



**HAL**  
open science

# Benchmarking Whole Body controllers on the TALOS Humanoid Robot

Noelie Ramuzat, Olivier Stasse, Sebastien Boria

► **To cite this version:**

Noelie Ramuzat, Olivier Stasse, Sebastien Boria. Benchmarking Whole Body controllers on the TALOS Humanoid Robot. *Frontiers in Robotics and AI*, 2022, 9, pp.826491. 10.3389/frobt.2022.826491 . hal-03614143

**HAL Id: hal-03614143**

**<https://laas.hal.science/hal-03614143v1>**

Submitted on 19 Mar 2022

**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.

# Benchmarking Whole Body controllers on the TALOS Humanoid Robot

N. Ramuzat<sup>1,2</sup>, O. Stasse<sup>1,\*</sup>, and S. Boria<sup>1</sup>

<sup>1</sup>LAAS-CNRS, Univ. of Toulouse, France

<sup>2</sup>Airbus, Toulouse, France

Correspondence\*:  
Corresponding Author  
ostasse@laas.fr

2 This paper is an evolved version of "Comparison of Position and Torque Whole Body Control Schemes on  
3 the TALOS Humanoid Robot" by N. Ramuzat, O. Stasse and S. Boria accepted to ICAR 2021 and available  
4 here: <https://hal.archives-ouvertes.fr/hal-03145141>. The novel part of this version is the experimental  
5 section.

## 6 ABSTRACT

7 This paper presents a comparison of three control schemes applied on the commercially  
8 available TALOS Humanoid Robot. The aim is to highlight the advantages and drawbacks of each  
9 model applied on three locomotion problems: walking on flat and non-flat terrain and climbing  
10 stairs. The different models are based on position control (first and second models) or torque  
11 control (third model). The first one uses a hierarchical quadratic program at velocity level. The  
12 second one employs a weighted quadratic program named Task Space Inverse Dynamic (TSID)  
13 at acceleration level. Finally, the last one also uses TSID but at torque level. The controllers  
14 performances are compared in simulation, using Gazebo, on the accuracy of their tracking, their  
15 energy consumption and their computational time execution.

16 **Keywords:** Humanoid robots, whole body control, benchmarking

## 1 INTRODUCTION

### 17 1.1 Goal

18 Bipedal locomotion of humanoid robots is considered as a difficult problem because of the complexity of  
19 the robot dynamics, the numerous constraints of the motion and the unknown environment. The design  
20 choice made when designing a robot may have a strong impact on the control law that are *really* working  
21 on the system, and the real performances. A recent example is the Digit robot which has very impressive  
22 capabilities by choosing a careful trade-off between the chosen actuation technology and the robot weight  
23 distribution Robotics (2022). The robot is very robust to impact, is allowing torque control but is slightly  
24 limited by the payload it can carries (10 kg). Realizing torque control on electric based bipedal system  
25 is challenging. If it was successfully realized on the TORO robot Engelsberger et al. (2014) for standing  
26 whole body control and walking, it is notoriously more difficult to achieve than position control. A striking  
27 example is giving by the iCub robot with which impressive Taichi motions have been realized Pucci et al.  
28 (2016) but where walking in torque control mode is still difficult to achieve Romualdi et al. (2019). The  
29 goal of this paper is to report a similar evaluation with the commercially available TALOS robot from the  
30 PAL-Robotics company.

## 31 1.2 Motion execution pipeline

32 Three stages are usually considered to execute a motion on a humanoid robot: the contact sequence  
33 generation, the trajectory planning and the whole-body control.

34 Most of trajectory planning methods use the centroidal dynamics to generate consistent behaviors for a  
35 legged robot. In this work we use preplanned trajectories provided either by a standard walking pattern  
36 generator or by a multi-contact planner Fernbach et al. (2020) The latter is used for a platform which can  
37 be easily rebuild for benchmarking walking on uneven terrain. This planner provides a centroidal trajectory  
38 that is dynamically balanced on uneven terrain, and does not assume that the robot behaves completely like  
39 a Linear Inverted Pendulum (LIPM). Because the centroidal dynamics is planned and the setup limits the  
40 number of contacts to one or two it is still possible to apply the concept of Divergent Component of Motion  
41 (DCM) Takenaka et al. (2009); Engelsberger et al. (2015) for control. The newly generated reference DCM  
42 is used for admittance control on the Center of Mass (CoM) as for Caron et al. (2019); Romualdi et al.  
43 (2019).

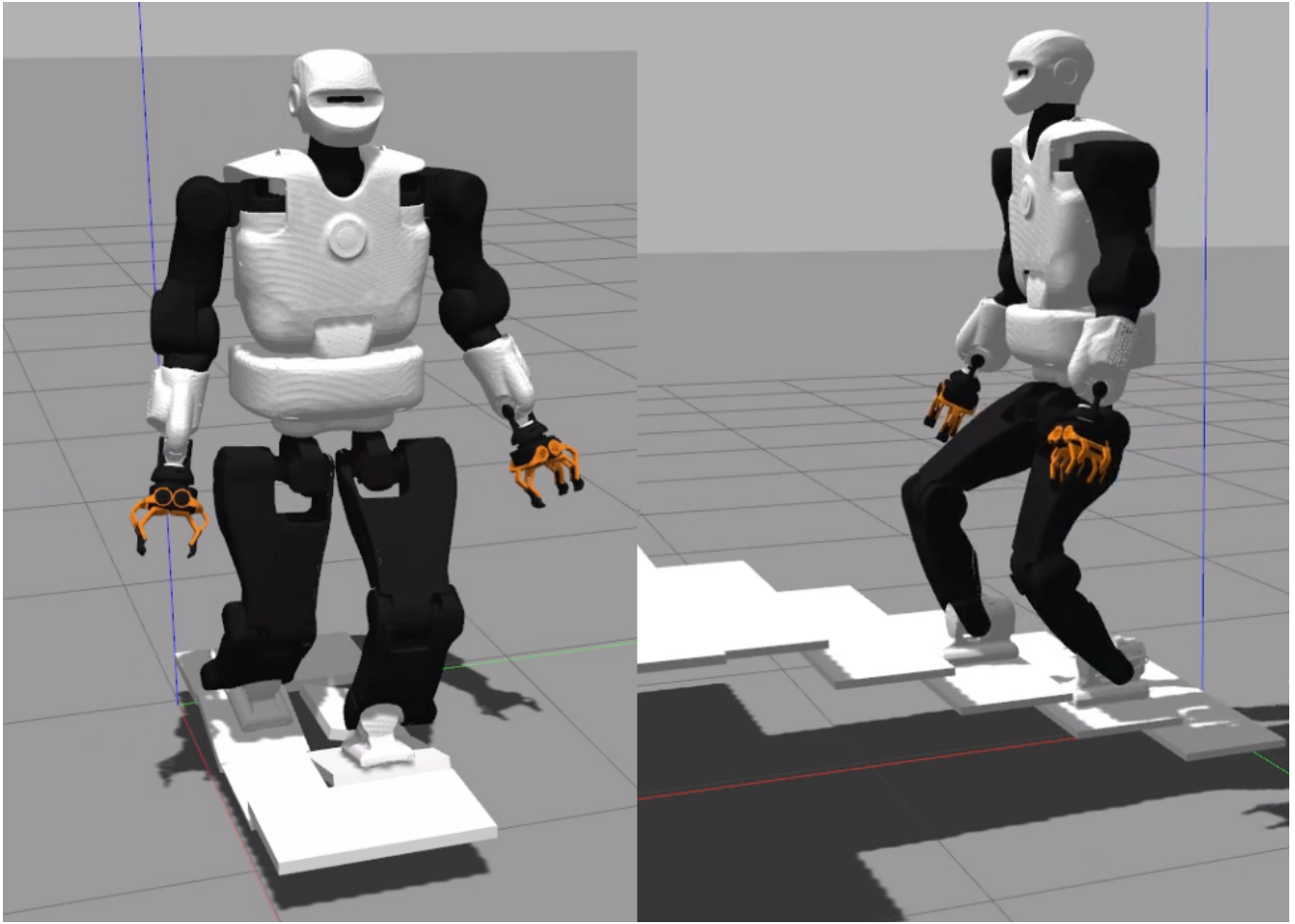
44 Then to track the reference trajectories a whole-body controller is needed. Whole-body controllers are  
45 based on the task function approach Samson et al. (1991); Escande et al. (2014) from which a quadratic  
46 program is formulated. Complex motions combine several nonlinear tasks and constraints. In this paper  
47 two types of QP formulations are compared, a Hierarchical QP which imposes a strict hierarchy between  
48 the tasks Henze et al. (2016); Herzog et al. (2014), and a weighted QP which sets weights to prioritise the  
49 tasks Koolen et al. (2016); Cisneros et al. (2018).

50 In the recent literature there is a growing number of implementations of torque based whole body control  
51 algorithms Koolen et al. (2016); Herzog et al. (2014); Lee and Goswami (2012); Engelsberger et al. (2015).  
52 Indeed, due to the intrinsic compliance of the torque control formulation, it is more suitable for interactions  
53 with humans and for multi-contact problems where external interactions and several contact points are  
54 needed. However, the transition from the simulations to the real experiments are harder due to inaccuracies  
55 on the actuation chain model Ramuzat et al. (2020). Such inaccuracies do not appear when using position  
56 control.

## 57 1.3 Contributions

58 Following the existing benchmarking of humanoid robots control architectures Romualdi et al. (2019);  
59 Stasse et al. (2018) this paper contributes by benchmarking the TALOS humanoid robot. It is done by  
60 comparing three whole-body control schemes on the TALOS robot in simulation. Two are using position  
61 control associated with DCM and CoM admittance controls and one using torque control. The first one  
62 is based on a Lexicographic QP using Inverse Kinematics (denoted *IK* in this paper), while the second  
63 and the third one use a Weighted QP (WQP) with Inverse Dynamics and an Angular Momentum (AM)  
64 task (denoted respectively *TSID position* and *TSID torque*). They are evaluated in Gazebo simulations  
65 on three locomotion problems: walking on flat, uneven terrains and stairs (Fig. 1), on the criterion of  
66 trajectory tracking, energy consumption, passivity and computational cost. As a first consequence of  
67 our torque control scheme, we achieve the highest walking velocity for the robot TALOS in simulation:  
68 0.6m/s. We believe that the motion on uneven terrain with the platforms is novel and offers an interesting  
69 new benchmark. Finally we also provides an evaluation of the Passivity Gait Measure that we believe is  
70 interesting to measure the efficiency of a balance strategy in terms of energy.

71 We organize the article as follows: Section 2 recalls the centroidal dynamics equations, the DCM control  
72 and the AM task. Section 3 details the three task-space whole-body control schemes compared in this paper.



**Figure 1.** Walking on Tilted Platforms and Climbing Stairs.

73 Section 4 presents the energy criterion employed. Section 5 describes the planning methodologies used to  
 74 obtain the reference trajectories for the simulations. Then, Section 6 presents these simulations results and  
 75 Section 7 describes the experiments achieved on TALOS and their limitations.

## 2 CENTROIDAL DYNAMICS

76 The under-actuated part of the robot whole-body dynamics is called the centroidal dynamics. It uses the  
 77 Newton-Euler equations of motion which couple the variations of the centroidal momentum with the  
 78 contact forces Orin et al. (2013):

$$\begin{cases} m\ddot{c} & = \sum_i f_i + mg & = \dot{l}_c \\ mc_\times(\ddot{c} - g) + \dot{L} & = \sum_i (p_i - c_i) \times f_i + \tau_i & = \dot{k}_c \end{cases} \quad (1)$$

79 with  $c, \dot{c}, \ddot{c}$  the CoM position  $c = [c_x, c_y, c_z]$ , velocity and acceleration,  $\dot{L} = \sum_k [R_k I_k \dot{w}_k - R_k (I_k w_k) \times w_k]$   
 80 and  $g = [0, 0, -9.81]^T$ , where  $R_k \in SO(3)$  is the 3d rotation matrix between the  $k^{th}$  body frame and the  
 81 inertial coordinate frame,  $I_k$  its inertial matrix,  $w_k$  its angular velocity,  $m$  is the mass of the robot,  $f_i \in \mathbb{R}^3$   
 82 the vector of contact forces at contact point  $i$ ,  $p_i \in \mathbb{R}^3$  their positions and  $\tau_i \in \mathbb{R}^3$  their contact torque

83 (represented at the inertial coordinate frame).  $l_c$  and  $k_c \in \mathbb{R}^3$  are the linear and angular momentum around

84 the CoM, and  $c_\times = \begin{pmatrix} 0 & -c_z & c_y \\ c_z & 0 & -c_x \\ -c_y & c_x & 0 \end{pmatrix}$ .

## 85 2.1 Divergent Component of Motion

86 We use the DCM formulation for the admittance control of the CoM. Under the assumptions of the LIPM,  
87 one can obtain the following set of equations Takenaka et al. (2009); Engelsberger et al. (2015):

$$\begin{aligned} \dot{c} &= \omega(\xi - c) \\ \dot{\xi} &= \omega(\xi - z) \\ \xi &= c + \frac{\dot{c}}{\omega} \end{aligned} \quad (2)$$

88 with  $z, \xi$  respectively the Zero Moment Point (ZMP) and DCM and  $\omega = \sqrt{g/c_z}$ . These equations show  
89 that the DCM is the divergent component of the LIPM model. Thus, the DCM needs to be controlled to  
90 stabilize the system Kajita et al. (2010); Sugihara (2009); Engelsberger et al. (2015); Mesesan et al. (2019).  
91 Caron et al. (2019) proposes to use a Proportional–Integral (PI) control on the DCM (the integral term is  
92 used to eliminate the steady-state error). Romualdi et al. (2019) proposes an asymptotical criteria but other  
93 techniques which guarantees stability can be used.

94 In terms of ZMP, the obtained control law is Caron et al. (2019):

$$\begin{aligned} z^* &= z^{ref} - \left[1 + \frac{kp_{dcm}}{\omega}\right](\xi^{ref} - \xi) \\ &\quad + \frac{kz_{dcm}}{\omega}(z^{ref} - z) - \frac{ki_{dcm}}{\omega} \int (\xi^{ref} - \xi) dt \end{aligned} \quad (3)$$

95 with  $z^{ref}, \xi^{ref}$  the respective ZMP and DCM reference values, given by the planning. Finally, this desired  
96 ZMP is used into a CoM admittance control as Caron et al. (2019):

$$\ddot{c}^* = \ddot{c}^{ref} + kp_{adm}(z - z^*) \quad (4)$$

97 The two position control schemes presented in this paper use this stabilization formulation. In the Fig.2,  
98 the Eq.3 is implemented in the *DCM Ctrl* blue block and the Eq.4 in the *CoM Admittance Ctrl* one. See  
99 Table 1 for the gains value used in the simulations.

## 100 2.2 Centroidal Momentum Tasks

101 The objective is to consider the angular momentum part of the Euler equation generated by the contact  
102 transition Kajita et al. (2003b). Using the equation Eq. 1, the centroidal dynamics is therefore defined by  
103  $h_c = [l_c \ k_c]^T \in \mathbb{R}^6$ . In Wensing and Orin (2013), the task formulation of the centroidal dynamics control  
104 is given by  $h_c = A_G(q)\dot{q}$  where  $q, \dot{q}$  are the joint position and velocity vectors of the robot and  $A_G$  is the  
105 Centroidal Momentum Matrix Orin et al. (2013).

106 The tasks dynamics are given by the following equations:

$$\begin{cases} \dot{l}_c &= m[\dot{c}^* + K_{Dcom}(\dot{c}^* - \dot{c}) + K_{Pcom}(c^* - c)] \\ \dot{k}_c &= \dot{k}_c^* + K_{Pam}(k_c^* - k_c) \end{cases} \quad (5)$$

107 The angular momentum task in TSID is expressed as in the equation Eq. 5, successfully implemented in  
 108 Lee and Goswami (2012) (the gains are defined in Table 1).

### 3 WHOLE-BODY CONTROLLER

#### 109 3.1 Lexicographic Quadratic Programming

110 The first controller used is a Lexicographic QP task-based inverse kinematics described in Mansard  
 111 et al. (2009). In this controller, the task errors  $e$  to be reduced in the cost function are implemented as  
 112 velocity-based tracking laws in the Lie group  $SE(3)$ . Having the robot configuration vector  $q$  and the joint  
 113 velocity  $\dot{q}$  as control input, a task-function is a derivable function  $x(q)$  whose space is named the task-space.  
 114 And the task errors  $e$  are expressed as:

$$\begin{aligned} \dot{e}(q, t) &= \dot{x}(q) - \dot{x}^*(t) \\ \dot{x}(q) &= J\dot{q} \end{aligned} \quad (6)$$

115 with  $J = \frac{\partial e}{\partial q} = \frac{\partial x}{\partial q}$  the Jacobian according to the robot state vector.

116 The following dynamics is imposed on these errors:

$$\begin{aligned} \dot{e}(q, t) &= K_P(x(q) \ominus x^*(q)) \\ \Leftrightarrow \dot{x}(q) &= \dot{x}^*(t) + K_P(x(q) \ominus x^*(q)) \end{aligned} \quad (7)$$

117 with  $\ominus$  the difference operator of Lie group.

118 *Inverse Kinematics QP: IK* - This control scheme is based on a DCM controller (Eq.3), a CoM admittance  
 119 controller (Eq.4) and a Lexicographic QP solving the inverse kinematics of the robot (see Fig. 2). The  
 120 authors have implemented this scheme in an open-source package GEPETTO Team LAAS-CNRS (2021c),  
 121 based on the QP in Mansard et al. (2009), adding the DCM and CoM admittance controllers.

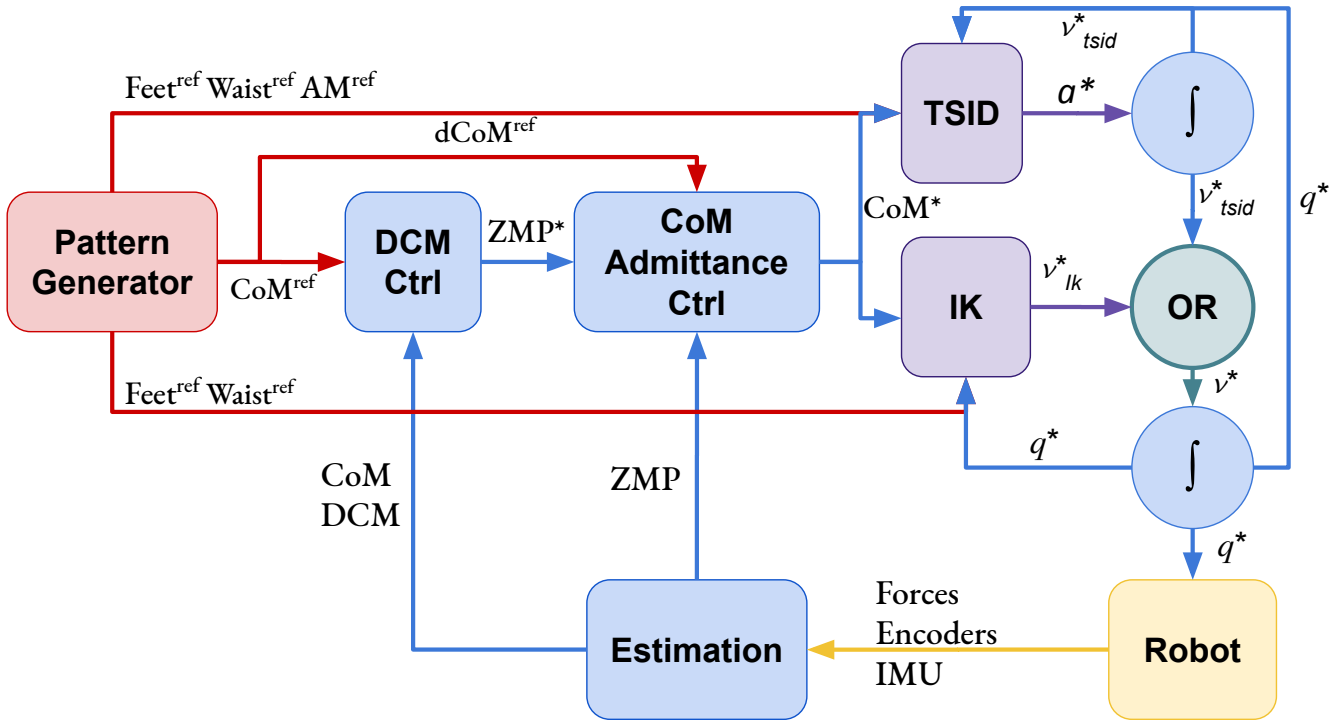
122 The tasks used during the simulations are (the priority 0 is the highest one) :

- 123 • Feet tracking (priority 0)
- 124 • CoM height tracking (priority I)
- 125 • CoM lateral-sagittal tracking (priority II)
- 126 • Waist orientation (priority III)
- 127 • Posture regularization in half-sitting (priority IV)

128 The respective task gains are defined in Table 1.

#### 129 3.2 Task Space Inverse Dynamics (TSID)

130 TSID Del Prete (2021) is a WQP which sums the task functions in a general cost function using weights  
 131 to define their priorities (as opposed to the *IK* controller it is not a strict hierarchy, it has only two strict  
 132 layers: the constraint and the cost). In this controller, the task errors  $e$  to be reduced are implemented as  
 133 acceleration-based tracking laws in the task space. Having the robot configuration vector  $q$  and the joint  
 134 acceleration  $\ddot{q}$  as control input, a task-function is a second-order derivable function  $x$  of  $q$ . And the task



**Figure 2.** Position control schemes: IK and TSID. The *OR* block is used to activate only one controller at a time.

135 errors  $e$  are expressed as:

$$\begin{aligned} \ddot{e}(q, t) &= \ddot{x}(q) - \ddot{x}^*(t) \\ \dot{e}(q, t) &= (J\dot{q} + \dot{J}q) - \dot{x}^*(t) \end{aligned} \quad (8)$$

136 The following dynamics is imposed on these errors:

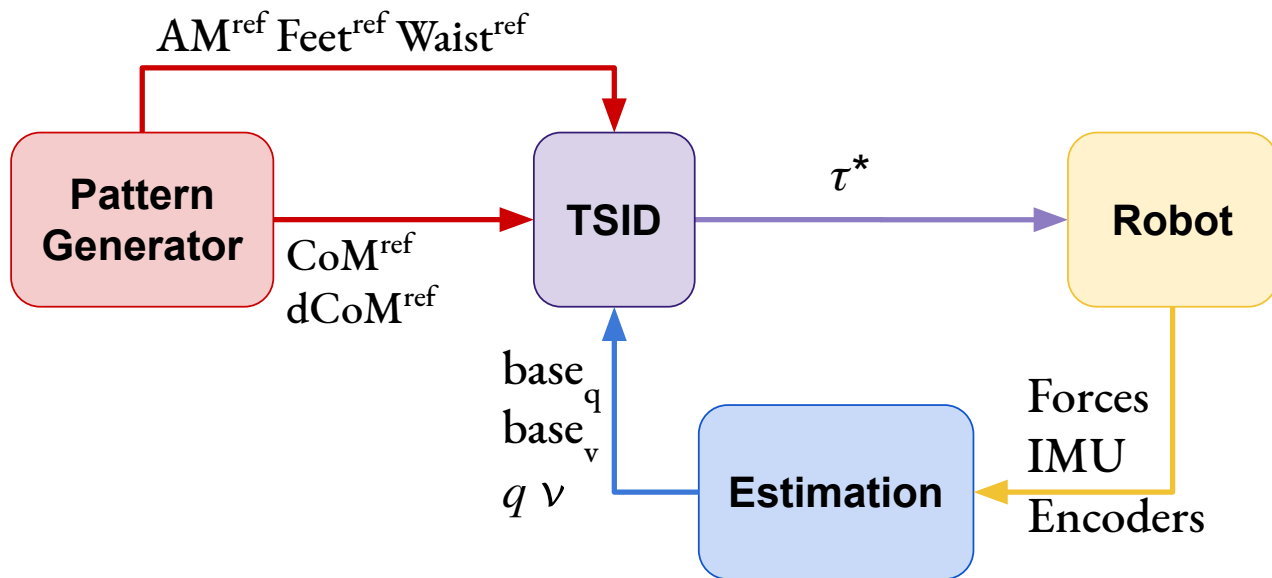
$$\begin{aligned} \ddot{e}(q, t) &= K_P(x(q) \ominus x^*) + K_D(\dot{x}(q) - \dot{x}^*(t)) \\ \Leftrightarrow \ddot{x}(q) &= \ddot{x}^*(t) + K_P(x(q) \ominus x^*(t)) + \\ &K_D(\dot{x}(q) - \dot{x}^*(t)) \end{aligned} \quad (9)$$

137 TSID solves the inverse dynamics of the robot in rigid contact with the environment Herzog et al. (2014)  
138 and has been successfully used on HRP-2 robot in Del Prete et al. (2016).

139 *Inverse Dynamics WQP: TSID Position* - This control scheme is based on a DCM controller (Eq.3), a CoM  
140 admittance controller (Eq.4) and a WQP solving the inverse dynamics of the robot, see Fig. 2. Compared  
141 to the previous controller, this one implements an AM task, which regulates the angular momentum to 0,  
142 using the formulation of Eq.5. The authors have implemented this controller using the TSID Del Prete  
143 (2021) library in the same package than the controller *TSID Torque*, with the DCM and CoM admittance  
144 controllers.

145 The tasks considered during the simulations are:

- 146 • Feet tracking (priority 0)
- 147 • Feet contacts (priority 0)
- 148 • CoM height tracking (priority I, weight  $10^3$ )



**Figure 3.** TSID torque control scheme.

- 149                   • CoM lateral-sagittal tracking (priority I, weight  $10^3$ )  
 150                   • Waist orientation (priority I, weight 1)  
 151                   • Posture regularization in half-sitting (priority I, weight 0.1)  
 152                   • AM velocity-acceleration regularization (priority I, weight  $2 \times 10^{-2}$ )

153 The respective task gains are defined in Table 1. The weights and gains have been chosen through trials and  
 154 errors with an a priori heuristic.

155 *Inverse Dynamics WQP: TSID Torque* - This control scheme is based on a WQP solving the inverse  
 156 dynamics of the robot (with an AM regularization task, using the formulation of Eq.5), as shown in Fig. 3.  
 157 From the desired acceleration computed by the QP, TSID retrieves the associated torque by using the robot  
 158 equation of the dynamics. The authors have implemented this controller using the TSID Del Prete (2021)  
 159 library in the open-source package GEPETTO Team LAAS-CNRS (2020).

160 The tasks considered in the simulations are the same as *TSID position*, with different gains (see Table 1).

### 161 3.3 Remark on the state feedback

162 For position control, it is needed to integrate the result of the QP (one time for *IK* and two times for *TSID*  
 163 *position*, see Fig.2) to obtain the desired command. To avoid instabilities, the control loop of both QP  
 164 use these integrated values in the next iteration instead of the measured ones. The measured position and  
 165 velocity of the robot are only used to compute the CoM, DCM and ZMP for the admittance control in the  
 166 position schemes. In contrary, the torque control scheme uses the measured values at each iteration of the  
 167 QP (see Fig.3) and in particular the position and velocity of the robot base (or free-flyer).



## 4 ENERGETIC COMPARISON CRITERION

### 168 4.1 Energy cost

169 Based on Torricelli et al. (2015), a relevant criteria to compare the energy consumption of the control  
 170 schemes is the cost of transport. It can be computed as the energetic cost of transport  $C_{et}$  using the whole  
 171 mechanical work of the actuation system  $E_m$  or as the mechanical cost of transport  $C_{mt}$  using only the  
 172 positive one  $E_{m+}$ .

$$C_{et} = \frac{E_m}{mgD}, \quad C_{mt} = \frac{E_{m+}}{mgD}, \quad E_m = \int_0^T \sum_{i=0}^N |\tau_i(t) \omega_i(t)| dt, \quad E_{m+} = \int_0^T \sum_{i=0}^N \varrho_i(t) dt, \quad \text{if } \varrho_i(t) > 0 \quad (10)$$

173 with  $m$  the mass of the system,  $g$  the gravity constant,  $D$  the distance traveled by the system and  $\tau_i, \omega_i$  the  
 174 respective torque and velocity of each robot joint for all ( $N$ ) joints, and  $\varrho_i(t) = \tau_i(t) \omega_i(t)$ .

### 175 4.2 Passivity Gait Measure

Another interesting energetic criteria is the ability to minimize joint torques to increase the passivity of the walk Torricelli et al. (2015). The Passivity Gait Measure (PGM) Mummolo and Kim (2012) quantifies the passivity of a biped walking motion:

$$PGM = 1 - \frac{RMS(\tau_{sa})}{RMS(\tau_{tot})} \quad (11)$$

$$RMS(\tau_{tot}) = \sqrt{\frac{\int_0^T \left[ \sum_{i=0}^N \tau_i(t)^2 \right] dt}{T}} \quad (12)$$

176 where  $RMS$  is the Root Mean Square along the period of time  $T$ ,  $\tau_{sa}$  stands for the torque on the stance  
 177 ankle joint and  $\tau_{tot}$  for the torque on all robot joints.

## 5 LOCOMOTION PLANNING

### 178 5.1 Walking Pattern Generator

179 The trajectories used in the straight walk simulations have been computed using the algorithm described  
 180 in Kajita et al. (2003a); Stasse et al. (2008); GEPETTO Team LAAS-CNRS (2021a). This algorithm  
 181 provides desired trajectories for the ZMP  $z^*$ , the CoM  $c^*$ , and the feet  $p_i^*$ , for a given set of foot steps  
 182 (pre-defined in these simulations). This implementation uses the centroidal dynamics and the dynamic  
 183 filter proposed in Kajita et al. (2003a) computed with the Recursive Newton-Euler Algorithm Featherstone  
 184 (2008) implemented in the Pinocchio library Carpentier et al. (2019). The CoM trajectory is modified to  
 185 take into account the momentum generated by the limbs motion. The desired DCM  $\xi^*$  is deduced from the  
 186 desired CoM  $c^*$  and desired ZMP  $z^*$  trajectories (see Eq.2).

### 187 5.2 Multi-contact-locomotion-planning

188 The trajectories used in the tilted platforms and stairs simulations have been computed using the open-  
 189 source framework *multi-contact-locomotion-planning* GEPETTO Team LAAS-CNRS (2021b). Given the

Tasks Gains	IK (20cm stairs)	TSID position (20cm stairs)	TSID torque (20-60cm stairs)
$K_{Pcom}$	100	1000	20 12
$K_{Dcom}$	-	300	3
$K_{Pcom.H}$	100	1000	-
$K_{Dcom.H}$	-	300	-
$K_{Pwaist}$	300	100	100
$K_{Dwaist}$	-	20	20
$K_{Pcontacts}$	1000	30	30-100 30
$K_{Dcontacts}$	-	11	11-0 11
$K_{Pfeet}$	1000	2000	1200 500
$K_{Dfeet}$	-	20	12
$K_{Pam}$	-	10	10
$K_{Pposture}$	100	see below	see below
$K_{Dposture}$	-	$2\sqrt{K_{Pposture}}$	$2\sqrt{K_{Pposture}}$
$K_{Pcom.Adm}$	15 45	12	-
$K_{Pdcm}$	8 25	8	-
$K_{1dcm}$	1	1	-
$K_{zdcm}$	1	1	-
<b>TSID Gains</b>		<b>Legs</b>	<b>Torso</b>
$K_{Pposture}$		[10, 5, 5, 1, 10, 10]	[100, 100]
		<b>Arms</b>	<b>Head</b>
$K_{Pposture}$		[50, 10, 10, 10, 50, 10, 10, 10]	[100, 100]

**Table 1.** Tasks gains of the control schemes. *tilted platforms* and *stairs* simulations use the same gains.

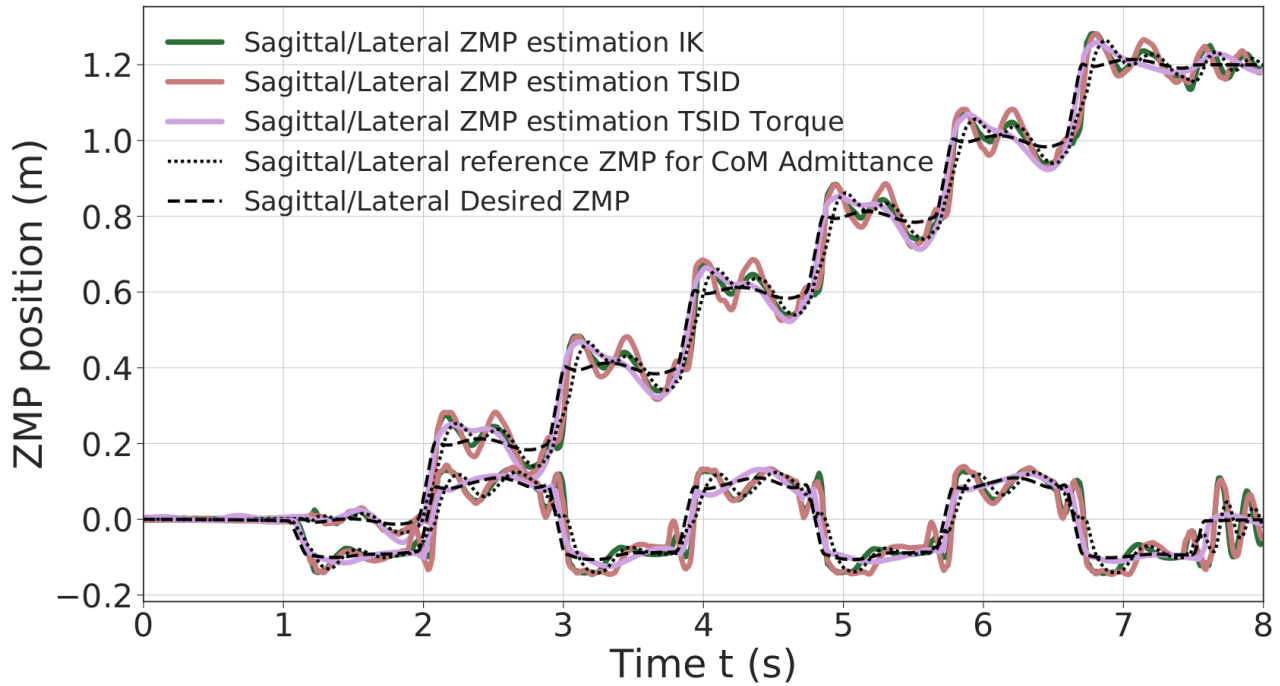
190 initial and final poses of the robot, the framework computes a reachability plan and a contacts sequence  
 191 as in Tonneau et al. (2020). Then it optimizes the centroidal dynamics (see Section 2) using two convex  
 192 relaxations based on trust regions Ponton et al. (2018). Similarly to the pattern generator method, it  
 193 takes into account the momentum generated by the swing leg owing to iterations between a kinematic  
 194 whole-body formulation and the centroidal dynamic optimization. In contrast, when solving Eq. 1, it does  
 195 not assume that  $\dot{L} = 0$  (see Section 2).

## 6 SIMULATION RESULTS

196 The simulations realized in this paper have been made using Gazebo. A video illustrating the  
 197 simulations is available at the following link: <https://peertube.laas.fr/videos/watch/4b5d3a5b-2355-47a0-8197-f41ed4f885c6>. The chosen simulations are walking on flat or uneven terrains and stair climbing. Based  
 198 on Torricelli et al. (2015), they cover different aspects of locomotion skills for a stationary environment  
 199 with and without unexpected disturbances.  
 200

### 201 6.1 Straight walk of 20 cm steps

202 In the simulation, the robot executes 6 steps forward at 0.2m/s and a final step (traveled distance of 1.2m).  
 203 The time distribution is 0.9s for single support phase and 0.115s for double support phase (leading to steps  
 204 of approx. 0.20m). The controllers have also been successfully tested on a faster walk with single/double  
 205 support time of 0.711/0.089s. The Fig. 4 presents a comparison of the three control schemes on their  
 206 estimated ZMP, on the sagittal (x-axis, top curves on the figure) and lateral (y-axis, bottom curves) planes  
 207 only, because the desired height of the CoM is constant. Fig. 5 shows the forces applied on the ground



**Figure 4.** ZMP estimation of the 20 cm step walk.

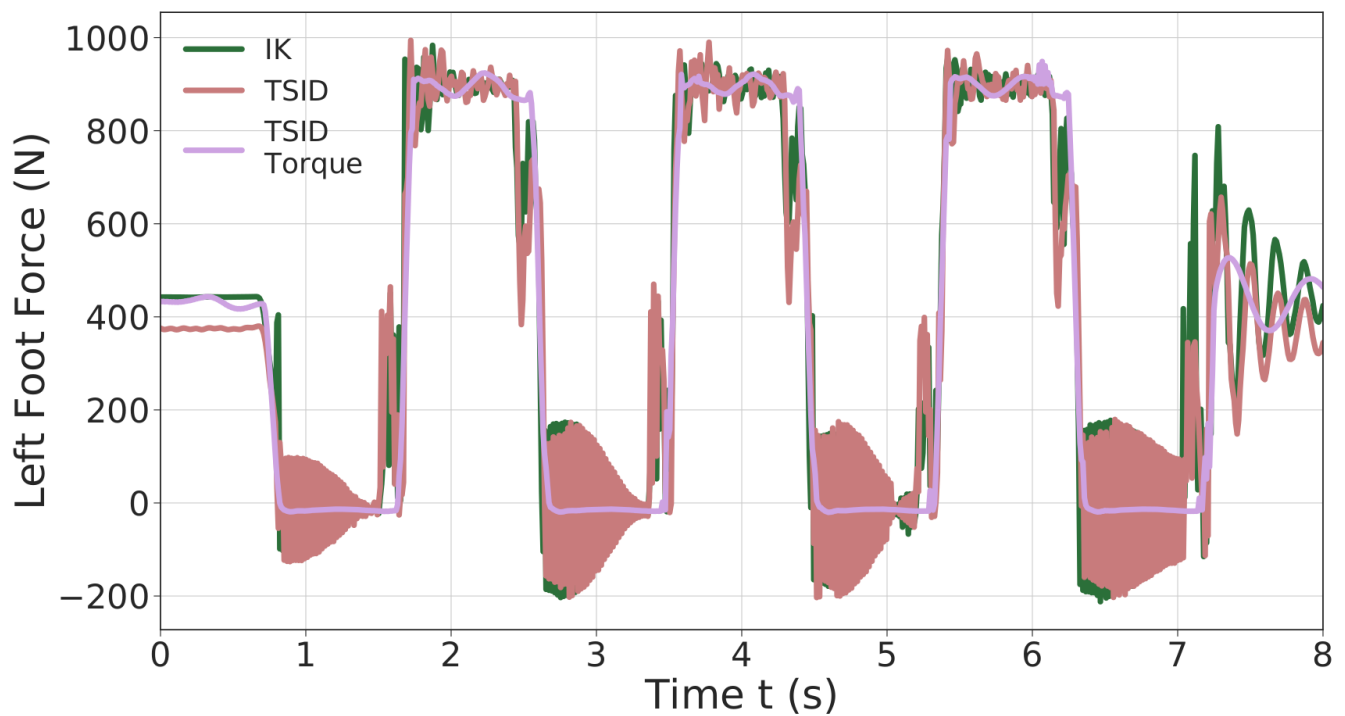
Control Scheme	Axis	Average	Standard deviation	Peaks
<i>IK</i>	x-axis	0.019m	0.022m	0.131m
	y-axis	0.022m	0.026m	0.150m
<i>TSID position</i>	x-axis	0.028m	0.025m	0.142m
	y-axis	0.025	0.027m	0.138m
<i>TSID torque</i>	x-axis	0.026m	0.021	0.078m
	y-axis	0.011m	0.014m	0.078m

**Table 2.** ZMP error of the 20 cm step walk simulation.

208 along the z-axis on the left foot. The tracking of the CoM and the feet are accurately followed by the three  
 209 controllers. The CoM tracking error is less than 1cm for a tolerance of up to 2cm due to the feet contact  
 210 surface. The tracking error depends on the force sensor error which is not handled in the present work.

211 The two position controllers achieve similar results, tracking correctly the ZMP reference of Eq. 3, with  
 212 an average error of 2cm (see Table 2). Noticeably, the torque control presents a ZMP which is close to  
 213 the position control results in Fig. 4 even though there is no explicit control on the ZMP nor the DCM. In  
 214 the Tables presenting the error on the ZMP, for the torque scheme, the estimated ZMP is compared to the  
 215 desired ZMP (from the planning). In particular, in the lateral plane, the error is quite low, 1cm in average.

216 The Fig. 5 illustrates the ground impacts problem in position control compared to the better foot landing  
 217 observed in torque control. Indeed, each time the left foot comes into contact with the ground (1.5s, 3.5s,...),  
 218 the *IK* and *TSID position* schemes show peaks in the foot force ( $\sim 400\text{N}$ ) which are avoided in *TSID*  
 219 *torque*. This explains also the peaks in the ZMP errors (around 15cm) because during an impact the foot  
 220 bounces on the ground. The force oscillations of the *IK* and *TSID position* controllers when the foot is in  
 221 the air are due to the high control gains on the ankle (Proportional–Integral–Derivative (PID) gains of the  
 222 low-level position control in Gazebo), it is mainly noises.



**Figure 5.** Z-axis left foot force of the 20 cm step walk.

	Axis	Average	Standard deviation	Peaks
CoM	x-axis	0.018m	0.013m	0.050m
	y-axis	0.004m	0.003m	0.015m
Left Foot	x-axis	0.014m	0.013m	0.063m
	y-axis	0.001	0.001m	0.005m
Right Foot	x-axis	0.016m	0.016	0.063m
	y-axis	0.001m	0.001m	0.006m

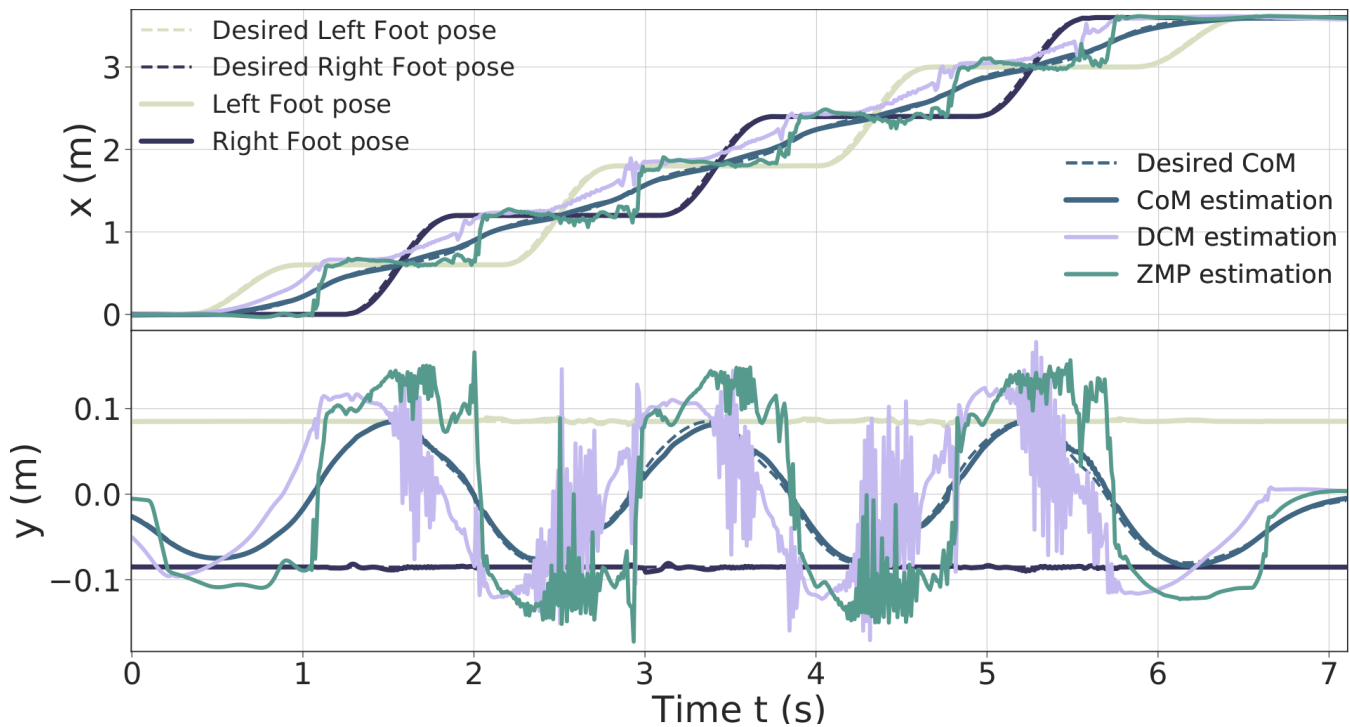
**Table 3.** CoM and Feet error of the 60 cm step walk.

## 223 6.2 Straight walk of 60 cm steps in torque control

224 In Mesesan et al. (2019) the humanoid robot TORO successfully performed a walk on flat terrain with  
 225 a step length of 55cm (single/double support time of 1.1/0.4s). In the following simulation, the torque  
 226 controller is pushed to its limits to show its capability to achieve a similar result. The robot TALOS executes  
 227 6 steps forward of 0.6m/s and a final one to go back to the initial position. The time distribution used is of  
 228 0.9s for single support phase and 0.115s for double support phase (leading to steps of approx. 60cm).

229 Figure 6 presents the results obtained on the tracking of the feet and the CoM (see Table 3); the ZMP  
 230 and DCM estimations. The feet tracks well the desired trajectories along the y-axis (maximum error of  
 231 6mm) however, along the x-axis, they show some delay (maximum error of 6cm). Thus, it induces greater  
 232 tracking errors on the x-axis for the CoM (peaks of 5cm along the x-axis and 1.5cm along the y-axis).

233 One can notice that the DCM and ZMP along the x-axis are more stable, whereas along the y-axis they  
 234 present large oscillations (which are caused by the feet impacts on the ground when landing).



**Figure 6.** Feet, CoM, DCM and ZMP of the 60 cm step walk.

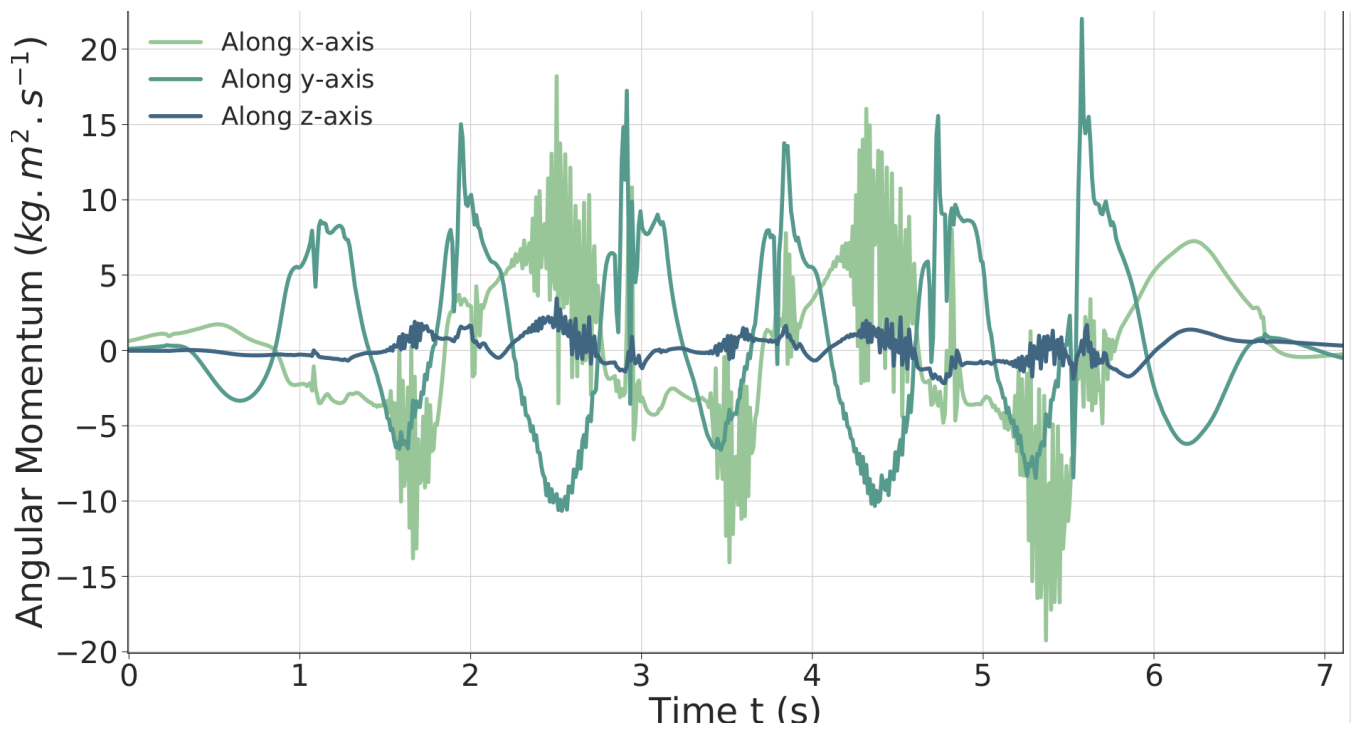
235 In Fig. 7, the AM behavior is shown along the three axes. The AM task minimizes the momentum to zero.  
 236 The  $x$  and  $y$  momentum components are the most solicited, leading to the inclination of the torso forward  
 237 and backward and to important moves of the arms to compensate the delay of the CoM and succeed the  
 238 60cm steps. The authors observed that without this AM task, the walk cannot be achieved.

### 239 6.3 Walk on the tilted platforms: Uneven terrain

240 In this third simulation, the robot walks on tilted platforms which represent uneven terrain (Fig. 1). This  
 241 walk is achieved using the *multi-contact-locomotion-planning* trajectories (see Section 5.2). The framework  
 242 ensures the stability of the controllers on non-flat terrain when the feet are tilted.

243 Figure 8 illustrates the tracking performance of the controllers. The ones in position present the largest  
 244 oscillations as *TSID torque* is the most stable (see Table 4). Both the *IK* and the torque control show  
 245 oscillations at  $t \approx 18$ s; it corresponds to the worst case where the robot has its two feet tilted to keep its  
 246 balance on two opposite platforms leading to small slippages of the feet (this behavior can be observed  
 247 in the linked video). These oscillations are larger in the case of the *IK* scheme. Similar oscillations on  
 248 the contact forces in this part of the motion have also been observed, which are smaller in the case of the  
 249 torque control. Increasing the gains on the feet only generates more instability, but raising the ones on the  
 250 DCM and admittance control lessen the oscillations (at the cost of a more rigid behavior).

251 Finally the same result on the feet forces is obtained in this simulation with respect to the 20cm steps one.  
 252 Due to the high gains on the DCM, to avoid the slippage of the robot, the *IK* control presents bigger peaks  
 253 of force.



**Figure 7.** AM behaviour during the 60 cm step walk in torque.

Control Scheme	Axis	Average	Standard deviation	Peaks
<i>IK</i>	x-axis	0.021m	0.024m	0.278m
	y-axis	0.016m	0.018m	0.118m
<i>TSID</i> <i>position</i>	x-axis	0.012m	0.017m	0.197m
	y-axis	0.015	0.019m	0.127m
<i>TSID</i> <i>torque</i>	x-axis	0.013m	0.021	0.107m
	y-axis	0.005m	0.006m	0.058m

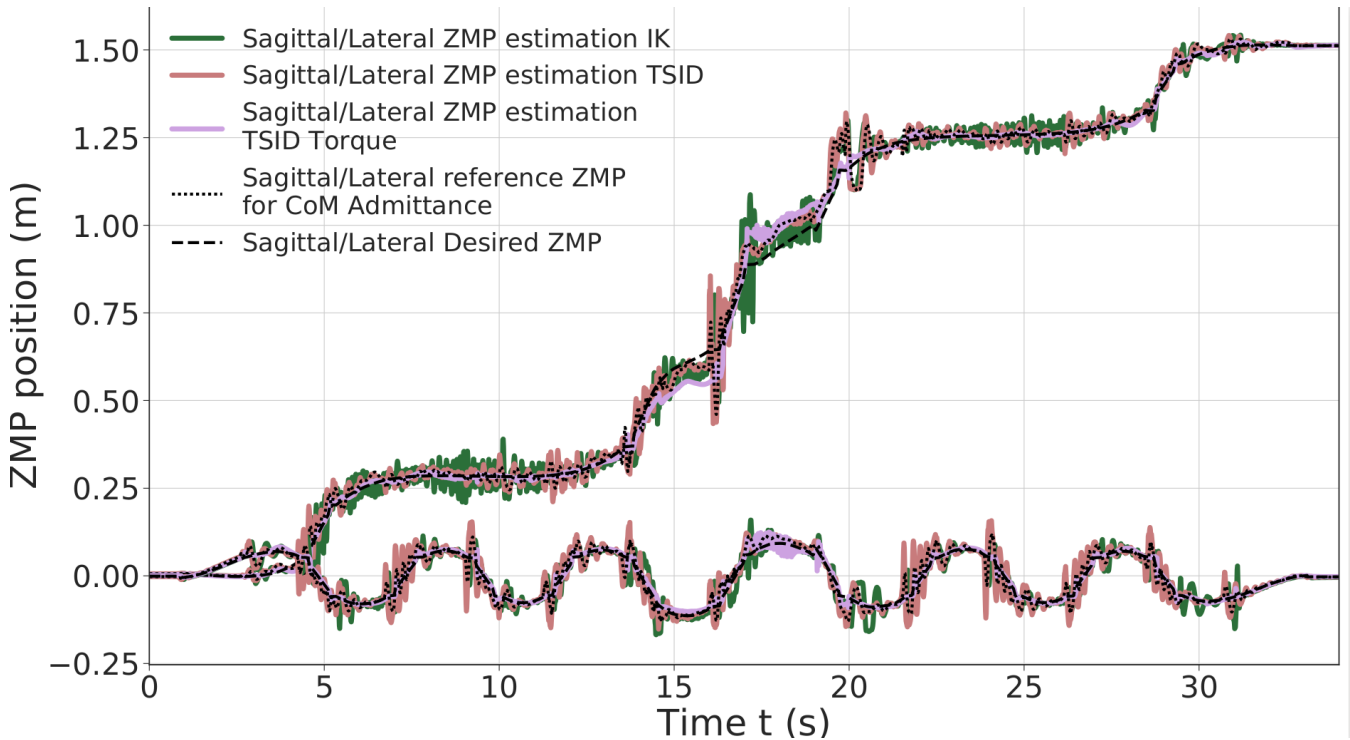
**Table 4.** ZMP error of the *tilted platforms* simulation.

## 254 6.4 Climbing Stairs

255 In the last simulation the robot is climbing 6 stairs of 10cm height and 30cm long (see Fig. 1). The  
 256 trajectories are planned with the *multi-contact-locomotion-planning*. Fig. 9 shows the ZMP evolution of  
 257 each controllers, where the result is similar to the *uneven terrain* simulation. The *TSID torque* scheme  
 258 behave significantly better than the others, with a ZMP matching the one planned (errors lesser than 1cm,  
 259 see Table 5). Noticeably, the *IK* scheme presents higher oscillations at the end of the move in the lateral  
 260 plane. The robot ends displaced on the right compared to the desired trajectories, due to slippages of the  
 261 feet when it finishes to climb a stair (shown in the linked video).

## 262 6.5 Energy cost and Passivity Gait Measure

263 The results obtained for the cost of transport of the four simulations are presented in the Table 6,  
 264 depending on the control scheme. The results obtained for iCub in Romualdi et al. (2019) are also presented  
 265 for comparison (computed using Eq.10), as the human ones. The lower the energy consumption is, the  
 266 better, and similarly, getting closer to the human cost of transport is an improvement.



**Figure 8.** ZMP estimation of the *tilted platforms* simulation.

Control Scheme	Axis	Average	Standard deviation	Peaks
<i>IK</i>	x-axis	0.022m	0.026m	0.257m
	y-axis	0.015m	0.017m	0.151m
<i>TSID position</i>	x-axis	0.009m	0.013m	0.151m
	y-axis	0.012	0.015m	0.119m
<i>TSID torque</i>	x-axis	0.008m	0.006	0.049m
	y-axis	0.006m	0.005m	0.047m

**Table 5.** ZMP error of the *stairs* simulation.

267 Compared to the results obtained on iCub, the control in torque has a similar cost for the 20cm steps  
 268 simulation. However, the cost of the position controllers presented in this paper is higher, because of their  
 269 higher gains. The human efficiency is closer to the torque control, walking with a  $C_{et}$  around  $0.2J/kg/m$   
 270 Collins et al. (2005). Noticeably, the energy costs in torque for the *tilted platforms* and *stairs* trajectories  
 271 are still less important than the simpler walk in position; the  $C_{mt}$  never exceeds 1, even for the 60 cm walk.  
 272 Overall, the controller *TSID position* consumes less energy than the *IK*.

273 The Passivity Gait Measure comparison of the different simulations is reported in Table 7 for three gait  
 274 stages: Single Support (Single S. corresponding to the stance ankle), Double Support (Double S.) and  
 275 Flying Foot (Flying F. where the foot has no contact with the ground). The human results is given as an  
 276 indicator Mummolo and Kim (2012), the robot behavior is expected to be similar during double support  
 277 and flying foot phase where the ankle should be passive.

278 The results of the position control schemes show a behavior which is the opposite of the human one. The  
 279 passivity of the ankle is higher during the stance phase because of the control of the ZMP which minimizes

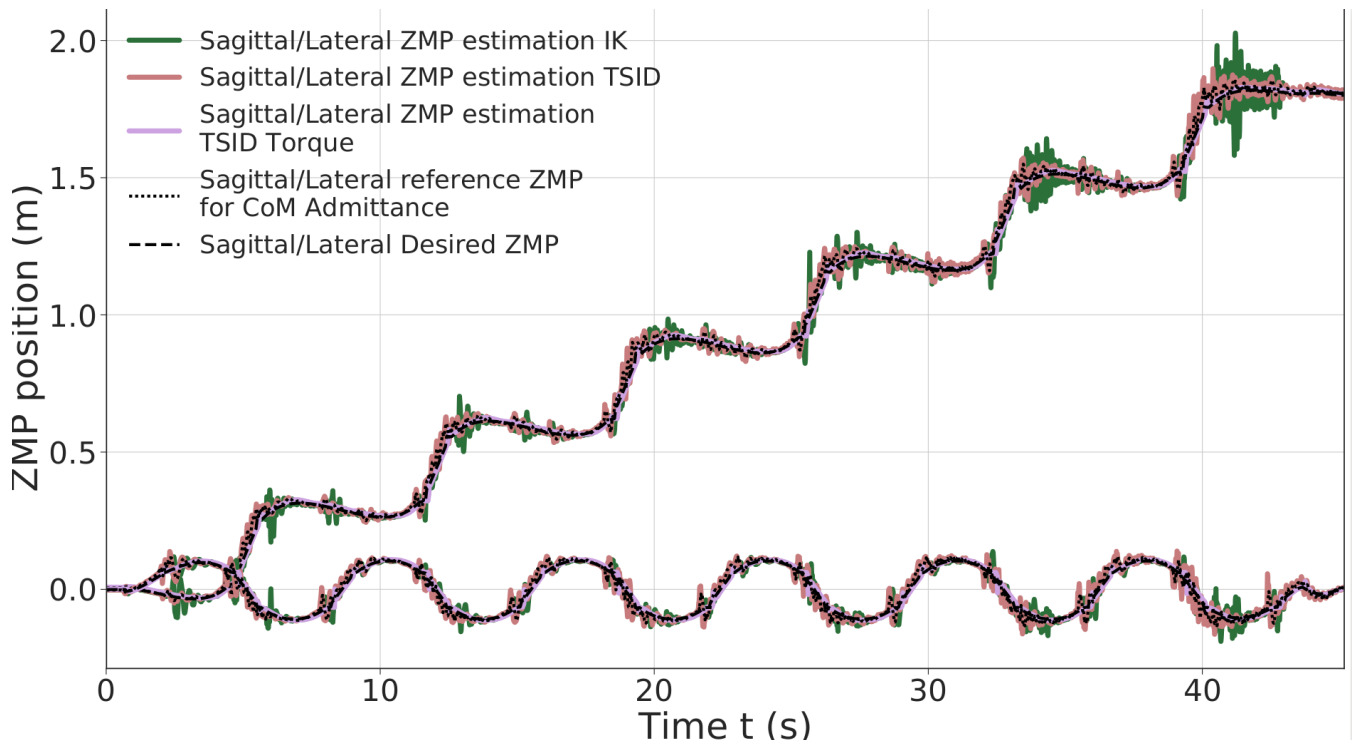


Figure 9. ZMP estimation of stairs climbing.

Control Scheme	Simulation	$E_m$ [J]	$E_{m+}$ [J]	$C_{et}$ [J/kg/m]	$C_{mt}$ [J/kg/m]
Human	-	-	-	0.2	0.05
<i>iCub</i>					
position	20cm	-	-	-	0.49
torque	20cm	-	-	-	0.26
<i>IK</i>	20cm	1983.9	1359.3	1.68	1.15
	platforms	5418.7	3769.2	3.7	2.6
	stairs	7249.5	2145.3	4.1	1.2
<i>TSID</i>	20cm	2324.5	764.1	1.97	0.65
position	platforms	5377.5	1413.6	3.6	2.0
	stairs	6812.6	2059.6	3.8	1.2
<i>TSID</i>	20cm	521.8	259.3	0.44	0.22
torque	60cm	3147.2	1583.8	0.89	0.45
	platforms	1378.6	668.5	0.93	0.45
	stairs	1861.1	1205.5	1.1	0.68

Table 6. Results of the specific cost of transport.

280 the ankle torque. And it is weaker during the double support and flying phases, due to the high PID gains  
 281 of the low-level position control.

282 The control scheme in torque shows much more passive behavior (except on the stance foot), with a  
 283 completely passive foot during the flying phase. During the double support phase, the ankle is almost  
 284 passive ( $PGM \sim 0.9$ ) which is close to the human result. These results are better than the one expected  
 285 in Mummolo and Kim (2012), where the torque controlled robot has a higher control on its stance ankle  
 286 ( $PGM = 0.2$ ).



	Simulation	Double S.	Single S.	Flying F.
<i>Human</i>	50cm	1.0	0.6	$\sim 1.0$
	20cm	0.35	0.89	0.24
<i>IK</i>	platforms	0.27	0.85	0.31
	stairs	0.46	0.86	0.36
<i>TSID</i>	20cm	0.37	0.74	0.37
	platforms	0.27	0.86	0.30
<i>position</i>	stairs	0.55	0.86	0.34
	20cm	0.93	0.87	1.0
<i>TSID</i>	60cm	0.87	0.79	1.0
	platforms	0.87	0.8	0.91
<i>torque</i>	stairs	0.97	0.89	1.0

**Table 7.** Results of the PGM on three gait stages.

Control Scheme	Simulation	20cm (60cm)	Platforms	Stairs
<i>IK</i>	Average	0.5ms	0.7ms	0.6ms
	Peaks	2ms	4ms	4ms
<i>TSID</i>	Average	1.2ms	1.2ms	1.2ms
	Peaks	4.5ms	4.3ms	4.2ms
<i>TSID</i>	Average	1ms (1.4ms)	1.2ms	1.1ms
	Peaks	2.8ms (6ms)	5ms	5.5ms

**Table 8.** Comparison of the execution time.

287 Finally, on the *uneven terrain*, the double support phase corresponds to the worst case where the robot  
 288 has its two feet tilted to keep its balance on two opposite platforms. This leads to a greater actuation than  
 289 on flat floor (decreasing the passivity). Similarly, the stance phase corresponds to the left support phase on  
 290 the final platform (highest slope), also leading to a bigger actuation of the ankle.

## 291 6.6 Execution time of the control schemes

292 The computational time obtained during the execution of one control loop of the three schemes are  
 293 presented in Table 8, according to the simulations.

294 The computational time of the *IK* is better due to the computational efficiency of the null space projectors  
 295 of the tasks. Exploiting this specific structure allows it to keep its control frequency higher than 1kHz in  
 296 average with 4 hierarchy levels. In *TSID* this method can only be used once because it is composed of two  
 297 strict layers: the constraints and the cost.

## 7 EXPERIMENTS REALIZED ON THE REAL ROBOT USING THE CONTROLLERS

298 In this section are presented the results we succeed to achieve on the real robot TALOS and the difficulties  
 299 we encountered. These experiments are intermediate steps towards transferring the whole simulated results  
 300 on the real robot. We detail the blocking points preventing us to successfully achieve these complete  
 301 experiments.

### 302 7.1 TALOS robot

303 Our robot TALOS is an humanoid robot of 1.75m tall and about 100kg, composed of 32 joints and  
 304 an under-actuated part called floating-base (38 Degrees-of-Freedom in total). It provides the possibility

305 to control the actuators in position control and torque control modes. It is performed owing to torque  
306 sensors on all the actuators but the head and the wrists. Humanoid robots often have flexible or compliant  
307 components. For instance, the actuators stiffness of the robot WALKMAN Negrello et al. (2017) can be  
308 directly tuned, creating an intended flexibility. Another example of humanoid robot with compliant material  
309 is HRP-2 Nakaoka et al. (2007). It includes a bush rubber in the ankle in order to smooth impacts. In the  
310 robot TALOS, a non-intended flexibility on the hip link has been observed and impacts meaningfully the  
311 control of its legs and, therefore, its balance and locomotion. Indeed, this flexibility (not modeled in the  
312 simulator) leads to errors in the landing positions of the feet on the real robot. However, the deflection is  
313 not directly measurable by the encoders and cannot be directly modified.

## 314 7.2 Position Control

### 315 7.2.1 Static stabilization

316 Using the whole body admittance control and the stabilizer described as the *IK* scheme in the Section.3,  
317 the team achieved good results for balancing during quasi-static moves and standing position. Indeed,  
318 the admittance control at the CoM allows a quick reaction when applying external perturbations such as  
319 pushing the robot. Fig.10 presents the reactive balancing of the TALOS humanoid robot when it is pushed  
320 from the front and from the side while standing on one foot. A video about this experiment is available at  
321 the following link: <https://peertube.laas.fr/videos/watch/2dec7dba-cc57-4df4-8f10-a7d387404301>

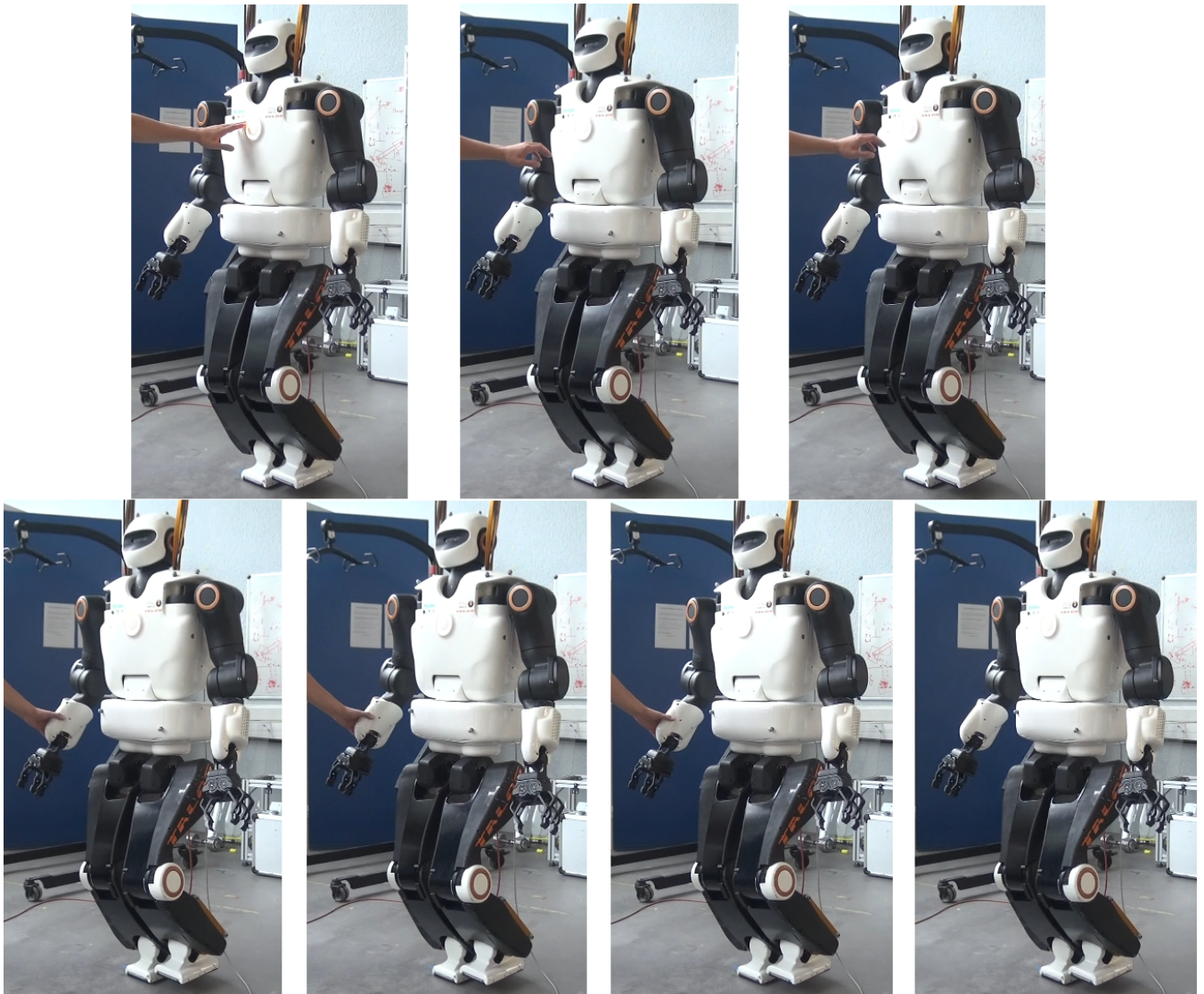
322 In the video is shown push-recovery experiments while the robot is standing on both feet and with one  
323 foot raised. One can notice that the robot is more stable with both feet on the ground, nonetheless, the *IK*  
324 scheme allows a good stabilization at the CoM level. The stabilizer correctly achieves the balance of the  
325 robot: it controls the DCM such that the CoM does not diverge and applies correct contact wrenches to  
326 avoid falling (no slipping, not too much forces on one foot which imbalance the robot). It is important to  
327 underline that the admittance control is only implemented on the CoM, thus the robot is stiff on its upper  
328 parts while more compliant on its lower parts (in particular the hips and ankles). This is why in the video  
329 pushing the robot arm produce motions on the whole robot and in particular its CoM.

330 The robot can achieve tasks with its upper body while external perturbations occur and keep its balance.  
331 It can also stabilize itself when non-dynamic trajectories are asked to the legs, or with no contact with the  
332 ground (for instance execute a swing on its foot). The difficulties appear when dynamic tasks are asked and  
333 involve the creation of contacts with the ground, typically during walking.

### 334 7.2.2 Dynamic stabilization

335 The dynamic stabilization of the robot is an ongoing work. The actual implementation of the stabilizer  
336 should allow the robot to achieve this goal, however this is compromised by the flexibility in the hip of  
337 the robot TALOS. By tuning the gains of the admittance controller, the team manages to achieve once a  
338 straight walk of 20cm using a WPG reference trajectories. The video of this success is available at the  
339 following link: <https://peertube.laas.fr/videos/watch/b56d80ed-7c6c-46a7-8750-fdb7ea6d1636>

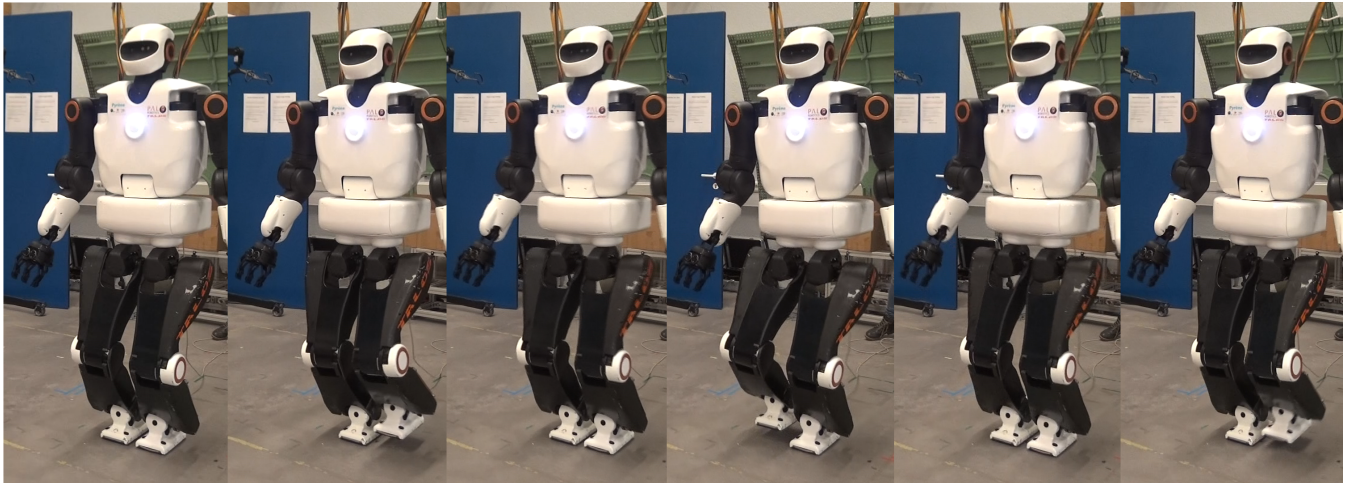
340 Later on, we successfully achieved a repeatable on spot walking which is quite stable  
341 (See Fig.11). The video of this on spot walking is available at the following link:  
342 <https://peertube.laas.fr/videos/watch/1a920902-c75f-4fb0-a638-33bb9b48d649>. One can notice that the  
343 left wrist of the robot is tilted, indeed, its absolute and relative encoders did not send the same value. Thus,  
344 when controlling its position, the wrist had an abnormal behavior as its returned position was not the good  
345 one. We had to deactivate its control for the experiment and then fix the offset of the relative encoder.



**Figure 10.** Experiments - Push recovery of the TALOS robot with one foot raised.

346 In both videos the impacts on the ground are large and lead to instabilities, in particular slippage (which  
 347 can also be caused by the flexibility in the hip). The robot has to move its upper body to compensate for  
 348 them, because of that, at the end of the 20cm walk the robot almost fall. These impacts are partly due to the  
 349 wrong positions of the feet when making contacts with the ground. The flexibility in each hip of the robot  
 350 cannot be measured by the encoders, then it is creating an error between the positions given by the encoders  
 351 and the real ones. These displacements at the hips are small, but transferred at the feet positions it can lead  
 352 to errors of up to 5cm. Thus, the controller is assuming a false position of the feet, and the robot enters  
 353 in contact with the ground at a wrong position (even at the wrong moment, sooner if the displacement is  
 354 in the direction of the walk or later in the opposite case). This is creating the large impacts and slippage,  
 355 which prevents us to achieve a successful walking, this is why compensating this flexibility is necessary. In  
 356 the next subsection is presented the experiment realized to compensate it with a fixed value.

357 An additional way to cope with the stabilization problem would be to reschedule the footsteps and their  
 358 location according the landing time.



**Figure 11.** Experiments - On spot walking with the TALOS robot.

### 359 7.2.3 Fixed compensation of the flexibility

360 We first try to compensate the flexibility by using a feed-forward on the commanded position of the  
361 hip taking into account the torsional stiffness and the measured torque. However, because of the noises  
362 on the torque sensors, we had to filter it which lead the compensation to be applied with delay. We also  
363 tried to activate this compensation only on single support phases and not on double support ones to avoid  
364 accumulation of internal efforts (on double support the robot will try to correct its hip position while having  
365 its feet in contact with the ground and thus not moving, leading to this accumulation of energy). Even with  
366 such modifications the results were not enough to successfully perform repeatable walk.

367 Thus, we then tried to impose a fix compensation of the flexibility without taking into account the  
368 measured torque. With a leg of 1m weighting 20kg, we fixed the compensation on the hip to  $\Delta q^{hip} = \frac{20}{K} \approx$   
369  $\frac{20}{973} = 0.021\text{rad}$ . Only a repeatable one step forward walk in position control has been successfully achieved  
370 with this method, see the video at the following link: [https://peertube.laas.fr/videos/watch/08db3177-372b-](https://peertube.laas.fr/videos/watch/08db3177-372b-43cc-85da-2009a267b5c9)  
371 [43cc-85da-2009a267b5c9](https://peertube.laas.fr/videos/watch/08db3177-372b-43cc-85da-2009a267b5c9).

## 372 7.3 Torque Control

### 373 7.3.1 PAL robotics low-level controller

374 To achieve torque control on the real robot, it is needed to transform the joint torque commands to  
375 motor currents. We decided to use the PAL robotics constructor low-level controller, which computes new  
376 commands respecting the robot actuators dynamics. This low-level controller is a proprietary black-box,  
377 which use a *ros-control* hardware interface to communicate with the robot. To interface our control scheme  
378 (based on the SoT with the WPG), we had to create a new version of the *roscontrol-sot* package. Indeed, our  
379 control scheme needs no more to communicate directly with the robot but with the PAL robotics controller,  
380 which implements different functions and formulations. One of the major difficulty is that the proprietary  
381 code source is not available, we only had access to its C++ headers and some basic tutorials. Developing  
382 this interface to keep all the functionalities implemented in the *roscontrol-sot* package (for instance to keep  
383 the recording of the logs and creating all the necessary signals needed by the SoT in the *dynamic-graph*  
384 structure), take us months of work (including the following remark).

Tasks	Priority	Weight
Feet contacts	0	-
Posture regularization in half-sitting	I	10

**Table 9.** Set of tasks for the torque control scheme on the posture task.

Experiments	Gains	Legs	Torso
Fail	$K_{Pposture}$	[800, 800, 800, 800, 800, 800]	[1000, 1000]
Success	$K_{Pposture}$	[50, 50, 50, 50, 50, 50]	[100, 100]
		Arms	Head
Fail	$K_{Pposture}$	[800, 800, 800, 800, 800, 800, 800, 800]	[100, 100]
Success	$K_{Pposture}$	[50, 50, 50, 50, 50, 50, 50, 50]	[10, 10]

**Table 10.** Tasks gains of the torque control scheme for the posture task.

385 Moreover, as the robot has a modified operating system called ferrum (equivalent to ubuntu), we created  
 386 a Docker Merkel (2014) container to have exactly the same environment as the one on the robot to test our  
 387 codes. Installing the SoT packages on this environment was not trivial as some packages had conflicting  
 388 dependencies with the PAL robotics packages. Finally, we succeeded to test in this Docker container our  
 389 interface and our torque controller using the PAL simulator available on ferrum. An additional difficulty is  
 390 that the simulator renders the behavior at a rate five times slower than the reality. Then, a small and slow  
 391 oscillation in the simulator is in fact a high frequency one in reality and can lead to dangerous behaviors.

392 One has to note that, in Dantec et al. (2021), the MPC is not embedded on the robot and is interfaced  
 393 with the PAL robotics low-level controller via a ROS topics. This simpler choice was made because it  
 394 is a stand-alone package (no SoT or *dynamic-graph* framework) and does not send commands at high  
 395 frequency (200Hz). ROS topics may induce latency and not allow to send high frequency commands  
 396 leading to real-time issues.

### 397 7.3.2 Experiment Results on a Posture Task

398 Once we achieve satisfying results on the PAL robotics simulator, we tested the classical formulation of  
 399 our torque controller using inverse dynamics on the real robot on a simple postural task. The tasks weights  
 400 and gains used are presented in the Tables.9 and 10, as the "Fail" experiment.

401 After few repetitions of a sinusoidal motion on the robot arm, the system diverged brutally and blocked  
 402 two of its harmonic drive: the waist and the right shoulder (we pushed the emergency button but the robot  
 403 had the time to reach the harmonic drive blocks). The Fig.12 presents the result failure. After investigation  
 404 it seems that the gains tuned in simulation (which simulates the actuation chains) were too high for the real  
 405 robot. Thus, tuning the gains even on a proper simulator with the model of the actuators is not enough to  
 406 ensure the safety of the solution. We know that some tuning is always necessary on the real robot, but we  
 407 wrongly assumed that the solution would remain quite stable. Thus, to provide a safe and reliable interaction  
 408 with the environment and possibly humans, we have looked for a way to ensure the system stability. The  
 409 video of the failed experiment is available at this link: <https://peertube.laas.fr/videos/watch/31fa2562-ba13-4043-a996-c2b8d5b21f4a>. Unfortunately, it was not possible to repair the robot at the laboratory because  
 411 the right shoulder and the torso were preventing the back cover of the robot to be removed (which needed  
 412 to be removed to access the shoulder harmonic drive). Thus the robot had to be send back to PAL robotics  
 413 for repair.



**Figure 12.** Failed experiment using torque control for a postural task.

414 By lowering the gains value on the posture task, as presented in the Table.10, we succeed to have a  
415 stable and compliant behavior of the robot. A video demonstrating this compliant behavior is available at  
416 the following link: <https://peertube.laas.fr/videos/watch/e9d8948d-08d5-4de9-8f42-2986fbbf0242>, and  
417 depicted by the Fig.13. At the end of the video, the robot falls because the contacts on the feet have been  
418 disturbed (the feet moved) breaking the constraint of the QP.

419 This small success encouraged us to add the CoM task for further tests. Unfortunately, this task does  
420 not work on the real robot. Instead of correcting the CoM error the QP seems to make it diverge. It is a  
421 behavior that is not appearing in the simulator, where the experiment is working. After investigations, this  
422 problem may be due to imprecise calibration or identification of the robot. The difficulty of performing this  
423 procedure once the robot is assembled, is to excite the parameters to be identified. For instance part the  
424 torso is particularly difficult to manipulate to observe the variables to be identified. It was the starting point  
425 of another research work outside the scope of this paper.

## CONCLUSION

426 The contribution of this paper is the benchmarking of three whole-body control implementations on the  
427 commercially available humanoid robot TALOS. Two of them are position based (with DCM and CoM



**Figure 13.** Successful experiment using torque control for a postural task.

428 admittance control): a Lexicographic QP using inverse kinematics and a WQP using TSID with an AM  
429 task. The last one is a WQP using TSID in torque with an AM task. They are evaluated in Gazebo on flat,  
430 uneven terrains and stairs climbing; on the criterion of trajectory tracking, energy consumption, passivity  
431 and computational cost.

432 In general, both position control schemes present the same results, with less energy consumption and  
433 higher passivity for the *TSID position* controller. A better tuning of the tasks gains may improve its results  
434 on the ZMP tracking.

435 On the other hand, the *TSID torque* controller shows better results in terms of smoothness of the trajectory  
436 tracking, energy consumption, passivity of the walk - without impacts and can achieve a 60cm walk with  
437 steps of 1s in simulation. This confirms the high capabilities of a torque control scheme coupled with  
438 an angular momentum regularization (see for instance Atlas in DARPA robotics challenge Koolen et al.  
439 (2016)). In average, the *TSID* controllers reach the 1kHz of control loop, necessary for real-time control,  
440 nonetheless, the *IK* scheme has the best computational time.

441 For our future works, we plan to control the hip flexibility of TALOS, so that we can evaluate the three  
442 controllers on the real robot. Moreover, it would be interesting to compare the controllers on different  
443 robotics platforms.

## CONFLICT OF INTEREST STATEMENT

444 The authors declare that the research was conducted in the absence of any commercial or financial  
445 relationships that could be construed as a potential conflict of interest.

## AUTHOR CONTRIBUTIONS

446 Noëlie Ramuzat is the first author of this work and did most of the new software development as well as  
447 the experimental results. Olivier Stasse and Sébastien Boria did the scientific and technical management  
448 for this project.

## FUNDING

449 This work is funded by the ROB4FAM research agreement.

## ACKNOWLEDGMENTS

450 TALOS is a robot created from the ACTANTHROPE ERC-Grant.

## REFERENCES

- 451 Caron, S., Kheddar, A., and Tempier, O. (2019). Stair climbing stabilization of the hrp-4 humanoid robot  
452 using whole-body admittance control. In *Int. Conf. on Robotics and Automation (ICRA)*
- 453 Carpentier, J., Saurel, G., Buondonno, G., Mirabel, J., Lamiroux, F., Stasse, O., et al. (2019). The pinocchio  
454 c++ library – a fast and flexible implementation of rigid body dynamics algorithms and their analytical  
455 derivatives. In *Int. Symp. on System Integrations*
- 456 Cisneros, R., Benallegue, M., Benallegue, A., Morisawa, M., Audren, H., Gergondet, P., et al. (2018).  
457 Robust humanoid control using a qp solver with integral gains. In *Int. Conf. on Intelligent Robots and  
458 Systems (IROS)*
- 459 Collins, S., Ruina, A., Tedrake, R., and Wisse, M. (2005). Efficient bipedal robots based on passive-dynamic  
460 walkers. *Science*
- 461 Dantec, E., Budhiraja, R., Roig, A., Lembono, T., Saurel, G., Stasse, O., et al. (2021). Whole body model  
462 predictive control with a memory of motion: Experiments on a torque-controlled talos. In *Int. Conf. on  
463 Robotics and Automation (ICRA)* (IEEE)
- 464 [Dataset] Del Prete, A. (2021). Tsid. <https://github.com/stack-of-tasks/tsid>
- 465 Del Prete, A., Mansard, N., Ponce, O., Stasse, O., and Nori, F. (2016). Implementing torque control with  
466 high-ratio gear boxes and without joint-torque sensors. *International Journal of Humanoid Robotics*
- 467 Engelsberger, J., Ott, C., and Albu-Schäffer, A. (2015). Three-dimensional bipedal walking control based  
468 on divergent component of motion. *IEEE Transactions on Robotics* 31
- 469 Engelsberger, J., Werner, A., Ott, C., Henze, B., Roa, M. A., Garofalo, G., et al. (2014). Overview of the  
470 torque-controlled humanoid robot toro. In *Int. Conf. on Humanoid Robotics (ICHR)*
- 471 Escande, A., Mansard, N., and Wieber, P. (2014). Hierarchical quadratic programming: Fast online  
472 humanoid-robot motion generation. *Int. Jour. of Robotics Research*
- 473 Featherstone, R. (2008). *Rigid Body Dynamics Algorithms* (Springer)
- 474 Fernbach, P., Tonneau, S., Stasse, O., Carpentier, J., and Taïx, M. (2020). C-croc: Continuous and  
475 convex resolution of centroidal dynamic trajectories for legged robots in multicontact scenarios. *IEEE  
476 Transactions on Robotics*



- 477 [Dataset] GEPETTO Team LAAS-CNRS (2020). talos-torque-control. [https://github.com/](https://github.com/stack-of-tasks/talos-torque-control)  
478 [stack-of-tasks/talos-torque-control](https://github.com/stack-of-tasks/talos-torque-control)
- 479 [Dataset] GEPETTO Team LAAS-CNRS (2021a). jrl-walkgen. [https://github.com/](https://github.com/stack-of-tasks/jrl-walkgen)  
480 [stack-of-tasks/jrl-walkgen](https://github.com/stack-of-tasks/jrl-walkgen)
- 481 [Dataset] GEPETTO Team LAAS-CNRS (2021b). multicontact-locomotion-planning. [https://](https://github.com/loco-3d/multicontact-locomotion-planning)  
482 [github.com/loco-3d/multicontact-locomotion-planning](https://github.com/loco-3d/multicontact-locomotion-planning)
- 483 [Dataset] GEPETTO Team LAAS-CNRS (2021c). sot-talos-balance. [https://github.com/](https://github.com/loco-3d/sot-talos-balance)  
484 [loco-3d/sot-talos-balance](https://github.com/loco-3d/sot-talos-balance)
- 485 Henze, B., Dietrich, A., and Ott, C. (2016). An approach to combine balancing with hierarchical  
486 whole-body control for legged humanoid robots. *IEEE Robotics and Automation Letters (RAL)* 1,  
487 700–707
- 488 Herzog, A., Rotella, N., Mason, S., Grimminger, F., Schaal, S., and Righetti, L. (2014). Momentum control  
489 with hierarchical inverse dynamics on a torque-controlled humanoid. *Autonomous Robots*
- 490 Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., et al. (2003a). Biped walking  
491 pattern generation by using preview control of zero-moment point. In *Int. Conf. on Robotics and*  
492 *Automation (ICRA)*
- 493 Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., et al. (2003b). Resolved  
494 momentum control: Humanoid motion planning based on the linear and angular momentum. In *Int.*  
495 *Conf. on Intelligent Robots and Systems (IROS)*
- 496 Kajita, S., Morisawa, M., Miura, K., Nakaoka, S., Harada, K., Kaneko, K., et al. (2010). Biped walking  
497 stabilization based on linear inverted pendulum tracking. In *Int. Conf. on Intelligent Robots and Systems*  
498 *(IROS)*
- 499 Koolen, T., Bertrand, S., Thomas, G., de Boer, T., Wu, T., Smith, J., et al. (2016). Design of a momentum-  
500 based control framework and application to the humanoid robot atlas. *Int. Journal of Humanoid Robotics*  
501 13
- 502 Lee, S. and Goswami, A. (2012). A momentum-based balance controller for humanoid robots on non-level  
503 and non-stationary ground. *Autonomous Robots*
- 504 Mansard, N., Stasse, O., Evrard, P., and Kheddar, A. (2009). A versatile generalized inverted kinematics  
505 implementation for collaborative working humanoid robots: The stack of tasks. In *International*  
506 *Conference on Advanced Robotics (ICAR)*
- 507 Merkel, D. (2014). Docker: Lightweight linux containers for consistent development and deployment.  
508 *Linux Journal*
- 509 Mesesan, G., Engelsberger, J., Garofalo, G., Ott, C., and Albu-Schäffer, A. (2019). Dynamic walking  
510 on compliant and uneven terrain using dcm and passivity-based whole-body control. In *Int. Conf. on*  
511 *Humanoid Robotics (ICHR)*
- 512 Mummolo, C. and Kim, J. H. (2012). Passive and dynamic gait measures for biped mechanism: formulation  
513 and simulation analysis. In *Robotica*
- 514 Nakaoka, S., Hattori, S., Kanehiro, F., Kajita, S., and Hirukawa, H. (2007). Constraint-based dynamics  
515 simulator for humanoid robots with shock absorbing mechanisms. In *IEEE Intelligent Conference on*  
516 *Intelligent Robots and Systems (IROS)*. 3641–3647
- 517 Negrello, F., Catalano, M. G., Garabini, M., Poggiani, M., Caldwell, D. G., Tsagarakis, N. G., et al. (2017).  
518 Design and characterization of a novel high-compliance spring for robots with soft joints. In *IEEE*  
519 *International Conference on Advanced Intelligent Mechatronics, AIM 2017, Munich, Germany, July 3-7,*  
520 *2017*

- 521 Orin, D., Goswami, A., and Lee, S.-H. (2013). Centroidal dynamics of a humanoid robot. *Autonomous*  
522 *Robot* 35, 161–176
- 523 Ponton, B., Herzog, A., Schaal, S., and Righetti, L. (2018). On time optimisation of centroidal momentum  
524 dynamics. In *Int. Conf. on Robotics and Automation (ICRA)*
- 525 Pucci, D., Romano, F., Traversaro, S., and Nori, F. (2016). Highly dynamic balancing via force control. In  
526 *Int. Conf. on Humanoid Robotics (ICHR)*
- 527 Ramuzat, N., Forget, F., Bonnet, V., Gautier, M., Boria, S., and Stasse, O. (2020). Actuator model,  
528 identification and differential dynamic programming for a talos humanoid robot. In *European Control*  
529 *Conference (ECC)*
- 530 [Dataset] Robotics, A. (2022). Meet digit
- 531 Romualdi, G., Dafarra, S., Hu, Y., Ramadoss, P., Chavez, F. A., Traversaro, S., et al. (2019). A  
532 benchmarking of dcm based architectures for position, velocity and torque controlled humanoid robots.  
533 *Int. Journal of Humanoid Robotics*
- 534 Samson, C., Leborgne, M., and Espiau, B. (1991). Robot control. the task-function approach. In *Oxford*  
535 *Engineering Science Series, vol. 22* (Oxford University Press)
- 536 Stasse, O., Giraud-Esclasse, K., Brousse, E., Naveau, M., Régnier, R., Avrin, G., et al. (2018).  
537 Benchmarking the hrp-2 humanoid robot during locomotion. *frontiers in Robotics and AI*
- 538 Stasse, O., Verrelst, B., Wieber, P.-B., Vanderborght, B., Evrard, P., Kheddar, A., et al. (2008). Modular  
539 architecture for humanoid walking pattern prototyping and experiments. *Advanced Robotics, Special*  
540 *Issue on Middleware for Robotics –Software and Hardware Module in Robotics System* 22, 589–611
- 541 Sugihara, T. (2009). Standing stabilizability and stepping maneuver in planar bipedalism based on the best  
542 com-zmp regulator. In *Int. Conf. on Robotics and Automation (ICRA)*
- 543 Takenaka, T., Matsumoto, T., and Yoshiike, T. (2009). Real time motion generation and control for biped  
544 robot-1st report: Walking gait pattern generation-. In *Int. Conf. on Intelligent Robots and Systems*  
545 *(IROS)*
- 546 Tonneau, S., Song, D., Fernbach, P., Mansard, N., Taïx, M., and Del Prete, A. (2020). SL1M: Sparse  
547 L1-norm Minimization for contact planning on uneven terrain. In *Int. Conf. on Robotics and Automation*  
548 *(ICRA)*
- 549 Torricelli, D., Gonzalez-Vargas, J., Veneman, J., Mombaur, K., Tsagarakis, N., del Ama, A., et al. (2015).  
550 Benchmarking bipedal locomotion: A unified scheme for humanoids, wearable robots, and humans.  
551 *IEEE Robotics Automation Magazine*
- 552 Wensing, P. M. and Orin, D. E. (2013). Generation of dynamic humanoid behaviors through task-space  
553 control with conic optimization. In *Int. Conf. on Robotics and Automation (ICRA)*

## FIGURE CAPTIONS