Auton Robot
DOI 10.1007/s10514-015-9422-7

Pattern generation and compliant feedback control for quadrupedal dynamic trot-walking locomotion: experiments on RoboCat-1 and HyQ

Barkan Ugurlu · Ioannis Havoutis · Claudio Semini · Kana Kayamori · Darwin G. Caldwell · Tatsuo Narikiyo

Received: 22 May 2014 / Accepted: 3 January 2015

Abstract In this paper, we introduce a method that synergistically combines an analytical pattern generator and a feedback controller frame, which are developed for the purpose of synthesizing dynamic quadrupedal trot-walking locomotion on flat and uneven surfaces. To begin with, the pattern generator analytically produces feasible and dynamically balanced joint motions in accordance with the desired trot-walking characteristics, with no empirical parameter tuning requirements. In concurrence with the pattern generation, a two-phased controller frame is constructed for closed-loop sensory feedback: (i) *virtual admittance controller* via force sensing, (ii) *upper torso angular momentum regulation* via gyro sensing. The former controller evaluates joint force errors and generates the corresponding joint displacement for a given set of virtual spring-damper couples. Together with the position constraints, these displacements are additionally fed-back to local servos for achieving compliant quadrupedal locomotion with which the position/force trade-off is addressed. The second controller, that is simultaneously used, evaluates the upper torso angular momentum rate change error using measured and reference orientation information. It then regulates the torso orientation in a dynamically consistent way as the rotational inertia is characterized. In order to validate the proposed methodology several experiments are conducted on both flat and uneven surfaces, using two robots with distinct properties; a ∼7 kg cat-sized electrically actuated quadruped (RoboCat-1), and a ∼80 kg Alpine Ibex-sized hydraulically actuated quadruped (HyQ). As a result we demonstrate continuous, repetitive, compliant and dynamically balanced trot-walking cycles in real-robot experiments, adequately confirming the effectiveness of the proposed approach.

Keywords Quadrupedal locomotion · Dynamic trot-walking · Active compliance · Pattern generation

1 Introduction

Recent advances in mechatronics technology allowed the creation of robotic systems with versatile and dexterous locomotion characteristics. Similarly, considerable effort has been devoted to the development of contemporary quadrupedal robotic platforms; several research groups and companies introduced systems that are capable of performing dynamic locomotion in challenging environments (Raibert et al. 2008), imitating their biological counterparts in terms of transportation cost (Sangok et al. 2013), and constituting a testbed for neuroscience studies (Yamada et al. 2011).

Electronic supplementary material The online version of this article (doi:10.1007/s10514-015-9422-7) contains supplementary material, which is available to authorized users.

B. Ugurlu (✉)
Department of Mechanical Engineering,
Ozyegin University, 34794 Istanbul, Turkey
e-mail: barkan.ugurlu@ozyegin.edu.tr

I. Havoutis · C. Semini · D. G. Caldwell
Department of Advanced Robotics, Istituto Italiano
di Tecnologia (IIT), 16163 Genoa, Italy
e-mail: ioannis.havoutis@iit.it

C. Semini
e-mail: claudio.semini@iit.it

D. G. Caldwell
e-mail: darwin.caldwell@iit.it

K. Kayamori · T. Narikiyo
Department of Advanced Science and Technology,
Toyota Technological Institute, Nagoya 468-8511, Japan
e-mail: n-tatsuo@toyota-ti.ac.jp

Published online: 04 February 2015

The so-called decentralized legged locomotion control problem can be tackled in two distinct phases: (i) online pattern generation, (ii) real-time sensory feedback control. This approach is proved to be experimentally efficient for various legged systems (Sugihara and Nakamura 2009). Keeping this in mind, we follow a similar strategy and propose an approach that consists of an analytical pattern generator coupled with a feedback controller framework, simultaneously operating in real-time for synthesizing quadrupedal dynamic trot-walking motions.

In light of the above-stated fact, the whole paper is structured on this dual basis. The current section continues with an overview of the related literature and our main contributions. The robots that are used in our experiments are introduced in Sect. 2. The overall locomotion control framework, the analytical pattern generator, the kinematics scheme and feedback controller units are explained in great detail in Sects. 3, 4, 5 and 6, respectively. Experimental results are presented and thoroughly discussed in Sect. 7. The paper is concluded in Sect. 8, presenting our concluding remarks and future research goals.

1.1 Related works: pattern generation

Research on many biological creatures, especially vertebrates, suggested that Central Pattern Generators (CPGs) play a key role in generating locomotory behavior. This fact motivated several roboticists, e.g. Rutishauser et al. (2008), exploited CPG models to synthesize quadrupedal gaits for a passively compliant robot. In a different example, Maufroy et al., utilized CPGs to modulate walking phases via leg loading/unloading sequences (Maufroy et al. 2010). Barasuol and his associates made use of ellipsoidal CPGs to achieve reactive locomotion behavior on a quadruped robot (Barasuol et al. 2013), an approach that has proven very successful and serves as the primary trot-controller of IIT's HyQ. Morimoto et al. implemented CPGs for bipedal walking and proved that their usage is not limited to multi-legged locomotion (Morimoto et al. 2008).

On one hand, CPGs are successfully applied in many different stages of the legged locomotion problem. On the other hand, CPG-based approaches are often criticized due to their inherent limitations. One of the major issues is the requirement of unintuitive empirical tuning that is often necessary for a successful implementation. CPGs also may not guarantee the generation of dynamically consistent motion, the success of the locomotion is usually experienced on the fly. In addition, it is still considered challenging to incorporate real-time sensory feedback despite impressive efforts (Righetti and Ijspeert 2008).

Based on these factors, a certain fraction of researchers investigated alternative options for the locomotion synthesis task. Moro et al. set an example toward this direction; they collected motion capture data from a horse and used it to generate a motion pattern for a passive compliant robot, in virtue of kinematic motion primitives (Moro et al. 2013). Koolen et al. proposed a legged locomotion technique based on capturability analysis (Koolen et al. 2012). Center of Pressure (CoP), a dynamic balance indicator, is used for quadrupedal trajectory generation, in which the governing CoP equation, a second order differential equation, is exploited either via polynomials (Kalakrishnan et al. 2011), pre-defined sinusoidal trajectories (Kurazume et al. 2002) or numerical integration (Ugurlu et al. 2013). Analytical solutions are also utilized (Yoneda et al. 1996); however, the Center of Mass (CoM) acceleration continuity is not yet investigated.

The operation of solving CoP differential equations to obtain the corresponding CoM trajectories are defined as an *inverse problem* by Byl et al. (2009). In solving this second order differential equation, one has to define initial CoM position and velocity values. As noted in Byl et al. (2009), selection of initial conditions poses a challenging problem; the generated CoM trajectory highly depends on initial conditions. An arbitrary selection for initial conditions, e.g., zero initial CoM velocity, would result with infeasible or physically unrealizable trajectories. This problem is thoroughly addressed in Sect. 2.

In an alternative solution, Kajita et al. tackled this issue by employing the optimal preview Zero Moment Point (ZMP) servo control principle (Kajita et al. 2003). Utilizing the future references, it is possible to obtain feasible CoM patterns while minimizing the ZMP error. This method is nowadays widely used in the humanoid research community and also reliably implemented for quadrupedal trajectory generation in Byl et al. (2009).

While the preview control appears to be quite efficient and functional, it has a couple of drawbacks on its own. As the ZMP-based pattern generation problem is treated as ZMP servo control, one needs to substitute optimal control gains, a performance index and future reference previewing period. As an alternative to Kajita's approach, we strongly defend that pattern generation task could be handled computationally rather than interpreting it as a ZMP servo tracking problem. It should be possible to generate ZMP-based CoM patterns in an open-loop strategy, in which the choice of feedback control can be freely designated, instead of interpreting the task as a requisite ZMP servo control.

1.2 Related works: control strategies

Regardless of terrain type, a legged robotic system needs to exhibit compliant locomotion behavior while interacting with the environment (Zheng and Hemami 1985). A straightforward implementation in this direction is the use of physically flexible elements to obtain passively compliant locomotion cycles (Rutishauser et al. 2008; Moro et al. 2013; Hutter et al. 2013; Kimura et al. 2007; Ugurlu et al. 2014; Sproewitz et al. 2011). This approach is founded on observations from biological structures; it potentially increases the energy efficiency and the environmental adaptability. In contrast, most passively compliant robots cannot modulate passive compliance in real-time, unlike their biological counterparts (Ferris et al. 1998). In particular, Galloway et al. experimentally documented that tunable passive compliance is of importance to succeed in synthesizing legged locomotion for their 1-DoF-per-leg salamander-like robot (Galloway et al. 2013). Nevertheless, utilization of flexible elements in a multi-DoF legged robot complicates the design and the control of the system.

As a different strategy, roboticists implement active compliance schemes where the virtual impedance properties are software-controlled, often based on sensory feedback. In this view, it is possible to enhance environmental interaction capabilities (Kim et al. 2007; Buschmann et al. 2009; Fujimoto et al. 1998), with enhanced locomotion robustness (Barasuol et al. 2013; Kalakrishnan et al. 2011; Fujimoto et al. 1998; Hyon 2009; Ott et al. 2011; Boaventura et al. 2012). In implementing active compliance control for legged locomotion and balancing, Hyon proposed a useful technique to optimally distribute the forces applied to the ground via predetermined contact points, in an attempt to achieve terrain adaptation (Hyon 2009). Aiming at a similar research objective, Ott and his colleagues make use of grasping formulations to optimize contact forces (Ott et al. 2011). The common ground in these works is to compute the desired wrench that is required for the compensation action, and then, to distribute it to the predetermined contact points which are assumed to be in contact with the terrain.

Inserting floating-base inverse dynamics feed-forward terms, Buchli and his collaborators succeed in locomotion control with relatively low servo control gains (Barasuol et al. 2013; Winkler et al. 2014; Kalakrishnan et al. 2011; Boaventura et al. 2012). In doing so, they are able to introduce virtual impedance in the joint space, without any requirement of predetermined contact points. What is more, the overall robot configuration may be covered in terms of compliant behavior when using this strategy. While impressively efficient in its own right, this method may not characterize the induced force error within the active compliance strategy. In other words, the compliant behavior of the joint does not change due to the force error. For instance, if servo gains are chosen relatively low, the joint is soft regardless of the force error. In response to this matter, this paper investigates that whether an admittance control approach may provide more favorable characteristics in managing the position/force trade-off, since it adapts the compliant behavior in relation to the force error.

1.3 Contributions

The contributions of this paper can be listed in the following three categories.

i) We propose a CoP-based analytical pattern generator that is capable of producing smooth, purely computational, and dynamically equilibrated CoM trajectories. It guarantees the continuity of the reference trajectories in position, velocity and acceleration, regardless of the walking phase. Despite existing attempts, the analytical trajectory generators in the literature may not be capable of possessing such properties. The proposed pattern generator imposes no requirement of any requisite feedback loop, nor a large set of empirically tuned free parameters. These features allow us to solve the trajectory generation problem without any time consuming trial-and-error parameter tuning. Moreover, the choice of online feedback can be freely designated.
ii) We present a control framework that includes a *virtual admittance controller* and an *orientation controller*.

ii-a) The virtual admittance controller introduces software controlled spring-damper couples in each joint to address the position/force trade-off within the entire robot configuration. In normal operation, robot joints are stiff to prioritize position tracking. Within the presence of external disturbances, due to environmental interactions, the force error may increase. In that instant the robot joint stiffness automatically decreases in a way to comply with the force constraints. This trade-off is managed by means of virtual spring-damper couples; for instance, the joints become more reactive to the induced force error if admittance coefficients are chosen to be smaller. The virtual admittance controller does not require any predetermined contact points and it is more reactive compared to impedance-based controllers in which joint stiffness does not change depending on the external impacts. To the best of our knowledge such features cannot be found in the existing active-compliance control techniques.
ii-b) In addition to this controller, an angular momentum based upper torso orientation control is implemented in which rotational inertia, a crucial element of the rotational dynamics, is characterized possibly for the first time. Friction and inverse dynamics compensation schemes are additionally implemented to enhance controller performance.

iii) We present substantial experimental evidence to support the proposed quadrupedal locomotion control framework. This way we verify that the combination of the above-mentioned contributions results in robust quadrupedal trot walking on level and uneven surfaces. We prove the feasibility of our approach by presenting results on two robots with completely different characteristics in terms of mass, size, and actuation.

An initial version of this work including preliminary evaluations has been presented in Ugurlu et al. (2013). The current paper is a significantly expanded and improved version to provide a thorough and complementary archival report. Enhancements over (Ugurlu et al. 2013) include the following: (i) exhaustive review of the related works, (ii) enriched and clearer mathematical analysis for the pattern generator and controllers, (iii) newly-conducted experiments, (iv) detailed analysis and discussion of experiment results.

2 Quadruped robots: RoboCat-1 and HyQ

2.1 RoboCat-1

In order to explore the dynamics of legged locomotion on uneven terrain, the quadruped robot RoboCat-1 was developed in Toyota Technological Institute, Japan. The main goal in designing RoboCat-1 is to create a robotic platform to emulate various terrain locomotion scenarios on a durable and low-cost system. Figure 1 depicts the actual system.

The main specifications of RoboCat-1 are summarized in Table 1. RoboCat-1 has four 2-DoF (Degrees of Freedom) planar legs, powered via AC servo motors. Each leg has a hip (hip_{fe}) and and a knee joint ($knee_{fe}$), rotating through pitch axis. An IMU sensor is located on the robot torso (Microstrain 3DM-GX3-25), which provide triaxial orientation and acceleration measurement. Moreover, 4 single axis force sensing units (0-500 N) are deployed in each foot to measure ground reaction forces.

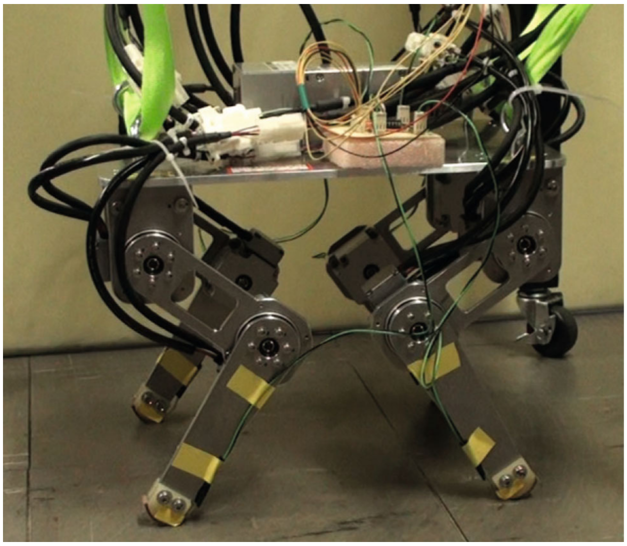


Fig. 1 Quadruped robot prototype RoboCat-1

Table 1 RoboCat-1 specifications

Size (L × H × W)	0.34 × 0.23 × 0.3 m
Total weight	6.9 kg
Thigh-knee	0.1 m
Knee-foot	0.13 m
Total DoF	2 in each leg; 2 × 4 = 8 DoF
Encoder resolution	0.0045°/count
Rated output torque	9 Nm (hip_{fe}, $knee_{fe}$)
Gear ratio	100
Sampling frequency	1 kHz

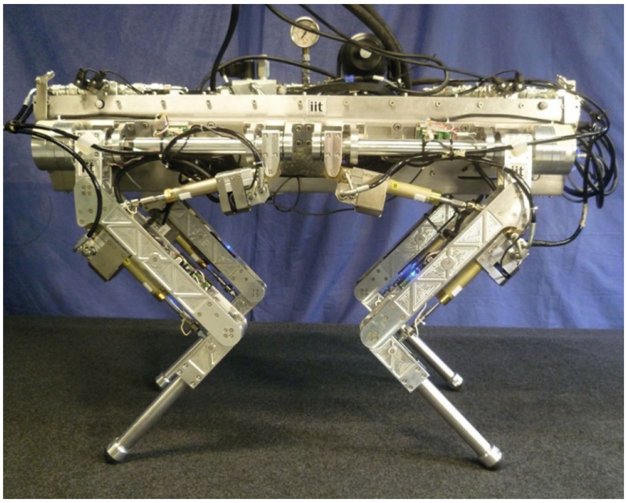


Fig. 2 The hydraulically actuated quadruped robot HyQ

2.2 HyQ: hydraulic quadruped

The quadruped robot platform HyQ (see Fig. 2) is a fully torque-controlled hydraulically actuated quadruped robot, comparable in size to a mountain goat (∼80 kg), e.g. an *Alpine Ibex*. It has been designed and built *in-house* in the Istituto Italiano di Tecnologia (Semini et al. 2011). It uses a combination of hydraulic cylinders and rotary motors for the actuation of its 12 joints. HyQ is capable of highly dynamic locomotion as hydraulic actuation allows the handling of large impact forces, high bandwidth control, high power-to-weight ratio and superior robustness.

Each leg has three DoFs, two in the hip (roll and pitch axes, namely, hip_{aa} and hip_{fe}) and one in the knee (pitch axis, $knee_{fe}$). The hip_{aa} joints are actuated by rotary hydraulic motors while all the hip_{fe} and the $knee_{fe}$ joints are actuated by hydraulic cylinders. HyQ's joints are equipped with high resolution encoders and load cells, which allow the smooth control of both position and torque. An IMU sensor (Micros-

Table 2 HyQ specifications

Size (L × H × W)	1.0 × 1.0 × 0.5 m
Total weight	80 kg
Thigh-knee	0.35 m
Knee-foot	0.33 m
Total DoF	3 in each leg; 3 × 4 = 12 DoF
Encoder resolution	0.0045°/count
Rated output torque	145 Nm (hip_{fe}, $knee_{fe}$)
Sampling frequency	1 kHz

train 3DM-GX3-25) is also deployed on its torso to provide triaxial orientation and acceleration measurements. In each leg, single axis load cell units (0-5000 N) output reaction force measurements. A brief overview of HyQ's characteristics is given in Table 2. The system is controlled by an on-board Pentium PC104 running a real-time patched Linux (*Xenomai*) operating system at a 1 kHz control frequency. For further details regarding the details of its mechatronic hardware, the interested reader can refer to Semini et al. (2011).

3 Overall locomotion control framework

Figure 3 provides an overview of the locomotion control framework. In this figure, leg numbers are denoted with the underscript $j (j = 1, 2, 3, 4)$, while the respective joints are numbered with the underscript i ($i = 0$: Hip Roll (hip_{aa}, only for HyQ), $i = 1$: Hip Pitch (hip_{fe}), $i = 2$: Knee Pitch ($knee_{fe}$)). X_{cop} refers to x-axis CoP (Center of Pressure), while $\mathbf{r_c} = (c_x, c_y, c_z)$ symbolizes CoM (Center of Mass) position with respect to the world frame (see Fig. 8). θ, ψ and ϕ point out upper torso roll, pitch and yaw orientation parameters. Position vectors $\mathbf{r_{fj}}$, $\mathbf{r_{cfj}}$, $\mathbf{r_{hfj}}$ indicate the jth foot position with respect to the world frame, CoM position with respect to the jth foot, the jth hip position with respect to the jth foot, respectively (see Fig. 8). F and q denote force sensor outputs and joint angle measurements. τ_{frji}, τ_{cgji}, τ_{pdji}, and τ_{cmdji} respectively denote friction compensation torque, inverse dynamics compensation torque, servo (PD) controller output and command torque for the jth leg, ith joint. Reference quantities are denoted with the ref underscript, whereas actual (measured) quantities have no underscript. Underscript c stands for the output signal of a controller.

The overall framework can be grouped in three categories which are explained in the following sections: (1) Trot-Walking Pattern Generation (see Sect. 4); (2) Leg Kinematics Scheme for Trot-Walking (see Sect. 5); (3) Feedback Controller Frame (see Sect. 6).

4 Dynamic trot-walking pattern generator

In quadrupedal trotting, the robot's feet are moved in such a way that diagonal legs perform simultaneously the same motion. Hence, quadrupedal trot-walking is induced via three subsequent phases: (a) Left front and right hind 2-legged support phase; (b) 4-legged support phase; (c) right front and left hind 2-legged support phase. As this does not include any 3-legged triangular support phase, we may assume that quadrupedal trot-walking is analogous to planar bipedal walking. Using this analogy, an equivalent planar biped model is constructed to simplify the pattern generation task. In other words, the equivalent planar biped model is a tool that allows the interpretation of quadrupedal trot-walking by means of bipedal walking, so as to facilitate the pattern generation task.

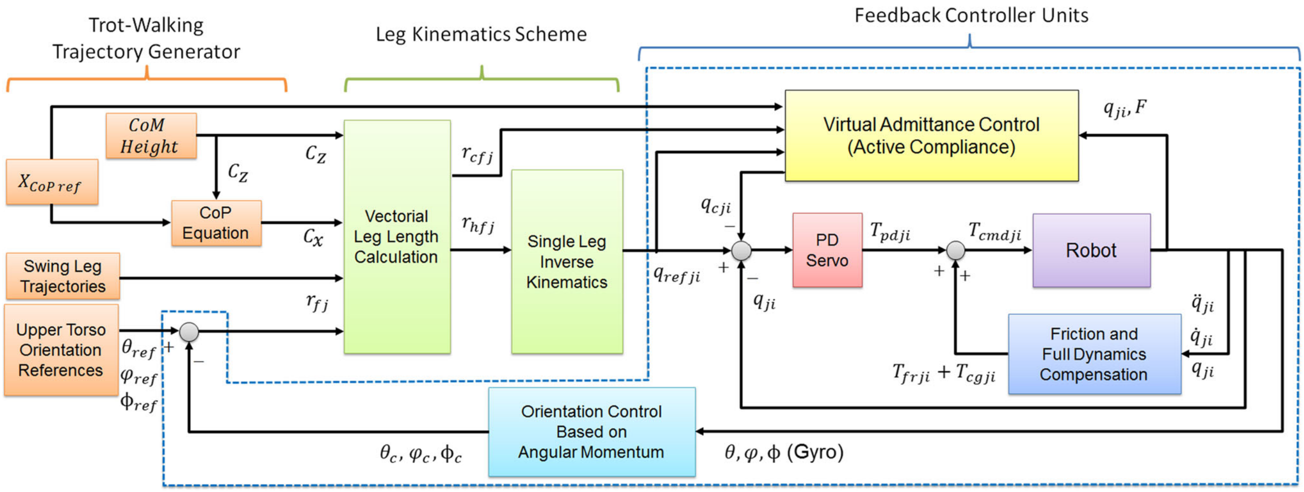

Trot-Walking Trajectory Generator
Leg Kinematics Scheme
Feedback Controller Units
CoM Height
$X_{CoP\,ref}$
c_z
c_z
CoP Equation
c_x
Swing Leg Trajectories
r_{fj}
Upper Torso Orientation References
θ_{ref}
φ_{ref}
ϕ_{ref}
+
−
Vectorial Leg Length Calculation
r_{cfj}
r_{hfj}
Single Leg Inverse Kinematics
Virtual Admittance Control (Active Compliance)
q_{ji}, F
q_{cji}
q_{refji}
+
−
−
q_{ji}
PD Servo
τ_{pdji}
+
+
τ_{cmdji}
Robot
$\ddot{q}_{ji}$
$\dot{q}_{ji}$
q_{ji}
Friction and Full Dynamics Compensation
$\tau_{frji} + \tau_{cgji}$
Orientation Control Based on Angular Momentum
$\theta_c, \varphi_c, \phi_c$
θ, φ, ϕ (Gyro)

Fig. 3 Overview of locomotion control framework. It consists of three blocks: (1) the trot-walking pattern generation (see Sect. 4); (2) the leg kinematics scheme for trot-walking (see Sect. 5); (3) the feedback controller frame (see Sect. 6)

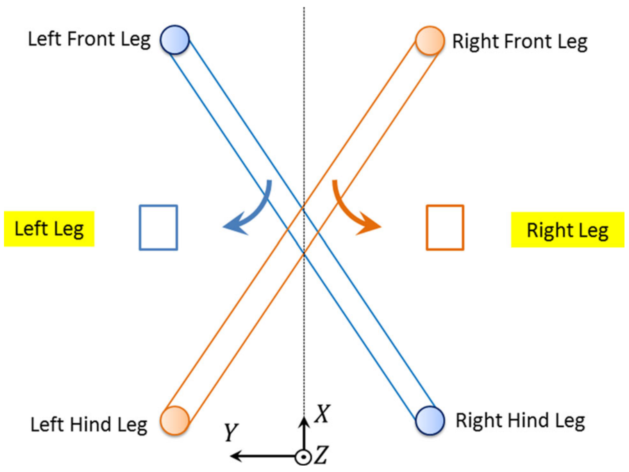


Fig. 4 Corresponding feet position during quadrupedal trot-walking, in the equivalent planar biped model. The *rectangular shapes* are used to indicate the equivalent planar biped feet, solely for illustration purposes. These do not indicate any support region

As illustrated in Fig. 4, the left foot of the equivalent biped corresponds to the middle of left front and right hind pair. Identically, the right equivalent biped foot corresponds to the middle of the right front and left hind pair. Using the equivalent planar biped model, the aforementioned phases can be interpreted as follows: (a) left foot single support phase; (b) double support phase; (c) right foot single support phase. Note that a similar approach was previously proposed as *virtual legs* in Yoneda et al. (1996).

4.1 CoM trajectory generation

In order to obtain a dynamically equilibrated CoM trajectory, we make use of the ZMP concept. Considering the classical point mass approach, the Center of Pressure (CoP) and ZMP are mathematically identical; therefore, we choose to use the term CoP. With this in mind, let us express the x-axis CoP formula when z-axis CoM position is constant (Kalakrishnan et al. 2011; Ugurlu et al. 2013; Byl et al. 2009; Kajita et al. 2003) as:

$$X_{cop} = c_x - c_z \frac{\ddot{c}_x}{g} \tag{1}$$

In the above equation, X_{cop} is the x-axis CoP, g is gravitational acceleration, (c_x, c_z) are horizontal and vertical CoM positions. Equation (1) appears to be a 2nd order differential equation which can be analytically solved depending on the X_{cop} input. In our trajectory generation algorithm, X_{cop} is a constant input during single support phases $(X_{cop} = p_x)$, and a ramp input during double support phases $(X_{cop} = p_x + K_x t)$.p_x and K_x are positive constants. Therefore, single and double support phases are separately analyzed.

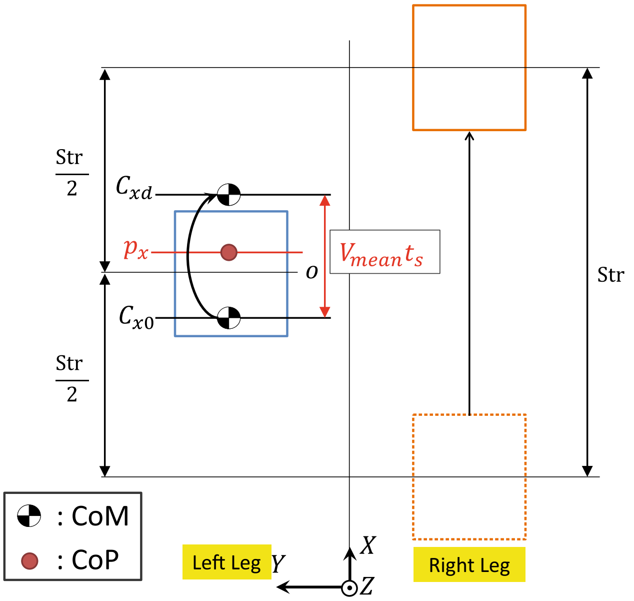


Fig. 5 Corresponding feet position during quadrupedal trot-walking, in the equivalent planar biped model. *Rectangular shapes* are used to indicate the equivalent planar biped feet, solely for the purpose of illustration. These do not indicate any support region

4.1.1 Single support phase

When the robot is in a single support phase, X_{cop} input is chosen as constant; $(X_{cop} = p_x)$. Figure 5 illustrates the corresponding CoP, CoM and feet positions in the equivalent planar biped model. Considering this case, we can analytically solve (1) as below:

$$c_x = (c_{x0} - p_x) \cosh \omega\tau + \frac{\dot{c}_{x0}}{\omega} \sinh \omega\tau + p_x, \tag{2}$$

$$\omega = \sqrt{\frac{g}{c_z}}, \quad \tau = t - t_0, \tag{3}$$

where c_{x0}, $\dot{c}_{x0}$, ω, t, t_0 are initial x-axis CoM position, initial x-axis CoM velocity, natural frequency of the equivalent pendulum, time variable and initial time, respectively. The x-axis CoM velocity and acceleration functions can also be obtained via differentiation, as in the following:

$$\dot{c}_x = \omega\,(c_{x0} - p_x) \sinh \omega\tau + \dot{c}_{x0} \cosh \omega\tau \tag{4}$$

$$\ddot{c}_x = \omega^2\,(c_{x0} - p_x) \cosh \omega\tau + \omega \dot{c}_{x0} \sinh \omega\tau. \tag{5}$$

For the purpose of synthesizing physically viable CoP-based CoM trajectories, a single support phase should be composed of two equal deceleration and acceleration phases. Figure 6 illustrates a numeric example from 0.0 to 0.88 period, both for x-axis CoM position, velocity and acceleration. As observed in this figure, initial and terminal CoM velocities should be equal. Initial and terminal acceleration values are equal in amplitude with opposite signs. These properties could be guaranteed by tuning initial conditions

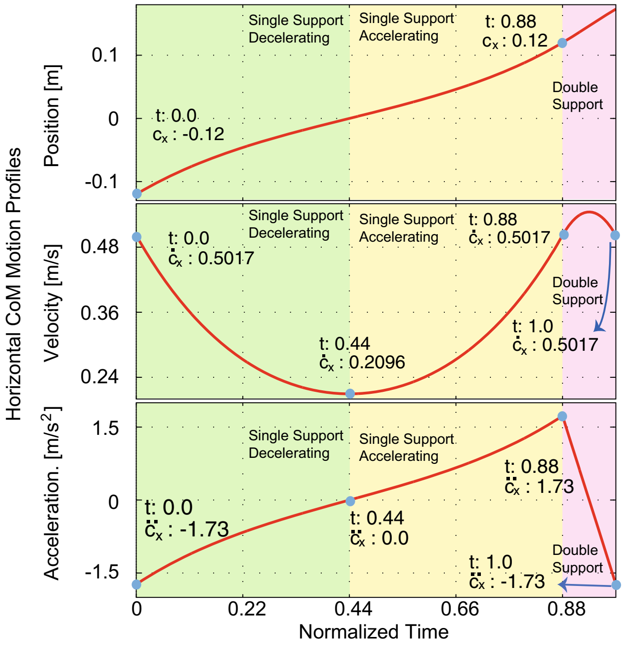


Fig. 6 Feasible x-axis CoM motion during a single (0.0–0.88) and a double (0.88–1.0) support phase periods. Time is normalized by considering a total trot-walking cycle (single + double). Note that this plot shows a mere numeric example

in accordance with the desired walking parameters. Since the nature of CoP equations are hyperbolic, a single support phase with unequal deceleration/acceleration periods would result in exponentially accumulated increases in CoM position and produce physically unrealizable trajectories.

Utilizing the symmetry feature discussed above, we may see that the velocity function has a minimum at the middle point of a single support period, i.e., when $t = t_m = t_0 + T_s/2$. Here, T_s refers to a single support time period.

$$\ddot{c}_x|_{t=t_m} = \omega^2(c_{x0} - p_x)\cosh\left(\frac{wT_s}{2}\right) + \omega\dot{c}_{x0}\sinh\left(\frac{wT_s}{2}\right) = 0 \tag{6}$$

Using (6), one can obtain $\dot{c}_{x0}$ as below.

$$\dot{c}_{x0} = \omega(p_x - c_{x0})\coth\left(\frac{wT_s}{2}\right) \tag{7}$$

During a single support phase, we may substitute a mean forward velocity, υ_{mean}. Considering this parameter, the terminal position, c_{xd}, can be computed (see Fig. 5).

$$c_{xd} = \upsilon_{mean}T_s + c_{x0}. \tag{8}$$

The terminal position c_{xd} can also be computed using (2), when $t = t_e = t_0 + T_s$ as:

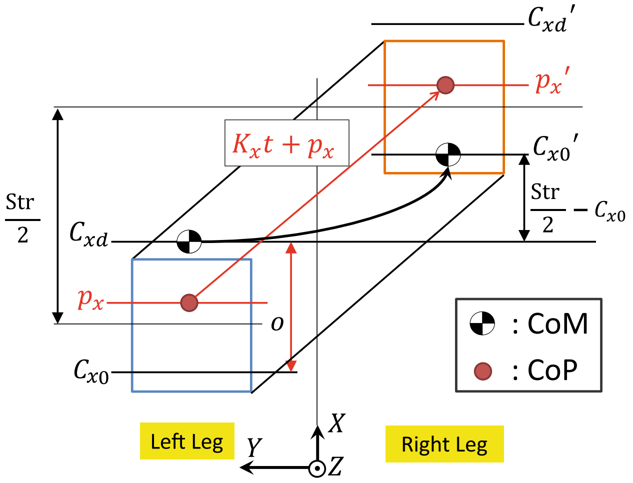


Fig. 7 The equivalent planar biped model during a double support phase. The *rectangular shapes* are used to indicate the equivalent planar biped feet, solely for illustration purposes. These do not indicate the support region

$$c_x|_{t=t_e} = c_{xd} = (c_{x0} - p_x)\cosh(\omega T_s) + \frac{\dot{c}_{x0}}{\omega}\sinh(\omega T_s) + p_x. \tag{9}$$

Plugging (7) into (9), the hyperbolic terms vanishes ($\cosh\omega T_s - \sinh\omega T_s\coth 0.5\omega T_s = -1$), and therefore, we arrive at the following equation:

$$c_{xd} + c_{x0} = 2p_x \tag{10}$$

The combination of (8) and (10) allows us to calculate the c_{x0} and c_{xd} terms in the following manner:

$$c_{x0} = p_x - \frac{\upsilon_{mean}T_s}{2}; \qquad c_{xd} = 2p_x - c_{x0} \tag{11}$$

For a given set of single support period (T_s), target mean velocity (υ_{mean}), and constant X_{cop} input (p_x), we may sequentially calculate initial conditions, ($c_{x0}, \dot{c}_{x0}$), via (11) and (7). c_{xd} can also be calculated beforehand for prior verification. Finally, using ($c_{x0}, \dot{c}_{x0}$), CoM trajectory can be generated with (2). A sequential algorithm for the above-described computations is given in Sect. 4.3.

4.1.2 Double support phase

During a double support phase, the main objective is to make sure that CoP travels from the preceding support foot to the proceeding support foot continuously, as illustrated in Fig. 7. What is more, at the end of a double support phase, the x-axis CoM position, velocity and acceleration values must be equal to the initial x-axis CoM position, velocity and acceleration values of the next single support phase; allowing us to link the sequential phases in a seamless fashion. Referring to Fig. 7, we can solve the 2nd order differential equation (1) for a linearly increasing X_{cop} input ($X_{cop} = K_xt + p_x$).

Differentiating it with respect to time also returns its velocity function as well.

$$c_x = (c'_{x0} - p_x)\cosh\omega\tau' + \frac{\dot{c}'_{x0} - K_x}{\omega}\sinh\omega\tau' + K_x\tau' + p_x \quad (12)$$

$$\dot{c}_x = \omega(c'_{x0} - p_x)\sinh\omega\tau' + (\dot{c}'_{x0} - K_x)\cosh\omega\tau' + K_x \quad (13)$$

In (12) and (13), the initial position and velocity terms for a double support phase are represented with $(c'_{x0}, \dot{c}'_{x0})$. As pointed out in (14), these terms are equal to the terminal position and velocity of the preceding single support phase. K_x is the inclination value for X_{cop} function. $\tau' = t'_0 + t$ stands for the shifted time variable with the initial time for double support period, namely t'_0. In order to yield a seamless connection, the terminal velocity of a double support phase should be equal to the initial velocity of the following single support phase when $t = t_{ed} = t'_0 + T_d$.

$$c'_{x0} = c_{xd}; \qquad \dot{c}'_{x0} = \dot{c}_{x0}; \quad (14)$$

$$\dot{c}_x|_{t=t_{ed}} = \dot{c}_{x0} = \omega(c_{xd} - p_x)\sinh\omega T_d + (\dot{c}_{x0} - K_x)\cosh\omega T_d + K_x. \quad (15)$$

Evaluating (15), K_x and stride length, Str, may be obtained as follows:

$$K_x = \dot{c}_{x0} + \omega(c_{xd} - p_x)\coth\frac{\omega T_d}{2}; \quad (16)$$

$$Str = 2T_d K_x. \quad (17)$$

Note that p_x constant was previously assigned and $\dot{c}_{x0}, c_{xd}$ were previously computed in the single support phase. Hence, for a given double support time period (T_d), K_x and stride length Str can be calculated using (16) and (17). Finally, using the initial conditions computed via (14) together with K_x, the CoM trajectory may be generated with (12). The stride length, Str, will be used for swing leg trajectory generation as described in the next subsection. An algorithm for the above procedure is given in Sect. 4.3.

4.2 Swing leg trajectories and torso orientation

The swing leg trajectories are calculated via polynomial functions with the stride length (Str) and maximum vertical swing foot clearance (F_h) in mind. Note that Str is previously computed in the CoM trajectory phase, while maximum vertical swing foot is freely assigned. In generating swing leg trajectories, both initial and terminal velocity and acceleration terms are set to zero to make sure that the swing foot tip arrives to the support surface in a hypothetically motionless state, so as to reduce the impact forces as much as possible, during the pattern generation phase. This issue will be further addressed in terms of force feedback in Sect. 5.

As described in Sect. 5, our leg kinematics scheme allows us to independently assign the upper torso orientation trajectories, without any concern of interference with the target CoM trajectory. In other words, regardless of the specified upper torso orientation, the desired CoM is always achieved provided that the actuator limits are not exceeded. In this paper, we choose to keep the upper torso orientation parallel to the support surface.

4.3 Summary: the CoM generation algorithm

In order to generate CoM patterns in accordance with the proposed method, the following algorithm is executed.

1. Determine the single and double support periods (T_s, T_d), constant CoM height (z), mean target forward velocity (υ_{mean}) and constant ZMP reference input for single support phases (p_x).
2. Compute ω, (c_{x0}, c_{xd}), $\dot{c}_{x0}$, K_x and Str via (3), (11), (7), (16), and (17), in a strictly sequential manner.
3. Once the aforementioned parameters are assigned, either use (2) (single support) or (12) (double support) to generate the CoM trajectory, depending on the phase information.

We would like to highlight the fact that the proposed CoM pattern generation algorithm exploits the symmetry properties explained above. This guarantees that reference c_x, $\dot{c}_x$, $\ddot{c}_x$ trajectories are seamlessly connected throughout the whole period, including phase transitions, such as from single to double support phase and vice versa. In other words, the pattern generator makes sure that there is no discontinuity through the generated reference CoM trajectories in position, velocity and acceleration levels, so that unexpected motions or undesired command torque jumps do not occur. The referential trajectory continuity guarantee may not necessarily mean that the actual robot motion will be seamless; it is the controller frame's (see Sect. 6) responsibility to track the reference trajectory even within the presence of disturbances or ground impacts.

5 Leg kinematics scheme for trot-walking

Having generated the necessary trajectories for CoM, swing legs and upper torso orientation, the next task is to obtain the corresponding joint angle references. In response to that matter, we make use of a vectorial calculation in which inputs (see Table 3) are processed to compute each joint angle. Referring to Fig. 8, the following expression can be obtained:

$$\mathbf{r_c} + R_t\mathbf{r_{ch}} + R_t\mathbf{r_{hf}} - \mathbf{r_f} = \mathbf{0} \quad (18)$$

In (18), $\mathbf{r_c}$ is the CoM position vector with respect to the world frame, $\mathbf{r_c} = (c_x, c_y, c_z)$. $\mathbf{r_{ch}}$ and $\mathbf{r_{hf}}$ are position vectors; the former is defined between the CoM and the respec-

Table 3 Inputs of the leg kinematics scheme

$\mathbf{r_c}$	CoM position with respect to the world frame
$\mathbf{r_{fj}}$	jth foot position with respect to the world frame
θ, ψ, ϕ	3 Euler angles that represent torso orientation

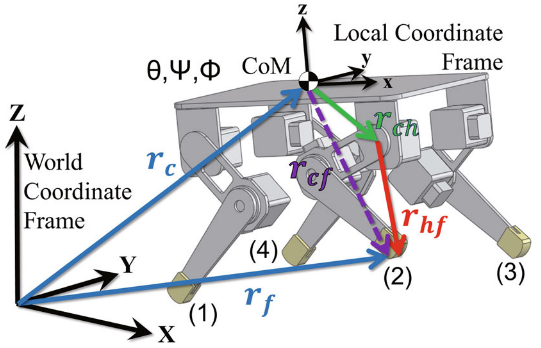


Fig. 8 Related parameters on RoboCat-1 model for the leg kinematics scheme. Vectors are shown only for a single leg

tive hip, the latter is defined between the respective hip and the foot. R_t is the upper torso orientation that is constructed using 3 Euler angles, namely, θ, ϕ and ψ. As previously stated, R_t, $\mathbf{r_c}$ and $\mathbf{r_f}$ are inputs and they are defined by the trajectory generator which was described in Sect. 4. Moreover, $\mathbf{r_{ch}}$ has a fixed value which can be obtained using CAD data. Keeping these in mind, $\mathbf{r_{hf}}$ is computed as follows:

$$\mathbf{r_{hf}} = R_t^T(\mathbf{r_f} - \mathbf{r_c}) - \mathbf{r_{ch}} \tag{19}$$

Once $\mathbf{r_{hf}}$ is obtained for a given leg, the associated joint angles are computed using an analytical inverse kinematics solution. In the case of RoboCat-1, there are no hip roll joints, so that all y-axis components are not needed. Moreover, in the case or RoboCat-1, yawing motion for the upper torso is not possible.

The main advantage of this scheme is that the upper torso orientation is kinematically decoupled. One may produce a trot-walking motion for a given CoM and feet positions, while the upper torso orientation can be freely designated. This allows us to introduce a separate orientation controller without interfering with the trot-walking motion. Any combination of rotations is applicable provided that the kinematic limits are not exceeded.

6 Feedback controller units

6.1 Compensation schemes: friction and dynamics

Prior to the synthesis of feedback controllers, it is advantageous to identify and compensate joint friction hysteresis, namely Coulomb and viscous friction. For this purpose, we make use of the technique proposed in Murakami et al. (1993) to experimentally determine joint Coulomb and viscous friction parameters. After the identification process, a friction compensation torque, τ_{fr} is computed and imposed to the system, as depicted in Fig. 3. Furthermore, the full dynamics compensation (inertia, coriolis&centrifugal and gravity) is added via a floating-base inverse dynamics computation scheme. In doing so, the PD servo controller gains can be set to low values, as the compensation schemes detailed above serve as feed-forward control inputs.

Moreover, inertia terms require the measurement of joint accelerations, i.e., the second derivative of encoder readings. Direct differentiation produces a considerably high amount of noise. In order to compute feasible joint acceleration data we utilized an approximate differentiation method that is proposed in Murakami et al. (1993), which performs well for practical applications.

6.2 Virtual admittance control

6.2.1 *Computation of joint force references*

In Virtual Admittance Control, position and force references are processed in a simultaneous fashion. Thus, we need to define joint force references in accordance with the given target trajectory. To achieve this, we compute the vertical foot force references in each foot, namely F_{zrefj}, as follows:

$$Mg = \sum_{j=1}^{4} F_{zrefj}; \tag{20}$$

$$X_{copref} = \frac{\sum_{j=1}^{4}\left(F_{zrefj}\, r_{cfxrefj}\right)}{\sum_{j=1}^{4} F_{zrefj}}; \tag{21}$$

$$Y_{copref} = \frac{\sum_{j=1}^{4}\left(F_{zrefj}\, r_{cfyrefj}\right)}{\sum_{j=1}^{4} F_{zrefj}}. \tag{22}$$

In (20)–(22), M is the total mass, g is the gravitational acceleration, F_{zrefj} is the jth foot vertical force reference, X_{copref} and Y_{copref} are assigned CoP references, $r_{cfxrefj}$ and $r_{cfyrefj}$ refer to the reference displacement between the CoM and the jth foot, through x-axis and y-axis respectively (see Fig. 9). In (20), the total vertical force is equalized to the robot weight, since there is no vertical acceleration during trot-walking. Equations (21) and (22) allow for a choice of force distribution; usually both CoP references are kept at zero.

Since we have 4 variables (F_{zref1}, F_{zref2}, F_{zref3}, F_{zref4}), we need to define one more equation, in addition to (20)–(22). This can be obtained from a zero-yawing moment (τ_z) which can be expressed as follows:

$$\begin{aligned}\tau_z = {} & F_{yref1}r_{cfxref1} - F_{xref1}r_{cfyref1} + F_{yref2}r_{cfxref2}\\ & -F_{xref2}r_{cfyref2} + F_{yref3}r_{cfxref3} - F_{xref3}r_{cfyref3}\\ & +F_{yref4}r_{cfxref4} - F_{xref4}r_{cfyref4} = 0.\end{aligned} \tag{23}$$

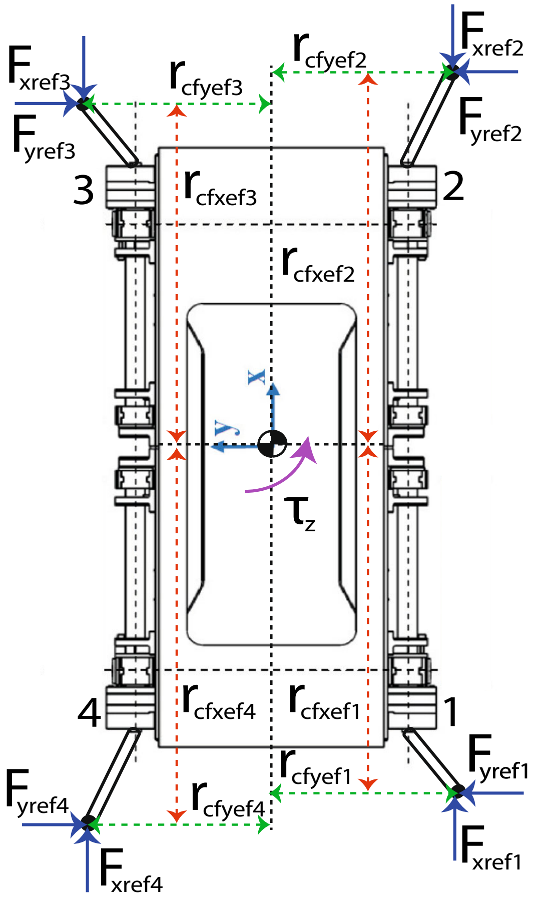


Fig. 9 CAD drawing of HyQ, top view (x–y plane). $r_{cfxrefj}$ and $r_{cfyrefj}$ vectors are illustrated on the drawing, together with x-axis and y-axis forces

In (23), $r_{cfxrefj}$ and $r_{cfyrefj}$ are the x-axis and the y-axis referential position values, defined between the CoM and the jth foot position. These parameters can be viewed in Fig. 9, and are obtained from the target trajectory generator (see Fig. 3). In contrast to (20)–(22), Eq. (23) is related to the x-axis (F_{xrefi}) and y-axis (F_{yrefi}) forces. Since the robot has point contact feet, the resultant ground reaction force is decomposed as illustrated in Fig. 10. Thus, (F_{xrefi}) and (F_{yrefi}) can be expressed in terms of F_{zrefi}, so as to yield the 4th equation:

$$F_{xrefj} = F_{zrefj} \tan(q_{1refj} + q_{2refj}) \sec q_{0refj}; \tag{24}$$

$$F_{yrefj} = -F_{zrefj} \tan q_{0refj}; \tag{25}$$

As shown in Fig. 10, q_{0refj}, q_{1refj}, and q_{2refj} subsequently symbolize hip_{aa}, hip_{fe}, and $knee_{fe}$ joints. These joint values are obtained from the leg kinematics model (see Fig. 3) prior to the joint force reference calculation task. Plugging (24) and (25) into (23), we finally acquire the 4th equation which, together with (20), (21), (22), allows us to calculate the vertical foot references for each foot, namely F_{zrefj}.

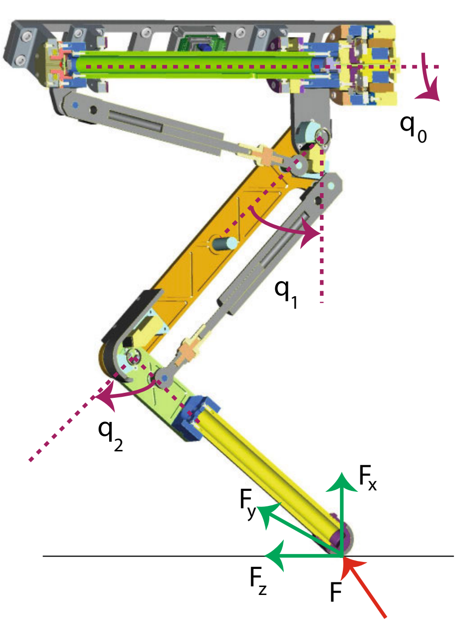


Fig. 10 Ground reaction force at the tip of the foot. Since the foot contact is considered as a point, only translational forces are considered

$$\overbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \\ r_{cfxref1} & r_{cfxref2} & r_{cfxref3} & r_{cfxref4} \\ r_{cfyref1} & r_{cfyref2} & r_{cfyref3} & r_{cfyref4} \\ \mu_1 & \mu_2 & \mu_3 & \mu_4 \end{bmatrix}}^{P_f} \overbrace{\begin{bmatrix} F_{zref1} \\ F_{zref2} \\ F_{zref3} \\ F_{zref4} \end{bmatrix}}^{F_z ref}$$

$$= \overbrace{\begin{bmatrix} Mg \\ MgX_{copref} \\ MgY_{copref} \\ 0 \end{bmatrix}}^{B_f} \tag{26}$$

$$\mathbf{F_{zref}} = P_f^{-1}\mathbf{B_f} \tag{27}$$

In (26), the following sub-expressions are used to ease the calculations.

$$\mu_j = \sec q_{0refj}(r_{cfxrefj} \sin q_{0refj} + r_{cfyrefj} \tan(q_{1refj} + q_{2refj})) \quad (j = 1, 2, 3, 4) \tag{28}$$

Equation (27) enables us to compute the vertical force references for each foot, during the double support phase where all 4 feet are on the ground. During the single support phases, we have only 2 feet on the ground; thus, we only utilize (21) and (22) to compute the vertical force references. In this case, the swing foot force references are naturally set to zero. Once we calculate the vertical force reference for the jth foot (F_{zrefj}), we can compute the horizontal and lateral components (F_{xrefj}, F_{yrefj}) using (24) and (25). Having computed all the 3-D referential force vectors for the ith leg, we make use of Jacobian transpose to obtain the referential joint forces.

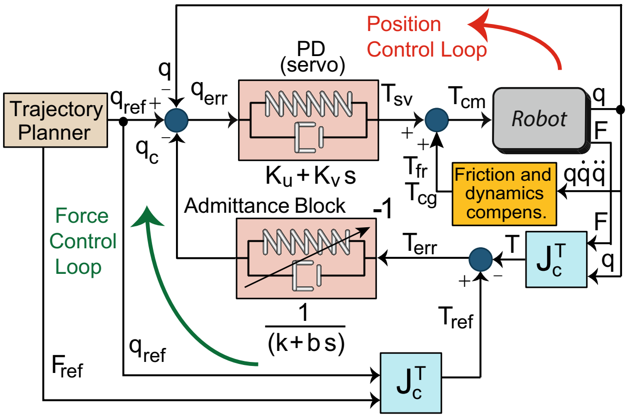


Fig. 11 Virtual admittance control in the servo loop. The *red* and *green arrows* respectively indicate the position and force control loops. When there is force error, joint position reference q_{ref} is updated ($q_{ref} := q_{ref} - q_c$) to comply with the force constraints, to the extent permitted by the virtual admittance coefficients (the smaller the coefficients, the larger q_c is generated for a given force error). Once force error vanishes, q_c converges to zero, and therefore, the system becomes a purely position controlled

Note that the above computation is explained for the case of HyQ, since it is a more generalized case due to the additional hip roll joint. For the case of RoboCat-1, there is no roll joint; therefore, its feet cannot move in the y-axis. Keeping this in mind, $r_{cfyrefj}$ are constant and q_{0refj} are set to zero for the case of RoboCat-1.

6.2.2 Hybrid joint position/force feedback: an antagonistic feedback method

The proposed controller diagram is depicted in Fig. 11 for a single leg. In this block diagram, T_{ref}, T_{sv}, T_{fr}, T_{cg}, T_{cm}, T, and T_{err} respectively indicate referential, PD output, friction compensation, full dynamics compensation, command, actual (measured) and error torques. q_{ref}, q and q_c denote referential joint angles, actual joint angles, and the output of Virtual Admittance Controller. K_p and K_v are diagonal matrices that store PD gains. Similarly, k and b are diagonal matrices which include the virtual admittance block coefficients. F is the measured force at the tip, F_{ref} is the referential tip force whose computation is carried out in accordance with the target trajectory as explained in the previous sub-subsection.

To begin with, the trot-walking trajectory generator and the leg kinematics scheme provide the joint angle references which are passed to the PD servo controller. On top of this scheme, we added a force control loop in which the joint force errors are processed via an admittance block to compute the corresponding joint displacements, namely q_c vector. This vector is computed for a single leg as follows:

$$\mathbf{T_{err}} = J_c^T(\mathbf{q_{refj}})\mathbf{F_{refj}} - J_c^T(\mathbf{q_j})\mathbf{F_j} \tag{29}$$

$$\mathbf{q_c} = \begin{bmatrix} \frac{1}{k_0+b_0 s} & 0 & 0 \\ 0 & \frac{1}{k_1+b_1 s} & 0 \\ 0 & 0 & \frac{1}{k_2+b_2 s} \end{bmatrix} \mathbf{T_{err}} \tag{30}$$

Since we compensate for the joint friction (both Coulomb and viscous friction) and the full dynamics load (inertia, coriolis&centrifugal, gravity) the Jacobian transpose performs well for the torque computation. In (30), (k_0, b_0), (k_1, b_1), (k_2, b_2) couples refer to virtual admittance coefficients for hip roll, hip pitch and hip knee joints. s is the Laplace domain variable.

By perturbing the joint reference for about q_c degrees, joint force feedback is provided in a parallel manner to the PD control loop. In this scheme, the PD controller is responsible to make sure that position tracking is achieved. However, within the presence of force errors (disturbances, ground impact, etc.), q_{ref} is updated via the secondary feedback ($q_{ref} := q_{ref} - q_c$) to comply with force constraints, by decreasing the joint stiffness. Once the external effect that causes the force error disappears, q_c term vanishes and the joint turns back to its initial stiffness. The trade-off between position and force tracking is adjusted via the admittance block parameters. In other words, the force control loop and the position control loop work in an antagonistic configuration; the joint becomes compliant only when necessary to handle force errors. If there is no force error, e.g., the leg is in a swing phase, the controller automatically prioritizes joint tracking as q_c converges to zero.

6.2.3 On the stability of virtual admittance control

The necessary and sufficient condition to guarantee stability for a robotic system that is interacting with a given passive environment is that the system is also passive, e.g., injecting no energy, at the interaction port (Colgate and Hogan 1989; Fasse 1987; Boaventura et al. 2013). Since most terrain types are passive, the robot leg must exhibit a passive behavior, so that the stability can be ensured.

Unlike passively compliant systems, active compliance controllers, such as virtual admittance control, inject energy into the system in a way so as to emulate the compliant behavior. While this prevents them to be *intrinsically* passive, they can still render a passive behavior depending on the compliance control gain selection. In order to see whether the system renders a passive behavior, we may use the manipulator admittance transfer function matrix ($Y_j(s)$) for the jth leg, which relates the output velocity (υ_j) and force (F_j) at the jth interaction port (foot).

$$\upsilon_j = Y_j(s)F_j \tag{31}$$

$Y_j(s)$ can be obtained using the block diagram in Fig. 11. In order to demonstrate passivity for the system, two following conditions must be met.

1. $Y_j(s)$ has no poles in the right hand plane; $\Re(s) > 0$.
2. $Y_j(s) + Y_j^*(s)$ is positive semi-definite in $\Re(s) > 0$, where $Y_j^*(s)$ is the conjugate transpose of $Y_j(s)$.

When $Y_j(s)$ has no poles in the right hand plane, the second condition can be reduced to this condition: $Y_j(\beth\omega) + Y_j^*(\beth\omega)$ is positive semi-definite for all real ω.

If the minimum eigenvalue of $Y_j(\beth\omega)+Y_j^*(\beth\omega)$ is positive, the system is considered to exhibit passive behavior, therefore, its interaction with the passive environment (terrain in our case) is stable.

In the case of sampled data control systems, Colgate proposed a method in which the port of interaction is assumed to be sampled. Using this approximation, the corresponding discrete-time admittance transfer function matrix, $Y_j(z)$ is computed with a phase lag correction term of $\frac{\omega dt}{2}$ rad, where dt is the sampling time (Colgate 1994). In order to guarantee passivity, and thus stability, the minimum eigenvalue of $Y_j(z) + Y_j^*(z)$ must be positive and the poles of $Y(z)$ should be within the unit circle. Therefore, one should define the compliance controller gains in such a way that the corresponding admittance transfer function satisfies the stability conditions mentioned above. For a systematic set of procedures to ensure stability in this manner, refer to Boaventura et al. (2013).

6.3 Orientation control based on angular momentum

While a quadruped robot performs periodic motions, such as trot-walking, the upper torso develops inevitable fluctuations. This issue potentially threatens the postural balance. Its influence becomes more significant for cyclic motions that are performed over uneven terrains. To overcome this issue, we propose a secondary controller unit that regulates the angular momentum rate change about upper torso CoM, via the characterization of rotational inertia.

Depending on the leg lengths, the upper body may develop rotations about the roll axis (θ), the pitch axis (ψ), and the yaw axis (ϕ). Keeping this in mind, the angular momentum rate of change about upper torso CoM is formulated in terms of θ, ψ and ϕ, along with their corresponding time derivatives.

First, the angular momentum rate change about upper torso CoM (centroidal torque about CoM) can be computed using Euler's equations of motion:

$$\dot{H}_x = I_{xx}\dot{\omega}_x - (I_{yy} - I_{zz})\omega_y\omega_z \tag{32}$$

$$\dot{H}_y = I_{yy}\dot{\omega}_y - (I_{zz} - I_{xx})\omega_z\omega_x \tag{33}$$

$$\dot{H}_z = I_{zz}\dot{\omega}_z - (I_{xx} - I_{yy})\omega_x\omega_y \tag{34}$$

In (32)–(34), I_{xx}, I_{yy}, I_{zz} are moments of inertia defined at the upper torso CoM, ω_x, ω_y, ω_z are angular velocity terms, $\dot{H}_x$, $\dot{H}_y$, $\dot{H}_z$ are rate changes of angular momentum about the torso CoM; all associated with the roll, pitch and yaw axes respectively. Products of inertia are not considered due to the fact that they are 500 times smaller than the diagonal elements for our robots. Evaluating (32)–(34), we should express angular velocity and angular acceleration vectors, in terms of θ, ψ and ϕ. To achieve this, the upper torso orientation, R_t, can be expressed by multiplying the rotation tensors about the roll (R_θ), pitch (R_ψ), and yaw (R_ϕ) axes:

$$\begin{aligned} R_t &= R_\theta R_\psi R_\phi \\ &= \begin{bmatrix} c_\psi c_\phi & -c_\psi s_\phi & s_\psi \\ (c_\theta s_\phi + c_\phi s_\theta s_\psi) & (c_\theta c_\phi - s_\theta s_\psi s_\phi) & -c_\psi s_\theta \\ (s_\theta s_\phi - c_\theta c_\phi s_\psi) & (c_\phi s_\theta + c_\theta s_\psi s_\phi) & c_\theta c_\psi \end{bmatrix} \end{aligned} \tag{35}$$

In (35), $c_\theta, s_\theta, c_\psi, s_\psi, c_\phi$ and s_ϕ sequentially refer to $\cos\theta$, $\sin\theta$, $\cos\psi$, $\sin\psi$, $\cos\phi$ and $\sin\phi$. Using a tensorial approach, angular velocity is expressed in the skew-symmetric form ($\omega^\dagger$):

$$\begin{aligned} \omega^\dagger &= \dot{R}_t R_t^T \\ &= \begin{bmatrix} 0 & -(\dot{\psi}s_\theta + \dot{\phi}c_\theta c_\psi) & (\dot{\psi}c_\theta - \dot{\phi}c_\psi s_\theta) \\ (\dot{\psi}s_\theta + \dot{\phi}c_\theta c_\psi) & 0 & -(\dot{\theta} + \dot{\phi}s_\psi) \\ (\dot{\phi}c_\psi s_\theta - \dot{\psi}c_\theta) & (\dot{\theta} + \dot{\phi}s_\psi) & 0 \end{bmatrix} \end{aligned} \tag{36}$$

The angular velocity vector can be simply acquired using the angular velocity vector-tensor identity, as shown in (37)–(39). Furthermore, the angular acceleration vector can be computed via differentiation as well.

$$\omega_x = \dot{\theta} + \dot{\phi}s_\psi \tag{37}$$

$$\omega_y = \dot{\psi}c_\theta - \dot{\phi}c_\psi s_\theta \tag{38}$$

$$\omega_z = \dot{\psi}s_\theta + \dot{\phi}c_\theta c_\psi \tag{39}$$

$$\dot{\omega}_x = \ddot{\theta} + \ddot{\phi}s_\psi + \dot{\psi}\dot{\phi}c_\psi \tag{40}$$

$$\dot{\omega}_y = \ddot{\psi}c_\theta - \dot{\theta}(\dot{\psi}s_\theta + \dot{\phi}c_\theta c_\psi) - \ddot{\phi}c_\psi s_\theta + \dot{\psi}\dot{\phi}s_\theta s_\psi \tag{41}$$

$$\dot{\omega}_z = \dot{\theta}(\dot{\psi}c_\theta - \dot{\phi}c_\psi s_\theta) + \ddot{\psi}s_\theta + \ddot{\phi}c_\theta c_\psi - \dot{\psi}\dot{\phi}c_\theta s_\psi \tag{42}$$

Subsequently, (37)–(42) are inserted to (32)–(34) to obtain ($\dot{H}_x$, $\dot{H}_y$, $\dot{H}_z$), namely, the rate change of angular momentum about the upper torso CoM (centroidal torque in short) associated with the roll, pitch and yaw axes.

$$\dot{H}_x = I_{xx}(\ddot{\theta} + \ddot{\phi}s_\psi + \dot{\psi}\dot{\phi}c_\psi) - I_{yy}\eta_2\eta_1 + I_{zz}\eta_2\eta_1 \tag{43}$$

$$\begin{aligned} \dot{H}_y &= I_{yy}(\dot{\psi}\dot{\phi}s_\theta s_\psi - \dot{\theta}\eta_2 + \ddot{\psi}c_\theta - \ddot{\phi}c_\psi s_\theta) \\ &\quad + I_{xx}\eta_2\eta_3 - I_{zz}\eta_2\eta_3 \end{aligned} \tag{44}$$

$$\begin{aligned} \dot{H}_z &= I_{zz}(\dot{\theta}\eta_1 + \ddot{\psi}s_\theta + \ddot{\phi}c_\theta c_\psi - \dot{\psi}\dot{\phi}c_\theta s_\psi) \\ &\quad - I_{xx}\eta_1\eta_3 + I_{yy}\eta_1\eta_3 \end{aligned} \tag{45}$$

η_1, η_2 and η_3 appear to be repeating sub-expressions which are expressed as in the following:

$$\eta_1 = \dot{\psi}c_\theta - \dot{\phi}c_\psi s_\theta \tag{46}$$

$$\eta_2 = \dot{\psi}s_\theta + \dot{\phi}c_\theta c_\psi \tag{47}$$

$$\eta_3 = \dot{\theta} + \dot{\phi}s_\psi \tag{48}$$

Equations (43)–(45) enable us to compute the centroidal torque for a given set of (θ, ψ, ϕ) parameters, together with their time derivatives. With the help of these equations, one can calculate the desired centroidal torque by considering the referential torso angle variations, $(\theta_{ref}, \psi_{ref}, \phi_{ref})$. Next, the actual centroidal torque can be computed using gyro sensor measurements, (θ, ψ, ϕ). By subtracting the referential centroidal torque from the actual one, it is possible to acquire the centroidal torque error for the roll and pitch axis, namely $\dot{H}_{xerr}$, $\dot{H}_{yerr}$, and $\dot{H}_{zerr}$. In order to regulate $\dot{H}_{xerr}$, $\dot{H}_{yerr}$, and $\dot{H}_{zerr}$, we insert them to admittance blocks to compute the necessary compensating rotational motion, and then feed it back to the orientation inputs, as illustrated in Fig. 3.

$$\theta_c = \frac{\dot{H}_{xerr}}{k_r + s\, b_r}; \quad \psi_c = \frac{\dot{H}_{yerr}}{k_p + s\, b_p}; \quad \phi_c = \frac{\dot{H}_{zerr}}{k_y + s\, b_y}. \qquad (49)$$

In (49), (k_r, b_r), (k_p, b_p) and (k_y, b_y) are virtual spring damper coefficients; $(\theta_c, \psi_c, \phi_c)$ are compensating rotational motions that are fed-back to orientation inputs (see Fig. 3). Virtual admittance couples, $(k_r, b_r, k_p, b_p, k_y, b_y)$, are defined by considering (θ, ψ, ϕ) angles and their rate changes (not necessarily the associated angular velocity). Figure 12 provides an illustration to explain the controller logic. The main reason of implementing the admittance blocks is that our orientation inputs are rotational angles, as may be examined in Fig. 3. From centroidal torque to angular position, one needs to introduce an admittance block to generate physically consistent compensating motion. In addition to physical consistency, such a controlling approach suppresses undesired upper torso fluctuations in an actively-compliant manner. Note that this admittance block is essentially different than the one we used in the Virtual Admittance Control block; the compliant effect is introduced in the Cartesian frame in the Angular Momentum Controller, via assessing gyro sensor information.

7 Experimental results and evaluations

In order to verify the proposed locomotion control framework, an extensive set of experiments were conducted on the RoboCat-1 and the HyQ. These experiments are listed as below:

- Experiment #1: RoboCat-1 was dropped from a height of 30 cm.
- Experiment #2: RoboCat-1 performed trot-walking on a level surface.
- Experiment #3: RoboCat-1 performed trot-walking on a terrain that was covered stones with various shapes and sizes.
- Experiment #4: HyQ performed trot-walking on a level surface.
- Experiment #5: HyQ performed trot-walking over gradually-placed wooden blocks with 3 cm thickness.

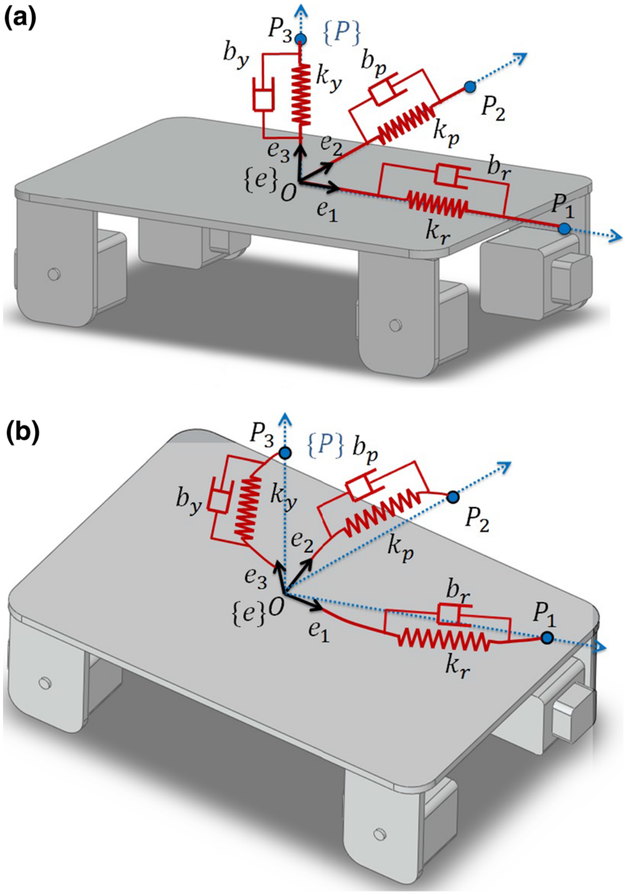


Fig. 12 Coordinate systems {P} and {e} are respectively defined with respect to desired and actual orientation profiles. **a** When both {P} and {e} coincide, i.e., orientation error is negligibly small, the spring-damper couples are in their natural (rest) position. **b** When the system experiences an orientation error, the orientation controller tries to regulate the upper torso attitude in a way that {e} converges towards {P}, via the utilization of the virtual spring-damper couples

Figure 13 displays snapshots for experiments #2–#5. Refer to the multimedia attachment to view the video that includes the experimental scenes.

7.1 Experiments on RoboCat-1

As explained in Sect. 2, RoboCat-1 is a low-cost and comparatively simple quadruped robot on which we can conduct relatively risky experiments. Such experiments include dropping from a height, and execute the locomotion with and without the specific controllers to observe their individual contributions in the locomotion behavior.

7.1.1 Experiment #1: dropping RoboCat-1 from a height of 30 cm

One of the useful properties of the virtual admittance controller lies on its shock-absorbing capability. To experimen-

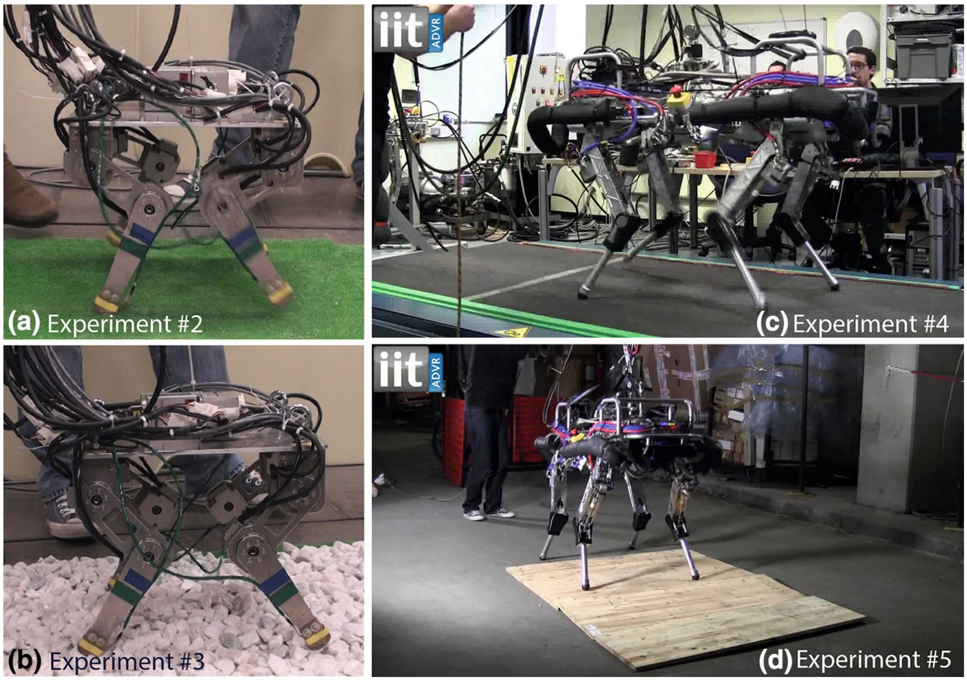


Fig. 13 Snapshots from the last 4 experiments. **a** Experiment #2, **b** Experiment #3, **c** Experiment #4, **d** Experiment #5

tally confirm this property, RoboCat-1 was released from a height of 30 cm to demonstrate the importance of active compliance that was provided by means of force feedback in our controller. In this experiment, no additional hardware modification was performed on RoboCat-1; the foot tips are covered using a very hard rubber tip with 0.5 cm thickness, as done in the usual operation (see Fig. 1). The robot was dropped onto a level surface.

The dropping experiment was conducted twice; (i) when the virtual admittance controller unit was activated, (ii) when the virtual admittance controller unit was deactivated. Ground reaction force and hip joint torque profiles for this experiment can be viewed in Fig. 14, where solid blue and red lines indicate variations from experimental trials with and without virtual admittance controller, respectively. The dotted black line indicates the admissible joint torque limit. The yellow hatched areas depict the variations after the robot hits the ground.

When the virtual admittance controller was deactivated, the robot could not exhibit compliant behavior, which then led to the a ground reaction force peak that was approximately 8 times larger than its weight. This is because the robot was stiffly controlled and not capable of handling large external forces. Consequently, its joints, for instance, right leg hip joint, did exceed the maximum torque limit when the robot hit the ground, and therefore, the motor drive was automatically halted.

When the virtual admittance controller was activated, RoboCat-1 was able to dissipate the excessive ground reaction force in an actively compliant manner as the robot hit the ground. This is depicted in Fig. 14; the ground reaction force profile indicates a peak of 180 N and joint torques were within the actuator limitations. Considering this result, we may claim that the use of virtual admittance controller enables the robot to compliantly handle unexpected disturbances and excessive ground reaction force peaks. This provides a clear advantage in the quadruped locomotion control context.

7.1.2 Experiment #2: RoboCat-1 trot-walked on a level surface

In this experiment, RoboCat-1 performed dynamic trot-walking on a level surface that was covered with an artificial grass carpet. A snapshot from this experiment can be seen in Fig. 13a. The torso orientation reference was assigned as level to the ground. 4-legged stance time (equivalent planar biped model double support phase), 2 diagonally paired legged stance time (equivalent planar biped model single support phase), constant torso height, stride length, forward velocity parameters were assigned as 0.2 s, 0.1 s, 0.2 m, 6.4 cm, 0.35 km/h, respectively. The experiment was restricted with 10 consecutive steps so as to display the cyclic data clearly, although the robot can walk as much as desired (see the multimedia attachment). When the virtual admittance controller is activated, RoboCat-1 can perform trot-walking up to the maximum velocity of 0.65 [km/h]; however, it cannot repeat this performance when the virtual admittance controller is deactivated. Keeping this in mind, we limited the forward velocity at 0.35 [km/h], so that we can repeat this experiment under the same conditions (with/without virtual

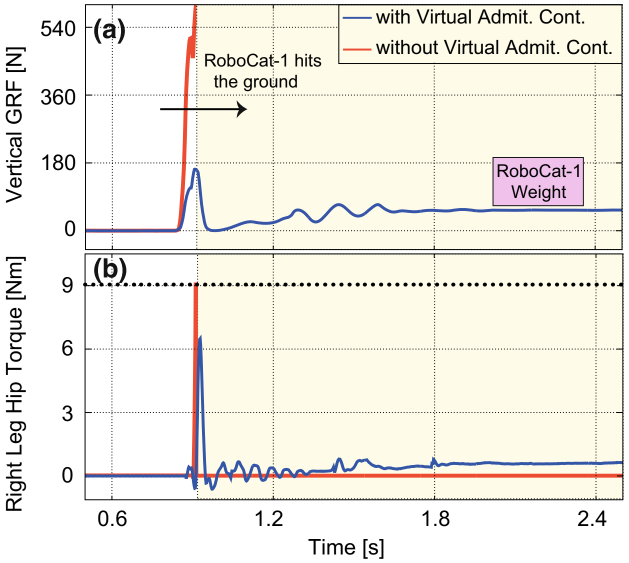


Fig. 14 Results of Experiment #1. **a** Vertical ground reaction force (GRF) profiles during Experiment #1. **b** Right leg hip joint torque variations during Experiment #1. Once the limit was exceeded without the virtual admittance controller, the motor was automatically halted and torque output was zeroed. The *dotted black line* indicates the admissible joint torque limit

admittance controller) to observe the virtual admittance controller's effectiveness on the locomotion behavior.

Referential and actual x-axis CoM trajectories with respect to the world frame are displayed in Fig. 15a with dotted black and solid red lines. Feet displacements are plotted with the equivalent planar biped model in mind (see Sect. 4), which are obtained using the geometric mean of diagonally-paired feet positions in x-axis. Scrutinizing this figure, it may be observed that the pattern generator was able to synthesize feasible, smooth and continuous feet and CoM trajectories which were seamlessly tied through all the phases (left front–right hind 2-legged support, 4-legged support, right front–left hind 2-legged support), as well as transitions between these phases, throughout the entire trot-walking period. CoM velocity and acceleration profiles were strictly continuous as well, however, not plotted.

The x-axis and y-axis CoP measurements are plotted in Fig. 15b, c, together with the associated support polygons. In these figures, solid red lines point out CoP measurements, while dotted green and blue lines indicate support polygon boundaries. Depending on the given stance phase (4-legged stance or 2 diagonally paired legged stance), the support polygon boundaries actively change. Examining this figure, the CoP response always stayed within the support polygon boundaries, indicating that the robot executed the trot-walking in a dynamically balanced manner.

The ground reaction force profiles are depicted in Fig. 15d, in which solid magenta lines shows the summation of the ground reaction force of diagonally-paired right front and left hind feet. Identically, the solid green line displays the summation of the ground reaction force of diagonally-paired left front and right hind feet. Observing this graph, we may see that no excessive ground reaction force peaks occurred throughout the entire trot-walking period. Moreover, the forces based on robot loading appeared to be distributed equally to the diagonally-paired legs, indicating that the cyclic trot-walking was executed in a consistent way. Since the virtual admittance controller introduced active compli-

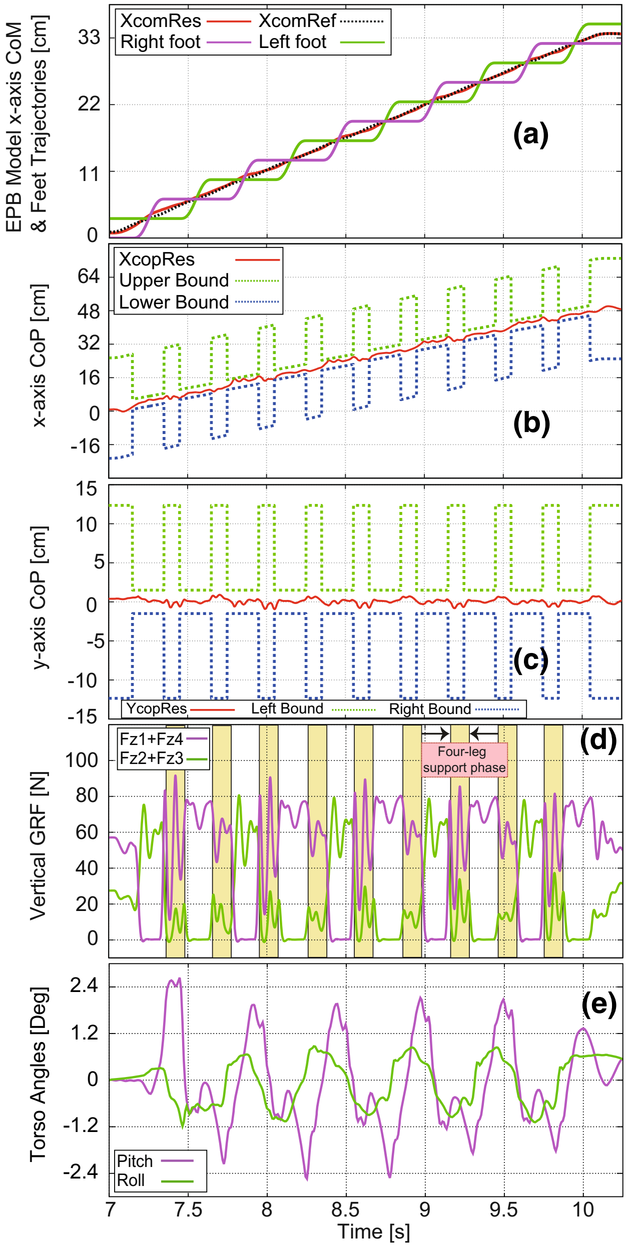


Fig. 15 Results of Experiment #2. **a** CoM and equivalent planar biped model (EPB) feet trajectories with respect to the world frame. **b** x-axis CoP response. **c** y-axis CoP response. **d** Ground reaction force (GRF) profiles of the diagonally-paired feet tips. **e** Torso angle variations

ance together with an associated force feedback control strategy, such favorable ground reaction force profiles were obtained.

The torso angle variations are shown in Fig. 15e, for pitch (solid magenta) and roll (solid green) axes. The pitch axis torso angle varied between ±2.4 degrees, while the roll axis torso angle was kept within a band of ±1.2 degrees. Evaluating this data, it can be judged that the robot did not suffer undesired torso angle variations which may have occurred due to large ground reaction force impacts or unequal reaction force distribution. Hence, the robot was able to execute trot-walking while preserving its postural balance.

The results displayed in Fig. 15 were obtained from an experiment in which the proposed controller was activated. In order to have a profound understanding on the effectiveness of the virtual admittance controller, we conducted additional trot-walking experiments of two different cases; (i) when the virtual admittance controller was activated, (ii) when the virtual admittance controller was deactivated. x-axis and z-axis torso acceleration variations, hip joint torque command and x-axis CoP error are displayed in Fig. 16, where solid blue and red lines indicate variations from experimental trials with and without the virtual admittance controller, respectively. Mean ± SD (Standard Deviation) values for 25 largest peak errors are displayed in Table 4, concerning the results given in Fig. 16.

When the virtual admittance controller was deactivated, the robot experienced relatively larger ground reaction force impacts, due to the absence of active compliance. This fact can be verified by the acceleration measurements which were directly collected using the on-board IMU sensor. Figure 16a, b display measured accelerations for x-axis and z-axis. When the virtual admittance controller was activated, the RMS (Root Mean Square) values of x-axis and z-axis acceleration profiles decreased for about 60 %. This result indicates that the virtual admittance controller is essential in dissipating ground reaction force impacts as it enabled the robot to exhibit compliant locomotion behavior.

Similar to the case in Experiment #1, the torque requirements were observed to decrease when the virtual admittance controller was activated. This is due to the fact that strict position control prioritizes joint tracking even within the presence of larger impacts, demanding relatively greater actuator torques. One may observe this phenomenon in Fig. 16c, in which the RMS value of torque command was 50 % less when the virtual admittance controller was activated. As described in Sect. 6, the virtual admittance controller decreases joint stiffness to handle large impacts by momentarily sacrificing joint tracking. In doing so, the locomotion is compliantly executed by adroitly managing the position/force trade-off.

As the virtual admittance controller manages the position/force trade-off thanks to its actively compliant architecture, ground reaction force was equally distributed well between robot legs. As a consequence, CoP errors appeared to be more containable compared to the case in which the virtual admittance controller was deactivated. Figure 16d illustrates the related result; CoP error showed more than a 50 % decrease in its RMS value, when the virtual admittance controller was activated. Therefore, the virtual admittance controller appears to play an important role in maintaining the dynamic equilibrium in the proposed locomotion control framework.

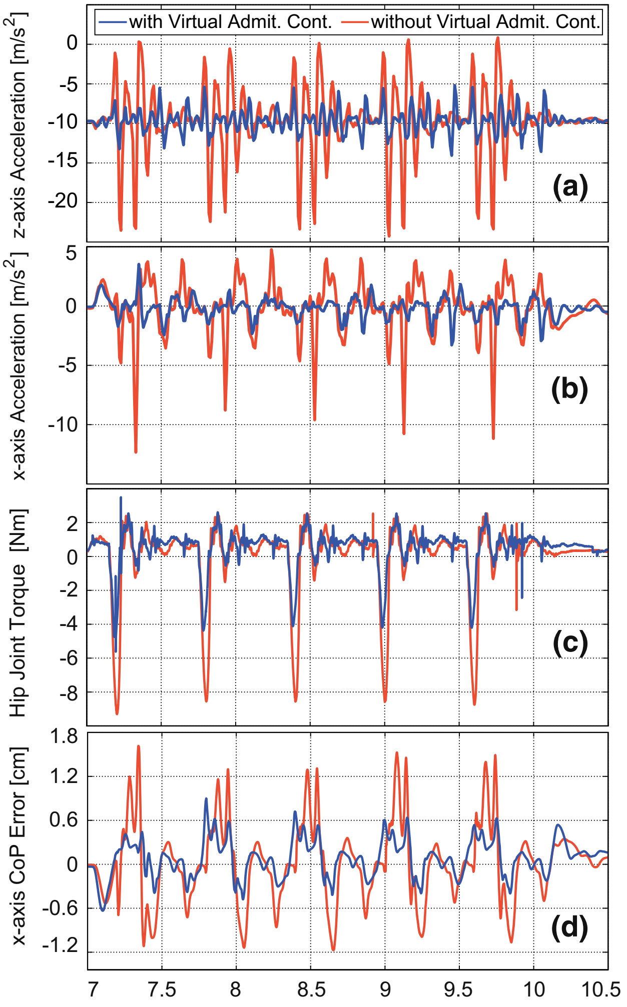


Fig. 16 Results from the virtual admittance controller verification experiments. **a** Vertical acceleration. **b** Horizontal acceleration. **c** Hip joint torque command. **d** x-axis CoP error

Table 4 Mean ± SD values for 25 largest peak errors in Fig. 16, both with and without VAC (Virtual Admittance Controller)

Measurement	No VAC	With VAC
z-axis acceleration (m/s^2)	9.1 ± 4.0	1.7 ± 0.9
x-axis acceleration (m/s^2)	4.7 ± 3.6	1.6 ± 1.2
Hip torque (Nm)	8.7 ± 0.4	4.2 ± 0.4
x-axis CoP error (cm)	1.0 ± 0.4	0.3 ± 0.2

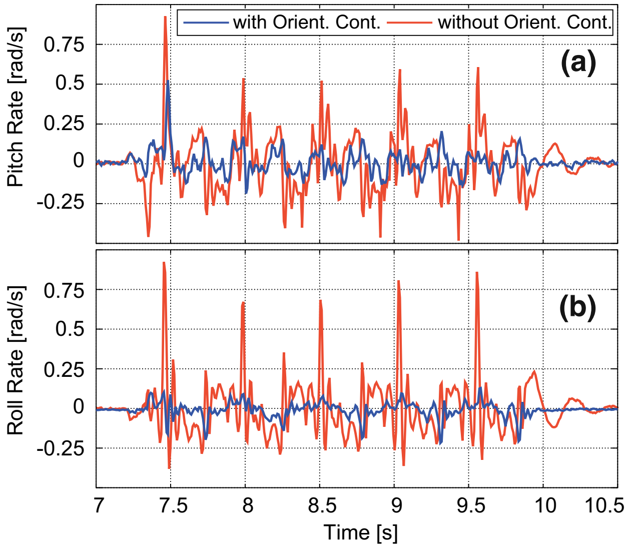


Fig. 17 Results from the verification of orientation controller experiments. **a** Pitch rate. **b** Roll rate

Table 5 Mean ± SD values for 25 largest peak errors in Fig. 17, both with and without Orientation Controller

Measurement	No orient. con.	With orient. con.
Pitch rate (rad/s)	0.5 ± 0.2	0.2 ± 0.1
Roll rate (rad/s)	0.5 ± 0.3	0.1 ± 0.0

In addition to the virtual admittance controller, we implemented a secondary feedback loop to control the torso orientation. In order to confirm its effectiveness, we conducted further trot-walking experiments with and without the orientation controller in the loop. In these experiments, the virtual admittance controller was activated in both cases. Pitch and roll rates, collected via the on-board IMU, can be viewed in Fig. 17, where solid blue and red lines indicate variations from experimental trials with and without the orientation controller, respectively. Mean ± SD values for 25 largest peak errors are displayed in Table 5, concerning the results given in Fig. 17.

As a result of these experiments, the RMS error of pitch rate showed more than a 60 % decrease when the orientation controller was in the loop. Similar to this result, the RMS error of roll rate decreased for about 70 %. Judging by this result, the orientation control appears to be very efficient in suppressing undesired torso angle fluctuations, providing a feasible trot-locomotion performance, on top of the virtual admittance controller feedback unit.

7.1.3 Experiment #3: RoboCat-1 trot-walked on a rocky terrain

Having completed trot-walking experiments on a level surface in Experiment #2, RoboCat-1 was employed to execute the same locomotion on a rocky terrain that was covered with stones of various shapes. A snapshot from this experiment can be seen in Fig. 13b. Trot-walking experiment parameters were kept identical with the ones used in Experiment #2. Results are given in Fig. 18. The terrain was unperceived, i.e., the robot did not utilize any terrain map to heuristically determine the surface shape. Since this experiment was conducted under comparatively challenging conditions (trot-walking on a rocky terrain), all the feedback control units were activated for safety reasons.

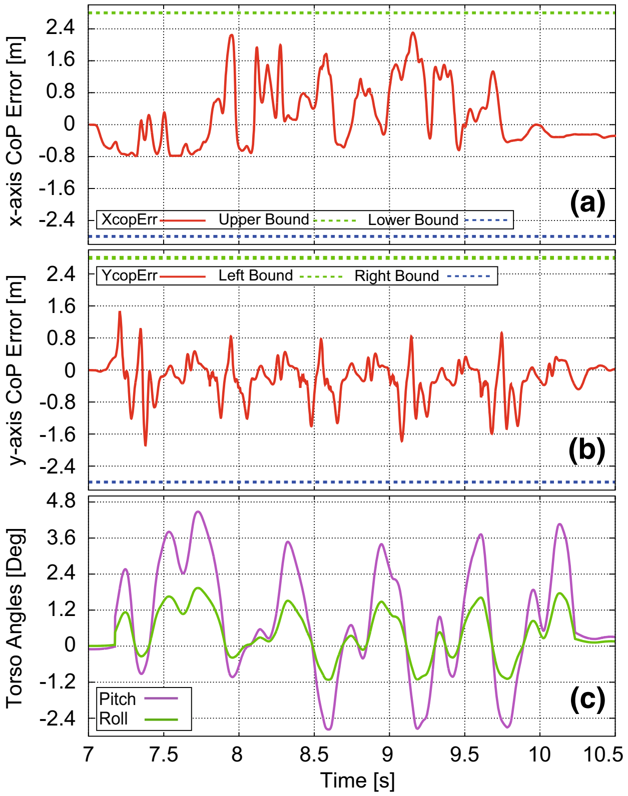


Fig. 18 Results of experiment #3. **a** x-axis CoP error. **b** y-axis CoP error. **c** Torso angle variations

CoP error variations are plotted in Fig. 18a and in Fig. 18b, for x-axis and y-axis. In these figures, solid red, dotted green and dotted blue lines respectively display CoP errors, and allowable CoP error boundaries. Examining this figure, the CoP error variations are always within the region of allowable boundaries, indicating that the robot exhibited dynamically balanced trot-walking locomotion characteristics throughout the whole experiment period. This is due to the fact that the virtual admittance controller introduces active compliance in conjunction with a force feedback strategy, enabling the robot to handle ground reaction force impacts which may severely occur while executing legged locomotion on an unperceived uneven terrain. Therefore, the robot gains enhanced environmental interaction capabilities and could maintain its dynamic balance when walking on different types of surfaces.

Torso angle variations are displayed in Fig. 18c, for pitch (solid magenta) and roll (solid green) axes. The figure shows that the pitch angle was kept within the +4.6 to −2.5 degrees band. The roll angle variation also appeared to be quite contained, it varied between +1.8 to −1.2 degrees. Based on these results, we may claim that the robot was able to exhibit consistent and feasible locomotion characteristics while maintaining its postural balance on the given uneven terrain, as the robot torso orientation did not suffer from severe variations.

7.2 Experiments on HyQ

Once we concluded a thorough experimentation study on RoboCat-1, the proposed locomotion control frame was implemented on the quadruped robot HyQ. In doing so, we had a chance to re-validate the effectiveness of the proposed approach, and therefore, enrich the evaluation, in an attempt to have an indication whether the method could be implemented to a quadrupedal robot with greater mass and size. Since HyQ is a relatively large quadruped with sophisticated hardware, all the feedback controller units were activated.

7.2.1 Experiment #4: HyQ trot-walked on a treadmill

In this experiment, HyQ was used to execute dynamic trot-walking on a treadmill. A snapshot from this experiment can be seen in Fig. 13c. The torso orientation reference was assigned as level to the ground. The 4-legged stance time (equivalent planar biped model double support phase), 2 diagonally paired legged stance time (equivalent planar biped model single support phase), constant torso height, stride length, forward velocity parameters were set to 0.28 s, 0.14 s, 0.68 m, 26.5 cm, 1.26 km/h, respectively. The experiment was restricted with 8 consecutive steps so as to display the cyclic data clearly, although the robot can walk as desired (see multimedia attachments). Results are presented in Fig. 19.

Referential and actual x-axis CoM trajectories with respect to the world frame are depicted in Fig. 19a, and are shown with dotted black and solid lines. Feet trajectories are also plotted with respect to the world frame using solid magenta and green lines. Feet trajectories were computed with the equivalent planar biped model in mind (see Sect. 4), obtained using the geometric mean of diagonally-paired feet positions in x-axis. This figure clearly displays that the pattern generator provided feasible, smooth and continuous feet and CoM trajectories which were crucial for the success of dynamic trot-walking. The trajectories were connected seamlessly regardless of the given phase (left front–right hind 2-legged support, 4-legged support, right front–left hind 2-legged support), and as well as transitions between the phases.

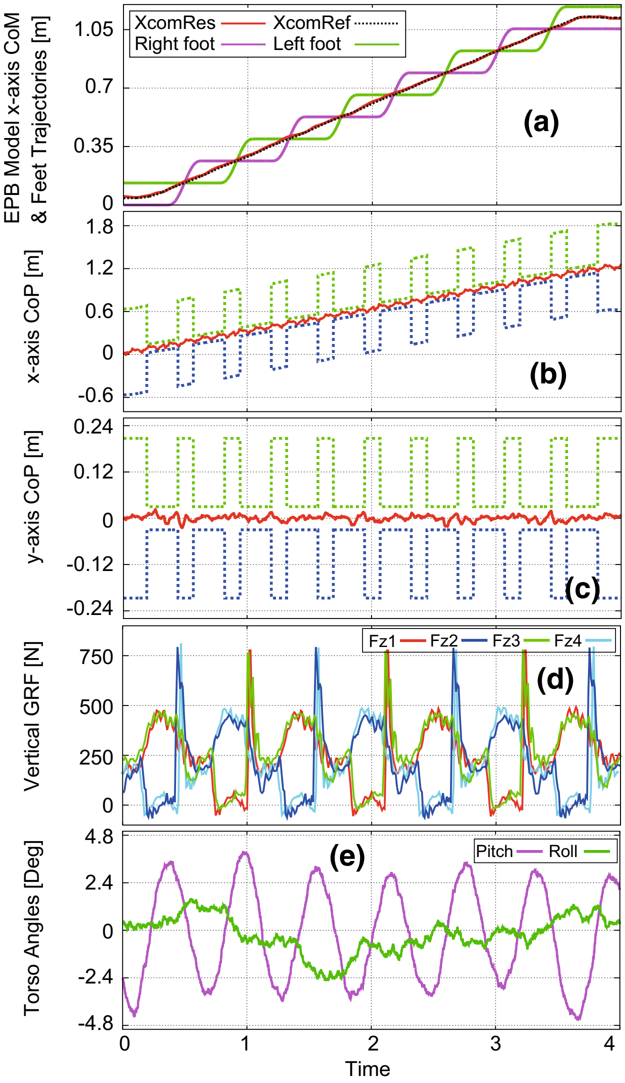


Fig. 19 Results of Experiment #4. **a** CoM and equivalent planar biped model (EPB) feet trajectories with respect to the world frame. **b** x-axis CoP response. **c** y-axis CoP response. **d** Ground reaction force (GRF) profiles on each feet. **e** Torso angle variations

The CoP measurements along the x-axis and the y-axis are illustrated in Fig. 19b, c, with the associated support polygons. Solid red line stands for the CoP response, while dotted green and blue lines point out the support polygon boundaries. It should be noted that the support polygon boundaries actively change depending on the specific stance phase (4-legged stance or 2 diagonally paired legged stance). Scrutinizing the figure, CoP responses strictly stayed within the support polygon boundaries, which substantiates the fact that HyQ maintained its dynamic balance throughout the whole locomotion period.

The ground reaction force profiles of each foot are shown in Fig. 19d. One of the prominent features of these ground reaction force profiles is that the forces based on robot loading were distributed equally to each leg, demonstrating that the locomotion was executed in a consistent manner. Moreover, there was no excessive ground reaction force peaks; the maximum peak was approximately 3 times larger than that of the robot weight. These positive results appear to be favorable characteristics of the virtual admittance controller which introduces active compliance that is associated with a force feedback control strategy.

The torso angle variations are displayed with solid magenta and green lines for pitch and roll axes, in Fig. 19e. Pitch angle showed a variance from +3.6 degrees to −4.8 degrees, while roll angle varied between +1.6 degrees to −2.4 degrees. These torso angle variation bands give us certain indications that the robot did not undergo abrupt changes while preserving the postural balance; therefore, it exhibited consistent and feasible trot-walking locomotion behavior.

Figs. 15 and 19 show that similar trot-walking performance was obtained from both robots, though they have distinct mechanical characteristics. Compared to RoboCat-1, leg length at rest is approximately 4 times longer for the case of HyQ; thus, stride length during walking was chosen larger. HyQ's mass and rotational inertia are approximately 11 times larger than that of RoboCat-1. Despite these apparent differences, both robots exhibited favorably similar trot-walking performances in which respective CoMs followed feasible trajectories, the dynamic equilibrium condition was always satisfied (see CoP measurements), ground reaction forces were equally distributed to the robot feet and torso angle variations were well controlled.

Among the quadrupedal robots in the literature, HyQ can be considered as a large sized robot, while RoboCat-1 is a medium to small size robot. With this in mind, Figs. 15 and 19, together, prove that the proposed approach could be implemented to most quadrupedal robots, regardless of its size. Although favorable results obtained from two different robots may not be enough to claim *generality* in the strict sense, this strongly indicates that the proposed approach is independent from the target robot mass and size.

7.2.2 Experiment #5: HyQ trot-walked on gradually placed wooden blocks

Upon the completion of level surface experiments on a treadmill, HyQ was used to conduct uneven surface trot-walking experiments. To this end, 3 wooden boards with a thickness of 3 cm were gradually placed on top of each other, both creating unevenness and a moderate slope. A snapshot from this experiment can be seen in Fig. 13d. Trot-walking parameters were kept the same; however, only target forward velocity was reduced to 0.36 [km/h]. Results are provided in Fig. 20.

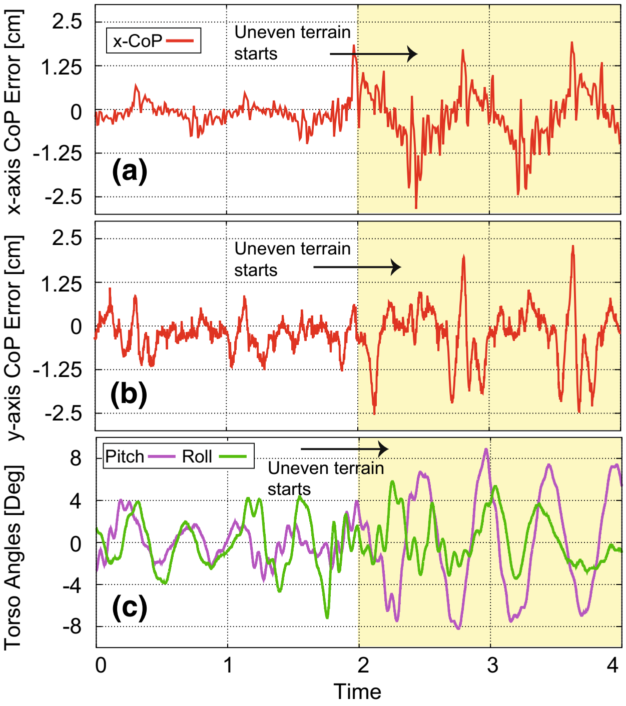


Fig. 20 Results of experiment #5. **a** x-axis CoP error. **b** y-axis CoP error. **c** Torso angle variations

CoP error variations are plotted for x-axis and y-axis in Fig. 20a, b. Initially, CoP error varied within the bands of ±0.7 cm and ±1.2 cm, respectively for x-axis and y-axis. As the robot started walking on the uneven terrain, x-axis CoP error band increased to +1.7 to −2.5 cm. y-axis CoP error followed a similar trend, it varied within ±2.5 cm. While the escalation in CoP error was expected as the robot started walking on the unperceived uneven terrain, it still resided within the safety margin of maximum allowable CoP error, that is ±3 cm. Therefore, the robot was able to maintain the dynamic balance while executing trot-walking on the unperceived uneven terrain.

Solid magenta and green lines present torso angle measurements in Fig. 20c, for pitch and roll axes. At the beginning, both angles varied within ±4 degrees. Once the robot started walking on the unperceived uneven terrain, roll angle altered within ±6 degrees. In an identical manner, pitch angle varied within ±8 degrees. Though the escalation in torso orientation variation was a natural consequence of executing trot-walking on the unperceived uneven terrain, the robot was able to preserve its postural balance and successfully completed the task.

8 Concluding remarks

In this paper we have demonstrated that the proposed pattern generator is able to generate smooth, continuous and dynamically consistent feet and CoM trajectories through the analytical solution of CoP equations. The algorithm automatically tunes several parameters (initial CoM position and velocity, stride length), so as to guarantee the seamless trajectories both in position, velocity and acceleration levels, through all the possible phases (left front–right hind 2-legged support, 4-legged support, right front–left hind 2-legged support) and as well as the transition between the phases, for the trot-walking locomotion task. To the authors' knowledge, such a property has not been introduced yet, when considering state-of-the-art trotting pattern generators.

The proposed pattern generator exploits the similarity between bipedal walking and quadrupedal trot-walking through the use of the equivalent biped model. Therefore, its implementation to bipedal walking systems is straightforward. That being said, other quadrupedal locomotion styles which may not resemble bipedal walking (static walk with 3 support legs, quadrupedal galloping) may not be generated, and this should be noted as a limitation.

The pattern generation utilizes the analytical solution of CoP equations (see Sect. 4) to synthesize dynamically balanced reference trajectories. In this task, phase information (single and double support periods) must be given beforehand. This shortcoming is not specific to the proposed pattern generator, it is an inherent characteristic in CoP-based CoM trajectory generation. If the actual state does not correspond to pre-planned state due to modeling imperfections, e.g., the robot swing leg arrives to the floor sooner than expected, it introduces force error to the system which is handled by the virtual admittance controller (see Sect. 6). While the requirement for the pre-planned phase information is a general shortcoming of CoP-based trajectory generation, and our controller is able handle possible issues due to this, it should be addressed as a limitation.

The virtual admittance controller is built upon an active compliance scheme, in which both position and force feedback are negotiated to address position/force trade-off by means of virtual admittance couples, introduced at each joint. The virtual admittance-based scheme of this controller naturally prioritizes position control when there is no force error, since the main purpose of trot-walking is to travel a certain distance. In other words, the joint is stiff when the robot is not subject to ground impacts and/or disturbances, e.g., during a swing phase. When the force error increases (e.g., due to unexpected impacts, external disturbances, unperceived uneven terrain, etc.), the joint stiffness is automatically decreased in order to comply with the force constraints. This ability allows the robot to handle large ground reaction force impacts, and enhance its environmental interaction capabilities which then leads it to exhibit enhanced locomotion behavior. Contributions of this controller were substantiated in Experiment #1 and Experiment #2 (see Figs. 14 and 16), in which the virtual admittance controller was both activated and deactivated. Compared to the case with no virtual admittance controller, the system showed a superior performance in handling ground reaction force impacts, reducing joint torque demands, decreasing torso acceleration fluctuations and CoP errors.

In classical virtual impedance control, the joints always keep the preset stiffness; they are either soft or stiff regardless of the induced force error which occurs due to the robot's interaction with the environment. Compared to this method, the proposed virtual admittance controller is more reactive, as it handles position/force trade-off depending on the induced force error. Therefore, it provides more favorable characteristics in legged locomotion control.

In addition to the virtual admittance controller, an orientation controller that is based on the regulation of angular momentum is presented. The controller generates orientation commands in a way so as to decrease the angular momentum error that is induced about the torso CoM. This strategy allows us to characterize the rotational inertia tensor of the torso, a crucial parameter in robot dynamics, so that a dynamically feasible control option could be implemented. Contributions of this controller were demonstrated in Experiment #2 (see Fig. 17). In this experiment, the undesired torso angle fluctuations were greatly decreased compared to the case in which the orientation controller was deactivated.

We presented a set experimental trials supporting that the proposed pattern generator and control framework could be useful assets in synthesizing dynamic trot-walking locomotion for two quadrupedal robots with different characteristics in terms of size and weight. The first set of experiments (Experiment #1, Experiment #2, Experiment #3) were conducted on a ∼7 kg cat-sized electrically actuated quadruped (RoboCat-1), validating the effectiveness the proposed method on a durable system. The relatively small size of this robot enabled us to conduct extensive experiments, in which the individual contribution of the controllers could be clarified. The robot was able to exhibit favorable trot-walking locomotion characteristics both on a level surface and on a rocky surface.

Upon the successful experimental verification with RoboCat-1, we proceeded to conduct experiments on a ∼80 kg Alpine Ibex-sized hydraulically actuated quadruped (HyQ). We were able to demonstrate that both the pattern generator and the controller framework were applicable to HyQ. Regardless of the robot actuator type, mass and size, the proposed methodology provided a feasible solution for the synthesis of dynamic trot-walking locomotion control. This was confirmed experimentally; HyQ was able to execute robust dynamic trot-walking locomotion, both on a treadmill and on an uneven surface with multiple wooden boards.

The Dynamic Legged Systems Laboratory at IIT recently addressed different aspects of locomotion control, namely, path planning with foothold adaptation, active impedance control, energy-efficient gait generation, stereo-vision-assisted locomotion, onboard perception, and local reflex generation (Winkler et al. 2014; Semini et al. 2013; Bazeille et al. 2014; Havoutis et al. 2013; Focchi et al. 2013). Thus, an integrative action will be taken to combine these technologies on HyQ in a synergistic fashion.

Acknowledgments In this study, the RoboCat-1 related portion is partially supported by Hitech Research Center, projects for private universities, supplied by the Ministry of Education, Culture, Sports, Science and Technology, Japan. The HyQ related portion is supported by Fondazione Istituto Italiano di Tecnologia, Genova, Italy. The authors would like to thank Takao Kawasaki, Kazuyuki Hyodo, Michihiro Kawanishi, Jesus Ortiz, Jake Goldsmith, Marco Frigero, Michele Focchi, Thiago Boaventura, Stephane Bazeille, Bilal Rehman, Hamza Khan, and the team of IIT Advanced Robotics technicians for their kind assistance and support.

References

Barasuol, V., Buchli, J., Semini, C., Frigero, M., De Pieri, E. R., & Caldwell, D. G. (2013). A reactive controller framework for quadrupedal locomotion on challenging terrain. In *IEEE International conference on robotics and automation (ICRA), Karlsruhe, Germany* (pp. 2539–2546).

Bazeille, S., Barasuol, V., Focchi, M., Havoutis, I., Frigerio, M., Buchli, J., et al. (2014). Quadruped robot trotting over irregular terrain assisted by stereo-vision. *Journal of Intelligent Service Robotics*, *7*(2), 67–77.

Boaventura, T., Medrano-Cerda, G. A., Semini, C., Buchli, J., & Caldwell, D. G. (2013). Stability and performance of the compliance controller of the quadruped robot HyQ. In *IEEE international conference on intelligent robots and systems (IROS), Tokyo, Japan* (pp. 1458–1464).

Boaventura, T., Semini, C., Buchli, J., Frigero, M., Focchi, M., & Caldwell, D. G. (2012). Dynamic torque control of a hydraulic quadruped robot. In *IEEE international conference on robotics and automation (ICRA), St. Paul, US* (pp. 1889–1894).

Buschmann, T., Lohmeier, S., & Ulbrich, H. (2009). Biped walking control based on hybrid position/force control. In *IEEE international conference on intelligent robots and systems (IROS), St. Louis, US* (pp. 3019–3024).

Byl, K., Shkolnik, A., Prentice, S., Roy, N., & Tedrake, R. (2009). Reliable dynamic motions for a stiff quadruped. *Springer Tracks in Advanced Robotics*, *54*, 319–328.

Colgate, E., & Hogan, N. (1989). An analysis of contact instability in terms of passive physical equivalents. In *IEEE international conference on robotics and automation (ICRA), Scottsdale, US* (pp. 404–409).

Colgate, J. E. (1994). Coupled stability of multiport systems—theory and experiments. *Transactions on ASME, Journal of Dynamic Systems, Measurement, and Control*, *116*(3), 419–428.

Fasse, E. (1987). *Stability robustness of impedance controlled manipulators coupled to passive environments*. Massachusetts Institute of Technology: Master's Dissertation.

Ferris, D. P., Louie, M., & Farley, C. T. (1998). Running in the real world: Adjusting leg stiffness for different surfaces. *Royal Society London*, *265*, 989–993.

Focchi, M., Barasuol, V., Havoutis, I., Buchli, J., Semini, C., & Caldwell, D. G. (2013). Local reflex generation for obstacle negotiation in quadrupedal locomotion. In *International conference on climbing and walking robots (CLAWAR), Sydney, Australia* (pp. 1–8).

Fujimoto, Y., Obata, S., & Kawamura, A. (1998). Robust bipedal walking with active interaction control between foot and ground. In *IEEE international conference on robotics and automation (ICRA), Leuven, Belgium* (pp. 2030–2035).

Galloway, K. C., Clark, J. E., & Koditschek, D. E. (2013). Variable stiffness legs for robust, efficient, and stable dynamic running. *ASME Journal of Mechanisms and Robotics*, *5*(1), 677–688.

Havoutis, I., Ortiz, J., Bazeille, S., Barasuol, V., Semini C., & Caldwell, D.G. (2013). Onboard perception-based trotting and crawling with the hydraulic quadruped robot (HyQ). In *IEEE international conference on intelligent robots and systems (IROS), Tokyo, Japan* (pp. 6052–6057).

Hutter, M., Remy, C. D., Hoepflinger, M. A., & Siegwart, R. (2013). Efficient and versatile locomotion with highly compliant legs. *IEEE Transactions on Mechatronics*, *18*(2), 449–458.

Hyon, S.-H. (2009). Compliant terrain adaptation for biped humanoids without measuring ground surface and contact forces. *IEEE Transactions on Robotics*, *25*(1), 677–688.

Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., & Hirukawa, H. (2003). Biped walking pattern generation by using preview control of zero-moment point. In *IEEE international conference on robotics and automation (ICRA), Taipei, Taiwan* (pp. 1620–1626).

Kalakrishnan, M., Buchli, J., Pastor, P., Mistry, M., & Schaal, S. (2011). Learning, planning and control for quadruped locomotion over challenging terrain. *International Journal of Robotics Research*, *30*(2), 236–258.

Kim, Y.-D., Lee, B.-J., Ryu, J.-H., & Kim, J.-H. (2007). Landing force control for humanoid robot by time-domain passivity approach. *IEEE Transactions on Robotics*, *23*(6), 1294–1301.

Kimura, H., Fukuoka, Y., & Cohen, A. H. (2007). Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts. *International Journal of Robotics Research*, *26*(5), 475–490.

Koolen, T., de Boer, T., Rebula, J. R., Goswami, A., & Pratt, J. E. (2012). Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models. *International Journal of Robotics Research*, *31*(9), 1094–1113.

Kurazume, R., Yoneda, K., & Hirose, S. (2002). Feedforward and feedback dynamic trot gait control for quadruped walking vehicle. *Autonomous Robots*, *12*(2), 157–172.

Maufroy, C., Kimura, H., & Takase, K. (2010). Integration of posture and rhythmic motion controls in quadrupedal dynamic walking using phase modulations based on leg loading/unloading. *Autonomous Robots*, *28*(3), 331–353.

Morimoto, J., Endo, G., Nakanishi, J., & Cheng, G. (2008). A biologically inspired biped locomotion strategy for humanoid robots: Modulation of sinusoidal patterns by a coupled oscillator model. *IEEE Transactions on Robotics*, *24*(1), 185–191.

Moro, F. L., Sproewitz, A., Tuleu, A., Vespignani, M., Tsagarakis, N. G., Ijspeert, A. J., et al. (2013). Horse-like walking, trotting, and galloping derived from kinematic motion primitives (kMPs) and their application to walk/trot transitions in a compliant quadruped robot. *Biological Cybernetics*, *107*(3), 309–320.

Murakami, T., Yu, F., & Ohnishi, K. (1993). Torque sensorless control in multidegree-of-freedom manipulator. *IEEE Transactions on Industrial Electronics*, *40*(2), 259–265.

Ott, C., Roa, M. A., & Hirzinger, G. (2011). Posture and balance control for biped robots based on contact force optimization. In *IEEE international conference on humanoid robots (humanoids), Bled, Slovenia* (pp. 26–32).

Raibert, M., Blankespoor, K., Nelson, G., Playtor, R., & the Big-Dog Team (2008). BigDog, the rough-terrain quadruped robot. In *The 17th world cong. The international federation automatic control, Seoul, Korea* (pp. 10822–10825).

Righetti, L., & Ijspeert, A. J. (2008). Pattern generators with sensory feedback for the control of quadruped locomotion. In *IEEE international conference on robotics and automation (ICRA), Pasadena, US* (pp. 819–824).

Rutishauser, S., Sproewitz, A., Righetti, L., & Ijspeert, A. J. (2008). Passive compliant quadruped robot using central pattern generators for locomotion control. In *IEEE international conference on biomedical robotics and biomechatronics (BioRob), Scottsdale, US* (pp. 710–715).

Sangok, S., Wang, A., Chuah, M. Y., Otten, D., Lang, J., & Kim S. (2013). Design principles for highly efficient quadrupeds and implementation on the MIT Cheetah robot. In *Proceedings of the IEEE conference on robotics and automation Karlsruhe, Germany* (pp. 3292–3297).

Semini, C., Barasuol, V., Boaventura, T., Frigerio, M., & Buchli, J. (2013). Is active impedance the key to a breakthrough for legged robots? In *IEEE international symposium on robotics research (ISRR), Singapore* (pp. 1–16).

Semini, C., Tsagarakis, N. G., Guglielmino, E., Focchi, M., Cannella, F., & Caldwell, D. G. (2011). Design of HyQ—A hydraulically and electrically actuated quadruped robot. *Institute of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, *225*(6), 831–849.

Sproewitz, A., Tuleu, A., Vespignani, M., Ajallooeian, M., Badri, E., & Ijspeert, A. J. (2011). Towards dynamic trot gait locomotion: Design, control, and experiments with Cheetah-cub, a compliant quadruped robot. *International Journal of Robotics Research*, *32*(8), 932–950.

Sugihara, T., & Nakamura, Y. (2009). Boundary condition relaxation method for stepwise pedipulation planning of biped robots. *IEEE Transactions on Robotics*, *25*(3), 658–669.

Ugurlu, B., Havoutis, I., Semini, C., & Caldwell, D. G. (2013). Dynamic trot-walking with the hydraulic quadruped robot—HyQ: Analytical trajectory generation and active compliance control. In *IEEE international conference on intelligent robots and systems (IROS), Tokyo, Japan* (pp. 6044–6051).

Ugurlu, B., Kotaka, K., & Narikiyo, T. (2013). Actively compliant locomotion control on rough terrain: Cyclic jumping and trotting experiments on a stiff-by-nature quadruped. In *IEEE international conference on robotics and automation (ICRA), Karlsruhe, Germany* (pp. 3298–3305).

Ugurlu, B., Saglia, J. A., Tsagarakis, N. G., Morfey, S., & Caldwell, D. G. (2014). Bipedal hopping pattern generation for passively compliant humanoids: Exploiting the resonance. *IEEE Transactions on Industrial Electronics*, *61*(10), 5431–5443.

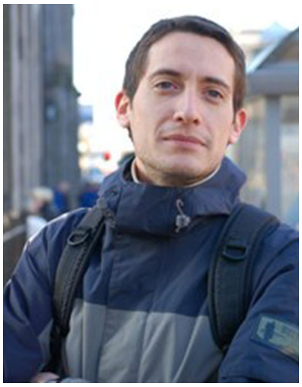
Winkler, A., Havoutis, I., Bazeille, S., Ortiz, J., Focchi, M., Dillmann, R., Caldwell, D. G., & Semini, C. (2014). Path planning with force-based foothold adaptation and virtual model control for torque controlled quadruped robots. In *IEEE international conference on robotics and automation (ICRA), Hong Kong* (pp. 6476–6482).

Yamada, Y., Nishikawa, S., Shida, K., Niiyama, R., & Kuniyoshi Y. (2011). Neural-body coupling for emergent locomotion: A musculoskeletal quadruped robot with spinobulbar model. In *IEEE international conference on intelligent robots and systems (IROS), San Francisco, US* (pp. 1499–1506).

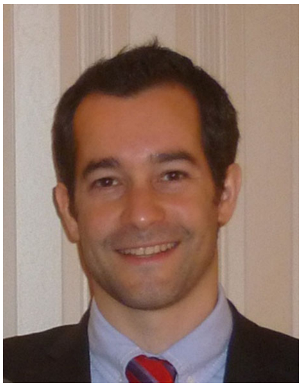
Yoneda, K., Iiyama, H., & Hirose, S. (1996). Intermittent trot gait of a quadruped walking machine dynamic stability control of an omnidirectional walk. In *IEEE international conference on robotics and automation (ICRA), Minnesota, US* (pp. 3002–3007).

Zheng, Y.-F., & Hemami, H. (1985). Mathematical modeling of a robot collision with its environment. *Journal of Robotic Systems*, *2*(3), 289–307.

Barkan Ugurlu received the Ph.D. degree in electrical and computer engineering from Yokohama National University, Yokohama, Japan, in 2010, with the Monbukagakusho scholarship granted by the Ministry of Education, Culture, Sports, and Technology of Japan. From May 2010 to March 2013, he was a Post-Doctoral Researcher, with the Istituto Italiano di Tecnologia, Genova, Italy, and Toyota Technological Institute, Nagoya, Japan. Between 2013 and 2015, he was a Research Scientist at the Computational Neuroscience Laboratories, Advanced Telecommunications Research Institute International (ATR), Kyoto, Japan. He currently holds an Assistant Professor position at the Department of Mechanical Engineering, Ozyegin University, Turkey. His research interests include biological sensorimotor control and motor recovery, active orthoses and exoskeletons, robot-aided rehabilitation, humanoid/quadruped locomotion control, and multibody dynamics, force, and compliance control.



Ioannis Havoutis is a postdoctoral researcher at the Department of Advanced Robotics in the Italian Institute of Technology. His research combines motion planning and control with machine learning, focusing on legged robots. Previously he received his PhD in Robotics and his M.Sc. in Artificial Intelligence with Distinction from the School of Informatics at University of Edinburgh and holds a Ptychion (B.Sc.) in Computer Science from the Department of Informatics at the Technological Educational Institute of Athens. His research interests include motion planning, humanoid robotics, legged locomotion, machine learning, (optimal) control and learning by demonstration.



Claudio Semini is currently the head of the Dynamic Legged Systems Lab that developed the HyQ robot platform, which is able to perform highly dynamic tasks such as running and jumping. Since the beginning of 2007 Claudio Semini has been a member of the Advanced Robotics Department at the Italian Institute of Technology (IIT). In spring 2007 he started the HyQ project as his PhD research under the supervision of Nikos Tsagarakis and Darwin Caldwell. By 2010 he finished the design and construction of the first version of the hydraulic quadruped robot HyQ with the support of a growing HyQ group. In April 2010, he finished his PhD with the title "HyQ - Design and Development of a Hydraulically Actuated Quadruped Robot".

Kana Kayamori received her Master's degree in Control Engineering from Toyota Technological Institute, Nagoya, Japan, in 2013. Throughout her graduate studies she focused on the development and locomotion control of quadruped robots. Her main research interest is the implementation of control engineering methods in real-world tasks.

Darwin G. Caldwell received the Ph.D. degree in robotics from the University of Hull, Hull, U.K., in 1990. From 1989 to 2006, he was with the University of Salford, Salford, U.K., as a Lecturer (1989-1996), Senior Lecturer (1996-1997), Reader (1997-1999), and finally Professor of Advanced Robotics in the Centre for Robotics and Automation (1999-2006). He is currently a Director with the Istituto Italiano di Tecnologia, Genova, Italy, and an Honorary Professor with the University of Sheffield, Sheffield, U.K.; University of Manchester, Manchester, U.K.; Bangor University, Bangor, U.K.; Kingüfs College London, London, U.K.; and Tianjin University, Tianjin, China. He is the author or coauthor of over 400 academic papers. He is the holder of 15 patents. His research interests include innovative actuators, humanoid and quadrupedal robotics and locomotion (iCub, HyQ, and CoMan), haptics, exoskeletons, dexterous manipulators, and medical robotics.

Tatsuo Narikiyo received his Ph.D. degree in Control Engineering from Nagoya University, Nagoya, Japan, in 1984. He was a research scientist in the Japanese Governmental Industrial Research Institute, Kyushu, Japan, from 1983 to 1990. In April 1990, he became an assistant professor in Toyota Technological Institute, Nagoya, Japan. Since April 1998, he has been a professor at the same institute. His research interests are control system design for nonlinear mechanical systems.