$$

where f_i^{max} is computed as a function of the stance and swing phases, and $f_i^{min} \geq 0$ is set to a fixed value. Equations (29) and (30) can be rewritten in compact form as $\mathbf{B}_i \mathbf{f}_i \leq \boldsymbol{\beta}_i$. Using eq.(7), the task constraints in terms of $\mathbf{x}_d$ will then be written as

$$\mathbf{B}\mathbf{R}^{-1}\mathbf{Q}_c^T \begin{bmatrix} \mathbf{M}(\mathbf{q}) & \mathbf{S}^T \end{bmatrix} \mathbf{x}_d \leq \mathbf{h}(\mathbf{q}, \mathbf{u}) - \mathbf{B}\mathbf{R}^{-1}\mathbf{Q}_c^T \boldsymbol{\beta}, \quad (31)$$

where $\mathbf{B} = \begin{bmatrix} \mathbf{B}_1^T & \dots & \mathbf{B}_{n_c}^T \end{bmatrix}^T$ and $\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^T & \dots & \boldsymbol{\beta}_{n_c}^T \end{bmatrix}^T$.

Task space motion limits: To limit task space positions, we make use of formulation (27). The main body height ${}_I\mathbf{r}_{IB_z}$ limits are set w.r.t. the footprint height h_z, which is computed as

$$h_z = \frac{1}{n_c} \sum_{k=0}^{n_{legs}} w_k {}_I\mathbf{r}_{IF_k}, \quad (32)$$

where n_c is the number of legs in contact. The weight w_k is 1 when leg k is in contact, 0 otherwise. The inequality motion task can be written as

$$h^{min} \leq {}_I\mathbf{r}_{IB_z} - h_z \leq h^{max}. \quad (33)$$

As previously discussed, the task can be rewritten in joint space as

$$\begin{bmatrix} \frac{\delta t^2}{2} {}_I\mathbf{J}_{B_z} & \mathbf{0}_{1\times n_\tau} \\ -\frac{\delta t^2}{2} {}_I\mathbf{J}_{B_z} & \mathbf{0}_{1\times n_\tau} \end{bmatrix} \mathbf{x}_d \leq$$
$$\begin{pmatrix} h^{max} - {}_I\mathbf{r}_{IB_z} + h_z \\ {}_I\mathbf{r}_{IB_z} - h_z - h^{min} \end{pmatrix} - \delta t \begin{pmatrix} ({}_I\mathbf{J}_{B_z} + \frac{\delta t}{2} {}_I\dot{\mathbf{J}}_{B_z}) \\ -({}_I\mathbf{J}_{B_z} + \frac{\delta t}{2} {}_I\dot{\mathbf{J}}_{B_z}) \end{pmatrix} \mathbf{u}. \quad (34)$$

Limb extensions limits can be seen as limits on the relative motion between feet and hips. To achieve terrain adaptation, we write these limits such that the hips will follow the z component of the feet position in I frame. Hence, for each leg k we write

$$\begin{bmatrix} \frac{\delta t^2}{2} {}_I\mathbf{J}_{H_z^k} & \mathbf{0}_{1\times n_\tau} \\ -\frac{\delta t^2}{2} {}_I\mathbf{J}_{H_z^k} & \mathbf{0}_{1\times n_\tau} \end{bmatrix} \mathbf{x}_d \leq$$
$$\begin{pmatrix} h^{max} - {}_I\mathbf{r}_{IH_z^k} + h_z^k \\ {}_I\mathbf{r}_{IH_z^k} - h_z^k - h^{min} \end{pmatrix} - \delta t \begin{pmatrix} ({}_I\mathbf{J}_{H_z^k} + \frac{\delta t}{2} {}_I\dot{\mathbf{J}}_{H_z^k}) \\ -({}_I\mathbf{J}_{H_z^k} + \frac{\delta t}{2} {}_I\dot{\mathbf{J}}_{H_z^k}) \end{pmatrix} \mathbf{u}, \quad (35)$$

where ${}_I\mathbf{J}_{H_z^k}$ is the third row of the Jacobian which projects the generalized velocities to $k-th$ hip linear velocities in operational space, ${}_I\mathbf{r}_{IH_z}$ is the z component of the position of the $k-th$ hip in world frame, h_z^k is the height of the foot of the $k-th$ leg, and h^{min} and h^{max} are the minimum and maximum user-defined relative distances between a hip and a foot.

V. Terrain adaptation

In a previous work [4] we have successfully shown locomotion over unperceived tilted terrain. However, the method described there fails when stepping down from a high step as shown in Fig.3. The robot's limb tries to blindly extend until kinematic limits are encountered. In such a situation, it is necessary to plan an appropriate whole-body motion which should take into account several factors such as balancing and kinematic limits. Instead, we can now exploit two advantages of the HO framework: each task can set both equality and inequality constraints in the Operational Space, and tasks are solved in a strict prioritized order. We defer the problem of designing a motion in the planning phase to the problem of reacting to task constraints in the control phase. To achieve this, we set an inequality constraint task which imposes limb extensions limits at a higher priority than torso orientation and height motion tracking, but at a lower priority than swing foot motion tracking (see Tab.I). As soon as the kinematic limits are violated, the main body will tilt towards the feet, achieving a reactive and perception-less terrain adaptation with no requirements on motion design (Fig.3). Balancing and contact force limit tasks are solved with the highest priority, which guarantees balance at all times.

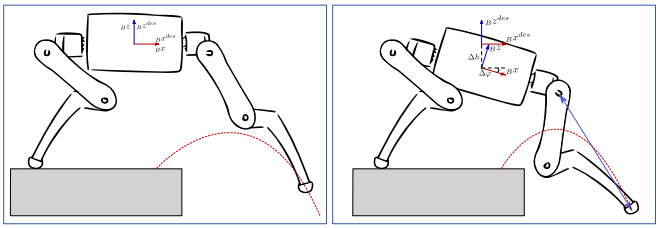

Fig. 3. When a regular set of tasks is employed, walking down a high step requires prior knowledge of the terrain and appropriate planning (left). With our method, a robot can reactively adapt to unperceived steps (right).

VI. Experiments

A. Setup

Our experiments[1] were conducted on ANYmal [9], an accurately torque-controllable quadrupedal robot. Control signals are generated in a $400Hz$ control loop which runs on a dedicated on-board computer together with state estimation [1] and communication with the drives. For modeling of kinematics and dynamics, we use the open-source Rigid Body Dynamics Library [15] (RBDL), which is a C++ implementation of the algorithms described in [3]. Table I describes which sub-tasks were implemented in each experiment that we describe in the following subsections. To numerically solve each optimization problem, we use a custom version of the QuadProg++ [14] library, a C++ open-source QP solver which implements the Active Set algorithm described in [6].

B. Full terrain adaptation

To test our framework, we are using a simple motion planner that implements a walking behavior based on the Zero Moment Point (ZMP) [21]. We plan the motion of the center of mass by solving a constrained Quadratic Programming problem, as shown in [12] and [22]. By

[1] https://youtu.be/AjiLCbJUYKI

TABLE I

THE TASKS USED IN THE EXPERIMENTS DESCRIBED IN SECTION VI-B. EACH TASK IS ASSOCIATED WITH A PRIORITY (1 IS THE HIGHEST).

Priority	Task
1	Equations of Motion
2	No contact motion
3	Torque limits
4	Friction cone and $\boldsymbol{\lambda}$ modulation
5	Desired torso x, y position
6	Swing foot motion tracking
7	Limb kinematic limits
8	Main body yaw
9	Main body height
10	Main body roll and pitch
11	Contact force minimization

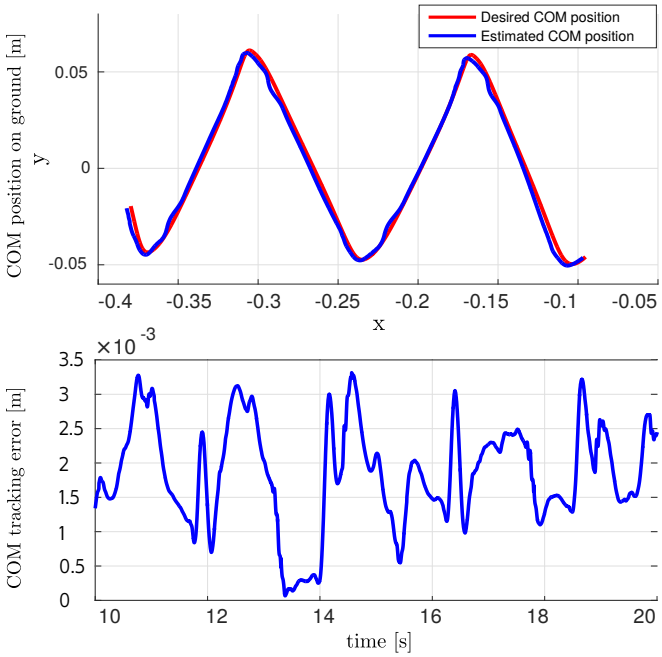

Fig. 4. The tracking behavior of the center of mass (COM) of the whole body during a ZMP based walking gait. During this experiment, the position error had an upper bound of 3.4 mm.

minimizing the desired accelerations of the motion plan, the optimization problem provides the coefficients of a fifth order Hermite spline which result in a smooth continuous motion. The results presented in Fig.4 show the tracking performance obtained while walking on flat terrain. To execute locomotion over uneven terrain, we implement the method discussed in section V, i.e. we constrain the motion of the main body to avoid over-extension of the limbs. We achieve this by setting an inequality task which has a higher priority than main body height and orientation tracking tasks described in section IV. The step-down sequence depicted in Fig.5 shows how the control framework allows to negotiate a $14cm$ step (23% of the maximum leg length). Automatic terrain adaptation is achieved when the orientation tracking task is subject to the higher priority limb extension kinematic limits. The 10-14 s time interval from Fig.6 shows how the orientation error is kept low as long as the relative distance on the ${}_I z$ direction is below the kinematic limits, which can be set by the user. When the foot motion is such that these limits are violated, the control framework tries to deactivate them by tilting the main body in the direction of the foot motion. Fig.6 additionally shows that the main body height tracking task performance also degrades during the adaptation phase. The entire adaptation behavior can be summarized in the following steps, clarified in Fig.5: 1) the desired height of the right front (RF) leg is gradually lowered until a touch-down event is registered; 2) when the limb extensions limits are violated for the RF leg, the main body torso orientation adapts by tilting forward; 3) the left hind leg (LH) kinematic limits are violated because of the main body tilting, so the height of the main body gradually decreases until the limits are no longer active.

VII. CONCLUSIONS AND FUTURE WORK

A whole body control framework was implemented which uses hierarchical optimization to solve a set of prioritized tasks. We present a clever combination of several inequality tasks in order to optimally exploit the system redundancy and hence adaptively shape the motion of the robot as a function of the terrain. The framework has allowed ANYmal, a fully torque-controllable quadrupedal robot, to overcome unperceived steps, with no requirement on motion design from the user. The framework has been thoroughly tested in a series of experiments, which include walking, manipulability optimization and terrain adaptation. Next steps involve using this framework to implement features such as self-collision avoidance and the use of additional limbs such as arms to allow for manipulation.

REFERENCES

[1] M. Bloesch, C. Gehring, P. Fankhauser, M. Hutter, M. A. Hoepflinger, and R. Siegwart. State estimation for legged robots on unstable and slippery terrain. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 6058–6064. IEEE, nov 2013.

[2] M. Bloesch, H. Sommer, T. Laidlow, M. Burri, G. Nuetzi, P. Fankhauser, D. Bellicoso, C. Gehring, S. Leutenegger, M. Hutter, and R. Siegwart. A Primer on the Differential Calculus of 3D Orientations. jun 2016.

[3] R. Featherstone. *Rigid Body Dynamics Algorithms*. Springer US, Boston, MA, 2008.

[4] C. Gehring, C. D. Bellicoso, S. Coros, M. Bloesch, P. Fankhauser, M. Hutter, and R. Siegwart. Dynamic trotting on slopes for quadrupedal robots. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5129–5135. IEEE, sep 2015.

[5] C. Gehring, S. Coros, M. Hutler, C. D. Bellicoso, H. Heijnen, R. Diethelm, M. Bloesch, P. Fankhauser, J. Hwangbo, M. Hoepflinger, and R. Siegwart. Practice makes perfect: An optimization-based approach to controlling agile motions for a quadruped robot. *IEEE Robotics Automation Magazine*, 23(1):34–43, March 2016.

[6] D. Goldfarb and a. Idnani. A numerically stable dual method for solving strictly convex quadratic programs. *Mathematical Programming*, 27(1):1–33, 1983.

[7] A. Herzog, N. Rotella, S. Mason, F. Grimminger, S. Schaal, and L. Righetti. Multi-Contact Interaction with Hierarchical Inverse Dynamics and Momentum Trajectory Generation. In *Proceedings of Dynamic Walking*, 2015.

[8] M. Hutter, C. Gehring, M. Hoepflinger, M. Bloesch, and R. Siegwart. Towards combining Speed, Efficiency, Versatility and Robustness in an Autonomous Quadruped. *IEEE Transactions on Robotics (in Press)*, 2014.

Fig. 5. A sequence showing ANYmal reacting to an unperceived $14cm$ step. The reactive behavior is obtained by imposing motion constraints between the hips and the feet. The terrain is initially assumed to be flat. The figure shows how the main body adapts to avoid over-extension of the right front (sequence 5-7) and right hind (sequence 16-18) legs. Main body motion tracking is sacrificed in favor of terrain adaptation.

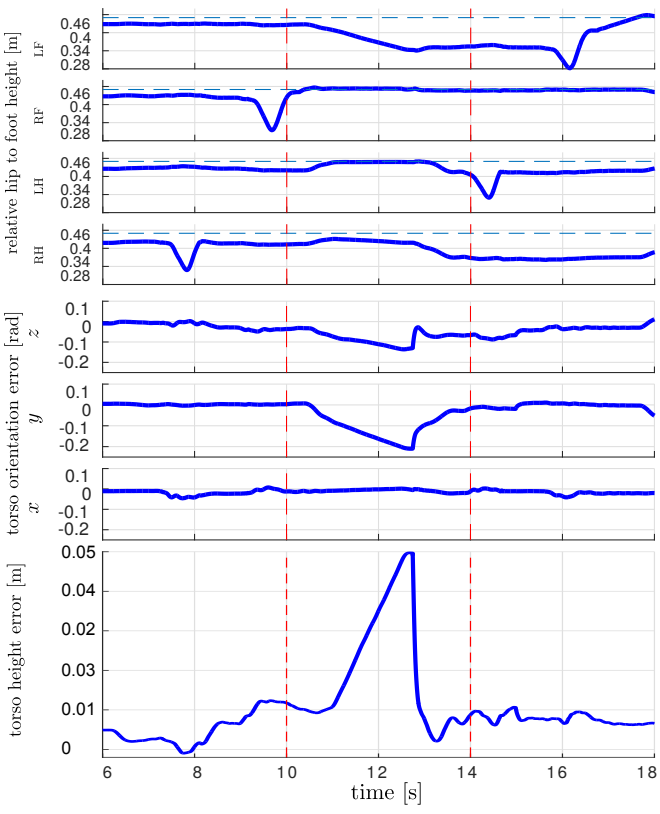

Fig. 6. The results obtained from the terrain adaptation experiment. With reference to the sequence of frames numbered 3-8 in Fig.5, it can be seen how the kinematic limits are reached for the right fore (RF) leg shortly after 10 s. At that point, the pitch and roll tracking tasks are taken over by the limb extension limits task which also overcome height tracking when the left hind (LH) leg violates the limits. The limits are soften by choosing $k_t t_s = 0.1$. When the swing foot makes contact, a new estimation of the terrain is available, which yields a new pose reference for the torso.

[9] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, P. Fankhauser, M. Bloesch, R. Diethelm, and S. Bachmann. ANYmal - A Highly Mobile and Dynamic Quadrupedal Robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

[10] M. Hutter, H. Sommer, C. Gehring, M. Hoepflinger, M. Bloesch, and R. Siegwart. Quadrupedal locomotion using hierarchical operational space control. *The International Journal of Robotics Research*, 33(8):1047–1062, may 2014.

[11] M. Iwamura and M. Nagao. A method for computing the Hessian tensor of loop closing conditions in multibody systems. *Multibody System Dynamics*, 30(2):173–184, 2013.

[12] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal. Fast, robust quadruped locomotion over challenging terrain. In *2010 IEEE Int. Conf. on Robotics and Automation*, pages 2665–2670. IEEE, may 2010.

[13] O. Kanoun, F. Lamiraux, and P. B. Wieber. Kinematic Control of Redundant Manipulators: Generalizing the Task-Priority Framework to Inequality Task. *IEEE Trans. on Robotics*, 27(4):785–792, 2011.

[14] Luca Di Gaspero. QuadProg++. Available at `http://quadprog.sourceforge.net/`, 1998.

[15] Martin Felis. Rigid Body Dynamics Library. Available at `http://rbdl.bitbucket.org/`.

[16] M. Mistry, J. Buchli, and S. Schaal. Inverse dynamics control of floating base systems using orthogonal decomposition. *2010 IEEE Int. Conf. on Robotics and Automation*, (3):3406–3412, 2010.

[17] L. Sentis. *Synthesis and Control of Whole-body Behaviors in Humanoid Systems*. PhD thesis, Stanford, CA, USA, 2007. AAI3281945.

[18] L. Sentis and O. Khatib. A whole-body control framework for humanoids operating in human environments. In *Proceedings 2006 IEEE Int. Conf. on Robotics and Automation, 2006. ICRA 2006.*, pages 2641–2648, May 2006.

[19] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. 2009.

[20] B. Siciliano and J. J. E. Slotine. A general framework for managing multiple tasks in highly redundant robotic systems. In *Advanced Robotics, 1991. 'Robots in Unstructured Environments', 91 ICAR., Fifth International Conference on*, pages 1211–1216 vol.2, 1991.

[21] M. Vukobratov and B. Borovac. Zero-moment point thirty five years of its life. *International Journal of Humanoid Robotics*, 01(01):157–173, 2004.

[22] A. W. Winkler, C. Mastalli, M. Focchi, D. G. Caldwell, and I. Havoutis. Planning and Execution of Dynamic Whole-Body Locomotion for a Hydraulic Quadruped on Challenging Terrain. *Icra*, pages 5148–5154, 2015.