Effect of Compliance on Morphological Control of Dynamic Locomotion with HyQ

Gabriel Urbain · Victor Barasuol · Claudio Semini · Joni Dambre · Francis wyffels

This is a preprint of an article published in Autonomous Robots. The final authenticated version is available online at: https://doi.org/10.1007/s10514-021-09974-9

Abstract Classic control theory applied to compliant and soft robots generally involves an increment of computation that has no equivalent in biology. To tackle this, morphological computation describes a theoretical framework that takes advantage of the computational capabilities of physical bodies. However, concrete applications in robotic locomotion control are still rare. Also, the trade-off between compliance and the capacity of a physical body to facilitate its own control has not been thoroughly studied in a real locomotion task. In this paper, we address these two problems on the state-of-the-art hydraulic robot HyQ. An end-to-end neural network is trained to control HyQ's joints positions and velocities using only Ground Reaction Forces (GRF). Our simulations and experiments demonstrate better controllability using less memory and computational resources when increasing compliance. However, we show empirically that this effect cannot be attributed to the ability of the body to perform intrinsic computation. It invites to give an increased emphasis on compliance and co-design of the controller and the robot to facilitate attempts in machine learning locomotion.

Gabriel Urbain
IDLab-AIRO, elis, Ghent University – imec, Ghent, Belgium
E-mail: gabriel.urbain@ugent.be

Victor Barasuol
Dynamic Legged Systems (DLS) Lab, Istituto Italiano di Tecnologia (IIT), Genova, Italy

Claudio Semini
Dynamic Legged Systems (DLS) Lab, Istituto Italiano di Tecnologia (IIT), Genova, Italy

Joni Dambre
IDLab-AIRO, elis, Ghent University – imec, Ghent, Belgium

Francis wyffels
IDLab-AIRO, elis, Ghent University – imec, Ghent, Belgium

Keywords Quadruped Locomotion · Embodiment · Morphological Computation · Morphological Control · Reflex-based Locomotion

1 Introduction

Robotic legged locomotion has potential applications in many situations with unfriendly terrain for wheeled robots. This covers a broad range of sectors, stretching through agriculture, construction, mining operations, search and rescue, hazardous operation, health-care, domestic help, or education (Siciliano & Khatib, 2008). This research domain has led to rapid improvements in the last decade. But paradoxically, while locomotion can be seen as a primitive behavior in biology, the successes in robotic locomotion are still far from achieving the level of complexity and robustness observed in nature. Among the most promising paths for improvement, we can mention producing more flexible and robust materials, to mimic living systems (Calisti et al., 2017), improving the ratio between size, weight, power, and price of the actuators (Eschenauer et al., 2012), or the better miniaturization and integration of electronics, actuators, and sensors. On top of that, the control mechanisms in biological brains are not yet fully understood and constitute a major disincentive to designing functional legged robots as adaptive and energy-efficient than their biological counterparts.

In a classical engineering approach, a locomotion controller processes raw sensor inputs to define an observable state of the robot, as described by the rigid-body theory. It then uses it to determine the next desired state, which is reprojected into the actuator space. This technique allows an analytical understanding of the process, which is useful to optimize the robot's performance, stability, or accuracy. It shows however different limitations (Pfeifer & Bongard,

2007). First, the projections between sensor, actuator, and Cartesian spaces add an important computation overhead. Secondly, this intermediate abstract representation creates sharp separation between the controller and the body, which could be an obstacle to mapping sensors and actions optimally (Brooks, 1991). Third, this kind of controller can not easily cope with underactuated, compliant, and flexible parts without adding extra complexity to model the dynamics.

In this context, it has been suggested that embodiment is a key concept to operate flexible robots adequately. This theory primarily advocates that the brain-body dynamics interaction should be the starting point in designing new robots and controllers. Among the various theoretical frameworks implementing this approach, morphological computation is certainly the most discussed in the literature (Füchslin et al., 2013). An important milestone in morphological computation for robotics has been conducted in Hauser et al. (2011), which provides a theoretical foundation to evaluate the computational capacity of compliant bodies in control and learning tasks. A generic implementation of these bodies can be achieved using recurrent networks of masses and springs, also called Mass-Spring Network (MSN) structures. This conclusion has been later extended for locomotion control of MSN using a similar framework in Urbain et al. (2017). Other demonstrations of morphological computation for robotics control include locomotion of underactuated tensegrity structures (Caluwaerts et al., 2013), gait embodiment in compliant quadruped robots (Zhao et al., 2013; Degrave et al., 2015), or the control of soft robotic arms (Nakajima et al., 2013; Eder et al., 2018). More recently, however, Müller & Hoffmann (2017) and Hoffmann & Müller (2017) have suggested an important differentiation between *morphological computation* on one side and *morphology that facilitates the control* on the other. In the latter, the body is not literally involved in a computational process. However, its non-linear complexity can provide more dynamical landscapes where the location of attractors could facilitate the performance on a given task. To determine to which extent the morphology can perform computation or just simplifies the control, Rückert & Neumann (2013) has used optimal control strategies to discard the contribution of the controller itself in the computation process. They discussed the impact of the morphology on the control problem using a simplified four-segment model. However, the exact role of compliance in this study is only addressed briefly and qualitatively.

In contrast, several works in traditional robotics and automation have provided analytical discussions on the importance of compliant and soft parts. They can be divided into three categories: active compliance, passive compliance, and soft robotics. Passive compliant robots are generally made of rigid parts with under-actuated joints and additional springs and dampers. Therefore, they require specific attention during the selection of their mechanical properties and are also limited by design: unlike real animals, they are unable to tune their stiffness properties in real-time to adapt to different gait behaviors and environmental constraints. Because they are cheap and easy to build, small-scale quadruped robots mostly use this technology to conduct research on dynamic gaits (wyffels et al., 2010; Spröwitz et al., 2013; Urbain et al., 2018; Hoffmann & Simanek, 2017). Soft robots operate with fully flexible parts or a mix of rigid and elastic segments (Calisti et al., 2017). By pushing the concept of compliance further than the simple joint level, they also define a new control paradigm, requiring either complex finite element inverse kinematics or a fully embodied approach. Finally, the idea of active compliance originates from the concept of impedance control in control theory (Hogan, 1984). Rather than implementing the compliance properties physically, a software PD controller is in charge of rendering stiffness and damping effects in a joint, Cartesian, or virtual space. The control accuracy and the capacity to deal with external impacts makes it suitable for big size quadrupeds. However, it assumes the use of actuators that can tolerate high-torques during a short amount of time. This has been handled using hydraulic actuators, e.g. in Big Dog (Raibert et al., 2008) and HyQ (Semini et al., 2011) robot or using Series Elastic Actuation (SEA), which combine passive and active properties, on the ANYmal robot (Hutter et al., 2016). Proprioceptive control on a leg level coupled with a custom actuator and mechanical design has also been used to mitigate this issue on Cheetah-2 and Cheetah-3 robots (Seok et al., 2012; Bledt et al., 2018).

Notwithstanding their contribution to the scientific knowledge and their improvements in traditional robotics, most works in compliant locomotion do not take full advantage of the dynamical interactions between the body and the controller as suggested in the embodiment theory. Furthermore, the role of the compliance value has been highlighted as a driving factor to characterize locomotion within mammals (McMahon, 1985) but classical mechanics modeling does not provide sufficient criteria to evaluate how it acts at a body level. A dynamical analysis seems, therefore, more appropriate for this investigation. Indeed, it has been shown that the concept of resonance disappears within nonlinear systems, which makes it hard to model optimal and stable solutions in complex systems (Carbajal, 2012) and to discuss optimal damping and stiffness values. Empirical studies on single-leg hopper robots have backed up this observation by showing that the eigenvalues of a compliant leg are not fully aligned with the optimal Cost of Transport (COT) for locomotion (Vu et al., 2015).

Therefore, the first goal of this paper is to clarify how compliance affects the trotting gait generated by the Reactive Framework Controller (RCF) (Barasuol et al., 2013) on the HyQ robot. This characterization has already been car-

ried out and discussed for separate legs alone (Boaventura, Semini, et al., 2012; Boaventura et al., 2013) and showed that some configuration of compliance cannot be theoretically achieved with the active impedance controller. Other works on HyQ have also investigated the effect of compliance (Semini et al., 2015) and the optimal impedance parameters (Heijmink et al., 2017) in trotting tasks on different surfaces. These results are extended in this paper through an extensive analysis of all possible stiffness and damping parameters in a trotting gait.

Previous work on locomotion has discussed the benefits of compliant robot's morphologies to improve energy consumption (Papadopoulos & Buehler, 2000; Vanderborght et al., 2009) or locomotion speed (Spröwitz et al., 2013; Galloway et al., 2011). However, to our knowledge, no paper has yet investigated the trade-offs between compliance in locomotion and the controller requirements in a closed-loop embodied approach. The second goal of this article is to provide such a discussion on HyQ with a perspective inspired by morphological computation. To carry out this analysis, the robot is driven by a very generic controller made of a Delay Line (DL), an Extreme Learning Machine (ELM), and a trained readout layer. It facilitates investigations to look for a potential transfer of computation between body and controller and to generate a robust closed-loop dynamic gait from simple biologically-inspired inputs.

In contrast to other works conducted in simulation only, we wanted to demonstrate the potential of morphological control on a real robot. Therefore, our third goal is to apply the same approach on the real robot HyQ, to provide an end-to-end neural controller that can be trained in a realistic time. In comparison to some recent research that has studied machine learning control on the robot ANYmal (Lee et al., 2019), the policies are learned directly on HyQ in a supervised way, which allows avoiding complex transfer procedures.

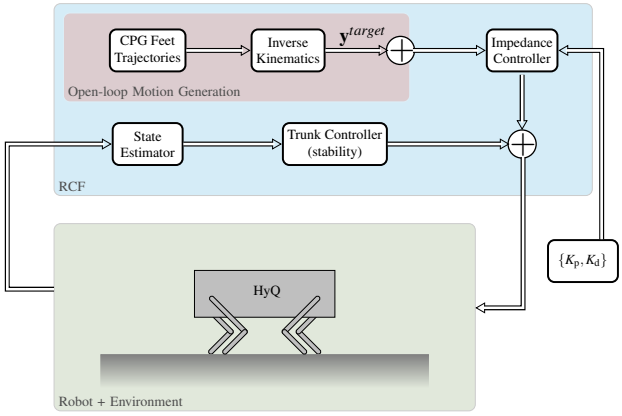


Fig. 1: The open-loop architecture is based on the RCF. In its core a CPG determines the evolution of the feet trajectories, which are translated in the joints space using inverse kinematics. A PD controller implements the active compliance module and the Trunk Controller helps to stabilize the robot during locomotion. Given the absence feedback, the dynamics of the motion generation and the robot are rather decoupled in this approach.

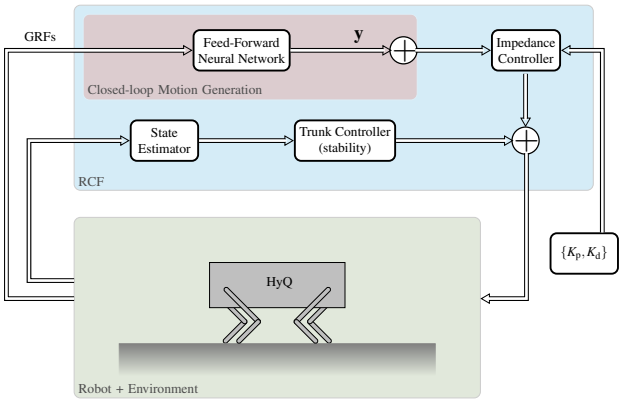


Fig. 2: In the closed-loop architecture, we use a feed-forward neural network that takes the GRF inputs from the robot and directly predicts the joints trajectories. The strong coupling between the robot and the controller dynamics makes it more appropriate to study the trade-offs between robot compliance and controller complexity.

2 Methods

2.1 Control Architectures

To support the research goals of this paper, two different architecture were elaborated: an *open-loop* architecture, represented in Fig.1 and a *closed-loop* architecture, represented in Fig. 2.

The open-loop architecture is based on an instance of the RCF controller (Barasuol et al., 2013) actuating the HyQ robot. The robotic platform is presented in more details in Section 2.2 and the internal mechanisms of the RCF are described in Section 2.3. The active compliance module is a key element in this architecture. It can virtually tune the stiffness and damping of the actuated joints, as explained in Section 2.4. The software also includes a framework written in *Python* to run parallel simulations with different sets of active joint stiffness and damping in *Docker*, and to investigate locomotion against the different metrics detailed in the Supplementary Materials.

The open-loop architecture is mostly used to produce the results of Section 3.1. However, it is not sufficient to study the dynamical interactions between the robot compliance and the controller complexity. Indeed, the motion genera-

tion of the RCF is performed by a CPG model, with its specific dynamic limit cycle. This architecture has two weaknesses in the scope of our study. First, the evolution of the full system is driven by the internal dynamics of the CPG, which explicitly controls the robot movements. The role of the feedback is only limited to correcting the stability of the robot trunk in our experiments. This decoupling between the generation of motion trajectories and the robot is not suited to study reciprocal interactions controller-morphology. Secondly, the CPG is a parametric model with a high-level representation. Therefore, in order to achieve embodied control, an agnostic model, such as a feed-forward neural network directly connected to the robot' sensors and actuators seems more appropriate. It creates a closed-loop control architecture. The neural network is presented in Section 2.5. Nonetheless, the open-loop motion trajectories are beneficial to train the neural network of the closed-loop architecture because it can provide reference trajectories for an efficient engineered locomotion. In Section 2.6, we explain how we can perform the training procedure to provide a continuous transition between the open- and the closed-loop architectures. This mechanism is key to train the HyQ platform in real conditions and to validate the results obtained in simulation in the real world.

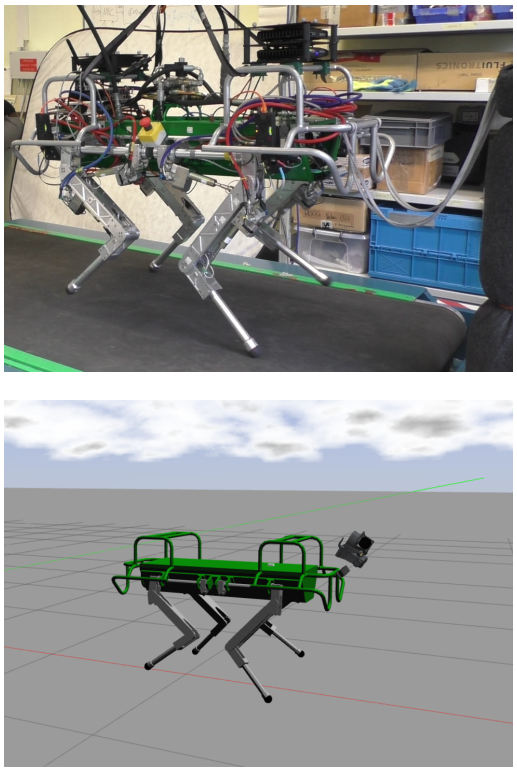


Fig. 3: In the top picture the real robot HyQ is trotting using the neural network presented in this work. Similar results can be achieved in simulation using the model from *ROS Gazebo* displayed in the bottom picture.

2.2 HyQ Robot and Simulation Model

The HyQ robot and its simulation model in *ROS Gazebo* are both represented in Fig. 3. HyQ is a state-of-the-art hydraulically powered quadruped platform of 1.3 m length and 90 kg weight (Semini et al., 2011). Its design allows the implementation of versatile gaits, ranging from static tasks like walking (Focchi et al., 2020) to dynamic trotting (Magana et al., 2019) and running, both on flat or uneven terrains. In contrast with cheap passive compliant robots on which morphological control has been already demonstrated (Iida & Pfeifer, 2004; Spröwitz et al., 2018; Urbain et al., 2018), HyQ has a bigger mass, a higher center of gravity (CoG) and a larger range of motion. To achieve robust stable locomotion control on this platform is therefore significantly more difficult but it could be a step in generalizing the works from morphological control to the field of advanced robotics.

HyQ has four legs with three degrees-of-freedom each, named Hip Abduction-Adduction (HAA), Hip Flexion-Extension (HFE), and Knee Flexion-Extension (KFE). All joints are hydraulically actuated. The advantage of this feature regarding the current work is twofold: first, the joints are capable of delivering or dissipating high torques, this allows fast actuation and makes the robot particularly robust for testing a feed-forward neural network controller, prone to oscillating behaviors that lead to larger Ground Reaction Forces (GRF); secondly, the actuation system can virtually produce adjustable levels of damping and stiffness as described in Section 2.3.

2.3 Reactive Controller Framework

To provide meaningful analysis, this work is limited to one single trot with constant gait parameters. The gait control signals are computed using a simplified version of the RCF (Barasuol et al., 2013). Sensory and visual feedback are used in other works to avoid obstacles and provide better foot placement (Magana et al., 2019) but these mechanisms are disabled here. In the simulations, the robot forward velocity is set to 0.25 m/s with a gait frequency of 1.7 Hz, a trot duty factor of 0.5, a desired step length of 7.3 cm, and a step height of 10 cm. These parameters are chosen heuristically to provide a robust trotting gait with a safe speed and foot trajectories. In the real experiments, the frequency has been lowered to 1.4 Hz with a duty factor of 0.6. This does not affect the comparison but helps to better monitor the robot to avoid inadvertent damage.

The different parts of the RCF controller can be seen in Fig. 1. In its core, a module produces four 2D CPG-inspired trajectories in the reference coordinates of each foot. They

are subsequently projected in the frame of reference of the robot's trunk. In this work, the correction of the foot trajectories according to the robot's attitude (*Kinematic Adjustment* module) has been deactivated to keep an open-loop behavior. Finally, the desired joint positions $\mathbf{q}_\mathrm{d}$, velocities $\dot{\mathbf{q}}_\mathrm{d}$ and accelerations $\ddot{\mathbf{q}}_\mathrm{d}$ are computed using inverse kinematics and transformed into joint torques via the PD impedance controller described in the next Sections. The robot body is stabilized in roll, pitch, and height using a module called *Trunk Controller*. It also produces joint torques that add up to the RCF contribution and regulate the stability of the robot.

2.4 Impedance Controller

The implementation of a torque controller for HyQ joints has been presented in Boaventura, Semini, et al. (2012) and Focchi et al. (2012) and its performance and stability when compensating external forces have been addressed both in simulation and on the real robot (Boaventura et al., 2013; Boaventura, Focchi, et al., 2012). To understand the role of the impedance controller, the joint control loop can be replaced by an equivalent rotational spring-damper system (Semini et al., 2015). This operation is represented for the KFE joint in Fig. 4. To this aim, we first hypothesize that the effect of the *Inner Torque Control Loop*, running at 1 kHz, is negligible compared to the oscillation of the impedance controller, which is true for the impedance gains considered in this paper. The equation describing the torques applied on the joints:

$$\boldsymbol{\tau}_\mathrm{f} = \boldsymbol{\tau}_\mathrm{ext} + K_\mathrm{p}\left(\mathbf{q}_\mathrm{d} - \mathbf{q}\right) + K_\mathrm{d}\left(\dot{\mathbf{q}}_\mathrm{d} - \dot{\mathbf{q}}\right) \tag{1}$$

and the one of the equivalent spring-damper system:

$$\boldsymbol{\tau}_\mathrm{f} = \boldsymbol{\tau}_\mathrm{ext} - k\left(\mathbf{q} - \mathbf{q}_\mathrm{ref}\right) - c\left(\dot{\mathbf{q}} - \dot{\mathbf{q}}_\mathrm{ref}\right) \tag{2}$$

therefore imply that the proportional and derivative gains K_p and K_d have respectively the same units as the virtual torsional spring stiffness k measured in N.m/rad and the damping c in N.m.s/rad, for which we would vary the vector of reference position $\mathbf{q}_\mathrm{ref}$ and the vector of reference rotational speed $\dot{\mathbf{q}}_\mathrm{ref}$ during actuation. In these equations, $\boldsymbol{\tau}_\mathrm{f}$ represents the torque vector experienced by the joints, and $\boldsymbol{\tau}_\mathrm{ext}$, external disturbing torques from the robot's interaction with its environment. $\mathbf{q}_\mathrm{d}$ and $\dot{\mathbf{q}}_\mathrm{d}$ are the vectors of desired position and velocities of the joints, and $\mathbf{q}$ and $\dot{\mathbf{q}}$, the real ones. For the sake of clarity, we will simply call K_p and K_d respectively stiffness and damping in the following sections.

2.5 Neural Network

The core element of the closed-loop architecture is the feed-forward neural network presented in Fig. 5. At each time step t_k, sampled at 250 Hz, the neural network model takes a set of inputs $\mathbf{x}_k$ and outputs a set of signals $\mathbf{y}_k$ that controls the flexion-extension actuators only:

$$\begin{aligned}\mathbf{y}_k \;=\; \{\; & q_{\mathrm{d},\,k}^{\mathrm{LH,\,HFE}},\; q_{\mathrm{d},\,k}^{\mathrm{RH,\,HFE}},\; q_{\mathrm{d},\,k}^{\mathrm{LF,\,HFE}},\; q_{\mathrm{d},\,k}^{\mathrm{RF,\,HFE}},\\ & q_{\mathrm{d},\,k}^{\mathrm{LH,\,KFE}},\; q_{\mathrm{d},\,k}^{\mathrm{RH,\,KFE}},\; q_{\mathrm{d},\,k}^{\mathrm{LF,\,KFE}},\; q_{\mathrm{d},\,k}^{\mathrm{RF,\,KFE}},\\ & \dot{q}_{\mathrm{d},\,k}^{\mathrm{LH,\,HFE}},\; \dot{q}_{\mathrm{d},\,k}^{\mathrm{RH,\,HFE}},\; \dot{q}_{\mathrm{d},\,k}^{\mathrm{LF,\,HFE}},\; \dot{q}_{\mathrm{d},\,k}^{\mathrm{RF,\,HFE}},\\ & \dot{q}_{\mathrm{d},\,k}^{\mathrm{LH,\,KFE}},\; \dot{q}_{\mathrm{d},\,k}^{\mathrm{RH,\,KFE}},\; \dot{q}_{\mathrm{d},\,k}^{\mathrm{LF,\,KFE}},\; \dot{q}_{\mathrm{d},\,k}^{\mathrm{RF,\,KFE}}\;\},\end{aligned} \tag{3}$$

where q_d and $\dot{q}_\mathrm{d}$ represent the desired position and velocities; LH, RH, LF, RF are the leg index, i.e., Left Hind, Right Hind, Left Front, Right Front; KFE and HFE refer to the different actuators, i.e., Knee Flexion-Extension and Hip Flexion-Extension.

The neural controller inputs are given by the normalized GRFs, a bias signal, and no clock:

$$\mathbf{x} = \{\; GRF_{\,k}^{\,\mathrm{LH}},\; GRF_{\,k}^{\,\mathrm{RH}},\; GRF_{\,k}^{\,\mathrm{LF}},\; GRF_{\,k}^{\,\mathrm{RF}},\; 1\;\}. \tag{4}$$

This critical choice of GRFs inputs is guided simultaneously by practical reasons and biological evidence about quadruped locomotion. On one hand, using real joints' positions and velocities as feedback to teach a feed-forward neural network to predict the next desired joints' positions and velocities could ultimately split the global architecture into a set of independent controllers at a joint level. In other words, instead of training a controller to predict the next robot pose, this approach could drift to training each actuator to predict its next pose individually. In that sense, GRFs are a better indicator of the full robot's dynamic than joints' status. For instance, an error of actuation on $q_\mathrm{d}^{\mathrm{LH,\,HFE}}$ can produce an strong direct effect on $GRF^{\,\mathrm{RF}}$ but less importantly on $q^{\mathrm{RF,\,HFE}}$. This signal is also very informative about the distribution of the robot weight on its different feet (hence, its stability) and the interaction with the environment (through the impacts with the ground).

On the other hand, the role of CPG is debated for complex locomotion behaviors within humans (Dzeladini et al., 2014). In contrast, GRFs have a serious plausibility to represent the biological implementation of reflex-based locomotion in mammals. For instance, simulation studies on cat

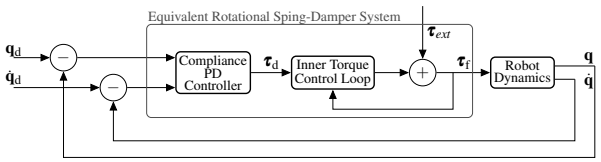


Fig. 4: The torque controller of the leg's actuators can accurately reproduce virtual stiffness and damping properties. The speed of the hydraulic actuation and the fast PD loop is a key factor to its high performance.

locomotion have indicated that the stability of gait coordination in the presence of external disturbance depends heavily on sensing GRFs in each separate leg (Ekeberg & Pearson, 2005). Demonstration of biologically plausible locomotion on simulated human models has also established the relevance of GRFs as the main input to design an efficient reflex-based controller (Geyer & Herr, 2010). The same principle has been also implemented to create natural recovery behaviors from tripping on unexpected obstacles (Murai & Yamane, 2011). Hybrid strategies implementing a mixture of centralized control and reflex mechanisms have been also achieved in robotics (Manoonpong et al., 2007). While limiting the input numbers, we are not aiming at an exact replication of the RCF dynamics. However, this representation should be able to learn a stable attractor with trajectories identical to the ones in the RCF, using no prior knowledge about the robot morphology and mechanics.

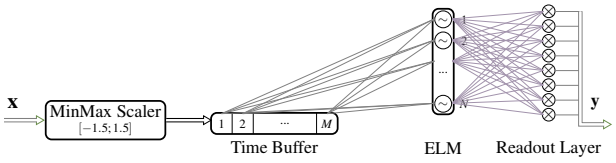


Fig. 5: Tuning the memory and the non-linearities of the neural network can be performed by setting respectively the size of a time buffer and a hidden layer of hyperbolic tangent neurons. The time buffer is fully connected to the hidden layer and only the connections between the hidden layer and the linear readout are trained.

In the generic controller, we opted for a very abstract representation: normalized inputs are sent to a Delay Line (DL), which acts as a first-in, first-out memory buffer. It is fully connected to a hidden layer of hyperbolic tangent neurons, followed by another fully connected readout layer of linear neurons. The layers' architecture is inspired by the Extreme Learning Machines (ELM) (Huang et al., 2004). This approach has been used in robotics before (Degrave et al., 2013) and has the benefit of exposing only two parameters M and N to tune respectively the memory and the nonlinear hidden projections of the controller's model. The absence of prior knowledge in the network structure does not restrict the range of motion to a defined subset but allows a continuous infinity of trajectories. Furthermore, due to the feed-forward processing, no temporal components are added in the controller dynamics.

2.6 Training

The weights of the connections between the hidden and readout layer $\mathbf{W}_{\text{out}}$ are updated using the FORCE learning method (Sussillo & Abbott, 2009) to learn to reproduce the trajectories provided by RCF target:

$$\mathbf{e}_{k+1} = (\ \mathbf{W}_k^{\text{out}}\ )^T \cdot \mathbf{x}_{k+1} - \mathbf{y}_{k+1}^{\text{target}} \tag{5}$$

$$\mathbf{P}_{k+1} = \mathbf{P}_k - \frac{\mathbf{P}_k \cdot \mathbf{x}_{k+1} \cdot \mathbf{x}_{k+1}^T \cdot \mathbf{P}_k}{1 + \mathbf{x}_{k+1}^T \cdot \mathbf{P}_k \cdot \mathbf{x}_{k+1}} \tag{6}$$

$$\mathbf{W}_{k+1}^{\text{out}} = \mathbf{W}_k^{\text{out}} - \mathbf{P}_{k+1} \cdot \mathbf{x}_k \cdot \mathbf{e}_{k+1}^T \tag{7}$$

$$\mathbf{y}_{k+1} = (\ \mathbf{W}_{k+1}^{\text{out}}\ )^T \cdot \mathbf{x}_{k+1}, \tag{8}$$

where $\mathbf{P}$ is the estimate of the inverse of the correlation matrix and $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{y}^{\text{target}}$ are the network input, the network output and the RCF target, respectively.

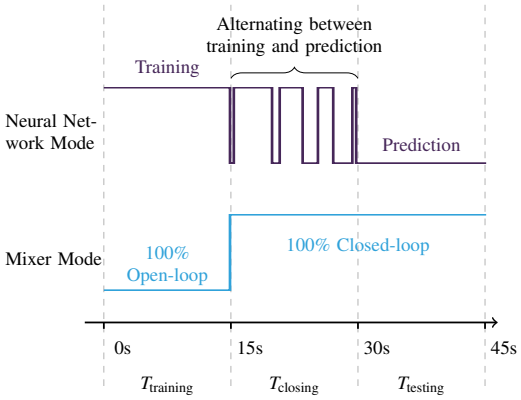


Fig. 6: The simulations and trials on the real robot are divided into three phases of 15 seconds. In the first phase, the robot is actuated in open-loop while the neural network is in training mode. In the second phase, we switch to the closed-loop architecture and the network alternates between training and prediction mode until it stays only in that mode in the third phase.

As presented in Fig. 6, the trials are divided into three phases of 15 seconds each. During the first phase, T_{training}, the robot is actuated by the RCF only and the signals are used together with the sensors to train the readout layer of the neural network. In the second phase, T_{closing}, the mixer depicted in the middle of the figure switches to the neural network output. The RCF target is however still used as a target to train the network. The closing phase alternates between prediction and training until only the prediction remains at the end of the phase. Finally, in the last phase T_{testing}, only the predictions of the neural network are used to investigate its performance.

2.7 Experimental Methodology

The experiments realized to collect the results are divided into three parts.

First, in Section 3.1, we are using the open-loop control architecture (presented in Fig. 1) in simulation to evaluate different virtual stiffness and damping coefficients. To estimate the performance in open-loop, the experiments run for 15 seconds and only the first phase T_{training} is carried. Performance indicators on stability, power, speed, and GRFs are measured to determine the behavior of the gaits and identify different locomotion regimes.

In the second set of experiments presented in Section 3.2, we evaluate the effect of compliance on the COT and the minimal controller requirements in a closed-loop architecture. In the stiffness-damping region with the best open-loop performance identified in the previous step, we train the neural network of the closed-loop architecture presented in Fig. 2 over the full duration of 45 seconds as discussed in Section 2.6. The interaction of four quantities (controller memory, controller non-linearity, joint stiffness, and joint damping) is evaluated for the dynamical system formed by the robot and the controller.

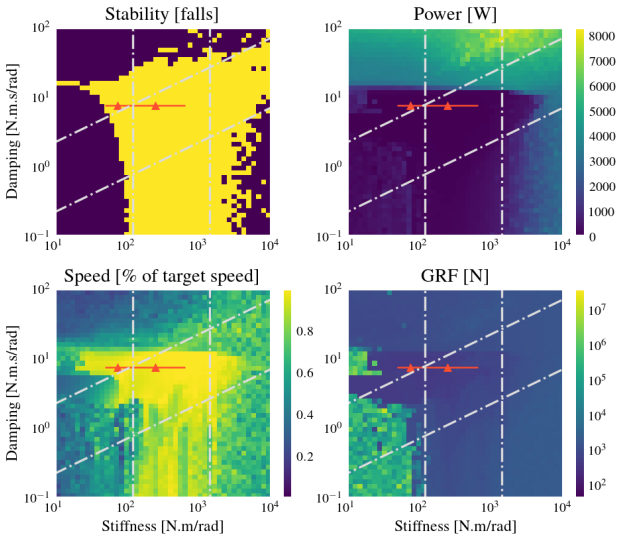
The third part, presented in Section 3.3, details the experimental trials on the real HyQ robot and discusses the qualitative observations made during these trials.

3 Results

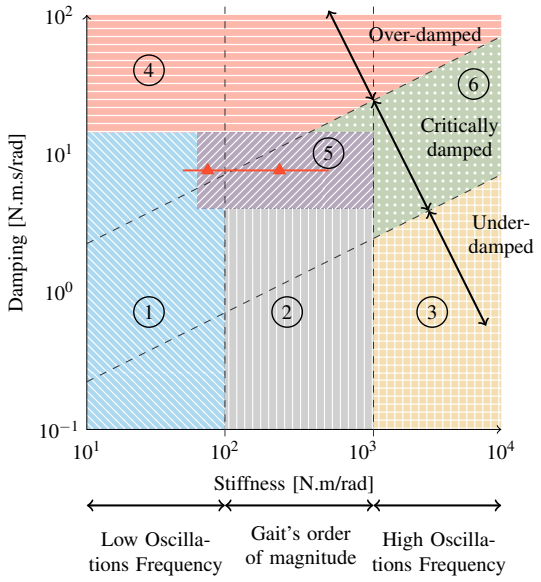
3.1 Compliance Regions for the Open-Loop Architecture

This section provides a characterization of the different trotting regimes on HyQ in function of its compliance. To this goal, 3.600 different settings in the HyQ's active impedance controller are sampled on a logarithmic space formed by the joints damping and stiffness. Ten trotting simulations with the open-loop architecture are averaged for each impedance, using the RCF as detailed in Section 2.3. Four metrics are represented as they together summarize the locomotion behavior: stability, power, speed, and GRFs. Motivation and an explicit definition for each of the metrics are provided in the Supplementary Materials.

From the results presented in Fig. 7a, we can cluster the different morphologies following their gait performance metrics. The graph in Fig. 7b provides such a partitioning into six different regions in the stiffness-damping space. To understand how these regimes are determined, the effects on the final gait are first discussed for increasing stiffness (X-axis). At low stiffness, the robot is falling and the GRFs observed in the simulation become unrealistic. This is caused by the difficulty of the physics engine to cope with the diverse impacts due to the undesired robot position, lying on the ground. For Region 1, the power consumption remains



(a) The gait metrics for different damping and stiffness values



(b) A diagram clustering the space in different gait regions

Fig. 7: Gait metrics for different damping and stiffness values. In (a), grid simulations represent the robot locomotion behavior in terms of stability (falling in blue and not falling in yellow), mechanical work power (in Watts), normalized speed (as a ratio of the desired target speed) and GRFs (in Newtons). Some unrealistic simulation results are discussed in the manuscript. A range of good performance on all four criteria is defined by the red line on the graphs. From left to right, the red triangles define two sets of impedance, called *compliant* and *stiff*, selected to evaluate a generic controller. In (b), the damping-stiffness space is divided into characteristic regions describing the robot locomotion.

small and the robot speed is reduced to zero. Increasing stiffness raises the frequency of the joint oscillations. For Region 2, these oscillations have the same order of magnitude as the locomotion frequency. The stability module presented in Section 2.3 is therefore able to prevent the robot from falling. This region is characterized by a correct forward speed and a stable yet shaky gait. When shifting to higher stiffness in Region 3, the oscillations are too fast and lead to locomotion instabilities, which explains the high falling rate and the low average speed.

The role of the damping coefficient can be better understood by browsing the graph diagonally through the different curves of iso-damping ratios. For lower ratios (Region 3 for instance), the oscillations around the desired joint positions are under-damped. The robot has more undesired impacts with the ground, causing it to fall. The power consumption is also increased due to the high velocities of the joint oscillations. On the other end, in the upper left corner of the graph, the damping ratios are too high and the actuators are over-damped. In Region 4, the dissipated power measured in simulation becomes unrealistic due to the small amplitude of the oscillations. Between these extremes, critical damping can be reached in Regions 5 and 6. Despite the high frequencies, the robot can maintain a positive forward speed without falling in most cases. Region 5 constitutes the most interesting area for our studies as it combines acceptable damping ratios with correct stiffnesses. It is characterized by adequate speed and energetic performance and realistic GRF for a transfer on the real platform. A red line is defined in this area of the graph. It will be considered in the next section to study the compliance effect in closed-loop locomotion. Two parameters sets, called *compliant* (left) and *stiff* (right) are also identified with red triangle on this line. They score equally for all the metrics in open-loop but have different stiffness values.

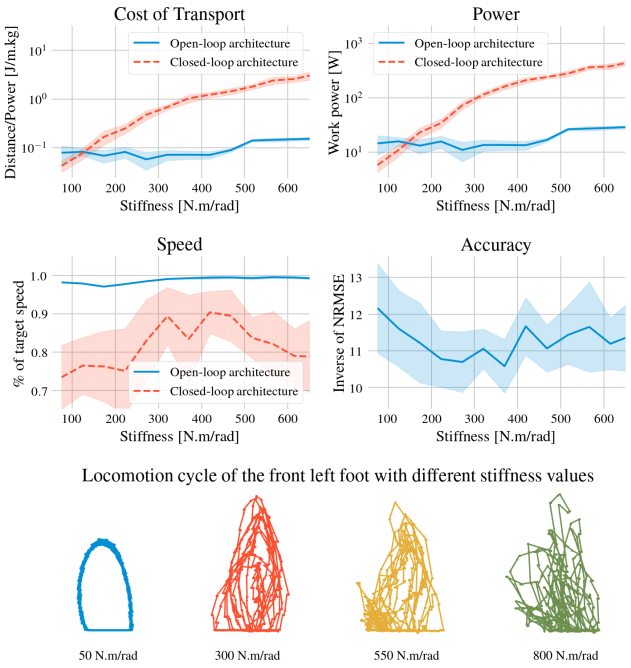


Fig. 8: The COT for the closed-loop architecture increases with stiffness (top left). This effect can be linked to an increment in power consumption, itself due to higher impacts on the ground. The speed, however, slightly increases with the stiffness before to drop at higher values, because of reduced stability and gait efficiency.

3.2 Effect of Compliance in Closed-Loop Control

The empirical evaluation of compliance in the last Section constitutes a basis to discuss major assumptions in morphological control. In this section, we aim at quantifying the effect of compliance in a closed-loop architecture by providing an answer to the two following questions:

1. Can compliance improve the COT in reflex-based locomotion? On one hand, it seems that compliant robots can be safer and use less energy (Papadopoulos & Buehler, 2000; Vanderborght et al., 2009; Kashiri et al., 2018; Seok et al., 2014). On the other hand, several works have discussed the presence of an optimal stiffness value to maximize the locomotion speed of a robot (Spröwitz et al., 2013; Galloway et al., 2011). Here, we evaluate how these two effects contribute to the COT in a closed-loop control architecture.
2. Can compliance facilitate morphological locomotion control on complex dynamic structures? While non-linearities and fading memory have been identified as essential features to perform computation with generic morphologies in Hauser et al. (2011), the role of compliance and its interaction with the controller complexity requires further investigation. Here, we contribute to this topic with empirical demonstrations on HyQ.

3.2.1 Cost of Transport

In this first investigation, a fixed neural network is selected in the closed-loop architecture by setting the memory and non-linearity parameters M and N to 50. These values are selected to provide enough complexity for the task with a correct real-time execution, as shown in the Supplementary Materials. Robots with different compliance are sampled from the red line introduced in Fig. 7. For each of them, we average ten simulations in which the neural network is trained as described in Section 2.6. The results for different metrics are displayed in Fig. 8.

In the top-left graph of Fig. 8, the evolution of the COT is represented for different compliance values. The blue line describes the open-loop architecture and the red line, the closed-loop architecture. The first observation is that the COT of the closed-loop system increases almost exponentially with stiffness. In other words, good performance with this control architecture can only be achieved with compliant robots. In the scope of HyQ's locomotion task, this is a serious motivation to combining morphological control with compliant morphologies in the search for optimal performance.

The COT is defined by the ratio between the robot's power consumption and speed, normalized by the robot's mass (see Supplementary Materials). The speed and power curves are, therefore, also represented in Fig. 8 to better understand the reasons behind the increase of COT with stiffness. The power value is only theoretical as it is obtained by multiplication of all actuators torque and speed as detailed in the Supplementary Materials, but calorific losses due to hydraulic compression and other mechanical drags need to be added in practice. The variations of COT seem to be mostly explained by the power contribution. To understand why, we rather need to consider the dynamic of the system as a whole. The last line of Fig. 8 shows an increment of the actuators' oscillations around their limit cycles at higher stiffness. Despite these oscillations, we can note that the neural network accuracy presented in the same figure, remains quite stable with the stiffness, i.e. it can keep track with the target trajectories. So, the oscillations are not due to the controller or the robot alone but rather to their interaction as a dynamic system. In turn, this is responsible for larger impacts between the feet and the ground. A plot of the maximal GRFs provided in Supplementary Materials substantiates this hypothesis. The increase of power consumption with the stiffness also corroborates the explanations presented in other works (Vanderborght et al., 2009; Hoffmann & Simanek, 2017; Kashiri et al., 2018) and advocates for the use of compliance to optimize energy efficiency and reduce the risks of impact with the environment, particularly in situations where humans are involved.

In comparison with power, speed seems to play a smaller role in its contribution to the COT variations for the closed-loop architecture. The dominant trendshows that speed is first rising with the stiffness before to drop when the joints become too stiff. The positive effect relates to a better lift-off phase and a stronger push on the ground to propel the robot forward when increasing the joint's rigidity. However, the negative effect starts to take over at stiffness higher than 400 N.m/rad. Additional graphs presenting the robot's step length and the number of steps in Supplementary Materials show that this is principally due to a reduced average step length (i. e. the feet are touching the ground sooner than expected). A visual analysis of the locomotive cycle in the last line of Fig. 8, in correlation with the trajectories displayed in Fig. 10 further demonstrates that this effect.

A last interesting observation from the COT graph suggests that a closed-loop controller becomes better than open-loop against this metric for really compliant robots (far left side of the curve). This highlights a positive point of morphological control: the ability to better handle compliant bodies, without forcing the joints to follow a given trajectory or computing complex intermediate representations with finite elements theory. Nonetheless, it is important to stress that this benefit also comes at a certain cost, as featured by the reduced speed at low stiffness.

3.2.2 Computational Requirements

In the second analysis, we evaluate the trade-offs between controller's complexity and robot compliance, to contribute to stable and efficient locomotion. To this goal, a qualitative evaluation of the gait is conducted for the two individuals with *compliant* and *stiff* parameters described in Section 3.1. For each of them, several controllers with a different amount of non-linearity, N and memory, M, are trained. This choice is inspired by the fact that nonlinear dynamical systems with fading memory have a finite computational capacity and display a trade-off between memory and non-linearity (Dambre et al., 2012). Furthermore, memory and non-linearities are not only meaningful in the controller but also at the body level. Therefore, we believe that these parameters are relevant to measure the complexity of the dynamical system formed by the robot and its controller and to discuss the influence on the final controllability of the system. For each neural architecture, ten simulations are performed to provide a meaningful estimate and three metrics are investigated in Fig. 9: accuracy, speed, and stability. In this figure, the left column represent the *compliant* robot and the right column, the *stiff* robot. In each graph, the X-axis describes the evolution of the controller's memory and the Y-axis, the controller's non-linearities. More metrics are discussed in the Supplementary Materials and can help to understand the underlying mechanisms of the experiment results. A qualitative evaluation of the gait obtained by the *compliant* and *stiff* robots is also presented in Fig. 10 to visualize how the gait trajectories predicted by the neural network are affected by the compliance. To draw these pictures, we have selected a controller with $M = 50$ and $N = 50$.

We can first observe in Fig. 9 that the prediction accuracy (defined as the inverse of Normalized Square Root Error between the neural network's prediction and the RCF target in the Supplementary Materials) increases with the controller memory and non-linearities (i.e., in the direction of the upper right part of the graph). As it can be seen in the second row of this figure, this improvement in prediction accuracy is also linked to a higher speed. It also has a beneficial

impact on gait behavior, as it seems to correlate with better stability in the third row. However, this effect is mostly observed for soft impedances and disappears for stiffer ones. This is expressed by the spreading of the accuracy, speed, and stability levels for the *stiff* robot on the graphs.

An important remark needs to be formulated here. In the hypothesis of a strict transfer of computation to the robot facilitated by compliance, we could have expected the migration of the region of interest (yellow points for accuracy and speed, and blue points for stability) on a diagonal axis between the *compliant* and *stiff* morphologies. However, we failed to observe such an effect in our experiments. Instead, these regions start spreading on the graphs. In other words, this means that the controller complexity, represented by its memory and non-linearities on the X and Y axis, has less influence on the locomotion metrics for a stiff robot than for a compliant one. The robot becomes less sensitive to the computation power of the controller at higher stiffness. Therefore, these empirical results rather support the idea that increasing stiffness leads to more decoupling between the controller and the robot dynamics in closed-loop, making morphological control a harder task with stiff robots, rather than demonstrating an explicit transfer of computation between the controller and the body.

In the scope of morphological control, we can conclude that the compliance is facilitating the control of locomotion on the HyQ robot. The feedback from a compliant body backs up the controller's memory and non-linearities to produce a better prediction in a virtuous cycle. This observation is fundamental in the scope of embodiment and morphological computing as it emphasizes how the choice of morphology is important to obtain a stable limit cycle.

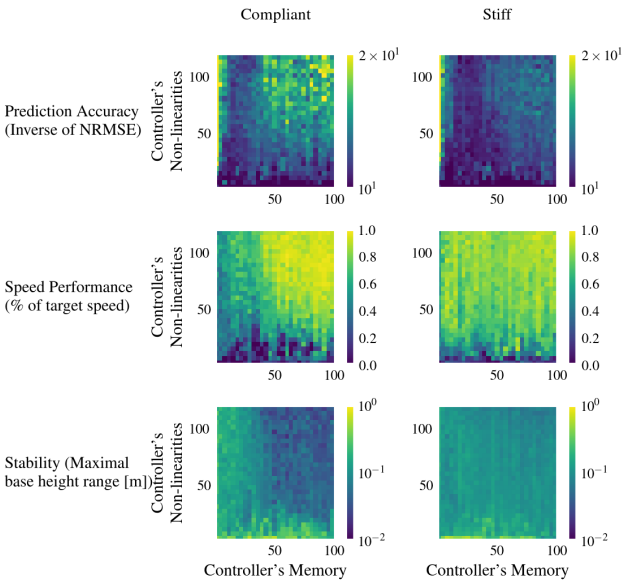


Fig. 9: In each of these six graphs, various controllers are evaluated by tuning the delay line size M (to increase or decrease the controller's memory) on the X-axis and the ELM layer size N (to change the non-linear richness) on the Y-axis. The columns correspond to the *compliant* and the *stiff* joint parameters. Each row represents a different locomotion metric ranging from blue for low values to yellow for high values. The spreading of the region of efficient locomotion with the stiffer robot can be interpreted as a decoupling between the controller and robot dynamics at higher stiffness.

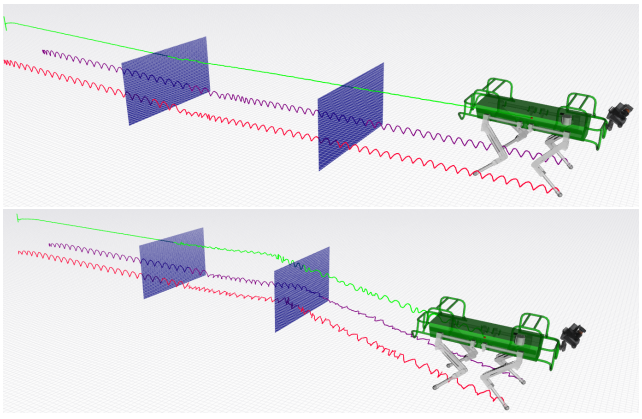


Fig. 10: This figure shows the end-effector trajectories of the right front foot (red), the left front foot (purple), and also the trunk's center of mass (green). The blue planes divide the trajectory in the time domain into three phases: *Training*, *Closing* and *Testing*. The experiment is repeated for the two compliance values discussed in the paper: *compliant* (top), and *stiff* (bottom). It shows a deteriorated locomotion cycle at higher stiffness.

3.3 Evaluation on the Real HyQ Robot

Despite recent progress on locomotion using machine learning, most works are conducted in simulation, and only a few results have been tested on real robots. This issue primarily arises from what is called the *reality gap*: simulation models cannot reproduce the real physics accurately enough without a prohibitive amount of computation or excessive parameter tuning. This paper, however, highlights the role of the body for efficient locomotion which makes a validation on the real HyQ platform inevitable. Minor parameter and algorithm adaptations have been added in comparison with the previous section to deal with the *reality gap*:

1. The stiffness gains discussed in Section 3.1 are defined empirically for the simulated model. Their relationship and the different defined areas are more meaningful than their absolute values. To cope with differences in weight, inertia, or actuator torques between the robot and its model, the stiffness values were raised to 250 N.m/rad

and 150 N.m/rad in swing phase and 150 N.m/rad and 100 N.m/rad in stance phase respectively for the hip and the knee.

2. The GRFs are indirectly acquired using encoders and load cells on different joints. The sensor accuracy at high torques and the propagation of errors in computing the contacts lead to noisy sensor data in comparison to the simulation baselines. This effect is corrected by applying a moving window filter using 15 samples followed by a threshold function on the GRF data. The values before and after processing are presented in Fig. 11 (top). The simulation counterpart is also represented on the same figure and demonstrates that the process can provide correct amplitude, smooth transitions, and keep the important phase information (gait synchronism).
3. The robot simulation model assumes a uniform distribution of the mass along the trunk. In reality, some subsystems located in the body are heavier than others, which creates a heterogeneous weight repartition. Therefore, the robot is less stable than its simulated counterpart and covers a larger range of roll, pitch, and height during locomotion. This effect is yet increased by external forces applied by the supply of hoses that provide pressurized oil to the robot and that randomly pull or push it to different directions during the gait. Finally, another disturbance comes from the communication delay between the robot and the computer where the neural network is running. For all these reasons, the experiment duration was increased to 180 seconds with 120 seconds of training, 30 seconds of closing phase, and 30 seconds of testing phase. The learning algorithm regularization was also raised from 0.001 to 1 to generalize on more samples.

The experimental validation results are shown in the video provided in the Supplementary Materials. The transfer can be considered successful as there is no qualitative difference between the trot using the target gait of the classical controller or the neural network gait. The stability of the trunk is contained in the same value range and the feet trajectories show no visible deviation from the targets. However, it has been observed that the robot is more sensitive to external disturbance. For instance, if there is a connection loss between the computer running the neural network and the robot (therefore freezing the actuators briefly) the motion mostly failed to recover back its attractor. To further understand the underlying mechanisms, the predicted trajectories are plotted together with the target in Fig. 11 (bottom). It can be observed that the shape of the locomotion cycle is similar but the neural network output starts to lose synchrony with the target phase when the training has stopped. From that moment on, the controller is completely disconnected from the classical CPG and uses no more temporal information, which explains why their phases desynchronize. This effect aside, this section provides a proof on the capacity of a neural controller to operate a large quadruped robot directly at a low-level joint level, to sustain an efficient embodied locomotion.

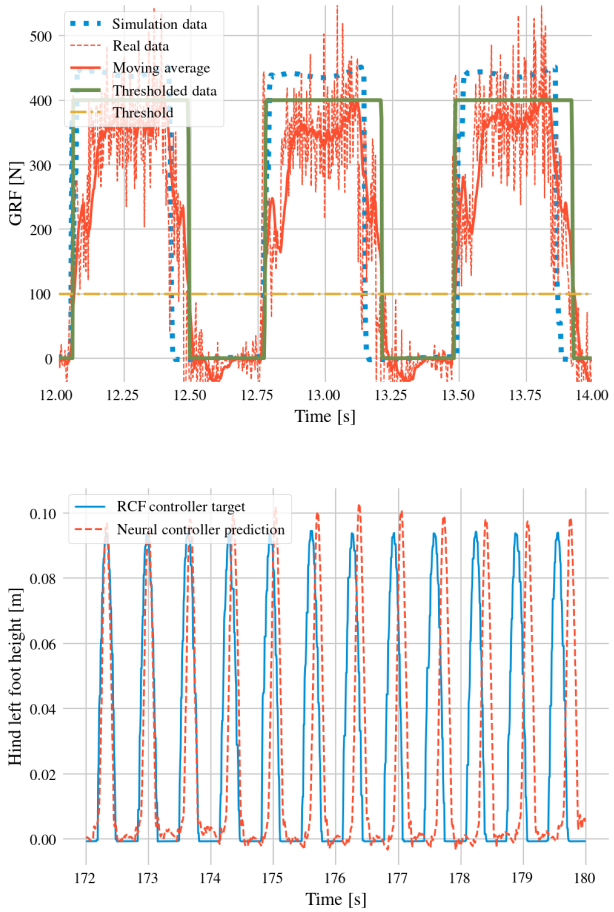


Fig. 11: Embodied locomotion on the real robot. On the top, the GRF are presented with the raw signal (red), after processing using a window filter and threshold (green) as well as a comparison with the signal in simulation (blue). The preprocessing method keeps the phase of the real signal but it generates levels that are closer to their simulation counterpart. On the bottom, the left hind foot height is shown for the target and prediction signals to highlight the dephasing in reflex-based locomotion.

4 Conclusions

In this work, we have suggested two different control architectures to achieve locomotion on the HyQ robot. The first, named open-loop, is based on a CPG and its dynamics controls tightly the motion of the body with limited feedback from the environment. The second, named closed-loop, is based on a reflex mechanism where the controller is im-

plemented by a simple feed-forward neural network taking GRF sensors as input and predicting directly the desired joints positions and velocities, which a strong coupling between the body dynamic and the controller. The motivation of this work is not to compare the open-loop and closed-loop architectures but rather to use the first one as a target and a benchmark to study morphological control in locomotion and the trade-offs with joint compliance.

The experiments presented in this paper advocate in favor of using compliant robots in conjunction with morphological control. They highlight the role of compliance for learning control and coordination of locomotion movement and the relevance of this approach on real robots. Beside providing advantages on the COT, simulations on HyQ show that compliant robots can simplify closed-loop control of robot locomotion. A possible interpretation of the role of compliance can be found in morphological computation theory. A robot can be seen as a computational entity that transforms actuation signals into sensor signals through its dynamical interactions with the environment. As remarked in Müller & Hoffmann (2017), it is nevertheless essential to differentiate *morphologies that facilitate control*, in which a good design reduces computation requirements, from *intrinsic morphological computation*, where there is clear outsourcing of the computing power. Our empirical study demonstrates that intrinsic morphological computation could be discarded in the locomotion task under consideration. At higher stiffness, however, we observe a decoupling between the control and the robot dynamics. Compliant robots have, therefore, *morphologies that facilitate control*. Rather than a concrete transfer of computation to the body as observed in other work on embodied locomotion (Urbain et al., 2017; Caluwaerts et al., 2013), compliance of HyQ helps to regulate the locomotion trajectories and better outsource the body feedback.

Built on the simulation work, a successful validation on real hardware has been investigated. The use of a stability module for PID-based attitude control and gravity compensation (called *Trunk Controller* and detailed in Section 2.3) in association with learning feet trajectories and coordination was crucial to facilitate the transfer. We believe that this is key in finding a methodology to bring simulation results on locomotion robots without damaging them. The experiment also showed the possibility to operate locomotion only in a reflex-based fashion. But the loss of phase synchronization with the motor target emphasizes the importance of oscillatory systems, like biological CPGs, to keep a stable locomotion phase in presence of external disturbance. By applying joint torques to keep the base stable and to ensure the stance timing, the stability module also plays a critical role in the system performance. A discussion on this question has been investigated in Urbain et al. (2020).

By quantifying the needs in memory and non-linearities of a neural network and interpreting them in the light of the mechanical properties of a robot, we hope that this paper contributes to bridging the gap between the fields of robotics and machine learning. In particular, the results could be used in further investigation of reinforcement learning methods to estimate the requirement of a recurrent network controller. This work also enriches the efforts in reflex-based locomotion with an effective demonstration on a real robot, which can bring some insights into the fundamental understanding of the neural mechanism used for locomotion in biology.

Author contributions statement

G.U., V.B., and F.w. conceived the study, designed the experiments, and interpreted the results. G.U. and V.B. conducted the experiments. G.U. wrote the manuscript and all authors reviewed it.

Acknowledgements This research was supported by the HBP Neurorobotics Platform funded from the European Union's Horizon 2020 Framework Programme for Research and Innovation under the Specific Grant Agreement No. 785907 (Human Brain Project SGA2). Experiments were conducted on the HyQ Platform hosted and funded by the Fondazione Istituto Italiano di Tecnologia. The authors would also like to thank Shamel Fahmi for his help during the experiments and Gennaro Raiola for his support with the software.

Conflict of interest

The authors declare that they have no conflict of interest.

References

Barasuol, V., Buchli, J., Semini, C., Frigerio, 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 2012* (pp. 2554–2561).

Bledt, G., Powell, M. J., Katz, B., Carlo, J. D., Wensing, P. M., & Kim, S. (2018). MIT cheetah 3: Design and control of a robust, dynamic quadruped robot. In *IEEE/RSJ international conference on intelligent robots and systems, IROS 2018* (pp. 2245–2252).

Boaventura, T., Focchi, M., Frigerio, M., Buchli, J., Semini, C., Medrano-Cerda, G. A., & Caldwell, D. G. (2012). On the role of load motion compensation in high-performance force control. In *IEEE/RSJ international conference on intelligent robots and systems, IROS 2012* (pp. 4066–4071).

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/RSJ international conference on intelligent robots and systems, IROS 2013* (pp. 1458–1464).

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

Brooks, R. A. (1991). Intelligence without representation. *Artificial intelligence*, *47*(1-3), 139–159.

Calisti, M., Picardi, G., & Laschi, C. (2017). Fundamentals of soft robot locomotion. *Journal of The Royal Society Interface*, *14*(130), 20170101.

Caluwaerts, K., D'Haene, M., Verstraeten, D., & Schrauwen, B. (2013). Locomotion without a brain: Physical reservoir computing in tensegrity structures. *Artificial Life*, *19*(1), 35–66.

Carbajal, J. P. (2012). *Harnessing nonlinearities: Generating behavior from natural dynamics* (Unpublished doctoral dissertation). Universität Zürich. Mathematisch-naturwissenschaftlichen Fakultät.

Dambre, J., Verstraeten, D., Schrauwen, B., & Massar, S. (2012). Information processing capacity of dynamical systems. *Scientific reports*, *2*, 514.

Degrave, J., Caluwaerts, K., Dambre, J., & wyffels, F. (2015). Developing an embodied gait on a compliant quadrupedal robot. In *IEEE/RSJ international conference on intelligent robots and systems, IROS 2015* (pp. 4486–4491).

Degrave, J., Cauwenbergh, R. V., wyffels, F., Waegeman, T., & Schrauwen, B. (2013). Terrain classification for a quadruped robot. In *12th international conference on machine learning and applications, ICMLA 2013* (pp. 185–190).

Dzeladini, F., Van Den Kieboom, J., & Ijspeert, A. (2014). The contribution of a central pattern generator in a reflex-based neuromuscular model. *Frontiers in human neuroscience*, *8*, 371.

Eder, M., Hisch, F., & Hauser, H. (2018). Morphological computation-based control of a modular, pneumatically driven, soft robotic arm. *Advanced Robotics*, *32*(7), 375–385.

Ekeberg, O., & Pearson, K. G. (2005, dec). Computer simulation of stepping in the hind legs of the cat: an examination of mechanisms regulating the stance-to-swing transition. *Journal of neurophysiology*, *94*(6), 4256–68.

Eschenauer, H., Koski, J., & Osyczka, A. (2012). *Multicriteria design optimization: procedures and applications*. Springer Science & Business Media.

Focchi, M., Boaventura, T., Semini, C., Frigerio, M., Buchli, J., & Caldwell, D. G. (2012). Torque-control based compliant actuation of a quadruped robot. In *12th IEEE international workshop on advanced motion control, AMC 2012* (pp. 1–6).

Focchi, M., Orsolino, R., Camurri, M., Barasuol, V., Mastalli, C., Caldwell, D. G., & Semini, C. (2020). In *Advances in robotics research: From lab to market: ECHORD++: Robotic science supporting innovation* (pp. 165–209). Springer International Publishing.

Füchslin, R. M., Dzyakanchuk, A., Flumini, D., Hauser, H., Hunt, K. J., Luchsinger, R. H., ... Walker, R. (2013). Morphological computation and morphological control: Steps toward a formal theory and applications. *Artificial Life*, *19*(1), 9–34.

Galloway, K. C., Clark, J. E., Yim, M., & Koditschek, D. E. (2011). Experimental investigations into the role of passive variable compliant legs for dynamic robotic locomotion. In *IEEE international conference on robotics and automation, ICRA 2011* (pp. 1243–1249).

Geyer, H., & Herr, H. (2010). A Muscle-reflex model that encodes principles of legged mechanics produces human walking dynamics and muscle activities. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *18*(3), 263–273.

Hauser, H., Ijspeert, A. J., Füchslin, R. M., Pfeifer, R., & Maass, W. (2011). Towards a theoretical foundation for morphological computation with compliant bodies. *Biological Cybernetics*, *105*(5-6), 355–370.

Heijmink, E., Radulescu, A., Ponton, B., Barasuol, V., Caldwell, D. G., & Semini, C. (2017). Learning optimal gait parameters and impedance profiles for legged locomotion. In *17th IEEE-RAS international conference on humanoid robotics, humanoids 2017* (pp. 339–346).

Hoffmann, M., & Müller, V. C. (2017). Simple or complex bodies? trade-offs in exploiting body morphology for control. In *Representation and reality in humans, other living organisms and intelligent machines* (pp. 335–345). Springer.

Hoffmann, M., & Simanek, J. (2017). The merits of passive compliant joints in legged locomotion: fast learning, superior energy efficiency and versatile sensing in a quadruped robot. *Journal of Bionic Engineering*, *14*(1), 1–14.

Hogan, N. (1984). Impedance control: An approach to manipulation. In *1984 american control conference* (pp. 304–313).

Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. (2004). Extreme learning machine: a new learning scheme of feedforward neural networks. *Neural networks*, *2*, 985–990.

Hutter, M., Gehring, C., Jud, D., Lauber, A., Bellicoso, C. D., Tsounis, V., ... Höpflinger, M. A. (2016). Anymal - a highly mobile and dynamic quadrupedal robot. In *IEEE/RSJ international conference on intelligent robots and systems, IROS 2016* (pp. 38–44).

Iida, F., & Pfeifer, R. (2004). Cheap rapid locomotion of a quadruped robot: Self-stabilization of bounding gait. In *Intelligent autonomous systems* (Vol. 8, pp. 642–649).

Kashiri, N., Abate, A., Abram, S. J., Albu-Schaffer, A., Clary, P. J., Daley, M., ... others (2018). An overview on principles for energy efficient robot locomotion. *Frontiers in Robotics and AI*, *5*, 129.

Lee, J., Dosovitskiy, A., Bellicoso, D., Tsounis, V., Koltun, V., & Hutter, M. (2019). Learning agile and dynamic motor skills for legged robots. *Science Robotics*, *4*(26).

Magana, O. A. V., Barasuol, V., Camurri, M., Franceschi, L., Focchi, M., Pontil, M., ... Semini, C. (2019). Fast and continuous foothold adaptation for dynamic locomotion through cnns. *IEEE Robotics and Automation Letters*.

Manoonpong, P., Geng, T., Kulvicius, T., Porr, B., & Wörgötter, F. (2007). Adaptive, fast walking in a biped robot under neuronal control and learning. *PLoS Computational Biology*, *3*(7).

McMahon, T. A. (1985). The role of compliance in mammalian running gaits. *Journal of Experimental Biology*, *115*(1), 263–282.

Müller, V. C., & Hoffmann, M. (2017). What is morphological computation? on how the body contributes to cognition and control. *Artificial Life*, *23*(1), 1–24.

Murai, A., & Yamane, K. (2011). A neuromuscular locomotion controller that realizes human-like responses to unexpected disturbances. In *IEEE international conference on robotics and automation, ICRA 2011* (pp. 1997–2002).

Nakajima, K., Hauser, H., Kang, R., Guglielmino, E., Caldwell, D. G., & Pfeifer, R. (2013). A soft body as a reservoir: case studies in a dynamic model of octopus-inspired soft robotic arm. *Frontiers in Computational Neuroscience*, *7*, 91.

Papadopoulos, D., & Buehler, M. (2000). Stable running in a quadruped robot with compliant legs. In *IEEE international conference on robotics and automation, ICRA 2000* (pp. 444–449).

Pfeifer, R., & Bongard, J. (2007). *How the body shapes the way we think - a new view on intelligence*. MIT Press.

Raibert, M., Nelson, K. B. G., & Playter, R. (2008). Bigdog, the rough-terrain quadruped robot. *IFAC Proceedings Volumes*, *41*(2), 10822–10825.

Rückert, E. A., & Neumann, G. (2013). Stochastic optimal control methods for investigating the power of morphological computation. *Artificial Life*, *19*(1), 115–131.

Semini, C., Barasuol, V., Boaventura, T., Frigerio, M., Focchi, M., Caldwell, D. G., & Buchli, J. (2015). Towards versatile legged robots through active impedance control. *The Interational Journal of Robotics Research*, *34*(7), 1003–1020.

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. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, *225*(6), 831–849.

Seok, S., Wang, A., Chuah, M. Y., Hyun, D. J., Lee, J., Otten, D. M., ... Kim, S. (2014). Design principles for energy-efficient legged locomotion and implementation on the mit cheetah robot. *Ieee/asme transactions on mechatronics*, *20*(3), 1117–1129.

Seok, S., Wang, A., Otten, D., & Kim, S. (2012). Actuator design for high force proprioceptive control in fast legged locomotion. In *IEEE/RSJ international conference on intelligent robots and systems, IROS 2012* (pp. 1970–1975).

Siciliano, B., & Khatib, O. (Eds.). (2008). *Springer handbook of robotics*. Springer.

Spröwitz, A., Tuleu, A., Ajallooeian, M., Vespignani, M., Möckel, R., Eckert, P., ... Ijspeert, A. J. (2018). Oncilla robot: A versatile open-source quadruped research robot with compliant pantograph legs. *Frontiers in Robotics and AI*, *2018*.

Spröwitz, A., Tuleu, A., Vespignani, M., Ajallooeian, M., Badri, E., & Ijspeert, A. J. (2013). Towards dynamic trot gait locomotion: Design, control, and experiments with cheetah-cub, a compliant quadruped robot. *The Interational Journal of Robotics Research*, *32*(8), 932–950.

Sussillo, D., & Abbott, L. F. (2009). Generating coherent patterns of activity from chaotic neural networks. *Neuron*, *63*(4), 544–557.

Urbain, G., Barasuol, V., Semini, C., Dambre, J., & wyffels, F. (2020). Stance control inspired by cerebellum stabilizes reflex-based locomotion on hyq robot. In *IEEE international conference on robotics and automation, ICRA 2020.*

Urbain, G., Degrave, J., Carette, B., Dambre, J., & wyffels, F. (2017). Morphological properties of mass-spring networks for optimal locomotion learning. *Frontiers in Neurorobotics*, *2017*.

Urbain, G., Vandesompele, A., wyffels, F., & Dambre, J. (2018). Calibration method to improve transfer from simulation to quadruped robots. In *15th international conference on simulation of adaptive behavior, SAB 2018* (pp. 102–113).

Vanderborght, B., Ham, R. V., Lefeber, D., Sugar, T., & Hollander, K. W. (2009). Comparison of mechanical design and energy consumption of adaptable, passive-compliant actuators. *The Interational Journal of Robotics Research*, *28*(1), 90–103.

Vu, H. Q., Yu, X., Iida, F., & Pfeifer, R. (2015). Improving energy efficiency of hopping locomotion by using a variable stiffness actuator. *IEEE/ASME transactions on mechatronics*, *21*(1), 472–486.

wyffels, F., D'Haene, M., Waegeman, T., Caluwaerts, K., Nunes, C., & Schrauwen, B. (2010). Realization of a passive compliant robot dog. In *3rd IEEE-RAS & EMBS international conference on biomedical robotics and biomechatronics, BioRob 2010* (pp. 882–886).

Zhao, Q., Nakajima, K., Sumioka, H., Hauser, H., & Pfeifer, R. (2013). Spine dynamics as a computational resource in spine-driven quadruped locomotion. In *IEEE/RSJ international conference on intelligent robots and systems, IROS 2013* (pp. 1445–1451).