Check for updates

OPEN

Learning plastic matching of robot dynamics in closed-loop central pattern generators

Felix Ruppert ✉ and Alexander Badri-Spröwitz

Animals achieve agile locomotion performance with reduced control effort and energy efficiency by leveraging compliance in their muscles and tendons. However, it is not known how biological locomotion controllers learn to leverage the intelligence embodied in their leg mechanics. Here we present a framework to match control patterns and mechanics based on the concept of short-term elasticity and long-term plasticity. Inspired by animals, we design a robot, Morti, with passive elastic legs. The quadruped robot Morti is controlled by a bioinspired closed-loop central pattern generator that is designed to elastically mitigate short-term perturbations using sparse contact feedback. By minimizing the amount of corrective feedback on the long term, Morti learns to match the controller to its mechanics and learns to walk within 1h. By leveraging the advantages of its mechanics, Morti improves its energy efficiency by 42% without explicit minimization in the cost function.

Animals can locomote with grace and efficiency due to intelligence embodied in their leg designs[1]. Owing to compliant mechanisms in their leg designs, animals can safely traverse rough and unstructured terrain[2,3] in the presence of neural delays and limited actuator power and bandwidth[4,5]. These compliant mechanisms are important components of the natural dynamics of a system. Natural, or passive, dynamics[6] describes the system's passive dynamic behaviour governed by its mechanical characteristics, such as impedance or inertia. More specifically, it describes the dynamics of the unactuated plant transfer function[7].

Compliant mechanisms help to mitigate the interaction forces between walking systems and the environment that are hard to model and are defined by a high degree of uncertainty[8].

To gain a better understanding of the underlying mechanics, bioinspired robots with passive compliant structures that provide the same advantages to robots and simplify the control task have been investigated[2,9–11]. By designing mechanical properties such as impedance[12–14] and spring-loaded inverted pendulum behaviour[3,15,16], the natural dynamics can be designed to achieve viable behaviour with no or reduced control effort, improved energy efficiency and robustness[17–19] comparable to nature.

In a system with strong natural dynamics, the mechanical elements produce forces comparable to the actuators. The challenge of how a controller learns to leverage those natural dynamics then arises. How can animals and bioinspired robots learn to match the control patterns (meaning the desired muscle or motor activation patterns) they produce to their natural dynamics to leverage advantageous passive characteristics?

If the control patterns do not match the natural dynamics, the controller requires additional energy to enforce a desired behaviour (see Supplementary Section 5) as it has to overcome the forces and torques produced by the passive mechanical elements. There is a lack of model-free learning formulations for the matching of control patterns to a given robot's dynamics, especially for robots with strong engineered passive compliant elements. Previous work focused on designing specific aspects of natural dynamics to fit a given control scheme[9,20,21]. In this work we focus on quantifying the match between control patterns and natural dynamics, and how to improve and learn matching in a bioinspired quadruped robot (Fig. 1a).

The neural structure and neuromuscular pathways of animals evolved over many generations and are inherent to each individual at birth[22]. In robotics, the control approach and electrical connections are hardcoded in the design phase before deployment. The timing and intensity of muscle activity patterns in animals have to be matched to the system's natural dynamics as a lifelong learning task[23–25], whereas in robotics the controller has to be learned or tuned for optimal performance during the testing phase or during the robot's lifetime[26–28].

To learn matching the control patterns to the natural dynamics, we separated system perturbations by their time horizon. A one-time stochastic perturbation, like stumbling, should not trigger a long-term adaptation. However, if stumbling occurs frequently, the system should adapt to this systematic discrepancy between the desired control patterns and the system's behaviour governed by the natural dynamics.

To implement this approach we took inspiration from the concept of neuroelasticity and long-term neuroplasticity from neuroscience[29], as well as the concept of elasticity and plasticity in mechanics[30] that describes the reaction to environmental stimuli based on its intensity. A one-time stimulus with low intensity will be mitigated and the control pattern elastically returns to its initial state (Fig. 2a, top). Permanent or frequent stimuli will plastically adapt the control pattern to remove the discrepancy between desired control pattern and natural dynamics behaviour (Fig. 2a, bottom).

In this study we implemented a quadruped robot, Morti, with engineered natural dynamics that is controlled by a central pattern generator (CPG). CPGs are neural networks found in animals that produce rhythmic output signals from non-rhythmic inputs[31,32] for tasks such as chewing, breathing and legged locomotion[33,34]. In robotics CPGs are used as joint trajectory generators[13,35,36] or bioinspired muscle activation pattern generators[37,38]. Feedforward CPGs dictate control and coordination of motor or muscle activation without knowledge of the system's dynamics. These model-free feedforward patterns work well in combination with passively compliant leg designs that provide passive stability and robustness[13,37,38]. By closing feedback loops in CPGs, the system can actively react to unforeseen influences from its environment and mitigate perturbations[32,35,39] such as unstructured terrain.

Dynamic Locomotion Group, Max Planck Institute for Intelligent Systems, Stuttgart, Germany. ✉e-mail: ruppert@is.mpg.de
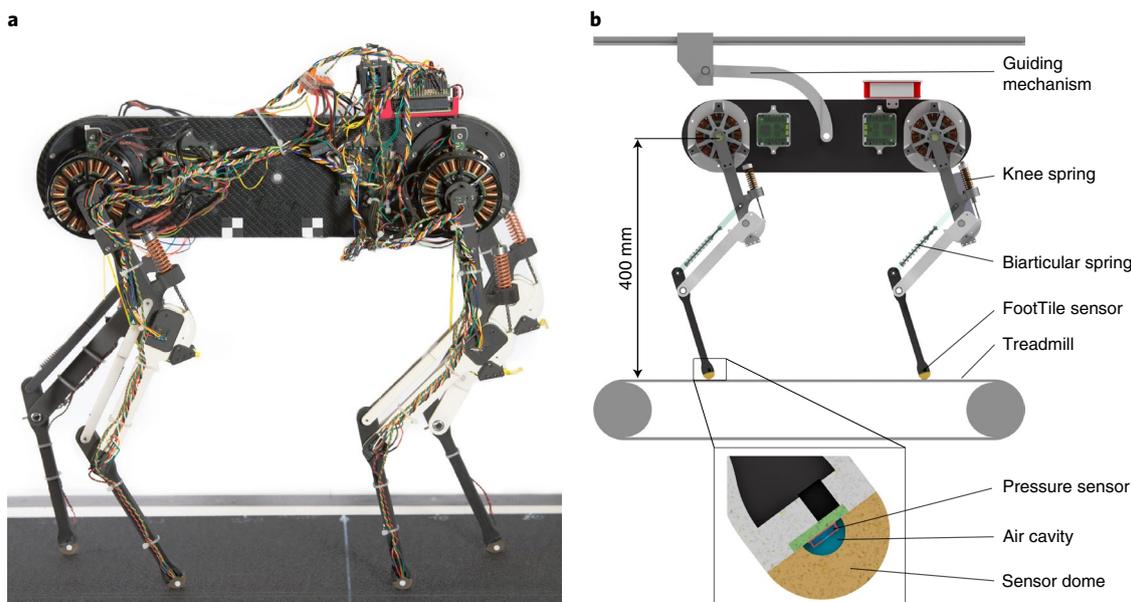
Fig. 1 | Quadruped robot Morti. a, Photograph of Morti. **b**, Render of Morti on top of the treadmill. Morti was constrained to the sagittal plane by a linear rail and lever guiding mechanism that allowed body pitch around its centre of mass. Contact by Morti was measured using four FootTile contact sensors. Inset: close-up cross-section of the FootTile sensor on the foot segment of Morti. The polyurethane sensor dome deforms under loading and the pressure sensor measures the increasing pressure in the air cavity to detect foot contact.

In our quadruped robot Morti, we implemented feedback loops based on continuous sensor data and feedforward reflexes triggered by discrete perturbation events. We then observed Morti's behaviour as measured using sparse feedback from contact sensors on its feet. This short-term feedback acts as a mechanism for mitigating elastic short-term perturbations.

To quantify how well the control pattern matched Morti's natural dynamics, we used the elastic feedback activity as a proxy. If the dynamics did not match, the feedback mechanisms constantly had to intervene to correct for the discrepancy between the commanded and measured behaviour of Morti. The matching of the control patterns needed to be increased plastically.

To improve the matching plastically, we optimized the CPG parameters that generate the control patterns by minimizing the amount of elastic feedback activity (Fig. 2b,c).

Different methods have been used to optimize and tune control patterns, such as optimization[40,41], self-modelling[42], adaptive CPGs[43-45] and machine learning techniques[45-51]. For this study, we applied Bayesian optimization[52,53] to minimize the amount of elastic feedback activity to plastically adapt the CPG parameters.

In previous work, Owaki and Ishiguro[54] presented a control approach that showed spontaneous gait transition based on mechanical coupling ('physical communication'). The CPG was coupled through mechanical coupling. Buchli et al.[45] presented an adaptive oscillator that adapted its frequency to the natural frequency of a spring-loaded inverted pendulum-like simulation model. In their adaptive frequency oscillator approach, the matching of control frequency and natural frequency led to performance improvements and a reduction in energy requirements. Fukuoka et al.[43] implemented short-term reflexes that adapted the robot's controller to the motion of the robot induced by external perturbations. Through a closed-loop CPG that incorporated the 'rolling body motion' the robot could actively adapt to its surrounding. Thandiackal et al.[38] showed that feedback from hydrodynamic pressure in CPGs can lead to self-organized undulatory swimming. However, there are no approaches for long-term plastic matching of control patterns and the natural dynamics of complex walking systems in passive elastic robots at present.

As learning and exploration in hardware are prone to critical failure, the control patterns were first optimized in simulation (Fig. 3a), as is common practice in robotics[47,50,51]. After successful optimization in simulation, the acquired optimal parameter set was applied in hardware (Fig. 3b). We transferred the optimized CPG parameter set into hardware to measure the performance of the real robot and validate the effectiveness of our approach by evaluating a performance measure.

Although optimization and learning in simulation are efficient and cheap, the transfer of control policies can be difficult due to the sim2real gap[50,51,55]. We examined the transferability of our approach by quantifying the sim2real gap by comparing simulation and hardware experiments.

Here we implemented elastic CPG feedback pathways triggered by foot contact. We utilized this elastic feedback activity to mitigate short-term perturbations. Over the long term, we used the feedback activity as a proxy for the mismatching between Morti's natural dynamics and the control pattern. We plastically minimized the required elastic feedback activity through model-free Bayesian optimization. Our approach enabled Morti to learn a trot gait at $0.3\,\mathrm{m\,s^{-1}}$ within 1 h. Matching improved energy efficiency without explicit formulation in the cost function. The improved energy efficiency is evidence of increased matching.

Results

We first examined the performance of the feedback mechanisms in simulation (Fig. 4). The feedback mechanism for late touchdown ($\mathbf{r}_{LTD}$), shown in red, decelerated the phase of the front left leg to wait until ground contact was established. The deceleration was visible in the flatter gradient of the oscillator phase when the mechanism was active.

The early touchdown mechanism (r_{ETD}) triggered a knee pull-up reflex (purple line) to shorten the leg to prevent further impact. In the event of early toeoff (r_{ETO}), shown in yellow, the knee flexion started earlier than instructed by the feedforward CPG. The late toeoff mechanism (r_{LTO}) measured the mismatching of control task and natural dynamics but did not trigger a feedback mechanism.
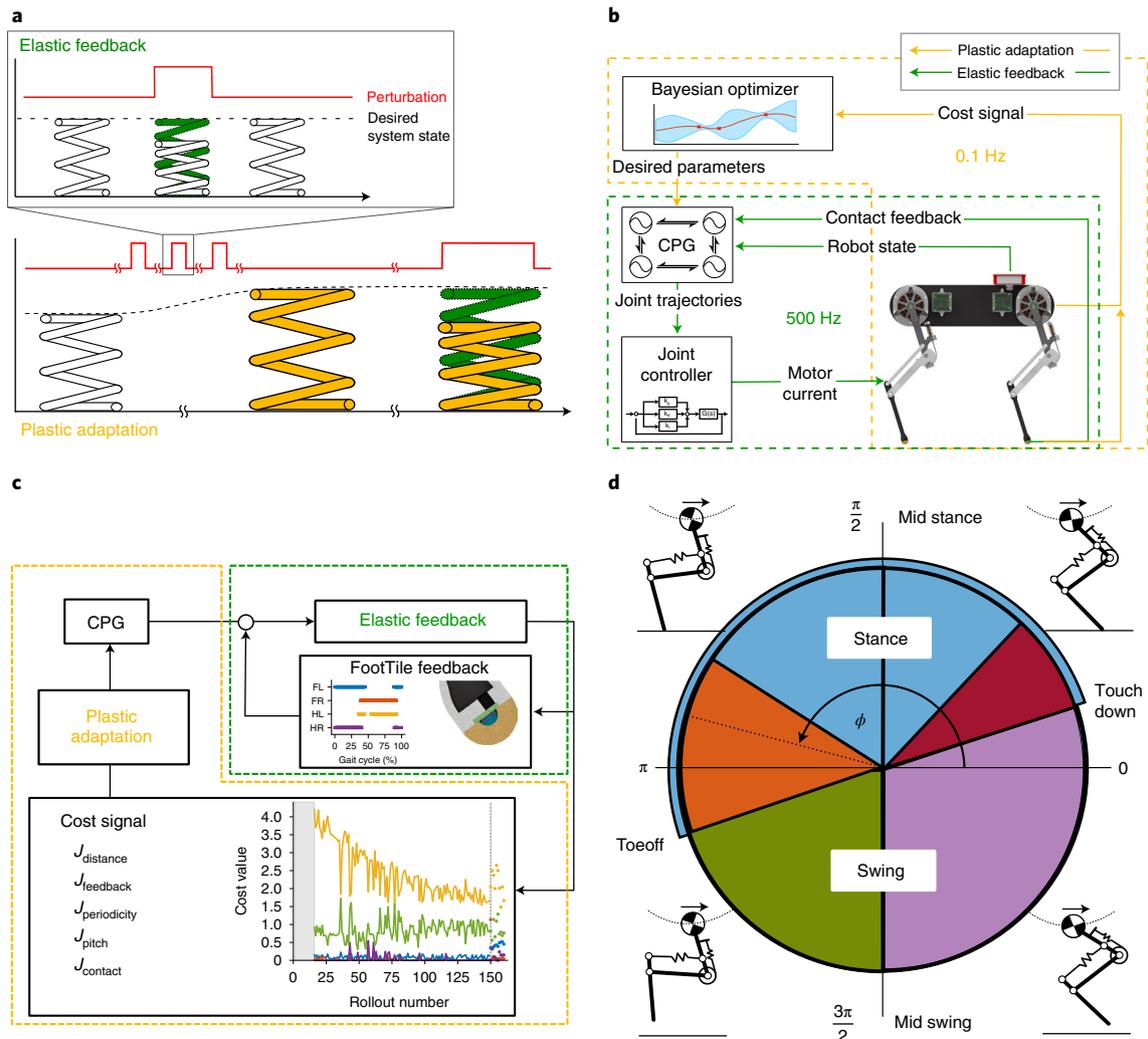
Fig. 2 | Schematic depiction of the elasticity and plasticity framework. a, Short-term elastic feedback (top) and long-term plasticity (bottom). Elastic feedback (green) mitigates stochastic short-term perturbations (red), such as pot holes, that disturb the system (spring) from its desired state (dashed line). Elastic activity is reversible and only active when a perturbation is present, just as a spring only deflects as long as an external force is active and then returns to its initial state. Plasticity (yellow) changes the system behaviour permanently to adapt to long-term active stimuli from the environment. If the same perturbation is frequently present, the system adapts to the perturbation. In our example the spring adapts its set point (spring length, dashed line) and stiffness (spring thickness). In this way, an initial desired system state that might be encoded in the initial control design can be adapted to better deal with perturbations throughout its life span, as well as changing environments. After plastic adaptation the spring deflects less (green, bottom right). **b**, Control structure of Morti. k_p, k_d and k_i are the joint controller gains; $G(s)$ is the plant transfer function in Laplace space s. **c**, Flowchart of the matching approach. The elastic feedback activity mitigates short-term perturbations through sparse contact feedback from the FootTile contact sensors. We measured the amount of elastic feedback activity as a proxy for the mismatching of dynamics. Over a longer time window, the optimizer minimizes the elastic feedback activity to plastically match the control pattern of the CPG to Morti's natural dynamics. F, front; L, left; H, hind; R, right. Colour representations are similar to those in Fig. 5b. **d**, Diagram of a step cycle in phase space (ϕ). The segments are colour-coded by feedback mechanism: late touchdown (red) later than the desired touchdown time ($\delta_{overSwing}$), late toeoff (yellow) later than the desired toeoff time ($\delta_{\phi,knee}$), early toeoff (green) and early touchdown (purple). The stance phase from touchdown to toeoff is shaded blue.

The feedback mechanisms helped Morti to mitigate perturbations stemming from dynamics mismatching. This mitigation effect was especially important in the first rollouts of the optimization, where good dynamics matching was not yet achieved.

In this rollout, the late touchdown mechanism was active for 7% of the step cycle, the early touchdown mechanism was active for 5% of the step cycle, the late toeoff mechanism was active for 8% of the step cycle and the early toeoff mechanism was active for 9% of the step cycle.

We found that 150 rollouts in simulation (Fig. 5a) were sufficient to learn a gait at a speed of 0.3 m s⁻¹. Each rollout in simulation took

an average of 23 s for 20 s of simulation runtime on an Intel i7 CPU, making the whole optimization duration roughly 1 h. The hardware rollouts were roughly 1 min long to ensure stable locomotion. When Morti reached stable behaviour, 10 s were evaluated, as in the simulation rollouts.

After initialization of the Gaussian kernel with 15 rollouts with random CPG parameters, the optimizer started to approximate the cost function and performance converged toward the optimum point.

During the whole optimization, Morti fell 16 times or 11% of rollouts. Nine of the failed rollouts occurred during the first 15 rollouts with random CPG parameters.
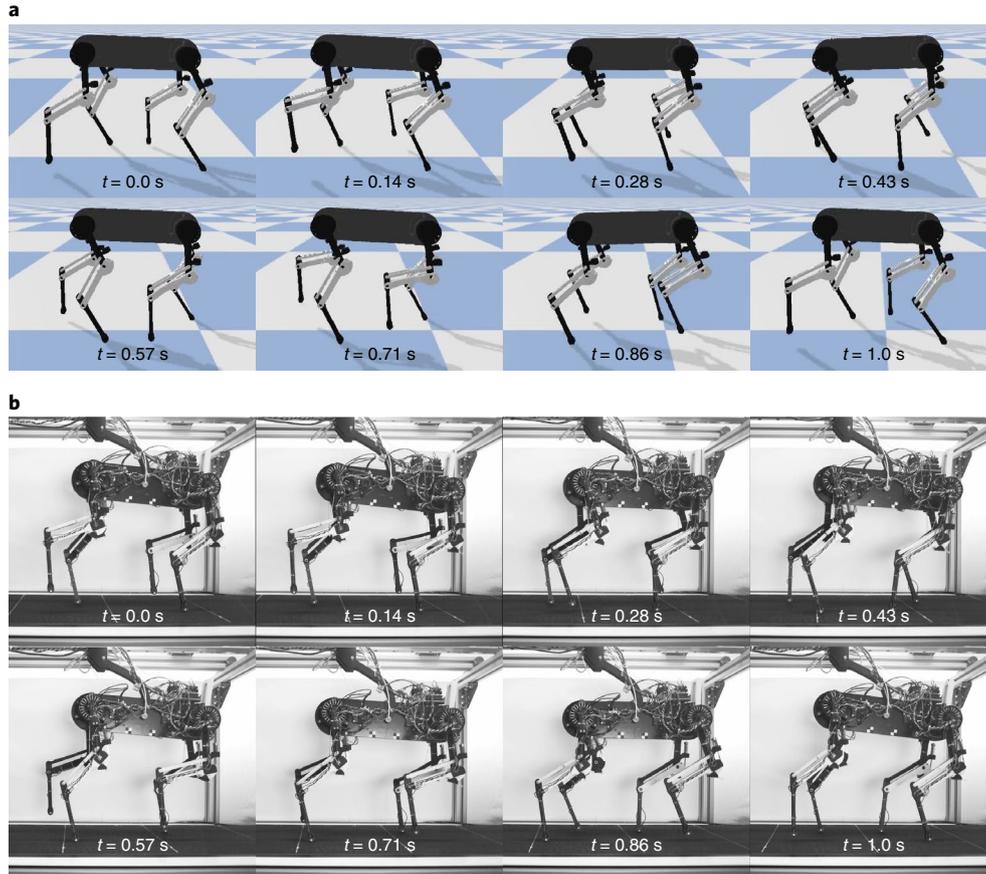
Fig. 3 | Snapshots of the simulation and hardware rollouts. a, Snapshots of the simulated Morti walking during one of the optimization rollouts in the PyBullet multibody simulation at the times (t) indicated. **b**, Snapshots of Morti walking on a treadmill during one of the hardware rollouts.

Through optimization, the simulated robot increased its performance from the least-performing rollout (rollout 107, cost 5.62) to the optimal rollout (rollout 109, cost 2.59) by 215%. In comparison, the simulation results transferred to hardware scored a cost between 5.65 and 4.41. The mean simulation cost was 3.49 ± 0.66 and the median simulation cost was 3.34. The mean hardware cost was 4.96 ± 0.38. The best simulation result was 41% lower than the lowest hardware result.

To validate the performance, as well as the differences between simulation and hardware rollouts in detail, we investigated the individual cost factors (Supplementary Table 3) for both simulation and hardware rollouts (Fig. 5a). We found that no single cost factor was responsible for the higher returned cost. Instead, all cost factors were slightly higher and their summation led to the higher cost returned for the hardware results. The distance cost term ($J_{distance}$) and the feedback cost term ($J_{feedback}$) contributed the highest difference between the simulation and hardware cost values: $J_{distance}$ had a mean hardware cost of 2.13 ± 0.36 compared with a simulation cost of 1.67 and $J_{feedback}$ had a mean hardware cost of 0.43 ± 0.06 compared with a simulation cost of 0.13. We assumed that the difference was due to modelling assumptions that were made in the simulation. The hardware robot showed a lower speed due to contact losses, gearbox backlash, friction and elasticity in the FootTile sensors. During touchdown, imperfect contact of the feet led to higher feedback activity, which imposed a penalty via the feedback cost term. The body pitch cost term J_{pitch} was in the range of the simulated cost; the mean hardware cost was 0.85 ± 0.22 compared to 0.80 in simulation. Morti showed more body pitch both during the optimization shown here and

initial tests for untuned CPG parameters, and it flipped over during several rollouts. This did not happen in hardware—even in early experiments the hardware robot never pitched more than 30°. The periodicity cost term ($J_{periodicity}$) (hardware: 0.15 ± 0.33; simulation: 0.0) and the contact cost term ($J_{contact}$) (hardware: 0.12 ± 0.09, simulation: 0.03) behaved similarly in simulation and hardware rollouts. This similarity was expected as both simulation and hardware gaits converged to the desired gait, and the latter three cost terms were introduced to guide the optimizer to find gaits similar to the desired CPG patterns, mostly during the first rollouts.

At the core of our approach, we hypothesized that matching dynamics improves energy efficiency. We therefore explicitly did not incorporate energy efficiency into the cost function. To quantify how matching dynamics improved energy efficiency we calculated a normalized torque as a measure of performance. We chose a normalized mean torque as we showed in previous work[12] that the torque signal has no major oscillations (Supplementary Fig. 5). It is therefore a sufficient representation of the system's energy requirement and is simple to measure both in simulation and in the hardware robot.

$$\tau_{normal} = \sum_{n=1}^{4} \frac{(\overline{\tau}_{hip,n} + \overline{\tau}_{knee,n})}{\overline{v}_{body}} \qquad (1)$$

where n is the leg index, $\overline{\tau}_{knee}$ and $\overline{\tau}_{hip}$ are the mean knee and hip torque per rollout per leg and $\overline{v}_{body}$ is the mean body velocity of the respective rollout.
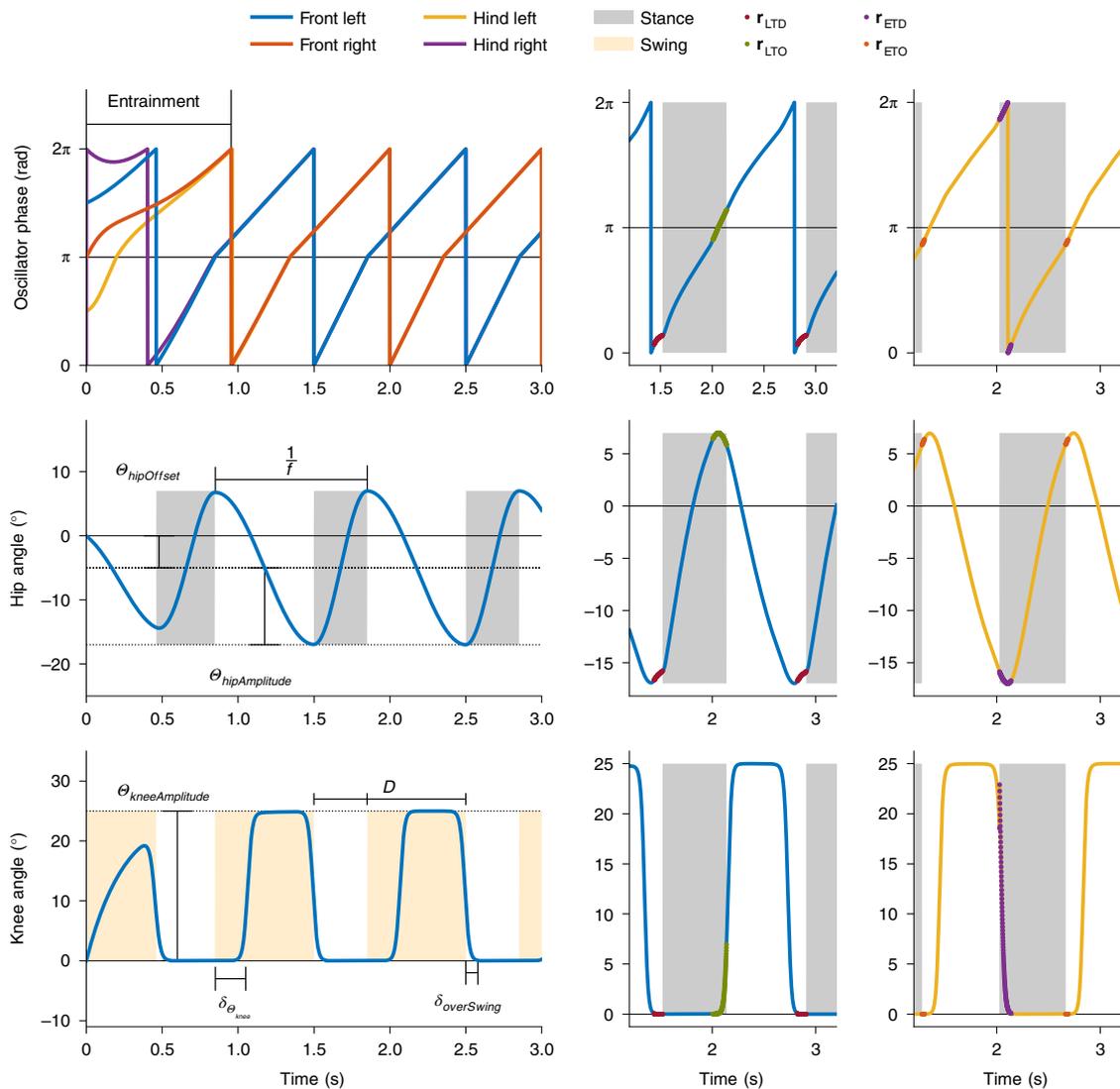
Fig. 4 | CPG parameters and elastic feedback activity. Left: examples of the CPG output for four coupled oscillators and the generated trajectories. The coupled phases are shown at the top, and the hip (middle) and knee (bottom) joint trajectories for one oscillator with their respective CPG parameters are shown below (Supplementary Table 2). Parameters here are $D = 0.35$, $\delta_{\phi,knee} = 0.3$, $\delta_{overSwing} = 0.2$, $f = 1$. Right: simulation results showing the four feedback mechanisms (same colour coding as Supplementary Fig. 2). Data are shown for the front left and front right legs. Late touchdown (red) on the front left leg phase shows the phase delay to wait for touchdown. Early touchdown (purple) on the right leg shows the knee pull-up reflex. Late toeoff (yellow) is shown on the left leg. Early toeoff (green) is shown on the right leg. The stance phase is shaded grey. $\Theta_{hipAmplitude}$, hip amplitude; $\Theta_{kneeAmplitude}$, knee amplitude; $\Theta_{hipOffset}$, hip offset; f, frequency; $\delta_{\Theta knee}$, knee phase shift; $\delta_{overSwing}$, knee overswing; D duty factor as described in Supplementary Section 2.

The initial normalized torque was 2.52, and the final value 1.02. The mean normalized torque was 1.7 ± 0.5 and the median normalized torque was 1.55 (Fig. 6). As expected, the normalized torque reduced over the optimization by 42% from plastically unmatched initial conditions (compare with Supplementary Section 5). The reduction in normalized torque as an efficiency measure confirmed our hypothesis, that matching the control pattern to the system's natural dynamics has beneficial effects on energy requirements.

Discussion

We suggested that enabling a locomotion controller to leverage the passively compliant leg structures could increase the energy efficiency indirectly. By minimizing the required elastic feedback activity, the controller learns to increase the matching between its control pattern and the natural dynamics. We showed that 150 optimization rollouts sufficed to learn a stable trot gait on flat ground at a speed of $0.3\,\mathrm{m\,s^{-1}}$ from random initial conditions with an optimization

duration of 1 h. In our experiment, we showed that matching dynamics is indeed beneficial for energy-efficient locomotion. We calculated a normalized performance measure that showed a decrease in power requirements.

In the normalized torque measure (τ_{normal}), Morti benefited from the increase in distance cost and a reduction in the required torque. Even though Morti increased its speed more than two-fold, the required normalized torque did not increase. Instead, the normalized torque decreased with a trend comparable to the cost function. The improved control pattern matching enabled the controller to leverage the natural dynamics to achieve better performance (Fig. 6).

The designed passive behaviour of Morti enabled a simple matched CPG control structure to leverage the natural dynamics of the leg design. Through sparse binary feedback from touch sensors, the controller was able to elastically mitigate the perturbations stemming from initial mismatching. Through synergy of the natural dynamics and matched CPG, Morti learned to walk on inexpensive
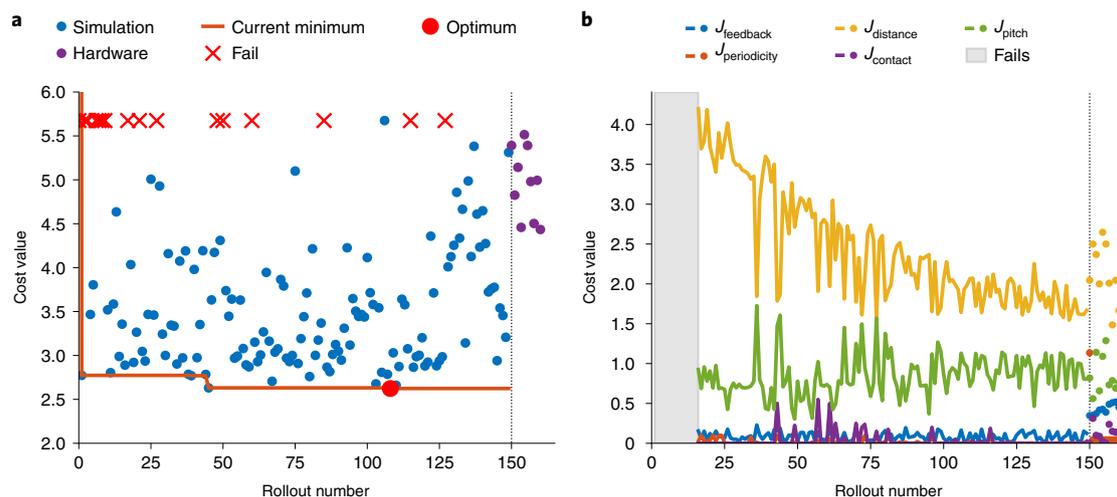
Fig. 5 | Results from plastic adaptation. a, Cost function for the Bayesian optimization. The Bayesian optimization did not monotonically minimize the cost function, so the current minimum is shown. **b**, Individual cost values from the different cost function terms (Supplementary Table 3). The individual cost terms show similar results between the simulation results (lines) and the hardware samples (dots). The mean hardware cost values are similar to the optimal costs from the simulation.
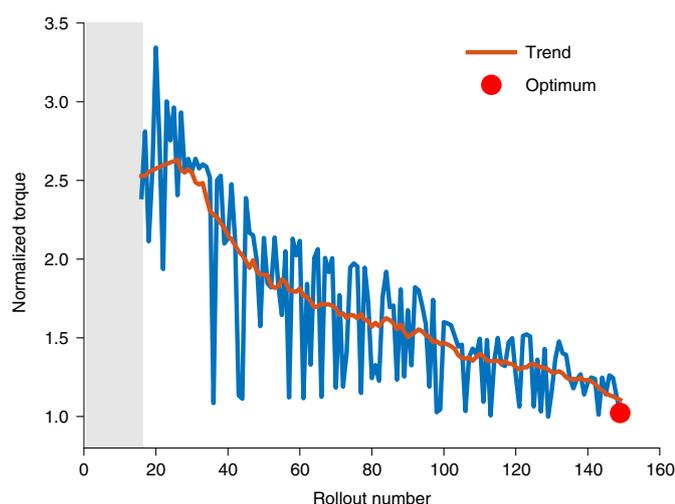

Fig. 6 | Normalized torque performance measure. Normalized torque (torque/speed) reduces over the optimization. Data is sorted by reward. Shaded area marks failed rollouts. By matching the control pattern to Morti's natural dynamics, the required energy for locomotion reduces. Because the controller learns to exploit Morti's natural dynamics, less energy is used to achieve the desired behaviour. The normalized torque was not part of the cost function but minimizes because the matching increases (compare with Supplementary Section 5).

hardware (<€4,000) with low computational power (5 W Raspberry power) and with lower control (500 Hz control loop) and sensor (250 Hz binary sensor signal) frequencies than state-of-the-art model-based locomotion controllers that require high-bandwidth computation and high control frequencies (>2 kHz control frequency, >17 W processor power)[50,56].

Closely examining the cost (Fig. 5b) showed that through dynamics matching and the minimization of $J_{feedback}$, Morti travelled longer distances in the given time, as shown by the improved $J_{distance}$. There was little change in J_{pitch} over the optimization, which was expected because the CPG cannot actively control body pitch. The $J_{periodicity}$ and $J_{contact}$ terms were used as penalty terms for undesired

gait characteristics. They are an order of magnitude lower than the remaining cost terms, and only peaked for less performant rollouts.

Although the gait learned in this work was simpler than state-of-the-art full-body control approaches, we provide evidence that minimizing feedback activity stemming from systematic mismatching between the control pattern and the system's natural dynamics provides an alternative learning approach.

Compared with end-to-end learning approaches[50,51], our method requires fewer rollouts. As the underlying control structure (the CPG network) was predefined, it required no approximation in the learning approach first. On the other hand, the versatility of CPG-based locomotion hinges on the complexity of the chosen CPG model. In this proof of concept, we chose a simplified CPG to limit the complexity of the underlying model and thus the technical implementation of our approach. However, we believe that our approach of minimizing elastic feedback activity to increase the matching between control pattern and natural dynamics is not limited by the choice of pattern generator, and could be transferred to other locomotion controllers that possess a metric for the amount of required elastic feedback.

The discrete feedback events (Fig. 2d) allowed the amount of feedback activity to be measured easily in our example. In systems with continuous feedback such as whole-body control[16,56,57], we believe control effort[58] could be an alternative measure. Our approach is also not limited to Bayesian optimization approaches, and we believe it could be used as part of the cost/reward function for different optimization or learning approaches. In this way, our model-free matching approach could scale to different compliant robots. More generally, our approach could be adapted for more versatile control approaches such as model-based full-body control[56,57] or CPGs with more versatile behaviours[43].

Although other studies reported problems with transferring simulation results to hardware (the sim2real gap)[59–61], we successfully transferred our simulation results to the Morti hardware without post-transfer modifications. The hardware performance was comparable both quantitatively (Fig. 5b) and in qualitative observation of the resultant gaits (Supplementary Videos 1 and 2). We believe that because the joint torques of Morti were not calculated from potentially inaccurate model parameters, the sim2real gap is not as evident here. Learned controllers that directly influence joint torques and leg forces might suffer more from the sim2real gap because smaller

inaccuracies between model and hardware behaviour can have a direct effect on the forces exerted on Morti. However, more research will be required to understand the transferability of results for underactuated robots with strong natural dynamics.

In future work, we intend to extend the CPG, taking body pitch into account when generating the hip trajectories as done by ref. [39]. With an inertial measurement unit the body pitch could be fed back into the CPG. In the current formulation, the CPG assumes no body pitch and relies on the robustness the passive elasticity adds to the system to compensate the existing body pitch. Abduction/adduction degrees of freedom with their respective feedback loops[39,62] could also be added to Morti to enable 3D locomotion without a guiding mechanism. The optimization loop could be implemented to run online on the hardware robot's computer. With online optimization and 3D locomotion, it would become possible to investigate the lifelong adaptation of the CPG control patterns to changing ground conditions and surface properties over extended time windows, as well as adaptations to wear and tear throughout Morti's lifetime.

In this Article, we examined how a walking system with limited control and sensor bandwidth could learn to leverage the intelligence embodied in its leg mechanics. Energy efficiency and speed are often used as criteria to evaluate performance in robotic systems. Here we proposed an additional measure that focuses on the synergy of passive mechanical structures and neural control. By separating feedback by its time horizon, we achieved perturbation mitigation in the short term and at the same time quantified the mismatching of control patterns and natural dynamics. We optimized the long-term performance of the system and adapted the controller to its mechanical system. Although investigated in a robotic surrogate, our findings could provide a new perspective on how learning in biological systems might happen in the presence of neural limitations and sparse feedback. Matching is probably not the sole driving factor in animal learning, but our study suggests that a quantitative measure for 'long-term learning from failure' could in part be influenced by the goal of maximizing the synergy between locomotion control and the robot's or animal's mechanical walking system. In contrast to task-specific cost functions such as speed or energy efficiency, our matching approach provides an intrinsic motivation to leverage the embodied intelligence in the natural dynamics as much as possible.

Methods

For both the experimentation and simulation, we designed and implemented quadruped robot Morti. Morti has a monoarticular knee spring and a biarticular spring between hip and foot that provide series elastic behaviour[12]. It was controlled by a closed-loop CPG. Through reflex-like feedback mechanisms, Morti could elastically mitigate short-term perturbations. To minimize the elastic activity, we implemented a Bayesian optimizer that plastically matched the control pattern to Morti's natural dynamics.

Robot mechanics. Morti consists of four 'biarticular legs' (Fig. 1b; ref. [12]) mounted on a carbon fibre body. Each leg has three segments: femur, shank and foot. The femur and foot segments are connected by a spring-loaded parallel mechanism that mimics the biarticular muscle–tendon structure formed by the gastrocnemius muscle–tendon group in quadruped animals[63]. A knee spring inspired by the patellar tendon in animals provided passive elasticity of the knee joint.

Morti walked on a treadmill and was constrained to the sagittal plane by a linear rail that allowed body pitch (Fig. 1b). It was instrumented with joint angle sensors, position sensors and the treadmill speed sensor. To measure ground contact, four FootTile sensors[64] were mounted on Morti's feet. Using a threshold, these analogue pressure sensors could be used to determine whether it established ground contact. Detailed descriptions of the experimental set-up can be found in Supplementary Section 1.

Simulation. We implemented the simulation in PyBullet[59], a multibody simulator based on the bullet physics engine (Fig. 3a). The robot mechanics were derived from the mechanical robot and its computer-aided design model (Supplementary Table 1). To increase the matching between the robot hardware and simulation, we imposed motor limits and set the motor controller to resemble the real actuator limits[55]. The simulation ran at 1 kHz, the CPG control loop ran at 500 Hz and ground

contacts are polled at 250 Hz to resemble the hardware implementation. The control frequency was chosen for technical reasons to guarantee stable position control and fast data acquisition. It could be lower, as shown in similar systems[5,13], to more closely resemble the neural delays and low technical complexity in animals.

CPG. The CPG used in this work was a modified Hopf oscillator[13,35,62] that was modelled in phase space. More biologically accurate and biomimetic CPG models do exist[37,43,54]; we chose this representation because of its reduced parameter space while retaining the functionality required to generate joint trajectories for locomotion. Similar to their biological counterparts, CPGs can be entrained through feedback from external sensory input[38] or from internal coupling to neighbouring nodes[54]. The CPG in this work consisted of four coupled nodes, representing the four legs (see Supplementary Fig. 1). The hip and knee of each leg were coupled through a variable phase shift. Depending on the desired phase shifts in between oscillator nodes (legs), a variety of gaits can be implemented by adapting the phase difference matrix while keeping the network dynamics identical (Supplementary Section 6). The joint trajectories generated by the CPG are described by eight parameters (Supplementary Table 6): the hip offset ($\Theta_{hipOffset}$) and hip amplitude ($\Theta_{hipAmplitude}$) describe the hip trajectory, the knee offset amplitude ($\Theta_{kneeOffset}$) describes the knee flexion, the frequency f determines the robot's overall speed, duty factors (D) describe the ratio of stance phase to flight phase, the knee phase shift ($\delta_{\phi,knee}$) describes the phase shift between hip protraction and knee flexion and overswing ($\delta_{overSwing}$) describes the amount of swing leg retraction[65]. The mathematical description of the CPG dynamics can be found in Supplementary Section 2.

Elasticity. As the CPG implemented here was written as a model-free feedforward network, it could be difficult to find parameters that lead to viable gaits with given robot dynamics. Essentially, the CPG commands desired trajectories without knowledge of the robot's natural dynamics. In the worst-case scenario the CPG would command behaviour that the robot cannot fulfil because of its own natural dynamics and mechanical limitations such as inertia, motor speed or torque limitations. To address this shortcoming, feedback can be used to mitigate the differences between desired and measured behaviour.

The feedback implemented here was an adaptation from Righetti et al.[35] that has been shown to aid in entrainment and can mitigate perturbations in foot contact information. This contact information can be integrated into the CPG to measure timing differences between the desired and measured trajectories.

The trajectories created by the CPG can be influenced through feedback by changing the CPG dynamics (meaning accelerating or decelerating the CPG's phases). Alternatively, feedback can influence the generated joint angle trajectories. During a step cycle (Fig. 2d), contact signals were used for several feedback mechanisms (Supplementary Fig. 2). The feedback mechanisms reacted to timing discrepancies in the touchdown and toeoff events and corrected the CPG trajectories if Morti established or lost ground contact earlier or later than instructed by the CPG. Righetti et al.[35] showed how these feedback mechanisms can actively stabilize a CPG controlled robot. We adapted these mechanisms to a phase-space CPG formulation and robot hardware to achieve similar performance in a different class of robot.

Using the feedback mechanisms implemented in the CPG, we corrected the timing of discrete events. If touchdown and toeoff happened earlier or later than commanded by the feedforward control pattern, the individual phases of each leg (CPG node) could be accelerated or decelerated to correct Morti's behaviour. We implemented a phase deceleration when touchdown was delayed (Fig. 2d, red). We accelerated knee flexion when a foot lost ground contact too early (Fig. 2d, orange), in addition to a phase deceleration when toeoff occurred later than commanded (Fig. 2d, green). We combined these feedback mechanisms with a knee pull-up reflex when a leg hit the ground too early to mimic a patellar reflex as adapted from ref. [66]. If a leg hit the ground during the swing phase (Fig. 2d, purple), the knee flexed in a predefined trajectory to generate more ground clearance. In addition to the mechanism in ref. [66], we disabled the hip motor from interfering in the passive impact mitigation of the mechanical leg springs. An in-depth description of the feedback mechanisms can be found in Supplementary Section 3 and the CPG output with active feedback can be seen in Fig. 4.

Plasticity. To match the CPG to the robot dynamics, we wanted to tune the CPG parameters p_m to achieve optimal performance. To do so, we evaluated the performance of Morti for a number of steps. The timescale of the optimization was designed to be much bigger than the frequency of the elastic feedback activity mechanisms (≤ 0.1 Hz versus ≥ 100 Hz). Consequently, the effects of the elastic feedback activity were minimized and small perturbations within one step were not captured in the plastic optimization that will only improve long-term performance.

To achieve long-term (close to) optimal behaviour we used Bayesian optimization for its global optimization capabilities, data efficiency and robustness to noise[52,53].

Bayesian optimization. Bayesian optimization is a black-box optimization approach that uses Gaussian kernels for function approximation. It is model-free, derivative-free and has been used successfully in many robotic optimization approaches[67–69]. Bayesian optimization is favoured over other data-driven

optimization and learning approaches because of its data efficiency for ten or more parameters.

We implemented a Bayesian optimizer that was based on skopt gp_minimize (ref. [70]). The optimizer evaluated the PyBullet simulation for 10 s (approximately ten step cycles) of each rollout with a cost function. Morti walked for 10 s to entrain the CPG from its initial condition (standing still; see Fig. 4, top) to achieve steady-state behaviour before the evaluation began. One complete rollout therefore took 20 s. We sampled 15 rollouts with random CPG parameters before approximating the cost function. Then we optimized for 135 rollouts with the gp_hedge acquisition function, which is a probabilistic choice of the lower confidence bound, negative expected improvement and negative probability of improvement.

To reduce complexity we limited the parameter space to six parameters. The six parameters are $\Theta_{hipOffset}$, $\Theta_{hipAmplitude}$, D_{front} and D_{hind}, $\delta_{\phi,knee}$ and $\delta_{overSwing}$ (Fig. 4). More parameters would probably improve performance more, but would also lead to more corner cases where the selected cost function could be exploited by the optimizer and result in undesired gait characteristics (such as skipping gaits) or gaits where the feet drag on the ground. For this proof of concept, we chose independent duty factors D_{front} and D_{hind} to allow some front–hind asymmetry that could help the optimizer find gaits that reduce body pitch. Where only one CPG parameter was selected, the parameter was used for all legs. For simplicity, we also fixed the frequency to $f = 1$ Hz to reduce experimental cost in terms of hardware wear from violent motions at high speed. The hip amplitude $\Theta_{kneeAmplitude}$ is set to 30° to ensure adequate ground clearance.

Cost function. We evaluated Morti using a cost function comprising three major components. The first component evaluated $J_{feedback}$, specifically the percentage of a step cycle that one of the elastic feedback mechanisms was active:

$$J_{feedback} = \frac{1}{fnT} \sum_{n=0}^{4} \sum_{t=0}^{T} \mathbf{r}_{ETO} + \mathbf{r}_{ETD} + \mathbf{r}_{LTO} + \mathbf{r}_{LTD} \qquad (2)$$

where $J_{feedback}$ is the average percentage of active feedback per step and leg, T is the evaluation time and $\mathbf{r}_{ETO}$, $\mathbf{r}_{ETD}$, $\mathbf{r}_{LTO}$ and $\mathbf{r}_{LTD}$ are the time vectors when the specific feedback was active for each leg.

The second component evaluates the distance travelled ($J_{distance}$ $(1/m)$) to encourage forward locomotion.

$$J_{distance} = \frac{1}{x_{body}} \qquad (3)$$

where x_{body} is the centre-of-mass position in the walking direction.

The third component penalized deviations from the commanded gait characteristics. It ensured that Morti moved with a low mean J_{pitch}:

$$J_{pitch} = \| \max(\alpha_{pitch}) - \min(\alpha_{pitch}) \| \qquad (4)$$

and was calculated as the difference between the mean minimum and the mean maximum of body pitch angle (α_{pitch}) of all strides during one rollout.

It also imposed a penalty if more than one ground contact phase per foot and step ($J_{contact}$ [% of step cycle]) occurred, as would take place during stumbling or dragging of the feet.

$$J_{contact} = \frac{1}{fnT} \sum_{n=0}^{4} \sum_{t=0}^{T} \left(\| \frac{d}{dt} \text{ contact } \| > 0 \right) \qquad (5)$$

where $J_{contact}$ is the mean number of flight-stance changes per step, t is time, T is the evaluation time and contact is the contact sensor data matrix for all four legs.

The third component penalized differences between the desired gait frequency and the measured gait frequency to prevent non-periodic gaits or multi-step gaits ($J_{periodicity}$ (Hz)).

$$S_{pitch} = \alpha_{pitch} \times \alpha_{pitch}$$
$$f_{bodyPitch} = \| \max S_{pitch} \| \qquad (6)$$

$$J_{periodicity} = \sqrt{\frac{1}{N} (f_{bodyPitch} - f_{cpg})^2} \qquad (7)$$

where S_{pitch} is the frequency spectrum of α_{pitch}, $f_{bodyPitch}$ is the frequency of the body pitch measurement, $J_{periodicity}$ is the standard deviation of the periodicity measure and f_{cpg} is the commanded CPG frequency.

Detailed descriptions can be found in Supplementary Section 4.

Hardware rollouts. To validate the optimal set of CPG parameters from simulation, we tested the same parameters on the hardware robot. The hardware controller had the same elastic mechanisms described in the 'Elasticity' section. We tested ten parameter sets and randomly varied the CPG parameters obtained from simulation by ≤10% to validate the hardware cost function around the optimal point found in simulation. We then evaluated Morti's performance with the same cost function

used for the simulation. As in the simulation, Morti ran for 10 s to entrain itself. The performance was measured for 10 s after Morti converged on a stable gait.

Videos of Morti walking are available as Supplementary Videos 1 and 2.

Data availability
The experimental data are available at https://doi.org/10.17617/3.XDOQNW (ref. [71]). The robot model and CAD design are available for non-commercial use at the same link.

Code availability
The code and a demo are available at https://doi.org/10.17617/3.XDOQNW (ref. [71]).

References
1. Iida, F. *Embodied Artificial Intelligence* (Springer, 2004).
2. Alexander, R. Elastic energy stores in running vertebrates. *Am. Zool.* **24**, 85–94 (1984).
3. Blickhan, R. The spring-mass model for running and hopping. *J. Biomech.* **22**, 1217–1227 (1989).
4. More, H. L. & Donelan, J. M. Scaling of sensorimotor delays in terrestrial mammals. *Proc. R. Soc. B* **285**, 20180613 (2018).
5. Ashtiani, M. S., Sarvestani, A. A. & Badri-Spröwitz, A. T. Hybrid parallel compliance allows robots to operate with sensorimotor delays and low control frequencies. *Front. Robot. AI* **8**, 645748 (2021).
6. Collins, S. Efficient bipedal robots based on passive-dynamic walkers. *Science* **307**, 1082–1085 (2005).
7. Franklin, G. *Feedback Control of Dynamic Systems* (Prentice Hall, 2002).
8. Daley, M. A. Running over rough terrain: guinea fowl maintain dynamic stability despite a large unexpected change in substrate height. *J. Exp. Biol.* **209**, 171–187 (2006).
9. Renjewski, D., Spröwitz, A., Peekema, A., Jones, M. & Hurst, J. Exciting engineered passive dynamics in a bipedal robot. *IEEE Trans. Robot.* **31**, 1244–1251 (2015).
10. Luo, X. & Xu, W. Planning and control for passive dynamics based walking of 3D biped robots. *J. Bionic Eng.* **9**, 143–155 (2012).
11. Ruina, A. Passive dynamics is a good basis for robot design and control, not! *Princeton University MAE* https://mae.princeton.edu/about-mae/events/passive-dynamics-good-basis-robot-design-and-control-not (2017).
12. Ruppert, F. & Badri-Spröwitz, A. Series elastic behavior of biarticular muscle-tendon structure in a robotic leg. *Front. Neurorobotics* **13**, 8 (2019).
13. Spröwitz, A. et al. Towards dynamic trot gait locomotion: design, control, and experiments with cheetah-cub, a compliant quadruped robot. *Int. J. Robot. Res.* **32**, 932–950 (2013).
14. Lee, H. & Hogan, N. Time-varying ankle mechanical impedance during human locomotion. *IEEE Trans. Neur. Syst. Rehab. Eng.* **23**, 755–764 (2015).
15. Tedrake, R. Zhang, T. W. & Seung, H. S. Learning to walk in 20 minutes. In *IEEE International Conference on Robotics and Automation* 4656–4661 (IEEE, 2004).
16. Bhounsule, P. A. et al. Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge. *Int. J. Robot. Res.* **33**, 1305–1321 (2014).
17. Geyer, H., Seyfarth, A. & Blickhan, R. Spring-mass running: simple approximate solution and application to gait stability. *J. Theor. Biol.* **232**, 315–328 (2005).
18. Geyer, H., Seyfarth, A. & Blickhan, R. Compliant leg behaviour explains basic dynamics of walking and running. *Proc. R. Soc. B* **273**, 2861–2867 (2006).
19. Rummel, J. & Seyfarth, A. Stable running with segmented legs. *Int. J. Robot. Res.* **27**, 919–934 (2008).
20. Rummel, J., Blum, Y. & Seyfarth, A. Robust and efficient walking with spring-like legs. *Bioinspir. Biomim.* **5**, 046004 (2010).
21. Kenneally, G., De, A. & Koditschek, D. E. Design principles for a family of direct-drive legged robots. *IEEE Robot. Autom. Lett.* **1**, 900–907 (2016).
22. Bicanski, A. et al. Decoding the mechanisms of gait generation in salamanders by combining neurobiology, modeling and robotics. *Biol. Cybernet.* **107**, 545–564 (2013).
23. Dominici, N. et al. Locomotor primitives in newborn babies and their development. *Science* **334**, 997–999 (2011).
24. Grasso, R. et al. Distributed plasticity of locomotor pattern generators in spinal cord injured patients. *Brain* **127**, 1019–1034 (2004).
25. Kudithipudi, D. et al. Biological underpinnings for lifelong learning machines. *Nat. Mach. Intell.* **4**, 196–210 (2022).
26. Marjaninejad, A., Urbina-Meléndez, D., Cohn, B. A. & Valero-Cuevas, F. J. Autonomous functional movements in a tendon-driven limb via limited experience. *Nat. Mach. Intell.* **1**, 144–154 (2019).
27. Mastalli, C. et al. Trajectory and foothold optimization using low-dimensional models for rough terrain locomotion. In *2017 IEEE International Conference on Robotics and Automation* 1096–1103 (IEEE, 2017).

28. Kwiatkowski, R. & Lipson, H. Task-agnostic self-modeling machines. *Sci. Robot.* **4**, 26 (2019).
29. Mitteroecker, P. & Stansfield, E. A model of developmental canalization, applied to human cranial form. *PLoS Comput. Biol.* **17**, e1008381 (2021).
30. Sadd, M. *Elasticity: Theory, Applications, and Numerics* (Elsevier/Academic Press, 2009).
31. Marder, E. & Bucher, D. Central pattern generators and the control of rhythmic movements. *Curr. Biol.* **11**, R986–R996 (2001).
32. Matsuoka, K. Mechanisms of frequency and pattern control in the neural rhythm generators. *Biol. Cybernet.* **56**, 345–353 (1987).
33. Bizzi, E., Tresch, M. C., Saltiel, P. & d'Avella, A. New perspectives on spinal motor systems. *Nat. Rev. Neurosci.* **1**, 101–108 (2000).
34. Dickinson, M. H. How animals move: an integrative view. *Science* **288**, 100–106 (2000).
35. Righetti, L. & Ijspeert, A. J. Pattern generators with sensory feedback for the control of quadruped locomotion. In *2008 IEEE International Conference on Robotics and Automation* 819–824 (IEEE, 2008).
36. Xie, F., Zhong, Y., Du, R. & Li, Z. Central pattern generator (CPG) control of a biomimetic robot fish for multimodal swimming. *J. Bion. Eng.* **16**, 222–234 (2019).
37. Ijspeert, A. J., Crespi, A., Ryczko, D. & Cabelguen, J.-M. From swimming to walking with a salamander robot driven by a spinal cord model. *Science* **315**, 1416–1420 (2007).
38. Thandiackal, R. et al. Emergence of robust self-organized undulatory swimming based on local hydrodynamic force sensing. *Sci. Robot.* **6**, eabf6354 (2021).
39. Sartoretti, G. et al. Central pattern generator with inertial feedback for stable locomotion and climbing in unstructured terrain. In *2018 IEEE International Conference on Robotics and Automation* 5769–5775 (IEEE, 2018).
40. Oliveira, M., Matos, V., Santos, C. P. & Costa, L. Multi-objective parameter CPG optimization for gait generation of a biped robot. In *2013 IEEE International Conference on Robotics and Automation* 3130–3135 (IEEE, 2013).
41. Yeganegi, M. H. et al. Robust humanoid locomotion using trajectory optimization and sample-efficient learning. In *International Conference on Humanoid Robots* 170–177 (IEEE, 2019).
42. Bongard, J., Zykov, V. & Lipson, H. Resilient machines through continuous self-modeling. *Science* **314**, 1118–1121 (2006).
43. Fukuoka, Y., Kimura, H., Hada, Y. & Takase, K. Adaptive dynamic walking of a quadruped robot 'Tekken' on irregular terrain using a neural system model. In *2003 IEEE International Conference on Robotics and Automation* IEEE Cat. No.03CH37422 (IEEE, 2003).
44. Buchli, J. & Ijspeert, A. J. Self-organized adaptive legged locomotion in a compliant quadruped robot. *Auton. Robots* **25**, 331–347 (2008).
45. Buchli, J., Righetti, L. & Ijspeert, A. J. in *Advances in Artificial Life* 210–220 (Springer, 2005).
46. Pearlmutter, B. A. Learning state space trajectories in recurrent neural networks. *Neur. Comput.* **1**, 263–269 (1989).
47. Heim, S., Ruppert, F., Sarvestani, A. A. & and Spröwitz, A. Shaping in practice: training wheels to learn fast hopping directly in hardware. In *2018 IEEE International Conference on Robotics and Automation* 5076–5081 (IEEE, 2018).
48. Matsubara, T., Morimoto, J., Nakanishi, J., Sato, M. & Doya, K. Learning CPG-based biped locomotion with a policy gradient method. In *5th IEEE-RAS International Conference on Humanoid Robots* 208–213 (IEEE, 2005).
49. Nakamura, Y., Mori, T. & Ishii, S. Natural policy gradient reinforcement learning for a CPG control of a biped robot. In *Parallel Problem Solving from Nature VIII* (eds Yao, X. et al.) (Springer, 2004).
50. Siekmann, J. et al. Learning memory-based control for human-scale bipedal locomotion. In *Robotics: Science and Systems Conference* (Robotics: Science and Systems Foundation, 2020).
51. Peng, X. B. et al. Learning agile robotic locomotion skills by imitating animals. In *Robotics: Science and Systems XVI* (Robotics: Science and Systems Foundation, 2020).
52. Calandra, R., Seyfarth, A., Peters, J. & Deisenroth, M. P. Bayesian optimization for learning gaits under uncertainty. *Ann. Math. Artif. Intell.* **76**, 5–23 (2015).
53. Mockus, J. *Bayesian Approach to Global Optimization* (Springer, 2012).
54. Owaki, D. & Ishiguro, A. A quadruped robot exhibiting spontaneous gait transitions from walking to trotting to galloping. *Sci. Rep.* **277**, 3 (2017).
55. Tan, J. et al. Sim-to-real: learning agile locomotion for quadruped robots. In *Robotics: Science and Systems XIV* (Robotics: Science and Systems Foundation, 2018).
56. Park, H.-W. & Kim, S. The MIT cheetah, an electrically-powered quadrupedal robot for high-speed running. *J. Robot. Soc. Jpn* **32**, 323–328 (2014).
57. Hutter, M. et al. ANYmal: a highly mobile and dynamic quadrupedal robot. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2016).
58. Haeufle, D. F. B., Günther, M., Wunner, G. & Schmitt, S. Quantifying control effort of biological and technical movements: an information-entropy-based approach. *Phys. Rev. E* **89**, 012716 (2014).
59. Coumans, E. & Bai, Y. pybullet version (3.0.7) (2016); http://pybullet.org
60. Rosser, K., Kok, J., Chahl, J. & Bongard, J. Sim2real gap is non-monotonic with robot complexity for morphology-in-the-loop flapping wing design. In *2020 IEEE International Conference on Robotics and Automation* 7001–7007 (IEEE, 2020).
61. Heiden, E., Millard, D., Coumans, E., Sheng, Y. & Sukhatme, G. S. NeuralSim: augmenting differentiable simulators with neural networks. In *IEEE International Conference on Robotics and Automation* 9474–9481 (IEEE, 2021).
62. Spröwitz, A. T. et al. Oncilla robot: a versatile open-source quadruped research robot with compliant pantograph legs. *Front. Robot. AI* **5**, (2018).
63. Witte, H. et al. Transfer of biological principles into the construction of quadruped walking machines. In *Second International Workshop on Robot Motion and Control* IEEE Cat. No.01EX535 245–249 (Poznan University Technology, 2001).
64. Ruppert, F. & Badri-Spröwitz, A. FootTile: a rugged foot sensor for force and center of pressure sensing in soft terrain. In *2020 IEEE International Conference on Robotics and Automation* 4810–4816 (IEEE, 2020).
65. Seyfarth, A., Geyer, H. & Herr, H. Swing-leg retraction: a simple control model for stable running. *J. Exp. Biol.* **206**, 2547–2555 (2003).
66. Focchi, M. et al. in *Nature-Inspired Mobile Robotics* (eds Waldron, K. J. et al.) 443–450 (World Scientific, 2013).
67. Gianni, M., Garcia, M. A. R. & Pirri, F. Learning the dynamics of articulated tracked vehicles. *Zenodo* https://doi.org/10.5281/zenodo.1124704 (2016).
68. Marco, A., Hennig, P., Bohg, J., Schaal, S. & Trimpe, S. Automatic LQR tuning based on Gaussian process global optimization. In *2016 IEEE International Conference on Robotics and Automation* 270–277 (IEEE, 2016).
69. Seyde, T., Carius, J., Grandia, R., Farshidian, F. & Hutter, M. Locomotion planning through a hybrid Bayesian trajectory optimization. In *2019 International Conference on Robotics and Automation* 5544–5550 (IEEE, 2019).
70. Head, T., Kumar, M., Nahrstaedt, H., Louppe, G. & Shcherbatyi, I. scikit-optimize version 0.9.0 (Python Software Foundation, 2020); https://pypi.org/project/scikit-optimize/
71. Ruppert, F. & Badri-Spröwitz, A. Learning plastic matching of robot dynamics in closed-loop central pattern generators: data. *Edmond* https://doi.org/10.17617/3.XDOQNW (2022).

Author contributions
F.R. contributed to the concept, design, experiments, data analysis and writing. A.B.S. contributed to the concept and provided feedback and supervision.

Competing interests
The authors declare no competing interests.

Additional information
Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42256-022-00505-4.

Correspondence and requests for materials should be addressed to Felix Ruppert.

Peer review information *Nature Machine Intelligence* thanks Joao Ramos, Fernando Perez-Peña, Ren Qinyuan and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.