



HAL
open science

Passive Inverse Dynamics Control using a Global Energy Tank for Torque-Controlled Humanoid Robots in Multi-Contact

Noelie Ramuzat, Sebastien Boria, Olivier Stasse

► To cite this version:

Noelie Ramuzat, Sebastien Boria, Olivier Stasse. Passive Inverse Dynamics Control using a Global Energy Tank for Torque-Controlled Humanoid Robots in Multi-Contact. *IEEE Robotics and Automation Letters*, 2022, 7 (2), pp.2787 - 2794. 10.1109/LRA.2022.3144767 . hal-03547136v1

HAL Id: hal-03547136

<https://hal.science/hal-03547136v1>

Submitted on 28 Jan 2022 (v1), last revised 24 Mar 2022 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Passive Inverse Dynamics Control using a Global Energy Tank for Torque-Controlled Humanoid Robots in Multi-Contact

N. Ramuzat^{1,2}, S. Boria¹, O. Stasse²

Abstract—This work presents a passivity-based inverse dynamics (ID) controller using a global energy tank. The proposed control approach allows us to achieve a safe multi-contact scenario on a torque controlled humanoid robot. The controller is primarily a task space ID quadratic programming (QP) which efficiently computes the reference torque satisfying a non-hierarchical set of tasks. Our work extends this controller by adding a global energy tank modulating the task gains, with power regulation, to ensure the passivity of the system. This method combines the benefits of the ID controller, which computes an optimal reference without joint torque feedback, and of the passivity-based system, which is robust to model uncertainties and external disturbances. The robustness of our framework is demonstrated in Gazebo simulations, where the robot TALOS achieves a multi-contact scenario and a 20cm step walk, with objectives in the Cartesian and configuration spaces, in torque control. The implementation of this controller is open-source.

Index Terms—Multi-Contact Whole-Body Motion Planning and Control; Humanoid Robot Systems; Humanoid and Bipedal Locomotion

I. INTRODUCTION

HUMANOID robots are complex systems designed to operate in similar environment than humans. One of their purposes is to achieve dangerous or too demanding tasks for a person, relying on their locomotion and manipulation skills. Yet, to achieve such tasks, these robots must interact with their environments, subject to unknown disturbances. Even performing simple tasks in known environment can lead to instabilities. The authors faced this issue while testing a classical Inverse Dynamics (ID) torque controller on TALOS on a posture task. After some motions, the system diverged and blocked some harmonic drive. The gains were tuned on a simulator modeling the actuators but the solution still became unstable. Thus, to have safe and reliable interactions with the environment (humans), ensuring the system stability is needed.

State of the art: This problem is tackled in the community using two strategies: the Lyapunov analysis or the passivity-based analysis. The former consists in analysing the stability of an equilibrium, looking if the solution remains in a bounded distance from the equilibrium or converges asymptotically or exponentially. This analysis ensures that the controller will not find a diverging solution. In [1], a momentum-based Quadratic

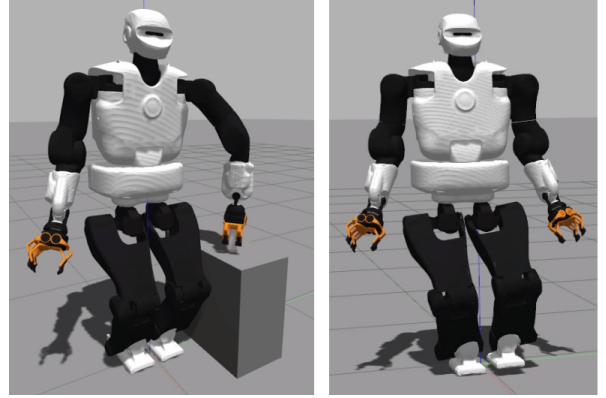


Fig. 1: Simulations with Passivity: Left: Contact-Force Task - Right: Walk of 20cm steps.

Programming (QP) with two strict hierarchies is proven to be Lyapunov stable by using integral gains in the task functions. Similarly, [2] adds an integral term to the computed torque while using it in the constraints of their QP.

The passivity-based analysis investigates the energy flows within a system [3]. Controlling the system energy without proving the passivity, [4] proposes a formulation of a kinetic energy constraint in a QP to prevent a manipulator robot from transferring dangerous amount of this energy during an impact. Yet, [5] proved that when the robot interacts with its environment, a passively-controlled robot is a necessary condition for stability [6]. The passivity ensures that the system is dissipative [7], proving that the system internal power is less than or equal to the power transferred to the system through its port. Using the system internal power, the passivity can be analyzed to ensure the convergence to a desired equilibrium or to detect which components have unbounded energy consumption. The energy generated by these components can be limited by adding an energy tank [6] in the controller.

The energy tanks are artificial energy storing elements that keep track of the energy naturally dissipated by the system agents [8]. This method was presented on manipulators for multi-tasks in a strict-hierarchy using null-space projection in [9]. Regulating the size and flow of the energy tank are challenges that have been expressed in the domain [10, 11]. Providing too much energy to the system may lead the controller to diverge from the equilibrium or to over-accumulate this energy. Likewise, an uncontrolled sharp release of the tank energy might lead to dangerous motions [12]. For manipulator robots with flexible joints, [13, 14] present a passive framework using impedance control. Another method, successfully implemented in [15], couples ID and passivity-based control

Manuscript received: September 9, 2021; Revised: December 3, 2021; Accepted January 1, 2022.

This paper was recommended for publication by Editor Clement Gosselin upon evaluation of the Associate Editor and Reviewers' comments.

This work is supported by the cooperation agreement ROB4FAM.

¹Airbus, Toulouse, France, noelie.ramuzat@airbus.com, sebastien.boria@airbus.com

²LAAS-CNRS, Université de Toulouse, CNRS, INSA, UPS, Toulouse, France, noelie.ramuzat@laas.fr, olivier.stasse@laas.fr
Digital Object Identifier (DOI): 10.1109/LRA.2022.3144767

by modifying the ID formulation to exploit the passivity-properties of the robot model.

Yet, using these methods on humanoid robots is not simple because humanoids robots are not ground-fixed to the environment, but under-actuated systems. This under-actuated part relies on external contact forces to control its motion. Besides, the penalization of the tracking tasks used in [11, 12] can compromise the balance of the robot. Thus, passivity-based control on humanoid robots is still an open-problem. In [16], the method of [11] was extended for a fully actuated humanoid robot for limit cycle control. Likewise, we extend the global energy tank approach of [11] to a non-strict hierarchy for under-actuated robots. The passivity method using impedance control was first implemented on humanoid robots by [17] and was lately extended on the DLR robot TORO, leading to impressive results in multi-contact scenarios in [18]. Finally, [19] designed a new controller respecting the Lyapunov stability and the system passivity even for conflicting sub-tasks, for fully actuated robots. Yet for under-actuated robot, if a constraint becomes active, the passivity is no longer guaranteed. Still, it achieves great results with the humanoid robot TORO. To the best of the authors knowledge, no proof of passivity on an ID controller using energy tank on an under-actuated robot has been published or implemented yet.

Contributions: Our contribution is two-fold: the task gains of our whole-body controller are non-constant and regulated by the global energy tank, and our scheme implements a constraint to respect the passivity even when other constraints become active in the QP. The classical ID formulation is transformed in a non-linear problem which first computes the energy tank and regulating coefficients using the method of [11]. Then, it solves the strictly constrained QP with these coefficients. It is a sort of alternating minimization problem. We propose the implementation of this framework in an open-source package [20] and validate it in simulations through a multi-contact scenario and a 20cm steps walk¹.

We organize the article as follows: Section II defines the formulation of the robot model and its dynamics. In Section III, the task-space QP TSID is detailed to explain where the energy tank will act. Then, Section IV recalls the passivity theory and Section V describes the mathematical process to add the energy tank and obtain the passivity proof in TSID. Finally, Section VI presents the simulations validating the controller's robustness and Section VII discusses them.

II. ROBOT MODEL

Our experimental platform is a robot TALOS from the PAL Robotics company, it is 1.75m tall weighting 100kg and is equipped with electric actuators. It has $n_j = 32$ joints and a floating base of $n_b = 6$ DOF, we denote $n = n_j + n_b$. We express the coordinates of the robot as:

$$p = [x_b, R_b, q]^T \quad (1)$$

with the under-actuated state of the floating base $[x_b, R_b]^T \in \mathbb{R}^6$ (x_b the position and R_b the orientation of the base frame

B relative to the world one W) and the joint configurations $q \in \mathbb{R}^{n_j}$. The equation of the robot dynamics can be written as:

$$M(p)a + \underbrace{C(p, v)v + g(p)}_h = \underbrace{[0, 0, \tau]^T}_{N^T \tau} + \underbrace{\tau_{ext}}_{J_c^T F} \quad (2)$$

$M \in \mathbb{R}^{n \times n}$ the symmetric and positive definite inertia matrix, $C \in \mathbb{R}^{n \times n}$ the Coriolis matrix and $g \in \mathbb{R}^n$ the gravity vector. $q \in \mathbb{R}^{n_j}$ is the joint configuration of the robot. $a, v, p \in \mathbb{R}^n$ are the accelerations, velocities and positions of the joint configuration of the robot including the base (free-flyer). The free-flyer information are estimated with a base-estimator from the configuration, IMU and force sensors of the robot. $\tau \in \mathbb{R}^{n_j}$ are the joint torques of the actuators and $\tau_{ext} \in \mathbb{R}^n$ are the external torques. N is a selector matrix associated to the actuated joints $N = [0_{n_b}, 1_{n_j}]$, such that $N^T \in \mathbb{R}^{n \times n_j}$.

Using the contact forces F we can write that $\tau_{ext} = J_c^T F$ with J_c the Jacobian of the contact points.

III. TASK SPACE INVERSE DYNAMICS (TSID)

TSID [20, 21] is a Weighted Quadratic Programming (WQP) which sums selected task functions in a general cost function using weights to define their priorities. It then optimizes this sum subject to several constraints.

A. Contact Constraints:

TSID solves the inverse dynamics of the robot in rigid contact with the environment [22], thus, these contacts constrain the motion. They are implemented as nonlinear functions, which are differentiated twice (with J_c the Jacobian at the contact point): $J_c^T a + \dot{J}_c v = 0$, such that the contact point acceleration is zero.

The equation of the dynamics is formulated as an equality constraint with τ_{ext} replaced by the forces applied at each contact, and these forces are optimized (see Eq.8). In this paper, we use the unilateral rectangular plane contact formulation of TSID. The contact is defined as a 6D motion task (Eq.4), an inequality constraint on the force bounds and a direct force task $T^T f = f^*$. With $T^T \in \mathbb{R}^{6 \times 12}$ the force-generator matrix mapping the forces at the vertices of the contact surface, $f \in \mathbb{R}^{12}$ (four 3D forces), to a 6D resultant force at the contact point (center of the surface). $f^* \in \mathbb{R}^6$ is the desired force for the contact. T creates the map between the two representations: $\tau_{contact} = J^T T^T f$. However in our formulation we do not use this direct force task, letting the QP optimize freely the forces at the contact, except for the Cartesian Force-Contact case.

B. Task Functions

Motion Task: Most of the task-function errors are expressed as motion task errors e . They are implemented as acceleration-based tracking laws in the Lie group $SE(3)$. With p and the acceleration a as control input, a task-function is a second-order derivable function $x(p)$ whose space is named the task-space. The motion task errors $e \in \mathbb{R}^l$ are expressed as [23]:

$$\ddot{e} = \ddot{x} - \ddot{x}^*, \quad \dot{e} = (Ja + \dot{J}v) - \dot{x}^* \quad (3)$$

¹A detailed video is available at the following link: peertube.laas.fr/videos/watch/d5534f56-1efb-41e3-9c19-10f1911e0bc2

with x^* , \ddot{x}^* the desired position and acceleration of the task, l is the dimension of the task error (for instance, for the 3D acceleration error on the CoM $l = 3$), $J = [J_u \ J_a] = [\mathcal{T}(SE(3), B) \frac{\partial e}{\partial q}] = [\mathcal{T}(SE(3), B) \frac{\partial x}{\partial q}]$ the Jacobian according to the robot coordinate vector (\mathcal{T} the transform between the frame of the task and the base frame). A PD+ dynamics is imposed on these errors to obtain the generic formulation for a motion task-function (K_P, K_D the diagonal matrices of the proportional and derivative gains) [23]:

$$\underbrace{J}_O \underbrace{a}_y = \underbrace{\ddot{x}^* - \dot{J}v + K_P e + K_D \dot{e}}_o \quad (4)$$

Angular Momentum Task: Based on the work of [24], the Angular Momentum (AM) (k_c) dynamics is expressed in TSID by the following equation, implemented in [23, 25]:

$$\dot{k}_c = \dot{k}_c^* + K_P(k_c^* - k_c) \quad (5)$$

Cartesian Force-Contact Task: This task is composed of two parts: the force application and the position control of the contact. Thus, it combines two tasks, a motion task defined in Eq.4 for the control of the position, and a force task for the application of the force.

The force task is defined as a Proportional Integral with a feed-forward (PI+) and anti-windup, inspired by [12]:

$$\begin{aligned} e_f &= F^* - F_{ext} \\ \underbrace{T_F^T}_O \underbrace{F_F}_y &= \underbrace{F^* + P_F e_f + I_F \int_0^t e_f ds}_o \end{aligned} \quad (6)$$

with $F^*, F_{ext} \in \mathbb{R}^6$, respectively the desired and measured or estimated external wrenches and $P_F, I_F \in \mathbb{R}^{6 \times 6}$ the task gains. T_F is the force-generator matrix defined as in the previous paragraph and F_F is the constrained wrench.

In TSID, as explained before, the forces in the equation of the dynamics are expressed at the contacts. Thus, the force task needs to be linked to the appropriate contact. In the TSID library the task is called "task-contact-force" and the motion task associated is the one of the contact task.

Considering that our system has c contacts and the Cartesian Force-Contact task, we can replace τ_{ext} by:

$$\tau_{ext} = \sum_{k=1}^c J_k^T T_k^T f_k - J_F^T T_F^T F_F \quad (7)$$

which is in fact the $J_c^T F$ term defined in Eq.2.

C. Quadratic Programming Formulation:

TSID sums all the task functions using scalar weights λ_i , $\forall i \in \mathcal{N}$ tasks, subject to constraints such as the rigid contacts and the dynamics:

$$\begin{aligned} \min_{y=[a, F]} & \sum_{i=0}^{\mathcal{N}} \lambda_i \|O_i y - o_i\|^2 \\ \text{s.t.} & \begin{bmatrix} J_c & 0 & 0 \\ M & -J_c^T & -N^T \end{bmatrix} \begin{bmatrix} a \\ F \\ \tau \end{bmatrix} = \begin{bmatrix} -\dot{J}_c v \\ -h \end{bmatrix} \end{aligned} \quad (8)$$

Tasks	Priority	Weight
Feet contacts	I	100
Feet tracking (walk simulation)	I	100
CoM tracking	I	50
Cartesian Force-Contact (force simulation)	I	10
Waist orientation	I	1
Posture regularization in half-sitting	I	0.1
AM regularization to 0	I	2×10^{-2}

TABLE I: Set of tasks used in the control scheme.

Where the free variables are the acceleration a and the force F . The QP is formulated to solve the acceleration and the forces and to then retrieve the torque using Eq.2. It has only two strict layers: the constraint layer (priority 0) and the cost layer (priority I). The set of tasks considered during the simulation are presented in the Table.I. Their respective task gains are set in Table II. The authors have implemented this controller using TSID [20] in the open-source package [26].

IV. PASSIVITY THEORY

A. Definition

A passivity-based control [3] is a dissipative [7] system respecting the following definition:

A system with the state space model of $\dot{x} = f(x, y)$ with initial state $x(0) = x_0 \in \mathbb{R}^n$, input vector $u \in \mathbb{R}^m$ and output $y = h(x, u)$ is said to be passive, if there exists a positive semi-definite function $H : \mathbb{R}^n \rightarrow \mathbb{R}^+$, called storage function, such that:

$$H(x(T)) - H(x(0)) \leq \int_0^T y^T(t) u(t) dt \quad (9)$$

$\forall y : [0, T] \rightarrow \mathbb{R}^m$, $x_0 \in \mathbb{R}^n$ and $T > 0$. The passivity is a stability criteria based on the power flow exchanged by the components of the system. $H(x)$ is often chosen as the energy of the system. One has thus to find an appropriate storage function $H(x)$ such that (in the following, we remove the state or time dependencies of the terms for clarity):

$$\dot{H} \leq y^T u \quad \forall (x, y) \quad (10)$$

Then, the system internal power (\dot{H} : derivative of the energy in the system) is lesser or equal to the power transferred to the system through its port ($y^T u$).

B. Analysis of our System

We have three components: the robot, the controller and the environment. We assume that the environment of the robot is a passive mapping ($v \rightarrow -\tau_{ext}$), in connection with impedance control [16, 14]. Then, the group robot-environment is passive, we have to make the group controller-robot passive. As in [14], the input of our controller is the velocity of the robot v . And the output of the controller is the control joint torque τ . Because the interconnection of passive systems leads to a general passive system we have two solutions: we can impose the passive mapping ($-Nv \rightarrow \tau$) between the robot and the controller, then because the group robot-environment is passive the overall system will be passive. Or, we can impose the passive mapping ($v \rightarrow \tau_{ext}$) between the system

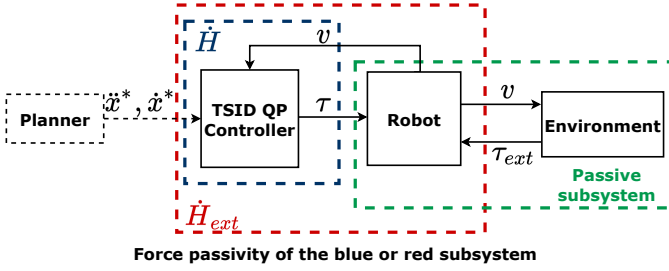


Fig. 2: Port-based modelling of the subsystems.

{robot+controller} and the environment, see Fig.2. Note that because $N^T \in \mathbb{R}^{n \times n_j}$ is the selector matrix we have (taking the Moore-Penrose pseudo-inverse) $N^\dagger \in \mathbb{R}^{n_j \times n} = N^T$, $N^\dagger N = I$ and $N^\dagger v = Nv = \dot{q}$ by definition (but $v \neq N^T \dot{q}$).

Then, as described in Section IV (see [16]), we have to find a storage function H such that:

$$\dot{H} \leq -(N^\dagger v)^T \tau = -\dot{q}^T \tau = -v^T N^T \tau \quad (11)$$

Or a storage function H_{ext} such that $\dot{H}_{ext} \leq v^T \tau_{ext}$. In the following, we prove the passivity of the system by building \dot{H} . Then we augment this function to obtain a working \dot{H}_{ext} .

C. Energy Tank

Let us introduce a virtual energy tank in the framework to store the energy dissipated by the tasks. The tank will then be used to transfer the stored energy when a task needs more energy to be completed, or, on the contrary if the tank is empty, to deteriorate the tracking of this task to ensure the passivity of the system. The energy tank is defined as a virtual storage element with flow variable \dot{s} and effort variable s such that [8, 9, 11] $E_{tank} = \frac{1}{2}s^2$.

V. FORMULATION USING ENERGY TANK IN TSID

A. Choosing E_{tank} for a set of tasks

We look at the proof of the passivity for a set of \mathcal{N} motion tasks (see Eq.4). We consider here only one level of hierarchy (the cost, priority D), and stack the tasks together into a single combined formulation (see Eq.15). For one task x , we define S the potential energy of the task:

$$\begin{aligned} S &= \frac{1}{2} e^T \Lambda K_P e \\ \dot{S} &= \dot{e}^T \Lambda K_P e + \frac{1}{2} e^T \dot{\Lambda} K_P e \end{aligned}, \quad \Lambda = J^\dagger M J^\dagger \quad (12)$$

with K_P from the controller task function (Eq.4), J^\dagger is the Moore-Penrose pseudo-inverse of J , $J^\dagger = U D^{-1} V^T$ with U , D and V respectively the left singular vectors, the diagonal matrix of singular values and the right singular vectors of J (using the Singular Value Decomposition), $e = (x - x^*)$ and Λ the positive definite Cartesian space inertia matrix. ΛK_P is semi-positive definite if K_P has equal positive elements (see Table.II) or under some specific conditions described in [27]. Then, S is always positive.

We define the storage function for one task as:

$$H = S + E_{tank} \quad (13)$$

Then, for a set of tasks we define $\mathbf{H} \in \mathbb{R}^{(\sum_0^{\mathcal{N}} 1) \times 1}$ as:

$$\mathbf{H} = [H_0 \dots H_{\mathcal{N}}]^T, \quad \dot{\mathbf{H}} = [\dot{H}_0 \dots \dot{H}_{\mathcal{N}}]^T \quad (14)$$

And accordingly, the vectors and matrix $\mathbf{J} \in \mathbb{R}^{L \times n}$, $\mathbf{K}_P, \mathbf{K}_D \in \mathbb{R}^{L \times L}$, $\mathcal{E} = \mathbf{X} - \mathbf{X}^* \in \mathbb{R}^L$, the respective stack of vectors and matrix of their non-bold value for the \mathcal{N} tasks, with $L = \sum_{i=0}^{\mathcal{N}} l_i$. From Eq.4 we obtain the following formulation for the set of motion tasks:

$$\dot{\mathbf{X}} = \mathbf{J} v, \quad \mathbf{J} a = \ddot{\mathbf{X}}^* - \dot{\mathbf{J}} v - \mathbf{K}_P \mathcal{E} - \mathbf{K}_D \dot{\mathcal{E}} \quad (15)$$

And the potential energy for the set of tasks as:

$$\begin{aligned} \mathbf{S} &= \frac{1}{2} \mathcal{E}^T \Lambda \mathbf{K}_P \mathcal{E}, \quad \Lambda = \mathbf{J}^\dagger M \mathbf{J}^\dagger \\ \dot{\mathbf{S}} &= \dot{\mathcal{E}}^T \Lambda \mathbf{K}_P \mathcal{E} + \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E} \end{aligned} \quad (16)$$

Using the equation of the dynamics (Eq.2) and the formulation of TSID for τ_{ext} (Eq.7), we have:

$$\begin{aligned} -v^T N^T \tau &= -v^T [M a + C v + g \\ &\quad - \sum_{k=1}^c J_k^T T_k^T f_k - J_F^T T_F^T F_F] \end{aligned} \quad (17)$$

We replace a by the Eq.15 and $M \mathbf{J}^\dagger$ by $\mathbf{J}^T \Lambda$ to obtain:

$$\begin{aligned} -v^T N^T \tau &= -v^T \left[\mathbf{J}^T \Lambda [\ddot{\mathbf{X}}^* - \dot{\mathbf{J}} v - \mathbf{K}_P \mathcal{E} - \mathbf{K}_D \dot{\mathcal{E}}] \right. \\ &\quad \left. + C v + g - \sum_{k=1}^c J_k^T T_k^T f_k - J_F^T T_F^T F_F \right] \end{aligned} \quad (18)$$

By definition $\dot{\mathbf{X}} = \mathbf{J} v$ and thus $\dot{\mathbf{X}}^T = v^T \mathbf{J}^T$. The previous equation is simplified in:

$$\begin{aligned} -v^T N^T \tau &= -\dot{\mathbf{X}}^T \Lambda (\ddot{\mathbf{X}}^* - \dot{\mathbf{J}} v) + \overbrace{\dot{\mathbf{X}}^T \Lambda \mathbf{K}_P \mathcal{E}}^{\Delta} \\ &\quad + \dot{\mathbf{X}}^T \Lambda \mathbf{K}_D \dot{\mathcal{E}} - v^T C v - v^T g \\ &\quad + v^T \sum_{k=1}^c J_k^T T_k^T f_k + v^T J_F^T T_F^T F_F \\ \Delta &= \dot{\mathbf{S}} + \dot{\mathbf{X}}^{*T} \Lambda \mathbf{K}_P \mathcal{E} - \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E} \end{aligned} \quad (19)$$

We know that \mathbf{S} is positive semi-definite, thus we want to choose the dynamics of our tank to contain the remaining terms (of which we do not know the sign):

$$\begin{aligned} \dot{E}_{tank} &= \overbrace{\dot{\mathbf{X}}^T \Lambda \mathbf{K}_D \dot{\mathcal{E}} + \dot{\mathbf{X}}^{*T} \Lambda \mathbf{K}_P \mathcal{E} - \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E}}^{\text{Damping of the tasks}} \\ &\quad - \dot{\mathbf{X}}^T \Lambda (\ddot{\mathbf{X}}^* - \dot{\mathbf{J}} v) - v^T [C v + g] \\ &\quad + v^T \left[\sum_{k=1}^c J_k^T T_k^T f_k + v^T J_F^T T_F^T F_F \right] \end{aligned} \quad (20)$$

With this definition, $\dot{\mathbf{H}} = \dot{\mathbf{S}} + \dot{E}_{tank} = -v^T N^T \tau$, ensuring the passivity of the system for the power-port $\{-Nv, \tau\}$. \square

B. Including protection and valve

Once we have the desired formulation of the dynamics of the tank, we define α, β, γ , the coefficients that connect the tank to the system and regulate the tasks. They are used to bound the energy in the tank and to limit the power transferred to the system.

Let $E_{tank}^{max}, E_{tank}^{min}$ be the upper and lower bounds of our tank and P_{low} the power transfer limit. We have $E_{tank} = \frac{1}{2}s^2$ thus, $\dot{E}_{tank} = s\dot{s}$ and we build \dot{s} to take into account the valves and satisfy $\dot{\mathbf{S}} + \dot{E}_{tank} \leq -v^T N^T \tau$. \dot{s} is separated in two components: \dot{s}^σ controllable, linked to the tasks and \dot{s}^ϕ

gathering the external components we cannot control (gravity, Coriolis and forces terms).

$$\dot{s}^\sigma = \frac{1}{s} \beta \gamma \underbrace{\left[\dot{\mathbf{X}}^T \Lambda \mathbf{K}_D \dot{\mathbf{E}} + \dot{\mathbf{X}}^{*T} \Lambda \mathbf{K}_P \mathcal{E} - \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E} - \dot{\mathbf{X}}^T \Lambda (\ddot{\mathbf{X}}^* - \dot{\mathbf{J}}v) + v^T J_F^T T_F^T F_F \right]}_{B^\sigma} \quad (21)$$

$$\dot{s}^\phi = -\frac{1}{s} v^T \underbrace{\left[Cv + g - \sum_{k=1}^c J_k^T T_k^T f_k \right]}_{B^\phi} \quad (22)$$

$$\dot{s} = (1 - \alpha)(\dot{s}^\sigma + \dot{s}^\phi) \quad (23)$$

Like in [12], to control the flow transferred from the tank to the tasks, we choose:

$$\gamma = \begin{cases} \frac{P_{low}}{B^\sigma} & \text{if } B^\sigma < P_{low} \leq 0 \\ 1 & \text{else} \end{cases} \quad (24)$$

$$\beta = \begin{cases} 0 & \text{if } (E_{tank} \leq E_{tank}^{min}) \ \& \ (\gamma B^\sigma - B^\phi < 0) \\ 1 & \text{else} \end{cases} \quad (25)$$

And like in [11], we choose α to be:

$$\alpha = \begin{cases} 1 & \text{if } (E_{tank}^{max} \leq E_{tank}) \ \& \ (\beta \gamma B^\sigma - B^\phi > 0) \\ 0 & \text{else} \end{cases} \quad (26)$$

α is the overflow valve, when $\alpha = 1$ the tank can be filled, otherwise the tank is already full and cannot be filled more [11]. β is the opposite, it controls the output flow of the tank, if $\beta = 0$ the tank at its lowest value and the controller cannot use it [11]. Finally, γ regulates the energy flow transferred from the tank to the controller. It avoids sharp release of the tank energy to the system, P_{low} is an empirical power threshold defining when the transfer should be regulated. When B^σ (the controllable part of the tank derivative energy) is lesser than this threshold (negative: the system takes energy from the tank), γ takes the value of $\frac{P_{low}}{B^\sigma}$ [28]. Then $0 \leq \gamma \leq 1$. We use the same approach as [29] to avoid discontinuities and smooth the coefficient transitions. The impact of the energy tank is propagated to the tasks with the parameters β, γ . They are introduced in the task functions such that we have new references $a^{\beta\gamma}$ and $F_F^{\beta\gamma}$:

$$\begin{aligned} J a^{\beta\gamma} &= J \beta \gamma a &= \beta \gamma \left[\ddot{x}^* - \dot{J}v - K_{Pe} - K_D \dot{e} \right] \\ T_F^T F_F^{\beta\gamma} &= T_F^T \beta \gamma F_F &= \beta \gamma \left[F^* + P_F e_f + I_F \int_0^t e_f ds \right] \end{aligned} \quad (27)$$

Thus, the parameters appear naturally in the equations when replacing the acceleration and the force by $a^{\beta\gamma}$ and $F_F^{\beta\gamma}$ in the equation of the dynamics. The Eq.19 is then:

$$\begin{aligned} -v^T N^T \tau &= \underbrace{\beta \gamma \left[\dot{\mathbf{X}}^T \Lambda \mathbf{K}_D \dot{\mathbf{E}} + \dot{\mathbf{X}}^{*T} \Lambda \mathbf{K}_P \mathcal{E} + v^T J_F^T T_F^T F_F - \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E} - \dot{\mathbf{X}}^T \Lambda (\ddot{\mathbf{X}}^* - \dot{\mathbf{J}}v) \right]}_{s \dot{s}^\sigma} \\ &\quad - \underbrace{v^T \left[Cv + g - \sum_{k=1}^c J_k^T T_k^T f_k \right]}_{s \dot{s}^\phi} + \beta \gamma \dot{\mathbf{S}} \end{aligned} \quad (28)$$

And then, the terms of the energy tank derivative appear ($\dot{E}_{tank} = s(1 - \alpha)(\dot{s}^\sigma + \dot{s}^\phi)$), as the one of the potential energy of the tasks ($\dot{\mathbf{S}}$).

Proof with valves for the power port $\{-Nv, \tau\}$: We first analyse the derivative of the storage function H in order to prove the passivity of the blue system in Fig.2.

Choosing $\mathbf{H} = \mathbf{S} + \mathbf{E}_{tank}$ we have:

$$\begin{aligned} \dot{\mathbf{H}} &= \dot{\mathbf{S}} + \dot{\mathbf{E}}_{tank} \\ \dot{\mathbf{H}} &= \beta \gamma \left[\Delta - \dot{\mathbf{X}}^{*T} \Lambda \mathbf{K}_P \mathcal{E} + \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E} \right] + \\ &\quad (1 - \alpha) \left[\beta \gamma \left[\dot{\mathbf{X}}^T \Lambda \mathbf{K}_D \dot{\mathbf{E}} + \dot{\mathbf{X}}^{*T} \Lambda \mathbf{K}_P \mathcal{E} - \right. \right. \\ &\quad \left. \left. \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E} - \dot{\mathbf{X}}^T \Lambda (\ddot{\mathbf{X}}^* - \dot{\mathbf{J}}v) + v^T J_F^T T_F^T F_F \right] \right] \\ &\quad + (1 - \alpha) \left[-v^T \left[Cv + g - \sum_{k=1}^c J_k^T T_k^T f_k \right] \right] \end{aligned} \quad (29)$$

Replacing the Eq.28 in Eq.29 we finally obtain:

$$\begin{aligned} \dot{\mathbf{H}} &= -v^T N^T \tau - \alpha \left[-v^T \left[Cv + g - \sum_{k=1}^c J_k^T T_k^T f_k \right] \right] \\ &\quad - \alpha \left[\beta \gamma \left[\dot{\mathbf{X}}^T \Lambda \mathbf{K}_D \dot{\mathbf{E}} + \dot{\mathbf{X}}^{*T} \Lambda \mathbf{K}_P \mathcal{E} - \right. \right. \\ &\quad \left. \left. \frac{1}{2} \mathcal{E}^T \dot{\Lambda} \mathbf{K}_P \mathcal{E} - \dot{\mathbf{X}}^T \Lambda (\ddot{\mathbf{X}}^* - \dot{\mathbf{J}}v) + v^T J_F^T T_F^T F_F \right] \right] \\ &= -v^T N^T \tau - \alpha \left[-B^\phi + \beta \gamma B^\sigma \right] \\ &\leq -v^T N^T \tau \end{aligned} \quad (30)$$

Indeed, here the role of α became clear, because of Eq.26, if $\alpha \neq 0$ then $\beta \gamma B^\sigma - B^\phi > 0$, thus: $-\alpha \left[\beta \gamma B^\sigma - B^\phi \right] \leq 0$. Then from Eq.30 we can conclude the passivity of the system with \mathcal{N} tasks for the power port $\{-Nv, \tau\} = \{-\dot{q}, \tau\}$. \square

Proof with valves for the power port $\{v, \tau_{ext}\}$: Using the previous result, we can also prove the passivity of the overall closed-loop dynamics, the red system of Fig.2. Let us augment the storage function \mathbf{H} with the kinetic energy of the tasks and the gravity potential to obtain \mathbf{H}_{ext} :

$$\mathbf{H}_{ext} = \mathbf{S} + \mathbf{E}_{tank} + \frac{1}{2} \dot{\mathbf{X}}^T \Lambda \dot{\mathbf{X}} + V_g(q) \quad (31)$$

We have by definition, $g(q) = \left(\frac{\partial V_g(q)}{\partial p} \right)^T$, leading to:

$$\frac{\partial V_g(q)}{dt} = \frac{\partial V_g(q)}{\partial p} \frac{\partial p}{dt} = \left(\frac{\partial p}{dt} \frac{\partial V_g(q)}{\partial p} \right)^T = v^T g(q) \quad (32)$$

Let us write the formulation of the equation of the dynamics for one task in the Cartesian space:

$$\Lambda \ddot{x} + \mu \dot{x} = J^{\dagger T} (-g + N^T \tau + \tau_{ext}) \quad (33)$$

with $\mu = J^{\dagger T} C J^{\dagger} - \Lambda \dot{J} J^{\dagger}$. Then, for a set of \mathcal{N} tasks we have, \mathbf{Z} designating the stack of μ :

$$\Lambda \ddot{\mathbf{X}} + \mathbf{Z} \dot{\mathbf{X}} = \mathbf{J}^{\dagger T} (-g + N^T \tau + \tau_{ext}) \quad (34)$$

By taking the derivative of Eq.31 and substituting Eq.34, we can write:

$$\begin{aligned} \dot{\mathbf{H}}_{ext} &= \dot{\mathbf{S}} + \dot{\mathbf{E}}_{tank} + v^T g + \frac{1}{2} \dot{\mathbf{X}}^T \dot{\Lambda} \dot{\mathbf{X}} + \\ &\quad \underbrace{\dot{\mathbf{X}}^T \left[\mathbf{J}^{\dagger T} (-g + N^T \tau + \tau_{ext}) - \mathbf{Z} \dot{\mathbf{X}} \right]}_{v^T} \end{aligned} \quad (35)$$

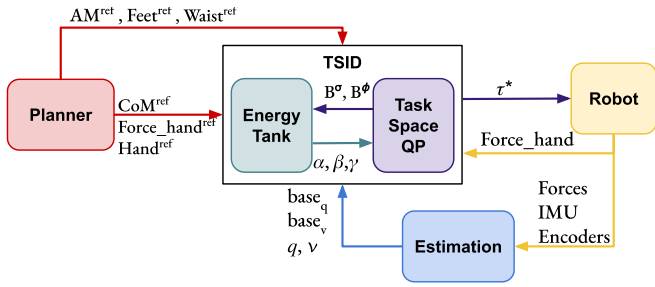


Fig. 3: TSID torque control scheme with global energy tank.

Using the fact that the Coriolis matrix is defined such that: $\frac{1}{2}\dot{M}(q) = \frac{C(q,v)+C(q,v)^T}{2}$, we have $\dot{X}^T(\frac{1}{2}\dot{\Lambda} - Z)\dot{X} = 0$ (because $\frac{1}{2}\dot{\Lambda} - Z$ is skew-symmetric).

Finally we obtain, using the Eq.30:

$$\dot{\mathbf{H}}_{\text{ext}} = \underbrace{\dot{\mathbf{S}} + \dot{\mathbf{E}}_{\text{tank}}}_{\leq -v^T N^T \tau} + v^T N^T \tau + v^T \tau_{\text{ext}} \leq v^T \tau_{\text{ext}} \quad (36)$$

Delivering the proof of passivity for the port $\{v, \tau_{\text{ext}}\}$. \square

C. Modified passive QP formulation

The classical inverse dynamics formulation is transformed in a non-linear problem with two stages. It is a sort of alternating minimization problem. The first stage (1) computes the energy tank dynamics and the regulating coefficients and then the second stage (2) solves the QP modified by these coefficients. One can notice that the first stage finds the coefficients for a torque which does not take into account the constraints of the QP. Thus as in [19], if a constraint became active, the passivity is no longer guaranteed because the computation used for the energy tank is not the same as the one of the QP. This is why we add a new constraint in the QP formulation, similarly to [4], enforcing the inequality $\dot{\mathbf{H}} \leq -v^T N^T \tau$ to maintain the passivity in any situation. The general control scheme is illustrated in Fig.3 and we obtain the new formulation:

$$\begin{aligned} (1) \quad & \text{Find } \alpha, \beta, \gamma, \dot{\mathbf{H}} \text{ s.t. Eq.21 - 26, 29} \\ (2) \quad & \min_{y=[a,F]} \sum_{i=0}^{\mathcal{N}} \lambda_i \|O_i y - \beta \gamma o_i\|^2 \\ \text{s.t.} \quad & \begin{bmatrix} J_c & 0 & 0 \\ M & -J_c^T & -N^T \end{bmatrix} \begin{bmatrix} a \\ F \\ \tau \end{bmatrix} = \begin{bmatrix} -j_c v \\ -h \end{bmatrix} \\ & \tau_{\min} \leq \tau \leq \tau_{\max}, F_{\min} \leq F \leq F_{\max}, \dot{\mathbf{H}} \leq -v^T N^T \tau \end{aligned} \quad (37)$$

VI. SIMULATIONS

This section details the Gazebo simulations demonstrating the validity of our implementation. The tests are performed on a humanoid robot TALOS under the set of tasks defined in Table.I and their weights reported in Table.II. The design parameters of the global energy tank are set as follows: $E_{\text{tank}}^{\max} = 5\text{J}$, $E_{\text{tank}}^{\min} = 0.1\text{J}$ and $P_{\text{low}} = -1\text{W}$. They are set as low values in order to keep small the contribution of the tank energy to the overall energy of the system. In the simulations, we use the robot base-estimator of [30]. In

Tasks Gains	K_P	K_D	K_I
CoM	$3 \times [50]$	$3 \times [5]$	-
Waist	$3 \times [100]$	$2\sqrt{K_P}$	-
Feet Contacts	$6 \times [30]$	$2\sqrt{K_P}$	-
Feet Motions (walk simulation)	$6 \times [100]$	$6 \times [5]$	-
Force-Contact (force simulation)	$6 \times [10]$	-	$0.5K_P$
AM	$3 \times [10]$	$2\sqrt{K_P}$	-
Posture	$32 \times [80]$	$2\sqrt{K_P}$	-

TABLE II: Tasks gains of the control scheme. We denote by $n \times [\cdot]$ a vector of n elements with the same value.

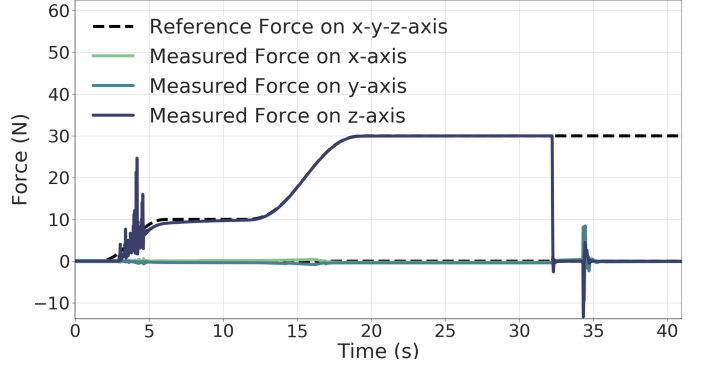


Fig. 4: Reference and measured forces of the pushing task.

the force task case, no planning methods were used as the robot remains in the same configuration. The force references were created by linear interpolation between the actual and final desired force. The reference trajectories of the 20cm step walk were computed with the multicontact-locomotion-planning framework [31] like in [32].

A. Force task with unplanned contact removal simulation

The robot makes a contact between a tool and a surface and applies a 30N force along the z-axis. In Gazebo, the tool is simulated by a cylinder and the surface is a simple square block, shown in Fig.1. This set-up is a first step toward more complex operations, requiring contact and a force application, such as drilling. It exposes the problem of unexpected broken contact when the drill bit exits the hole (or caused by slippage/disturbances).

In this simulation, the reference force is raised in two steps, the first one reaches 10N to stabilize the contact, in Fig.4 some oscillations can be seen in the first slope, creating instability in the tank coefficients in Fig.5. Then, it is set to 30N, and at 32.3s the block is removed. The energy tank is initialized

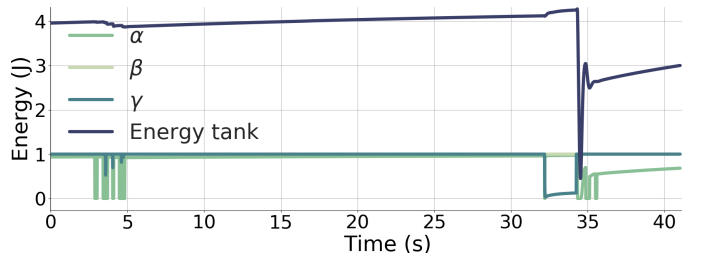


Fig. 5: Energy tank and coefficients of the force simulation.

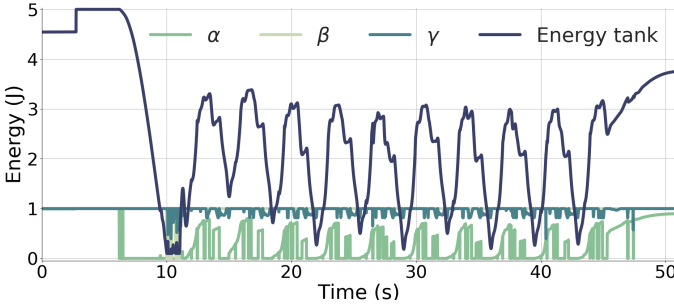


Fig. 6: Energy tank and coefficients of the walking simulation.

at 3J. As shown in Fig.4, the measured force along the z-axis falls to 0N when the block is removed. First, a simulation without the energy tank and the passivity constraint is done. The robot quickly falls and cannot react in time to balance itself or to remove the force task when detecting the loss of the contact.

On the contrary, with the passive formulation, when the block is removed the robot does not fall immediately. Indeed, as presented in Fig.5, the α and γ coefficients are triggered. With the γ closes to 0, the force task is highly penalized and thus the hand slowly comes down. As a result, the system has the time to detect the loss of the contact and remove the task, leading to the pick in the consumption of the energy tank at 34.4s. By removing the task, the robot goes back to its initial position because of the postural task and thus raises its hand, going against the gravity and consuming energy.

B. 20cm step walk simulation

In this simulation the robot walks 1 meter forward with 20cm steps, starting with the right foot. The first and last steps are only 10cm long, and the first CoM swing is quasi-static. The double support duration is of 0.2s while the single support is of 1.2s. The set of tasks and gains used are the same as the previous simulation without the Cartesian Force-Contact task and with feet tracking tasks during the single support stages.

Fig.6 presents the energy tank evolution during the simulation and the regulating coefficients. At the beginning of the motion during the quasi-static swing the energy tank is emptied and the β parameter decreases to 0 around the 10th seconds. The oscillations of the parameters lead the robot to oscillate as well because the β, γ parameters are used in the task errors formulation (see Eq.30). As a result, the potential function derivative \dot{H} is really noisy around the 10th seconds. To avoid this concerning behavior one can increase the budget of the energy tank (set at its maximum value, 5J, at the beginning of the motion), or try to replace the quasi-static part by a dynamic one to see if the tank is emptied. We think that the better solution would be to design the energy tank and P_{low} parameters in function of the reference trajectories [12].

After the concerning beginning, the robot achieves the walking motion efficiently (good reference trajectories tracking). The energy tank is filled and emptied in a sinusoidal manner. It increases when the CoM is taken back in the middle of the support polygon, thus, when the feet are in double support. And it decreases when the robot CoM reaches the edge of the

support polygon above the supporting foot. The authors think that it comes from the postural regularization task (in half-sitting position) and the CoM/feet motion tasks (consuming energy due to small tracking delays).

VII. DISCUSSION

We first discuss the positivity of S in order to have H positive. In this paper, K_p has equal elements in function of the tasks to achieve this result. However, this is restrictive, one may want to find properties on ΛK_p to achieve the positivity with different gains [27] or to reformulate $S = \frac{1}{2}e^T K_P^{\frac{1}{2}} \Lambda K_P^{\frac{1}{2}} e$ to keep the symmetry (the presented proofs need to be updated accordingly but should follow the same structure).

During the first simulation we observe oscillations on the α and γ coefficients due to oscillations of the tool on the surface and after removing the force task. This behavior is mitigated thanks to smoothness functions [29] but can be improved. Indeed, in [12] the P_{low} value varies in function of the task progression, leading to a smoother γ because it has been learnt and built in coherence with the task to achieve.

One problem of the presented formulation is to maintain the robot balance: if the coefficient β falls to 0, each task will be degraded to a zero tracking. Thus, nothing will keep the balance of the robot and it may fall. To palliate this problem, one can choose to not multiply the β coefficient on the postural task and setting the reference posture to the current one, in order to have the robot only compensates the gravity force when $\beta = 0$. However, this will prevent the postural task to be regulated by the energy tank and may lead to a non-passive behavior. Moreover, this solution is unreliable if the robot has a reference posture not guaranteeing the balance. Using a reference computed on-line by the planning to ensure the balance can be a solution. As said previously, a solution would be to design the energy tank and the coefficients in function of the reference trajectories [12], then β falling to 0 should be rare.

Another point of concern is the addition of the passivity constraint in the QP which may lead to a too constrained problem. For instance, a case may arise where the passivity needs a torque which exceeds the limits imposed on it by the QP. The two constraints will be in conflict and lead to oscillations in the solution. To solve this problem it will be necessary to take into account the constraints of the QP in the formulation of the energy tank. It may also be possible to reformulate the problem in order to compute the two stages of our formulation in Eq.37 simultaneously. Otherwise, it will be necessary to relax the passivity constraint. This is a limit of the passivity formulation at control level; the authors think that on-line planning can help for this kind of issue. For instance, using Model Predictive Control with a constraint on the passivity may be a future aim.

Finally, the transition from the simulations to the real experiments is not straightforward. In the Gazebo simulator, the joint torque control is almost perfect because the dynamics of the motors are neglected. However, not taking these dynamics into account will lead to unrealistic and dangerous behaviors on the real robot. Moreover, the real robot is subject to

imperfections such as errors in the actuation chain model, sensor noise, limited torque bandwidth or delays [33]. In the presented implementation, the signals retrieved from the robot are filtered and the force sensors on the feet are used to improve the robustness of the base-estimator. The authors are currently testing the simulation on the simulator of the TALOS constructor, PAL robotics, which models the actuator dynamics of the robot.

CONCLUSION

In this paper a novel control formulation for the passivation of an inverse dynamics framework is presented. It involves a global energy tank which monitors the exchange of energy between a set of tasks and regulates their gains. This method is close to the passive hierarchical impedance control relying on a strict hierarchy and null-space projection of [11]. It adds a protection valve for power flow variations and is adapted for a non-strict hierarchy on an under-actuated humanoid robot. The robustness of the controller is evaluated in simulations on Gazebo through a multi-contact scenario and a 20cm walk involving six and five tasks. Thus, the proposed method extends the repertoire of multi-objectives non-hierarchical controllers for applications requiring physical interactions with the environment or a human. Its passive formulation ensures a stable, safe and robust behavior of the system.

For our future works, we plan to improve the behavior of the coefficients, in particular, the one on the power flow regulation can be computed as in [12], to vary in function of the task progression. The impact of these coefficients on the balance of the robot will be evaluated, for now, the robot may falls if all the system is penalized to respect the passivity. Finally, our new control formulation will be evaluated on the real robot.

REFERENCES

- [1] G. Nava, F. Romano, F. Nori, and D. Pucci, "Stability analysis and design of momentum-based controllers for humanoid robots," in *IROS*, 2016.
- [2] R. Cisneros, M. Benallegue, A. Benallegue, M. Morisawa, H. Audren, P. Gergondet, A. Escande, A. Kheddar, and F. Kanehiro, "Robust humanoid control using a qp solver with integral gains," in *IROS*, 2018.
- [3] A. Schaft, *L2-Gain and Passivity in Nonlinear Control*. Springer International Publishing, 2000.
- [4] L. Joseph, V. Padois, and G. Morel, "Towards x-ray medical imaging with robots in the open: safety without compromising performances," *ICRA*, 2018.
- [5] M. Camlibel, A. Julius, R. Pasumarthy, and J. Scherpen, Eds., *Mathematical Control Theory I: Nonlinear and Hybrid Control Systems*, ser. Lecture Notes in Control and Information Sciences. Springer, 2015.
- [6] G. A. Folkertsma and S. Stramigioli, "Energy in robotics," *Foundations and Trends® in Robotics*, 2017.
- [7] J. C. Willems, *Dissipative dynamical systems part I: General theory*, A. for Rational Mechanics and Analysis, Eds. Springer, 1972.
- [8] P. R. Giordano, A. Franchi, C. Secchi, and H. H. Bühlhoff, "A passivity-based decentralized strategy for generalized connectivity maintenance," *IJRR*, 2013.
- [9] A. Dietrich, C. Ott, and S. Stramigioli, "Passivation of projection-based null space compliance control via energy tanks," *RAL*, 2016.
- [10] T. S. Tadele, T. De Vries, and S. Stramigioli, "Combining energy and power based safety metrics in controller design for domestic robots," in *ICRA*, 2014.
- [11] A. Dietrich, X. Wu, K. Bussmann, C. Ott, A. Albu-Schäffer, and S. Stramigioli, "Passive hierarchical impedance control via energy tanks," *RAL*, 2017.
- [12] E. Shahriari, L. Johannsmeier, E. Jensen, and S. Haddadin, "Power flow regulation, adaptation, and learning for intrinsically robust virtual energy tanks," *RAL*, 2020.
- [13] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *IJRR*, 2007.
- [14] C. Ott, A. Albu-Schaffer, A. Kugi, and G. Hirzinger, "On the passivity-based impedance control of flexible joint robots," *ITRO*, 2008.
- [15] A. Giusti, J. Malzahn, N. G. Tsagarakis, and M. Althoff, "On the combined inverse-dynamics/passivity-based control of elastic-joint robots," *ITRO*, 2018.
- [16] G. Garofalo and C. Ott, "Passive energy-based control via energy tanks and release valve for limit cycle and compliance control," *IFAC-PapersOnLine*, 2018, 12th IFAC Symposium on Robot Control SYROCO 2018.
- [17] G. Cheng, S.-h. Hyon, J. Morimoto, A. Ude, G. Colvin, W. Scroggin, and S. C. Jacobsen, "Cb: A humanoid research platform for exploring neuroscience," in *2006 6th IEEE-RAS ICHR*, 2006.
- [18] B. Henze, M. A. Roa, and C. Ott, "Passivity-based whole-body balancing for torque-controlled humanoid robots in multi-contact scenarios," *IJRR*, 2016.
- [19] J. Engelsberger, A. Dietrich, G. Mesesan, G. Garofalo, C. Ott, and A. Albu-Schäffer, "Mptc – modular passive tracking controller for stack of tasks based control frameworks," in *RSS*, 2020.
- [20] GEPETTO Team LAAS-CNRS, "Tsid," https://github.com/NoelieRamuzat/tsid/tree/topic/task_energy_contact.
- [21] A. Del Prete, N. Mansard, O. Ponce, O. Stasse, and F. Nori, "Implementing torque control with high-ratio gear boxes and without joint-torque sensors," *IJHR*, 2016.
- [22] A. Herzog, N. Rotella, S. Mason, F. Grimmering, S. Schaal, and L. Righetti, "Momentum control with hierarchical inverse dynamics on a torque-controlled humanoid," *Autonomous Robots*, 2014.
- [23] N. Ramuzat, G. Buondonno, S. Boria, and O. Stasse, "Comparison of position and torque whole body control schemes on the humanoid robot talos," in *ICAR*, 2021.
- [24] P. M. Wensing and D. E. Orin, "Generation of dynamic humanoid behaviors through task-space control with conic optimization," in *ICRA*, 2013.
- [25] S. Lee and A. Goswami, "A momentum-based balance controller for humanoid robots on non-level and non-stationary ground," *Autonomous Robots*, 2012.
- [26] GEPETTO Team LAAS-CNRS, "talos-torque-control," <https://github.com/NoelieRamuzat/talos-torque-control/>

tree/topic/task_energy.

- [27] N. Ramuzat, “On the semi-positive definition of the product of a positive diagonal matrix and a symmetric semi-positive definite matrix,” in *HAL open archive*, 2022.
- [28] C. Schindlbeck and S. Haddadin, “Unified passivity-based cartesian force/impedance control for rigid and flexible joint robots via task-energy tanks,” in *ICRA*, 2015.
- [29] K. Kronander and A. Billard, “Passive interaction control with dynamical systems,” *RAL*, 2016.
- [30] T. Flayols, A. D. Prete, P. Wensing, A. Mifsud, M. Benallegue, and O. Stasse, “Experimental evaluation of simple estimators for humanoid robots,” in *ICHR*, 2017.
- [31] GEPETTO Team LAAS-CNRS, “multicontact-locomotion-planning,” <https://github.com/loco-3d/multicontact-locomotion-planning>.
- [32] N. Ramuzat, F. Forget, V. Bonnet, M. Gautier, S. Boria, and O. Stasse, “Actuator model, identification and differential dynamic programming for a talos humanoid robot,” in *European Control Conference (ECC)*, 2020.
- [33] J. Engelsberger, G. Mesesan, A. Werner, and C. Ott, “Torque-based dynamic walking - a long way from simulation to experiment,” in *ICRA*, 2018.