THE UNIVERSITY of EDINBURGH

Edinburgh Research Explorer

Online Optimal Impedance Planning for Legged Robots

Citation for published version:
Angelini, F, Xin, G, Wolfslag, W, Tiseo, C, Mistry, M, Garabini, M, Bicchi, A & Vijayakumar, S 2020, Online Optimal Impedance Planning for Legged Robots. in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* . Institute of Electrical and Electronics Engineers (IEEE), pp. 6028-6035, 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, Macau, China, 4/11/19. https://doi.org/10.1109/IROS40897.2019.8967696

Digital Object Identifier (DOI):
10.1109/IROS40897.2019.8967696

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

OPEN ACCESS

Download date: 05. May. 2024

Online Optimal Impedance Planning for Legged Robots

Franco Angelini[1,2,3], Guiyang Xin[4], Wouter J. Wolfslag[4], Carlo Tiseo[4],
Michael Mistry[4], Manolo Garabini[1,3], Antonio Bicchi[1,2,3] and Sethu Vijayakumar[4]

***Abstract*— Real world applications require robots to operate in unstructured environments. This kind of scenarios may lead to unexpected environmental contacts or undesired interactions, which may harm people or impair the robot. Adjusting the behavior of the system through impedance control techniques is an effective solution to these problems. However, selecting an adequate impedance is not a straightforward process. Normally, robot users manually tune the controller gains with trial and error methods. This approach is generally slow and requires practice. Moreover, complex tasks may require different impedance during different phases of the task. This paper introduces an optimization algorithm for online planning of the Cartesian robot impedance to adapt to changes in the task, robot configuration, expected disturbances, external environment and desired performance, without employing any direct force measurements. We provide an analytical solution leveraging the mass-spring-damper behavior that is conferred to the robot body by the Cartesian impedance controller. Stability during gains variation is also guaranteed. The effectiveness of the method is experimentally validated on the quadrupedal robot ANYmal. The variable impedance helps the robot to tackle challenging scenarios like walking on rough terrain and colliding with an obstacle.**

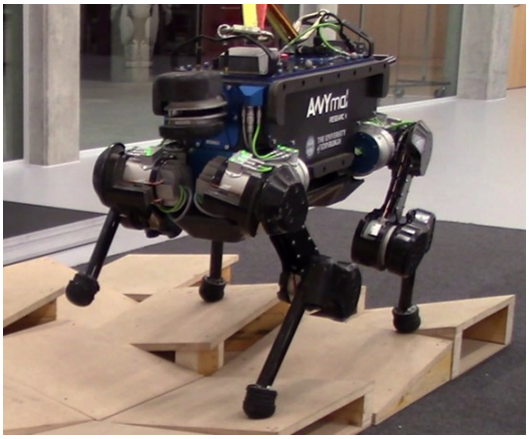


Fig. 1. Quadrupedal robot ANYmal walking on rough terrain with online impedance planning.

I. INTRODUCTION

The development of more flexible and adaptable control architectures and technologies has initiated a shift of paradigm in robotics. This new trend aspires to move robots from traditional industrial cages to unstructured environments. For example, there has been an increasing interest in the deployment of robots in hazardous environments, such as mines, disaster area, nuclear sites, space and offshore platforms. Despite the great improvements in controlling robots in challenging dynamic condition, they are still not sufficiently reliable and adaptable to be employed in such locations. Especially, the intrinsic unpredictability of such environments makes the exteroceptive information not reliable. Thus, moving around in such conditions may severely affect the robot: there could be unexpected interactions or disturbances, which may destabilize or damage the robot. An adaptive behavior may help to mitigate or even avoid these dangerous scenarios. Impedance control [1] tackles these issues by directly setting the robot stiffness and damping. In other words, this technique allows to specify the force produced in response to a motion imposed by the environment, leading to the desired robust and adaptive behavior.

Applying impedance control to legged robots is a common practice in literature [2], [3], [4], [5], [6]. However, there is currently no established basis upon which to choose the impedance gains for a specific system or scenario. Tuning these parameters is generally tedious and time- consuming. An experienced robot programmer may be able to find some fine-tuned fixed gains that achieve the desired behavior, but different tasks and different robot configurations may require different parameters. Thus, it is of paramount importance to replace the typical trial and error impedance modulation with an automatic one.

Literature proposes several approaches to solve this problem. Learning techniques [7], [8], [9], [10], [11] are an effective method to achieve this goal. The price of these strategies is that they require demonstrations from humans, rich data sets, or, generally, a learning process that may require several iterations to converge. Thus, despite their efficacy, these methods can not be employed in every scenario.

In [12] the authors formulate a robust optimization problem to compute the optimal control action and compliance value for a robot-environment interaction scenario under uncertainties. This techinique is validated on a handover task, however, it is an offline approach.

This work was funded by the European Commission Horizon 2020 Work Programme: THING ICT-2017-1 780883 and MEMMO ID: 780684, as well as EPSRCs RAI Hubs for Extreme and Challenging Environments: ORCA EPR026173/1 and NCNR EPR02572X/1.

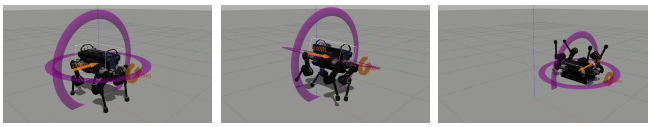
[1]Centro di Ricerca "Enrico Piaggio", Università di Pisa, Largo Lucio Lazzarino 1, 56126 Pisa, Italy

[2]Soft Robotics for Human Cooperation and Rehabilitation, Fondazione Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy

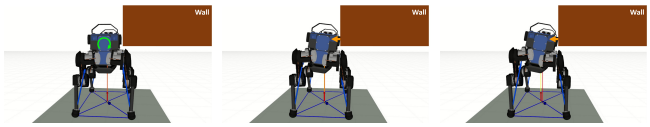
[3]Dipartimento di Ingegneria dell'Informazione, Università di Pisa, Largo Lucio Lazzarino 1, 56126 Pisa, Italy

[4]Institute for Perception, Action, and Behaviour, School of Informatics, The University of Edinburgh, Informatics Forum, 10 Crichton Street, Edinburgh, EH8 9AB, United Kingdom

frncangelini@gmail.com

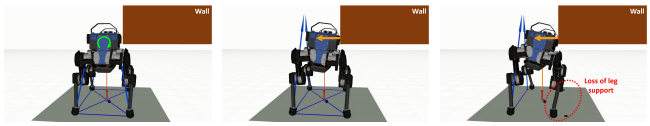


(a) Step force disturbance in the low impedance case.



(b) Wall interaction in the low impedance case.

(c) Step force disturbance in the high impedance case.



(d) Wall interaction in the high impedance case.

Fig. 2. Interactions in low and high impedance cases. (a,c) step force disturbance with module equal to 170N. (b,d) interaction with a wall (brown box)) due to a roll movement. The impedance of the robot is set as low in (a) and (b), and it is set as high in (c) and in (d). In the case of low impedance the robot falls in case of an external force disturbance, but it is more robust to contacts with the environment. In the case of high impedance the robot is able to reject force disturbances, but the contact with the wall destabilizes it. This is highlighted by the dashed red circle: the robot loses support on two legs.

Other strategies focus on an online adaptation of the impedance based on sensory feedbacks (mostly force sensors). In [13] the authors employ adaptive dynamic programming to find the impedance gains that minimize the tracking error and interaction forces with an unknown environment. The results are validated with simulations. In [5] an impedance tuning method based on time-varying Lyapunov stability margins is proposed and experimentally verified. The goal is to guarantee the balancing of a humanoid robot during walking. In [14] is presented an impedance controller that automatically regulates the impedance of the robot based on expected interaction forces and local sensory data (vision, tracking error and interaction forces). The authors experimentally validate it on a manipulation task.

All these works rely on force measurements to modify the impedance of the robot accordingly, which may not be available on the robot. In this paper we propose a method for online Cartesian impedance planning of a robot without utilizing direct force measurements. The idea is to leverage the mass-spring-damper model resulting from the employment of a task-space impedance controller. Instead of changing the impedance based on force measurements, we define a constrained optimization problem in which the cost index aims at highest compliance (corresponding to the most interaction-tolerant robot behavior), and the constraints guarantee the desired task performance. Such constraints take into account: desired maximum tracking error, robot configuration, robot stability, expected disturbances and accuracy with which the environment is perceived. To the best of authors' knowledge, this is the first time a legged robot without external force measurements is able to modulate its impedance online.

In this work we show that variable gains lead to better performance and robustness to unexpected interactions. We will rely on the impedance controller proposed in [3], developing a module that will plan the gains of this controller online. In particular, we will focus on the impedance modulation of the torso of the robot. Note that in this case the employment of any force sensor at the point of contact is particularly challenging since the interaction force could be applied at any point of the robot body. The proposed method will be experimentally validated on the quadrupedal robot ANYmal [6] (Fig. 1).

II. PROBLEM DEFINITION

Legged robots are developed to solve tasks in unstructured environment, thus their impedance modulation is a trade-off between performance in achieving the task and robustness to unexpected environmental interactions. Low impedance improves the compliance to undesired contacts, while high impedance improves tracking performance and disturbance rejection. Fig. 2 depicts two examples of this trade-off. The case of study is the response of the robot ANYmal in presence of a step force disturbance (Fig. 2(a,c)) or an interaction with the environment (Fig. 2(b,d)). Both cases are analyzed with low (Fig. 2(a,b)) and high impedance (Fig. 2(c,d)). Comparing Fig. 2(a) and Fig. 2(c) shows that high impedance allows to reject force disturbances. On the other hand, high impedance (Fig. 2(d)) causes large contact forces that destabilize the robot, while low impedance allows smaller contact forces (Fig. 2(b)), thus more robustness.

The aim of this paper is thus to plan the impedance of the robot online, without relying on direct force sensor measurements. Applying a task-space impedance control technique to a generic robotic system leads to a resulting behavior described by the model [15], [3]

$$\Lambda_c(\mathbf{q})\ddot{\tilde{\mathbf{x}}} + \mathbf{D}\dot{\tilde{\mathbf{x}}} + \mathbf{K}\tilde{\mathbf{x}} = -\mathbf{F}_{\text{ext}}\,, \tag{1}$$

where $\tilde{\mathbf{x}} \in \mathbb{R}^n$, $\dot{\tilde{\mathbf{x}}} \in \mathbb{R}^n$ and $\ddot{\tilde{\mathbf{x}}} \in \mathbb{R}^n$ are the task-space positioning error vector w.r.t. the desired position vector $\hat{\mathbf{x}} \in \mathbb{R}^n$ and its derivatives, n is the dimension of the task (usually $n = 3$ or $n = 6$), $\mathbf{q} \in \mathbb{R}^{n_q}$ are the generalized coordinates, $\Lambda_c \in \mathbb{R}^{n\times n}$ is the constraint-consistent operational space inertia matrix [3], and $\mathbf{D} \in \mathbb{R}^{n\times n}$ and $\mathbf{K} \in \mathbb{R}^{n\times n}$ are the damping and the stiffness matrices (i.e. the gain matrices of the impedance controller), respectively. Finally, $\mathbf{F}_{\text{ext}} \in \mathbb{R}^n$ is the vector of the external force disturbances. Note that the constraint-consistent operational space inertia matrix Λ_c

is usually configuration dependent. This is a more general case compared to studying a fixed inertia matrix $\mathbf{M}$. Such generality is necessary, because impedance control requires a force sensor to modulate the inertia matrix.

Fig. 3 shows an overview of the control architecture. The impedance controller computes the input to have the desired impedance behavior. This behavior is chosen by the impedance planner, i.e. an optimization problem that we are going to formulate. The cost function is designed to reduce the contact forces resulting from a displacement imposed by the environment, i.e. to reduce the system impedance. The constraints are designed to guarantee the desired performance in terms of tracking error and stability. The performance is set by boundaries that are chosen online by an other module. This module merges the information given by the robot state, the environment and the user inputs. During the robot motion, the configuration, the surrounding environment and the user inputs (e.g. task) will change, leading to different controller gains.

To formulate the optimization problem we leverage the simplified model (1) resulting from the application of an impedance control technique. This solution allows to compute and modify online the impedance parameters. The goal of this paper is to improve the robustness of the robot, thus we will define a cost function $J(\mathbf{D},\mathbf{K})$ that minimizes the impedance. Decreasing the impedance will lower accuracy, thus we will rely on the constraints to guarantee the desired minimum performance. In particular, we can relate tracking performance and stability (in the case of a legged robot) to the position error, bounding it accordingly. We will assume that the disturbances are due to non-zero initial conditions, i.e. $\tilde{\mathbf{x}}_0 \neq \mathbf{0}$ and $\dot{\tilde{\mathbf{x}}}_0 \neq \mathbf{0}$. Furthermore, we will assume that each element $\tilde{x}_{0,i}$ and $\dot{\tilde{x}}_{0,i}$ of the initial condition vectors is bounded. Each element $d_{i,j}$ and $k_{i,j}$ of the matrices $\mathbf{D}$ and $\mathbf{K}$ will also have its own bound due to the actuation system and robot design. Hence, the problem formulation is

$$
\begin{aligned}
&\arg\min_{\mathbf{D},\mathbf{K}} J(\mathbf{D},\mathbf{K}) \\
&\text{s.t.}\ \ l_{\mathrm{d}_{i,j}} \leq d_{i,j} \leq u_{\mathrm{d}_{i,j}} \quad \forall\, i,j = 1\dots n \\
&\qquad l_{\mathrm{k}_{i,j}} \leq k_{i,j} \leq u_{\mathrm{k}_{i,j}} \quad \forall\, i,j = 1\dots n \\
&\qquad \max_{(\tilde{\mathbf{x}}_0,\dot{\tilde{\mathbf{x}}}_0)} |\tilde{x}_i(t)| \leq b_i \quad \forall\, i = 1\dots n,\, \forall\, t \in [0,\infty) \\
&\qquad \text{s.t.}\ \ \Lambda_c \ddot{\tilde{\mathbf{x}}} + \mathbf{D}\dot{\tilde{\mathbf{x}}} + \mathbf{K}\tilde{\mathbf{x}} = \mathbf{0} \\
&\qquad\qquad l_{\tilde{x}_{0,i}} \leq \tilde{x}_{0,i} \leq u_{\tilde{x}_{0,i}} \quad \forall\, i = 1\dots n \\
&\qquad\qquad l_{\dot{\tilde{x}}_{0,i}} \leq \dot{\tilde{x}}_{0,i} \leq u_{\dot{\tilde{x}}_{0,i}} \quad \forall\, i = 1\dots n,
\end{aligned}
\tag{2}
$$

where a boundary b_i is defined (and changed online) for each of the n axes of the robot. These values define the desired performance of the robot and are linked to the external environment, stability (i.e. support polygon) or desired tracking performance (i.e. maximum tracking error).

Eq. (2) is a minimization problem with a maximization constraint. Since the initial conditions are uncertain, this is a robust optimization problem, similar to [12]. In Sec. III we will show how to transform this robust optimization problem in a deterministic optimization problem.

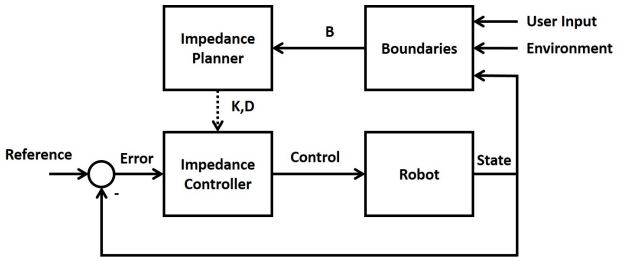


Fig. 3. Scheme of the method. The robot is controlled with an impedance controller, whose gains are chosen online by the proposed impedance planner. Robot configuration, environment and user inputs determine the performance boundaries that need to be guaranteed.

During the execution of the task, the configuration of the robot (thus $\Lambda_c(\mathbf{q})$) will change, together with the bounds b_i, leading to different optimal gain matrices $\mathbf{D}$ and $\mathbf{K}$.

It is worth noting also that in (2) $\mathbf{F}_{\text{ext}} = \mathbf{0}$, i.e. there are no force disturbance, hence we are assuming that all the disturbances are due to non-null initial conditions. The initial velocity error $\dot{\tilde{x}}_{0,i}$ can be interpreted as an external impulsive disturbance $F_{\text{ext},i} = A\delta(t)$ acting along the axis $\tilde{x}_i$, where A is the amplitude of the impulse and $\delta(t)$ is the Dirac Delta.

III. METHOD

In this section we describe the proposed impedance modulation method. Although this method could be applied to any impedance controller, we will rely on the one proposed in [3], in particular, we will focus on the impedance modulation of the center of mass (COM) of the quadrupedal robot ANYmal. Under this circumstance, (1) refers to the task-space dynamics of the COM, thus $\tilde{\mathbf{x}}$ is the position error of the COM w.r.t. the desired position given by the motion planner. Note also that, in this case, $n = 6$.

In the following we will transform the robust optimization problem (2) in a deterministic optimization one. Furthermore, we will report a sufficient condition to guarantee the stability of the system with time varying gains.

A. System Decoupling and Transient Response

The constraint-consistent operational space inertia matrix Λ_c depends on the current configuration $\mathbf{q}$. Modulating also the inertia matrix would require a torque sensor that, generally, is not installed on the robot. Here we assume that Λ_c can be considered constant between two gain updates, i.e. that we can replace Λ_c with a constant matrix $\mathbf{M} \in \mathbb{R}^{n\times n}$. Furthermore, since we are focusing on the impedance of the torso, we have that Λ_c is diagonally dominant, thus, we can approximate $\mathbf{M}$ as a diagonal matrix. Since $\mathbf{D}$ and $\mathbf{K}$ are the gain matrices of the impedance controller, we can set them as diagonal too, decoupling (1) into n SISO sub-systems and (2) into n distinct optimization problems. Note that Λ_c can not be approximated as diagonal in every system. Later, we will briefly discuss how to extend the proposed method to full Λ_c matrices, leaving a deeper investigation to future works.

The resulting SISO model of the system under analysis is

$$m_i\ddot{\tilde{x}}_i + d_i\dot{\tilde{x}}_i + k_i\tilde{x}_i = -F_{\text{ext},i}\,, \tag{3}$$

where $m_i \in \mathbb{R}$ is the i-th diagonal element of the constant inertia matrix $\mathbf{M}$, $d_i \in \mathbb{R}$ is the i-th diagonal element of the damping matrix $\mathbf{D}$, and $k_i \in \mathbb{R}$ is the i-th diagonal element of the stiffness matrix $\mathbf{K}$. $\tilde{x}_i \in \mathbb{R}$, $\dot{\tilde{x}}_i \in \mathbb{R}$ and $\ddot{\tilde{x}}_i \in \mathbb{R}$ are the i-th elements of $\tilde{\mathbf{x}}$, $\dot{\tilde{\mathbf{x}}}$ and $\ddot{\tilde{\mathbf{x}}}$, i.e. the positioning error vector and its derivatives. $F_{\text{ext},i} \in \mathbb{R}$ is i-th element of the external force disturbance vector $\mathbf{F}_{\text{ext}}$, i.e. the disturbance acting on the axis $\tilde{x}_i$. Hereinafter, if it is not strictly necessary, we will omit the subscript i for the sake of readability.

The system (3) will present one of the three well-known behaviors: *under damped*, *critically damped* or *over damped*. Since the optimizer will iteratively change the optimal value for d and k, the system could switch between these three cases. We opt to restrict our analysis to the critically damped case because it is the fastest to converge without oscillations, and, in legged robot stance phases, overshoots are usually avoided [2], [16], [5]. Under this choice, the impedance controller gains are related, reducing the dimensionality of the parameter space of (2). We choose to optimize over the damping term d because it simplifies the resulting expression. Consequently, the stiffness k is

$$k = \frac{d^2}{4m}\,. \tag{4}$$

***Remark** 1:* Note that the proposed method could be applied also to set under-damped or over-damped gains, which are not reported here for the sake of space. Their formulations differ only due to slight variations in the constraints.

B. Cost Function

Our goal is to achieve a behavior robust to environmental interactions. This can be obtained by decreasing the robot impedance, thus the values of the gains d and k. Since we are focusing on the critically damped case, the gains are related by (4). Hence, we can adopt the quadratic cost function

$$J(d) = d^2\,. \tag{5}$$

The idea is to have a robot as soft as possible, while relying on the constraints to guarantee stability and desired minimum performance.

C. Constraints

To guarantee the desired performance and the robot stability, the idea is to compute the worst possible position error and to tune the impedance gains accordingly. The evolution of the position error is defined by (3). Given the boundary conditions $\tilde{x}_0$ and $\dot{\tilde{x}}_0$, the analytical solution of (3) is

$$\tilde{x}(t) = \left(\tilde{x}_0 + \left(\dot{\tilde{x}}_0 + \frac{\tilde{x}_0 d}{2m}\right)t\right)e^{-\frac{dt}{2m}}\,. \tag{6}$$

Substituting (6) into (2) allows to remove the dynamics from the optimization problem. Also considering the simplifications discussed in Sec. III-A and the cost function (5), the optimization SISO sub-problem is

$$\begin{aligned}
&\arg\min_{d}\ d^2\\
&\text{s.t.}\ \ l_{\text{d}} \le d \le u_{\text{d}}\\
&\qquad \max_{(\tilde{x}_0,\dot{\tilde{x}}_0)} \left|\left(\tilde{x}_0 + \left(\dot{\tilde{x}}_0 + \frac{\tilde{x}_0 d}{2m}\right)t\right)e^{-\frac{dt}{2m}}\right| \le b \quad \forall\, t \in [0,\infty)\\
&\qquad \text{s.t.}\ \ l_{\tilde{x}_0} \le \tilde{x}_0 \le u_{\tilde{x}_0}\\
&\qquad\qquad l_{\dot{\tilde{x}}_0} \le \dot{\tilde{x}}_0 \le u_{\dot{\tilde{x}}_0}\,,
\end{aligned} \tag{7}$$

where b is the strictest bound for the axis under analysis.

Our goal is to respect the boundaries b (stability and performance) at every time step in presence of unknown initial conditions, hence we have to focus on the worst case, i.e. the case where the signs of $\tilde{x}_0$ and $\dot{\tilde{x}}_0$ are equal. Thus, without loss of generality, we assume that $\tilde{x}_0 \ge 0$ and $\dot{\tilde{x}}_0 \ge 0$. This allows us to remove the absolute value from the maximization constraint. We also add the time t as a decision variable in the maximization constraint of (7). In other words, instead of considering every position error $\tilde{x}(t)$, we will consider only its peak value over time. Thus, defining

$$\tilde{x}_{0,\max} \triangleq \max(|l_{\tilde{x}_0}|,\, u_{\tilde{x}_0})\,, \quad \dot{\tilde{x}}_{0,\max} \triangleq \max(|l_{\dot{\tilde{x}}_0}|,\, u_{\dot{\tilde{x}}_0})\,, \tag{8}$$

the maximization constraint of (7) can be rewritten as

$$\begin{aligned}
&\max\nolimits_{(\tilde{x}_0,\dot{\tilde{x}}_0,t)} \left(\tilde{x}_0 + \left(\dot{\tilde{x}}_0 + \tfrac{\tilde{x}_0 d}{2m}\right)t\right)e^{-\frac{dt}{2m}}\\
&\text{s.t.}\ \ 0 \le \tilde{x}_0 \le \tilde{x}_{0,\max}\\
&\qquad 0 \le \dot{\tilde{x}}_0 \le \dot{\tilde{x}}_{0,\max}\\
&\qquad t \ge 0\,.
\end{aligned} \tag{9}$$

At this point, we transform the robust optimization problem (7) into a deterministic optimization problem. Using KKT conditions [17] we prove that the value

$$\begin{cases}
\tilde{x}_0 = \tilde{x}_{0,\max}\\
\dot{\tilde{x}}_0 = \dot{\tilde{x}}_{0,\max}\\
t = \dfrac{4m^2\dot{\tilde{x}}_{0,\max}}{d(2m\dot{\tilde{x}}_{0,\max} + d\tilde{x}_{0,\max})}\,,
\end{cases} \tag{10}$$

is the argument of the maximum of (9).

Proof: The KKT conditions applied to (9) are

$$\begin{cases}
\left(1 + \frac{dt}{2m}\right)e^{-\frac{dt}{2m}} + \mu_1 - \mu_2 = 0\\
te^{-\frac{dt}{2m}} + \mu_3 - \mu_4 = 0\\
\left(\left(1 - \frac{dt}{2m}\right)\dot{\tilde{x}}_0 - \frac{d^2 t}{4m^2}\tilde{x}_0\right)e^{-\frac{dt}{2m}} + \mu_5 = 0\\
-\tilde{x}_0 \le 0\\
\tilde{x}_0 - \tilde{x}_{0,\max} \le 0\\
-\dot{\tilde{x}}_0 \le 0\\
\dot{\tilde{x}}_0 - \dot{\tilde{x}}_{0,\max} \le 0\\
-t \le 0\\
\mu_i \ge 0 \quad \forall i = 1\dots 5\\
\mu_1 \cdot (-\tilde{x}_0) = 0\\
\mu_2 \cdot (\tilde{x}_0 - \tilde{x}_{0,\max}) = 0\\
\mu_3 \cdot (-\dot{\tilde{x}}_0) = 0\\
\mu_4 \cdot (\dot{\tilde{x}}_0 - \dot{\tilde{x}}_{0,\max}) = 0\\
\mu_5 \cdot (-t) = 0\,.
\end{cases} \tag{11}$$

Given (11) we have two stationary points, i.e. $(\tilde{x}_0,\, \dot{\tilde{x}}_0, t) = (\tilde{x}_{0,\max},\, 0,\, 0)$ and (10). Recalling that, for the examined problem, $d > 0$ and $m > 0$ and given (6), direct computation shows that (10) is the maximum of the function under analysis. ∎

Using (10) allows us to replace the robust optimization problem with a deterministic one. The maximization constraint in (7) is replaced by

$$\max_{(\tilde{x}_0,\dot{\tilde{x}}_0,t)} (\tilde{x}(t)) = \frac{2m\dot{\tilde{x}}_{0,\max} + d\tilde{x}_{0,\max}}{d} e^{\left(\frac{-2m\dot{\tilde{x}}_{0,\max}}{2m\dot{\tilde{x}}_{0,\max}+d\tilde{x}_{0,\max}}\right)} \leq b\,. \tag{12}$$

It is worth noting that (12) is a nonlinear constraint. Applying a conservative approach is possible to find an upper bound of (12) that is a linear constraint, i.e.

$$\max_{(\tilde{x}_0,\dot{\tilde{x}}_0,t)} (\tilde{x}(t)) \leq \tilde{x}_{0,\max} + \frac{2m\cdot\dot{\tilde{x}}_{0,\max}}{e\cdot d}\,. \tag{13}$$

Proof: Given the linearity of $\tilde{x}_0$ and $\dot{\tilde{x}}_0$ in (6), we have

$$\tilde{x}(t) = \tilde{x}_0\eta_1(d,t) + \dot{\tilde{x}}_0\eta_2(d,t)\,, \tag{14}$$

where

$$\eta_1(d,t) \triangleq \left(1+\frac{d}{2m}t\right)e^{-\frac{dt}{2m}}\,, \quad \eta_2(d,t) \triangleq te^{-\frac{dt}{2m}}\,. \tag{15}$$

Maximizing (14), and recalling that the maximum of the sum of two terms is lesser or equal to the sum of the maximum of the two terms, we obtain

$$\begin{aligned}\max_{(\tilde{x}_0,\dot{\tilde{x}}_0,t)} (\tilde{x}(t)) &= \max_{(\tilde{x}_0,\dot{\tilde{x}}_0,t)} (\tilde{x}_0\cdot\eta_1(d,t) + \dot{\tilde{x}}_0\cdot\eta_2(d,t)) \leq \\ &\leq \max_{(\tilde{x}_0,t)}(\tilde{x}_0\cdot\eta_1(d,t)) + \max_{(\dot{\tilde{x}}_0,t)}(\dot{\tilde{x}}_0\cdot\eta_2(d,t))\,.\end{aligned} \tag{16}$$

Assuming $\tilde{x}_0$, $\dot{\tilde{x}}_0$, η_1 and η_2 as positive we have

$$\begin{aligned}\max_{(\tilde{x}_0,t)}(\tilde{x}_0\cdot\eta_1(d,t)) &= \max_{(\tilde{x}_0)}(\tilde{x}_0)\cdot\max_{(t)}(\eta_1(d,t)) \\ \max_{(\dot{\tilde{x}}_0,t)}(\dot{\tilde{x}}_0\cdot\eta_2(d,t)) &= \max_{(\dot{\tilde{x}}_0)}(\dot{\tilde{x}}_0)\cdot\max_{(t)}(\eta_2(d,t))\,.\end{aligned} \tag{17}$$

From (8) we have that $\max_{(\tilde{x}_0)}(\tilde{x}_0) = \tilde{x}_{0,\max}$ and $\max_{(\dot{\tilde{x}}_0,t)}(\dot{\tilde{x}}_0) = \dot{\tilde{x}}_{0,\max}$. Then, we can analytically compute the maximum of η_1 and η_2 differentiating over t and equating the derivative to 0, obtaining

$$\max_{(t)}(\eta_1(d,t)) = 1\,, \quad \max_{(t)}(\eta_2(d,t)) = \frac{2m}{e\cdot d}\,. \tag{18}$$

This leads to the thesis. ∎

D. Closed Form Solution

Eq. (13) can be substituted for (12) in (7), leading to the deterministic optimization problem

$$\begin{aligned}&\arg\min_{d}\ d^2 \\ &\text{s.t.}\ \ d \geq \frac{2m\cdot\dot{\tilde{x}}_{0,\max}}{(b-\tilde{x}_{0,\max})\cdot e} \\ &\qquad l_{\mathrm{d}} \leq d \leq u_{\mathrm{d}}\,,\end{aligned} \tag{19}$$

that has a closed form solution

$$d = \min\left(\max\left(l_{\mathrm{d}},\quad \frac{2m\cdot\dot{\tilde{x}}_{0,\max}}{(b-\tilde{x}_{0,\max})\cdot e}\right),\quad u_{\mathrm{d}}\right)\,. \tag{20}$$

Eq. (20) and (4) allow the robot programmer to directly map the task requirements in the impedance parameters of the robot. l_{d} is fixed by the actuators, m is given by the configuration of the robot, and $\tilde{x}_{0,\max}$ and $\dot{\tilde{x}}_{0,\max}$ are linked to the expected disturbances, in particular, $\dot{\tilde{x}}_{0,\max}$ is linked to the expected impulsive force disturbance.

Finally, b is related to the stability margin of the robot and the performance/task requirements. For example, b could be set as the distance between the edges of the support polygon and the planned COM position or as the distance between the hip and the foot of the robot to avoid limb singularities. The user can also manually set the desired tracking performance by imposing a b value. This is useful, for example, if the task changes from walking to manipulation. The boundary b for each axes $\tilde{x}$ will be set accordingly as the strictest of the set. During the execution of the task, m and b will change, leading to different gains values.

***Remark** 2:* The structure of (20) ensures the existence of a valid solution. However, dynamic model inaccuracies, hardware limitations and overly stringent task requirements might make it impossible to satisfy the desired performance in practice.

E. Stability Analysis

To prove that the gain variation does not lead to instability, we will rely on Theorem 1 presented in [18]. This theorem states that a sufficient condition for the asymptotic stability of the equilibrium at the origin of the system

$$\ddot{x}(t) + \frac{d(t)}{m(t)}\dot{x}(t) + \frac{k(t)}{m(t)}x(t) = 0\,, \tag{21}$$

is that $d(t)/m(t)$ is continuous and bounded over time, $k(t)/m(t)$ is positive, differentiable and bounded, and that

$$\frac{1}{2}\frac{\mathrm{d}(k(t)/m(t))}{\mathrm{d}t}\left(\frac{k(t)}{m(t)}\right)^{-1} + \frac{d(t)}{m(t)} > 0\,. \tag{22}$$

Let us consider system (3). In this case all the conditions of Theorem 1 in [18] are satisfied except for (22). Given (4), condition (22) becomes

$$\dot{d}(t) > -\frac{d^2(t)}{m(t)} + \frac{d(t)\dot{m}(t)}{m(t)}\,. \tag{23}$$

Eq. (23) imposes a maximum changing rate for the damping coefficient. Fulfilling condition (23) assures the stability of the equilibrium at the origin (i.e. zero tracking error) during the gain variation.

Eq. (23) is always fulfilled during a damping increment, but it may not be during a damping reduction. To fulfill this condition we approximate $\dot{d}$ as a difference quotient, thus (23) becomes

$$d(t+T_s) > d(t) - \frac{d^2(t)T_{\mathrm{s}}}{m(t)} + \frac{d(t)\dot{m}(t)T_{\mathrm{s}}}{m(t)}\,, \tag{24}$$

where T_{s} is the sampling time. The term $d(t+T_{\mathrm{s}})$ is the solution (20) of the optimization problem. If (24) is satisfied, we can set the next damping value as (20), otherwise, we will limit it to the maximum variation imposed by (24).

Note that we can not compute exactly $\dot{m}(t)$ online. We could approximate it using the planned motion, but since the term $d(t)\dot{m}(t)T_s/m(t)$ is very small (the inertia matrix varies slowly in the system under analysis), we can substitute it with a constant value tuned on the worst case scenario.

IV. SIMULATION RESULTS

In this section we test this method with two simulations. In each simulation we used $T_s = 0.0025$s $\tilde{x}_{0,\max} = [0.0218\text{m}, 0.0160\text{m}, 0.0057\text{m}, 0.0192\text{rad}, 0.0097\text{rad}, 0.0301\text{rad}]^T$ and $\dot{\tilde{x}}_{0,\max} = [0.1729\text{m/s}, 0.1240\text{m/s}, 0.0635\text{m/s}, 0.4331\text{rad/s}, 0.2396\text{rad/s}, 0.1633\text{rad/s}]^T$. These values are the maximum position and velocity error occurred while walking on an uneven terrain with the simulated robot using the fixed impedance gains employed in [3]. The impedance limits are set as $\mathbf{l}_k = [300\cdot\mathbf{1}_3^T\text{N/m}, 200\cdot\mathbf{1}_3^T\text{Nm/rad}]^T$, $\mathbf{u}_k = [1800\cdot\mathbf{1}_3^T\text{N/m}, 400\cdot\mathbf{1}_3^T\text{Nm/rad}]^T$, $\mathbf{l}_d = [230\cdot\mathbf{1}_3^T\text{Ns/m}, 19\text{Nms/rad}, 36\text{Nms/rad}, 38\text{Nms/rad}]^T$ and $\mathbf{u}_d = [450\cdot\mathbf{1}_3^T\text{Ns/m}, 34\text{Nms/rad}, 85\text{Nms/rad}, 69\text{Nms/rad}]^T$, where $\mathbf{1}_3^T \triangleq [1, 1, 1]$.

In the first simulation we test the variation of the impedance gains due to a change in configuration, thus a change in Λ_c. We simulate the robot approaching a table and crawling under it. For this simulation the boundaries imposed by the user as task requirements are $\mathbf{b} = [0.04\cdot\mathbf{1}_3^T\text{m}, \pi/6\cdot\mathbf{1}_3^T\text{rad}]^T$. The results are reported in Fig. 4. Different background colors define the different task phases: walking, transition and crawling. For the sake of space and readability we report only the results for the translational axes, analogous result were obtained for the rotational ones. Fig 4(a) shows the change in the inertia matrix diagonal elements due to the change of configuration. This is considered by the optimization problem that increases the damping in the x axis (Fig. 4(b)). Thus, the tracking error continues to fulfill the margin $\mathbf{b}$ imposed by the user (Fig. 4(c)). Note that also the translational stiffness changes as described by (4), but we do not report its behavior here for the sake of space.

In the second simulation we test the variation of the impedance gains due to a change in the task requirements. We simulate that the user increases the required tracking performance from $\mathbf{b} = [0.06\text{m}, 0.055\text{m}, 0.05\text{m}, \pi/6\cdot\mathbf{1}_3^T\text{rad}]^T$ to $\mathbf{b} = [0.025\text{m}, 0.04\text{m}, 0.04\text{m}, 0.01\cdot\mathbf{1}_3^T\text{rad}]^T$ and then decreases them again. Fig. 5(a) shows that the imposed tracking performance continues to be satisfied, while in Fig 5(b) and 5(c) the variation of the impedance gains is reported. This shows how the user can easily change task without any tedious tuning of the gains. An increment in the gains does not cause instability as discussed in Sec. III-E. This means that the abrupt change in the gains does not compromise the stability, while the change at $t = 37$s must be slowed down to preserve it. The inserts in Fig. 5(b) highlight this difference. Note that there is an analogous behavior for the rotational impedance gains.

V. EXPERIMENTAL RESULTS

In this section we test the proposed method on the robot ANYmal [6]. This is a quadrupedal platform with point feet. This system weighs ~ 30kg, is ~ 0.5m tall, and each link of its legs is ~ 0.3m long. The robot presents a total of 12 identical actuators (ANYdrive [6]), three per legs. These joint units are compact series elastic actuators, i.e. actuators where a mechanically compliant element is embedded between the gearbox output and the joint output shaft.

We used $T_s = 0.0025$s $\tilde{x}_{0,\max} = [0.0337\text{m}, 0.0360\text{m}, 0.0193\text{m}, 0.0231\text{rad}, 0.0279\text{rad}, 0.0211\text{rad}]^T$ and $\dot{\tilde{x}}_{0,\max} = [0.2157\text{m/s}, 0.1812\text{m/s}, 0.1264\text{m/s}, 0.2712\text{rad/s}, 0.2447\text{rad/s}, 0.1409\text{rad/s}]^T$. These values are the maximum position and velocity error occurred while walking on an uneven terrain with the real robot using the fixed gains employed in [3]. The impedance limits are set as $\mathbf{l}_k = [300\cdot\mathbf{1}_3^T\text{N/m}, 200\cdot\mathbf{1}_3^T\text{Nm/rad}]^T$, $\mathbf{u}_k = [1800\cdot\mathbf{1}_3^T\text{N/m}, 400\cdot\mathbf{1}_3^T\text{Nm/rad}]^T$, $\mathbf{l}_d = [230\cdot\mathbf{1}_3^T\text{Ns/m}, 19\text{Nms/rad}, 36\text{Nms/rad}, 38\text{Nms/rad}]^T$ and $\mathbf{u}_d = [450\cdot\mathbf{1}_3^T\text{Ns/m}, 34\text{Nms/rad}, 85\text{Nms/rad}, 69\text{Nms/rad}]^T$. Please, for what concern the experiments, refer also to the video attached to this paper.

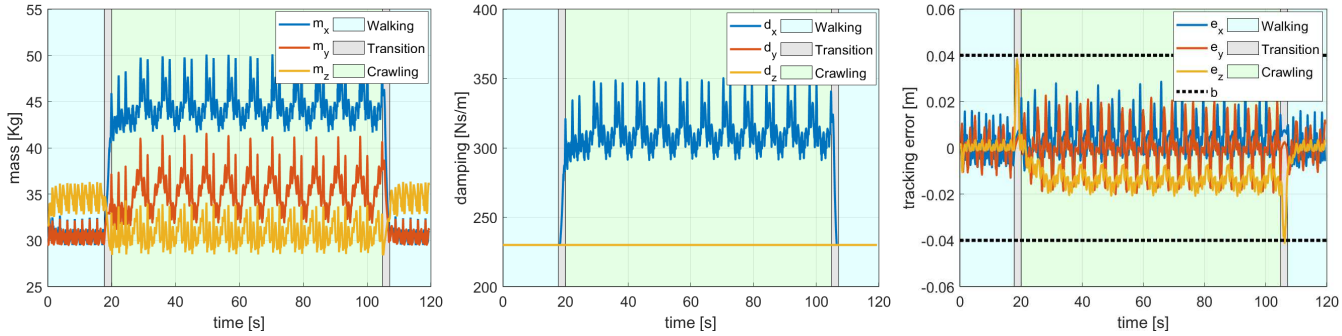
A. Identification Procedure

Since the system model used in the impedance controller is not accurate enough, there is a mismatch between the simulated dynamics and the real one. In particular, we have that the damping behavior of the system along the translational axes is different from the modeled one. In order to reduce this issue, we used a least square approach to identify the intrinsic damping and the mass of the system along the translational axes. We set the damping gain of the impedance controller to zero, and we applied several displacements to the robot using different stiffness gains. In particular, we applied the displacements $\tilde{x}_0 = \{10, 20, 35, 50\}$mm for the x axis, $\tilde{x}_0 = \{10, 20, 35, 50\}$mm for the y axis and $\tilde{x}_0 = \{50, 100, 150, 170\}$mm for the z axis. Each displacement was applied three times for each tested stiffness value. In total we tested three stiffness values: $k = 800$N/m, $k = 1000$N/m and $k = 1200$N/m.The result of this identification process is that $\tilde{d}_x = 50.4885$Ns/m, $\tilde{d}_y = 64.4859$Ns/m and $\tilde{d}_z = 75.6533$Ns/m, while $\tilde{m}_x$, $\tilde{m}_y$ and $\tilde{m}_z$ are approximately equal to the diagonal elements of the matrix Λ_c in the default configuration. The identified damping values were then imposed as an offset to the optimal values (20).

B. Obstacle Interaction Experiment

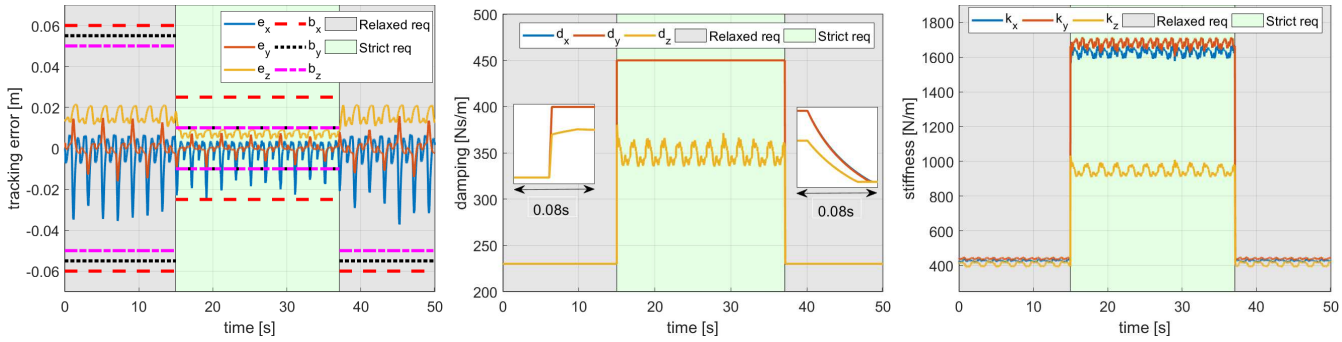
In this experiment the robot touches an unperceived obstacle (placed at ~ 0.6m from the ground) while walking. We repeated the experiment with low[1] ($\mathbf{K} = \text{diag}([500\cdot\mathbf{1}_3^T\text{N/m}, 200\cdot\mathbf{1}_3^T\text{Nm/rad}])$, $\mathbf{D} = \text{diag}([90\cdot\mathbf{1}_3^T\text{Ns/m}, 10\cdot\mathbf{1}_3^T\text{Nms/rad}])$), high ($\mathbf{K} = \text{diag}([1500\cdot\mathbf{1}_3^T\text{N/m}, 300\cdot\mathbf{1}_3^T\text{Nm/rad}])$, $\mathbf{D} = \text{diag}([200\cdot\mathbf{1}_3^T\text{Ns/m}, 80\cdot\mathbf{1}_3^T\text{Nms/rad}])$) and variable impedance ($\mathbf{b} = [0.06\text{m}, 0.055\text{m}, 0.05\text{m}, \pi/6\cdot\mathbf{1}_3^T\text{rad}]$). Fig. 6 shows a photo-sequence for the high and variable impedance cases. The low impedance case is omitted due to similarity to the variable gains scenario. In the case of high impedance the interaction force is too large causing a failure on the robot. Conversely, in the case of

[1]Note that these values are the same employed in [3].



(a) Diagonal term of the inertia matrix. (b) Translational damping gains. (c) Translational tracking error.

Fig. 4. Simulation results of the crawling task. (a) The change of configuration causes a change in the inertia matrix of the robot. This is considered by the optimization problem that increases the damping gain for the x axis (b). (c) The tracking error does not exceed the performance margins imposed by the user. Analogous results were obtained for the stiffness gains, not reported here for the sake of space.



(a) Translational tracking error. (b) Translational damping gains. (c) Translational stiffness gains.

Fig. 5. Simulation of a change in the task requirements. The user imposes a new desired maximum tracking error (a) and the method automatically set the new impedance gains (b,c). The inserts in (b) highlight the difference between increasing and decreasing the gains. The latter may cause instability so the variation must be slowed down as discussed in Sec. III-E. The fluctuation of the stiffness gains in (c) are caused by the variation of the inertia matrix, via (4). Analogous results were obtained for the rotational impedance gains, not reported here for the sake of space.

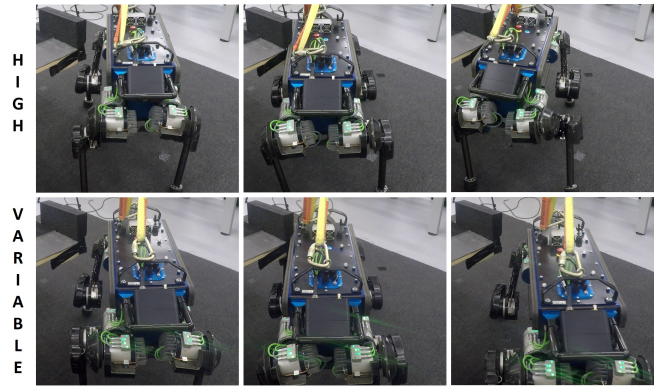
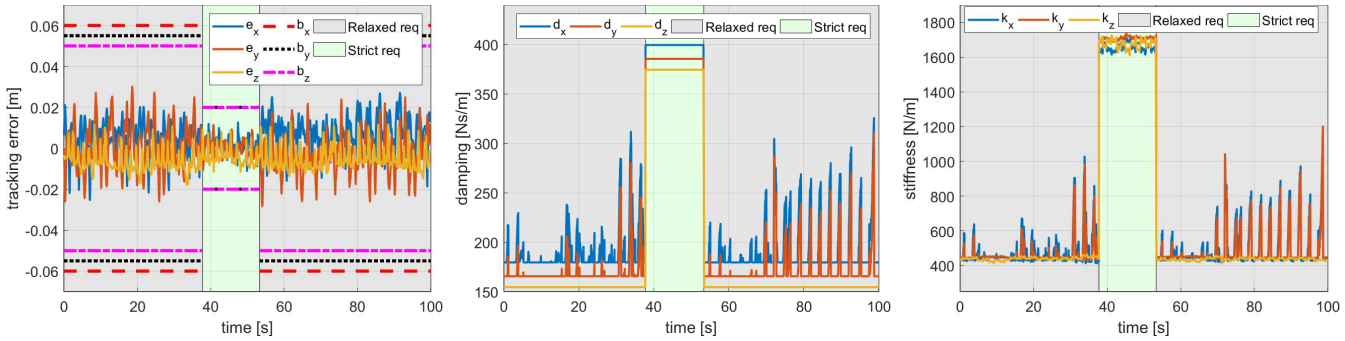


Fig. 6. Photo-sequence of the robot touching an unperceived obstacle. In the case of high impedance the robot can not cope with the motion displacement caused by the environment. In the case of variable impedance, the robot is able to handle that disturbance.

variable impedance the robot is able to resist to the motion displacement caused by the environment. This shows the benefit of minimizing the impedance, which is achieved thanks to the choice of the cost function (5).

C. Rough Terrain Experiment

In this experiment the robot walks on rough terrain (Fig. 1) using the controller proposed in [3] combined with locomotion planner described in [19]. The terrain is realized randomly combining 0.4×0.4m wooden tiles with different slope steepness ($\{\pi/36, \pi/18, \pi/12\}$rad) and orientation. During the motion, the task requirements are changed from $\mathbf{b} = [0.06\text{m}, 0.055\text{m}, 0.05\text{m}, \pi/6 \cdot \mathbf{1}_3^T \text{rad}]^T$ to $\mathbf{b} = [0.02 \cdot \mathbf{1}_3^T \text{m}, \pi/6 \cdot \mathbf{1}_3^T \text{rad}]^T$ and then reseted to the previous values. Fig. 7 reports the tracking error and the impedance variation during the experiment. The error stays in between the boundaries even when these boundaries are tightened by the operator between $t = 38$s and $t = 53$s. The roughness of the terrain and the changes in the support polygon cause variation on the impedance gains throughout the motion. This allows to keep balance and to achieve the desired tracking performance. Note that the damping coefficient reach smaller values than $\mathbf{l}_\text{d}$ due to the offset imposed by the intrinsic property of the system identified in Sec. V-A. Despite the different conditions between simulation and this experiment the results



(a) Translational tracking error. (b) Translational damping coefficients. (c) Translational stiffness coefficients.

Fig. 7. Walking on a rough terrain experiment. (a) The user imposes a new desired maximum tracking error at $t = 38$s and resets it at $t = 53$s. The tracking error is in between the boundaries. (b,c) The method automatically set the new impedance gains. The roughness of the terrain and the changes in the support polygon cause variation on the impedance gains throughout the motion.

are qualitatively similar. Therefore, this experiment supports the soundness of the proposed method.

VI. CONCLUSIONS

In this work we presented a method to plan the impedance of legged robots online to improve their robustness to unexpected interactions with the environment. This method optimizes the impedance based on the desired performance and expected disturbances, without relying on direct force measurements. Theoretical analysis provided a closed form solution to the robust optimization problem and guarantees of the system stability during the gain variation.

The proposed method was tested on simulation and on hardware in challenging scenarios involving rough terrain and unperceived obstacle interaction. Results show that this technique can vary the controller gains online, adapting them to different robot configurations or task requirements. Indeed, the user can change the desired tracking performance with no need of tedious and time-consuming tuning procedure.

Future works will focus on extending this method to system with full inertia matrices such as manipulators. In the case this matrix is not diagonal, the decoupled approach used in this paper can not be applied. However, the discussed optimization problem could be tackled evaluating the maximum error via mixed integer optimization.

REFERENCES

[1] Neville Hogan. Impedance control: An approach to manipulation: Part iiimplementation. *Journal of dynamic systems, measurement, and control*, 107(1):8–16, 1985.

[2] Jong Hyeon Park. Impedance control for biped robot locomotion. *IEEE Transactions on Robotics and Automation*, 17(6):870–882, 2001.

[3] Guiyang Xin, Hsiu-Chin Lin, Joshua Smith, Oguzhan Cebe, and Michael Mistry. A model-based hierarchical controller for legged systems subject to external disturbances. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4375–4382. IEEE, 2018.

[4] Claudio Semini, Victor Barasuol, Thiago Boaventura, Marco Frigerio, Michele Focchi, Darwin G Caldwell, and Jonas Buchli. Towards versatile legged robots through active impedance control. *The International Journal of Robotics Research*, 34(7):1003–1020, 2015.

[5] Emmanouil Spyrakos-Papastavridis, Peter RN Childs, and Nikos G Tsagarakis. Variable impedance walking using time-varying lyapunov stability margins. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 318–323. IEEE, 2017.

[6] Marco Hutter, Christian Gehring, Andreas Lauber, Fabian Gunther, Carmine Dario Bellicoso, Vassilios Tsounis, Péter Fankhauser, Remo Diethelm, Samuel Bachmann, Michael Blösch, et al. Anymal-toward legged robots for harsh environments. *Advanced Robotics*, 31(17):918–931, 2017.

[7] Haifa Mehdi and Olfa Boubaker. Impedance controller tuned by particle swarm optimization for robotic arms. *International Journal of Advanced Robotic Systems*, 8(5):57, 2011.

[8] Jonas Buchli, Freek Stulp, Evangelos Theodorou, and Stefan Schaal. Learning variable impedance control. *The International Journal of Robotics Research*, 30(7):820–833, 2011.

[9] Miao Li, Hang Yin, Kenji Tahara, and Aude Billard. Learning object-level impedance control for robust grasping and dexterous manipulation. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6784–6791. IEEE, 2014.

[10] Tasuku Yamawaki, Hiroki Ishikawa, and Masahito Yashima. Iterative learning of variable impedance control for human-robot cooperation. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 839–844. IEEE, 2016.

[11] Chenguang Yang, Gowrishankar Ganesh, Sami Haddadin, Sven Parusel, Alin Albu-Schaeffer, and Etienne Burdet. Human-like adaptation of force and impedance in stable and unstable interactions. *IEEE transactions on robotics*, 27(5):918–930, 2011.

[12] G Maria Gasparri, Filippo Fabiani, Manolo Garabini, Lucia Pallottino, M Catalano, Giorgio Grioli, R Persichin, and Antonio Bicchi. Robust optimization of system compliance for physical interaction in uncertain scenarios. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 911–918. IEEE, 2016.

[13] Shuzhi Sam Ge, Yanan Li, and Chen Wang. Impedance adaptation for optimal robot–environment interaction. *International Journal of Control*, 87(2):249–263, 2014.

[14] Pietro Balatti, Dimitrios Kanoulas, Giuseppe F Rigano, Luca Muratore, Nikos G Tsagarakis, and Arash Ajoudani. A self-tuning impedance controller for autonomous robotic manipulation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5885–5891. IEEE, 2018.

[15] Luigi Villani and Joris De Schutter. Force control. In *Springer Handbook of Robotics*, pages 195–220. Springer, 2016.

[16] Tomomichi Sugihara and Yoshihiko Nakamura. Contact phase invariant control for humanoid robot based on variable impedant inverted pendulum model. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 1, pages 51–56. IEEE, 2003.

[17] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[18] Alexander O Ignatyev. Stability of a linear oscillator with variable parameters. *Electronic Journal of Differential Equations*, 1997(17):1–6, 1997.

[19] Péter Fankhauser, Marko Bjelonic, C Dario Bellicoso, Takahiro Miki, and Marco Hutter. Robust rough-terrain locomotion with a quadrupedal robot. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 2018.