Title: A neuromechanics solution for adjustable robot compliance and accuracy

Authors: Ignacio Abadía[1*], Alice Bruel[2], Grégoire Courtine[3], Auke J. Ijspeert[2], Eduardo Ros[1], Niceto R. Luque[1*]

Affiliations:
[1]Research Centre for Information and Communication Technologies, Department of Computer Engineering, Automation and Robotics, University of Granada, Spain.
[2]Biorobotics Laboratory, EPFL, Lausanne, Switzerland.
[3]NeuroX Institute and .NeuroRestore, EPFL/CHUV/UNIL, Lausanne, Switzerland.
*Corresponding author: iabadia@ugr.es (I.A.); nluque@ugr.es (N.R.L.)

Abstract: Robots have to adjust their motor behavior to changing environments and variable task requirements in order to successfully operate in the real world and physically interact with humans. Thus, robotics strives to enable a broad spectrum of adjustable motor behavior, aiming to mimic the human ability to function in unstructured scenarios. In humans, motor behavior arises from the integrative action of the central nervous system and body biomechanics; motion must be understood from a neuromechanics perspective. Nervous regions such as the cerebellum facilitate learning, adaptation and coordination of our motor responses, ultimately driven by muscle activation. Muscles, in turn, self-stabilize motion through mechanical viscoelasticity. Besides, the agonist-antagonist arrangement of muscles surrounding joints enables cocontraction, which can be regulated to enhance motion accuracy and adapt joint stiffness, thereby providing impedance modulation and broadening the motor repertoire. Here, we propose a control solution that harnesses neuromechanics to enable adjustable robot motor behavior. Our solution integrates a muscle model replicating mechanical viscoelasticity and cocontraction, together with a cerebellar network providing motor adaptation. The resulting cerebello-muscular controller drives the robot through torque commands in a feedback control loop. Changes in cocontraction modify the muscle dynamics; and the cerebellum provides motor adaptation without relying on prior analytical solutions, driving the robot in different motor tasks, including payload perturbations and operation across unknown terrains. Experimental results show that cocontraction modulates robot stiffness, performance accuracy and robustness against external perturbations. Through cocontraction modulation, our cerebello-muscular torque controller enables a broad spectrum of robot motor behavior.

One-Sentence Summary: Muscle cocontraction and cerebellar adaptation adjust robot accuracy and compliance, enabling a wide motor behavior spectrum.

Main Text:

INTRODUCTION

Robotics development is pursuing autonomous intelligent robots able to dynamically adapt their motor behavior to the environment and task requirements (*1*). Traditional industrial robots are designed for specific tasks and controlled, structured environments, prioritizing performance accuracy and operation speed. Nevertheless, robotics applications are expanding beyond manufacturing plants into the real world, defining new applications that involve collaboration through physical human-robot interactions (HRI). The evolving constraints of physical HRI require these collaborative robots to change their priorities over time, thus demanding adjustable motor behavior. For example, assistive robots must perform motor tasks accurately and discern objects by their fragility for delicate manipulation, but also allow the human to freely drive their motion if required;

rehabilitation robots must adapt to the evolution of the patient abilities (*2*). Even a seemingly simple task as an object handover involves different phases that require both the giver and receiver to dynamically adapt their behavior (*3*); when receiving a tool from a human, the robot motion should be soft when approaching the tool to not destabilize nor compromise the human safety, whereas the subsequent task execution using the tool demands more robustness from the robot. Thus, the range of tasks and environments encountered by collaborative robots requires them to adaptively transition between robust and softer, compliant motor behavior depending on the circumstances.

These motor requirements arise from the deployment of robots in the unstructured real world, and can be addressed by developing robots that mimic the living organisms that inhabit it (*4*). In nature, adaptive motor behavior is facilitated by the central nervous system (CNS). Robotics has thus looked to neuroscience to develop brain-inspired technologies that endow robots with some sort of biomimetic behavior (*5*): generalization to unseen scenarios (*6*), precise navigation (*7*), multitasking (*8*), adaptation to changes in the dynamics (*9*), or robustness against nondeterministic time delays (*10*). However, motor behavior is not shaped exclusively by what occurs inside the brain; it is also substantially influenced by body biomechanics (*11, 12*). This is exemplified by the passive-dynamics walker robots, in which mechanical properties enable stable locomotion with minimal actuation and rather primitive control (*13*). Consequently, robotics also looks to biomechanics to incorporate and harness some of its inherent features: agile locomotion across different terrains (*14*), variable stiffness actuators (*15*), adhesive behavior to diverse surfaces (*16*), or the lifelike capabilities to change shape, modulate stiffness, self-heal, or grow, qualities that soft robotics aims to replicate (*17*). Therefore, combining brain-inspired and biomechanics solutions shall foster robots with ever more biomimetic behavior.

The versatility of human motor behavior very well reflects this interaction between neural control and body morphology. For instance, our distinctive hand dexterity, crucial for precise and adaptive manipulation, emerges from the synergy of the CNS and biomechanics (*18*). Actually, our physical interactions with the world are mainly mediated through the upper limbs, whose functionality is directly linked to their morphology (*19*). The human upper limb is actuated by no less than twenty muscles that control seven degrees of freedom (DOF) (three DOF in the shoulder, two in the elbow, and two in the wrist, excluding the hand DOFs) (*20*). This muscle redundancy and over-actuation add to the complexity of the musculoskeletal dynamics, already sophisticated due to the nonlinear properties of muscles, tendons, and other soft tissues (*21*). Such dynamic intricacy requires fine control mechanisms in the CNS to govern motion, but at the same time muscle actuation offers inherent benefits. The nonlinear muscle viscoelasticity induces self-stability (*22, 23*), provides an intrinsic zero-delay response to perturbations (*24*), and simplifies neuronal information processing by off-loading computation to the morphological structure (*25-27*). Furthermore, muscle passive viscoelasticity can be actively controlled to modify behavior (*12*). Apart from these intrinsic mechanical properties, the arrangement of agonist-antagonist muscle groups to actuate joints enables cocontraction: the simultaneous activation of antagonistic muscles surrounding a joint. Cocontraction can be actively modulated to increase performance accuracy, regulate joint stiffness and, thereby, modify the dynamic response to perturbations, thus broadening the motor repertoire (*28-30*).

These muscle mechanical properties and their agonist-antagonist distribution have inspired new robotic actuators (*15, 31-34*); the challenges of performing in unstructured environments can be approached from a hardware perspective that mimics functional biological tissues (*35*). Nonetheless, hardware-based solutions face manufacturing challenges and are often limited to specific platforms, which can constrain their wide-scale adoption (*1, 35, 36*), and introducing nonlinear dynamics complicates accurate and reliable control (*37*).

Besides hardware implementations, biological tissue can also inspire software-based control approaches. Indeed, muscle redundancies, viscoelasticity, and cocontraction inspired impedance control (*38-40*), which was developed to regulate the dynamic interactions between a manipulator and its environment. Impedance control posits that when two physical systems interact one must physically complement the other: if one system behaves as an impedance (accepts motion inputs and yields force outputs) the other must be an admittance (accepts force inputs and yields a motion response). In its original conception, this control paradigm considers the environment as an admittance since it can always be pushed upon but does not always move. Hence, the manipulator is considered an impedance for complementarity, reflecting the causality from detected motion to applied force (*39, 40*). Impedance control can be implemented to maintain a specific impedance across interactions or to regulate and adjust the impedance of the robot depending on the encountered circumstances through variable impedance control (*41, 42*).

Impedance control implementations depend on well-designed control laws addressing the robot dynamics, a complex issue without a straightforward solution for nonlinear robots, and the desired impedance model, which depends on the environment and task requirements (*41*). In unstructured scenarios, the environment properties are unknown; thus, reliable measurements of the contact force between the robot and the environment are usually required to model the desired impedance (*43*). Consequently, impedance control often relies on extensive analytical development, including robot dynamics, desired impedance, environment information and/or specialized hardware for sensing capabilities. Both control-based and learning-based approaches have been developed to achieve variable impedance control, but the limitations imposed by their technical requirements reduce their applicability and spread as broad robotic solutions (*42*).

The ultimate goal of impedance control lies on adjusting the dynamic behavior of the robot (*44*). Anthropomorphic solutions can facilitate this achievement. To reduce dependency on reliable sensory feedback, humans adjust our interactions with the environment by regulating limb stiffness (*45*). Rather than directly controlling the force exerted on physical objects, modulating limb stiffness through muscle cocontraction allows for active regulation of our interactions, without prior knowledge about the environment and without accurate modeling information.

Inspired from these biological features, we model muscle mechanical properties and include them in a robot torque controller. We replicate muscle viscoelasticity and agonist-antagonist actuation drawing from Ekeberg's computational muscle model (*46*); a model previously applied to study animal locomotion (*47*), and robust undulatory swimming in robots (*48*). We expand the muscle model by incorporating tunable cocontraction and a spinal cord (SC) reflex mechanism, and apply it to six DOFs of the Baxter robot arm (*49*) performing different motor tasks: continuous trajectories at different speeds, external perturbations, and operation across different terrains. By adjusting muscle cocontraction, which also modulates the SC-based reflex, we can regulate the robot stiffness, the performance accuracy, and the response to interactions. Thus, we achieve muscle-like actuation using electric motors without needing antagonistic actuators. We also integrate a cerebellar spiking neural network (SNN) to control the muscle dynamics and drive the robot toward its motor goal. The cerebellum, key for biological motor control and learning (*50-54*), has been previously validated for robot control (*9, 10*); however, these previous implementations lacked motor behavior regulation. In the current implementation, the cerebellar SNN generates high level muscle activation signals, in contrast to previous implementations of muscle dynamics in robots, which typically relied on oscillatory activation signals (*48, 55*). Subsequently, the muscle dynamics generates the joint torque signals that drive the robot motion. The cerebellar SNN is equipped with synaptic plasticity, allowing for motor adaptation to control the unknown plant dynamics without prior analytical knowledge; model-free control, as both the muscle and robot dynamics are initially unknown to the cerebellum.

In this Article, we propose a neuromechanics approach for adjustable robot motor behavior (Fig. 1). By harnessing muscle dynamics and cerebellar adaptability, our approach achieves adjustable motor behavior through the regulation of the robot stiffness, without prior analytical modeling (model-free), without environment information and hence applicable to unknown environments, and without contact force sensing or specific hardware as it relies just on position sensory feedback. Integrating brain-inspired and biomechanics solutions, our cerebello-muscular torque controller enables the adjustment of the robot motor behavior, striking a dynamic trade-off between performance accuracy and compliance: high accuracy and robustness against perturbations, vs. adopting lower accuracy and softer, compliant behavior in response to interactions.

RESULTS

Integration of cerebellum, muscle dynamics, and robot

We implemented a cerebellar SNN and a software-based muscle model, both combined to build a cerebello-muscular torque controller operating within a feedback control loop with the Baxter robot as the front-end body (Fig. 1B-C). The cerebellar network adapted the motor commands needed to perform different tasks involving six DOFs, where each robot DOF was governed by the dynamics of the implemented muscle model, in addition to Baxter built-in series elastic actuators (*49*). The cerebellar network, built upon previous work (*9, 10, 56*), used the following sensory inputs: joint coordinates describing the desired trajectory for the motor task (position θ_d, and velocity θ'_d); actual joint coordinates of the robot (position θ_a, and velocity θ'_a); instructive signal (ε) computing the mismatch between the desired and actual joint states. Based on this input sensory information, the cerebellar network generated the agonist and antagonist muscle activation commands (A_F for flexion, and A_E for extension, per joint), which were then sent to the muscle model. Our muscle model, derived from (*46*), simulated a pair of antagonistic muscles and was developed to include adjustable cocontraction (c) and a reflex component based on SC circuitry. Hence, each robot joint had its corresponding muscle model which simulated an agonist-antagonist muscle pair and included: active control of agonist-antagonist dynamics (flexion-extension, commanded by cerebellar signals A_F and A_E), cocontraction level, an SC-based reflex response (R_{SC}), and passive viscoelastic properties. The combination of the different muscular components produced a torque signal per joint (τ) which was then sent to the robot, thus closing the feedback loop. The desired joint coordinates and the adjustable cocontraction level were fed into the loop by a module representing motor cortex functionality (*57-60*). Please, refer to Materials and Methods for a comprehensive description of the muscle model, the cerebellar SNN, and the robot.

To adjust the robot motor behavior, we modified the muscle dynamics by varying the cocontraction value (*c*), impacting both the active and passive muscle properties (see Materials and Methods). To explore a rich spectrum of robot behavior, we conducted tests within a wide cocontraction range: $c = [x0, x1, x3, x5, x7, x9, x10]$, where *x0* corresponds to $c = 0.0$, *x1* is $c = 0.1$, *x3* is $c = 0.3$, and so on. The following sections detail our cerebello-muscular torque controller performing with different cocontraction levels and its effect on key tenets of HRI: performance accuracy, torque response, and reaction against external perturbations (soft vs. robust behavior). An overview of our neuromechanics approach is shown in Movie 1.

Enhancement of performance accuracy through cocontraction

First, we assessed the influence of cocontraction on performance accuracy. Two motor tasks constituted our motor control benchmark: a circular trajectory executed in the vertical plane; a figure eight trajectory executed in the horizontal plane. Each task was performed in 2.5 s; please refer to Supplementary Materials for the desired joint coordinates specific to each motor task (fig. S1). For each cocontraction value tested, the cerebello-muscular controller underwent 1000 consecutive trials of the trajectory. The trial-and-error cerebellar learning process started without prior knowledge of the plant dynamics and, as cerebellar adaptation progressed, it allowed operation of the given muscle dynamics to command the robot in the execution of the motor tasks, leading to a gradual reduction of the performance error (Fig. 2A and B). As the motor adaptation process concluded, our cerebello-muscular torque controller accurately tracked the desired joint positions (Fig. 2C and D).

Increased cocontraction levels correlated with a more consistent execution of the motor task as trajectory variability decreased (Fig. 2E and F), demonstrating a significant enhancement in performance accuracy measured as the position mean absolute error (MAE) (Fig. 2G and H). For all cocontraction values, the performance improved the reference provided by the factory-default position controller. Nevertheless, it was observed that increasing cocontraction does not yield an endless enhancement of performance accuracy; the highest cocontraction values (*x10* for the circular trajectory, *x9* and *x10* for the figure eight trajectory) did not significantly reduce the error beyond what was achieved by preceding cocontraction values.

Enhancement of performance accuracy through cocontraction is limited by motion speed

Previous studies have reported that, biologically, cocontraction stabilizes the upper limb and enhances motion accuracy (*61, 62*); however, its benefits diminish during high-speed ballistic movements, which become spastic in the presence of cocontraction (*63*). To test whether these findings still apply to our robotic setup, we evaluated the influence of cocontraction on performance accuracy when increasing the motion speed. We extended the original motor benchmark by testing the following trajectory durations: vertical circle performed in 2.5, 2.0, 1.5, 1.0, and 0.8 s; horizontal figure eight trajectory performed in 2.5 and 2.0 s. Note that the figure eight trajectory performed in 2.0 s forced the joint speed close to the hardware limitations specified by the robot manufacturer, thus preventing further increments in speed of the figure eight trajectory for safety reasons. Refer to Supplementary Materials for a description of the joint coordinates of each motor task performed at different speeds (fig. S1).

For the slower executions of the circular trajectory (2.5, 2.0, and 1.5 s), increasing cocontraction improved the performance accuracy, except for the higher cocontraction value (*x10*) in the 2.5 s and 1.5 s trajectories, and cocontraction value *x9* in the 2.0 s trajectory, which did not significantly modify the tracking error (Fig. 3A). Conversely, for faster trajectories (1.0 and 0.8 s), the highest cocontraction value (*x10*) did not enhance but significantly worsened the trajectory execution. This deterioration was already noticeable with cocontraction value *x7* for the fastest trajectory (Fig. 3A). Regarding the figure eight trajectory, previous observations showed that for the slowest trajectory (2.5 s) cocontraction values *x9* and *x10* did not significantly affect the trajectory tracking. For the fastest trajectory (2.0 s), we now observed that cocontraction *x5* significantly increased the tracking error compared to the immediately lower cocontraction value, and higher values did not further modify the performance accuracy (Fig. 3B).

Muscle torque modifications induced by cocontraction

The joint torque commanded to the robot was given by the output of the muscle dynamics. Varying the cocontraction value modified the muscle dynamics (see Materials and Methods), thereby affecting the output muscle torque (Fig. 4A and fig. S2 for the circular trajectory, and fig. S3 for the figure eight trajectory). To assess the torque changes induced by cocontraction, we computed the joint torque time-integral (TTI) as a measurement of the delivered energy (*64*), a metric also used in neuromuscular studies to account for the ability to maintain force over time (*65-67*). This involved calculating the integral of the joint torque over the motor task duration (2.5 s for both the circular and figure eight trajectories; see Materials and Methods). Although statistically significant variations were observed for certain joints (Fig. 4B), the overall joint TTI change was relatively small; the maximum TTI increment among all cocontraction values was 0.24 Nm·s for joint *j1* in the circular trajectory and 1.18 Nm·s for joint *j2* in the figure eight trajectory (see Fig. 1B and fig. s12 for joint labels). As reference for joint torque magnitudes, the robot manufacturer specifies a maximum torque of 50 Nm for joints *j1*, *j2*, *j3* and *j4*, and 15 Nm for joints *j5* and *j6*.

To provide a deeper insight into the torque response, we broke down the output muscle torque into its different muscle dynamics components, and measured the corresponding time-integral. According to biological plausibility, the final muscle torque resulted from the interplay of both active and passive components of the muscle dynamics. The cerebellum actively controlled the flexion-extension direction of movement (Fig. 4C), which experienced a maximum time-integral increment among all cocontraction values of 2.27 Nm·s (joint *j1*) for the circular trajectory, and 1.90 Nm·s (joint *j4*) for the figure eight trajectory (Fig. 4D). The SC-based reflex component corrected deviations from the desired trajectory, exhibiting an increase with cocontraction (Fig. 4E). The maximum time-integral increment among all cocontraction values was 2.66 Nm·s (joint *j2*) for the circular trajectory, and 3.59 Nm·s (joint *j4*) for the figure eight trajectory (Fig. 4F). On the passive side of the muscle dynamics, we considered two components accounting for the muscle mechanical elasticity and viscosity, respectively. The elastic component exhibited an increase with cocontraction (Fig. 4G), showing a peak increment of 3.68 Nm·s (joint *j1*) for the circular trajectory, and 3.71 Nm·s (joint *j1*) for the figure eight trajectory (Fig. 4H). Conversely, the viscous component exhibited relatively smaller variations with cocontraction changes (Fig. 4I), showing a peak increment among all cocontraction values of 0.04 Nm·s (joint *j6*) for the circular trajectory, and 0.21 Nm·s (joint *j6*) for the figure eight trajectory (Fig. 4J). Importantly, the changes induced by cocontraction to the different components of the muscle dynamics were compensatory: when combining all components to generate the output muscle torque, the time-integral changes were mitigated; all muscular components, except the viscous one, exhibited higher time-integral changes than the resulting output muscle torque.

Enhancing robot physical agency through cocontraction: response to external perturbations

After analyzing the influence of cocontraction on performance accuracy, we investigated its influence on the robot behavior when reacting to external perturbations. Once motor adaptation had been settled, we attached a payload to the end-effector during the execution of the motor tasks and measured the deviation in trajectory tracking for each cocontraction value. We used a 0.5 kg payload for the circular trajectory (Fig. 5), and a 1 kg payload for the figure eight trajectory (fig. S4). To obtain a precise measurement of the payload-induced deviation and isolate the role of cocontraction, we initially disabled cerebellar synaptic plasticity, thus preventing cerebellar adaptation to dynamic changes and inhibiting convergent behavior (*9*). Hence, the payload effect remained over time and the perturbation response could only be attributed to the cocontraction level, providing a comprehensive assessment of the performance variations (before and after payload attachment) for each cocontraction value (Fig. 5A and fig. S4 A). We observed that as cocontraction increased, the robot motor response exhibited greater robustness to perturbations; higher cocontraction values resulted in smaller deviations when attaching the payload. In accordance with biological motor behavior, a direct relation between cocontraction and limb stiffness was found. We measured the limb stiffness as the ratio between the applied force (the mass of the attached payload times the gravitational acceleration) and the corresponding deformation (the vertical displacement of the end-effector produced by the attached payload). The differences in the robot arm configuration when performing each trajectory (especially at the position of joint *j6*) lead to the differences in the limb stiffness for each trajectory. Nonetheless, a direct relation between cocontraction and limb stiffness was found in both cases (see table s8 for the limb stiffness results).

To better contextualize the cerebellar control of the muscle-robot dynamics, and the behavioral spectrum enabled by our cerebello-muscular controller across the cocontraction range (that is, the degree of motor variability achieved), we tested the response to the payload of a proportional-derivative (PD) controller governing the muscle dynamics under each cocontraction value (Fig. 5A and fig. S4 A). The differences between low and high cocontraction values were less pronounced with PD control than with cerebellar control, thereby indicating a narrower range of enabled robot behavior. For the circular trajectory, the cerebellar control of the muscle dynamics allowed a 15.6 times larger range of payload reactions than the PD control: from cocontraction *x0* to *x10*, the cerebellar control showed a deviation range from 0.099 to 0.021 rad, and the PD control ranged from 0.014 to 0.009 rad. Similarly, for the figure eight trajectory, the cerebello-muscular controller showed a payload reaction range 29 times larger than the PD control: from cocontraction *x0* to *x10*, the cerebellar control case allowed a deviation range from 0.069 to 0.011 rad, and the PD control ranged from 0.006 to 0.004 rad (fig. S4 B and fig. S5 B).

To further deepen the analysis of the behavioral spectrum allowed by our cerebello-muscular controller, we defined a motor behavior metric based on the trajectory deviation induced by the payload. In an HRI environment, physical interactions are determined by the reactions of both agents; the human and the robot. When the physical intentions of the agents are in conflict, the stronger intention will prevail. The more robust agent will dominate the interaction, thus indicating higher "physical agency". Conversely the softer agent will allow a greater deviation from its path, indicating lower physical agency. To quantify this concept, we defined a robot physical agency (RPA) metric based on the trajectory deviation resulting from the applied perturbations. The maximum deviation in position MAE observed for each trajectory (0.099 rad for the circular trajectory; 0.069 rad for the figure eight trajectory, both allowed by the cerebello-muscular controller with cocontraction *x0*) constituted our RPA value of 0.0; the softest reaction to the perturbation. Conversely a hypothetical deviation of 0.0 rad represented our maximum RPA value of 1.0; the most robust reaction to the perturbation. The MAE deviation associated with each cocontraction value resulted in an RPA value normalized within this range (see Materials and Methods). Results revealed that the cerebello-muscular controller enabled the robot to exhibit low and high RPA behaviors (both soft and robust reactions), whereas the PD controlling the muscle dynamics only enabled the robot to exhibit high RPA behaviors (Fig. 5B and fig. S4 C). Also note that the already narrow behavior range provided by the PD would be further restricted if the PD were to be tuned with higher gains to improve its accuracy. We observed an inverse relationship between the PD gains and the payload reaction range; higher PD gains resulted in a narrowed spectrum of enabled motor behavior (fig. S5). We also tested direct PD control without muscle dynamics, which also demonstrated an inverse relationship between PD gains and payload induced deviation. Importantly, even with low PD gains, which resulted in a significant reduction of accuracy, a soft and compliant behavior was not achieved. Therefore, similar to PD-muscular control, direct PD control did not provide the motor behavior versatility enabled by our cerebello-muscular controller (fig. S5). Please refer to movie s1 for visualizing the different robot motor behaviors allowed by the cocontraction range.

We then activated cerebellar synaptic plasticity to enable the cerebellum to adapt to the attached payload. Again, higher cocontraction values consistently produced a more robust response, resulting in smaller deviations from the target trajectory. After the initial response to the external payload perturbation, cerebellar adaptation progressed effectively, showing a convergent behavior and allowing successful performance of the motor task despite the presence of the payload for all concontraction values (Fig. 5C and fig. S4 D).

Finally, we verified a known biological motor behavior: reacting with high cocontraction as the initial response to new dynamics, followed by a gradual decrease in cocontraction as learning and adaptation evolve (*68-70*). In our experimental setup, the robot initially performed the motor task with high cocontraction values when the payload was attached. As the cerebellum adapted to the new dynamics, we were able to gradually decrease cocontraction (Fig. 5D and fig. S4 E).

Motor adaptation to low cocontraction enables a seamless transition to higher cocontraction levels

Biological motor learning begins with high muscle cocontraction to minimize initial errors, and it is followed by a gradual decrease in cocontraction levels as learning progresses (*68*). In the early stages of learning, motion is confined to high cocontraction levels, as low-cocontraction motor control skills have not been yet acquired; as learning advances, cocontraction can be reduced, and hence the metabolic cost of motion diminishes. This implies that by the time low cocontraction motion becomes part of the motor repertoire, high cocontraction has been previously mastered, and a seamless transition between cocontraction levels is facilitated. We validated this biological behavior in our robotic setup by applying the learned cerebellar solution developed for a specific cocontraction value, to control the muscle dynamics corresponding to other cocontraction values, for both the circular (Fig. 6) and figure eight (fig. S6) trajectories.

We first applied the cerebellar solution developed for controlling the muscle dynamics at the lowest cocontraction value (*x0*), to control the muscle dynamics for the higher cocontraction values. Cerebellar learning was disabled and the performance started with cocontraction value *x0*, and after 50 trials cocontraction was modified, thus illustrating the continuity of the cerebellar solution. Results showed that mastering the control of the muscle dynamics for cocontraction *x0* enabled successful control of all the higher cocontraction values (Fig. 6A and fig. S6 A). Subsequently, we applied the cerebellar solution developed for controlling the muscle dynamics at an intermediate cocontraction value (*x5*), to control the muscle dynamics for both lower and higher cocontraction values. After 50 trials performed with cocontraction *x5*, the cocontraction value was modified. Results showed that the cerebellar solution provided precise motor control for higher cocontraction values, but performance declined for lower cocontraction values. Since performance at cocontraction *x1* substantially deteriorated, providing a compelling result, the cerebellar solution acquired for cocontraction *x5* was not further applied to control cocontraction *x0*, hence prioritizing the integrity of the robot (Fig. 6B and fig. S6 B). Finally, applying the cerebellar solution developed for controlling the muscle dynamics at the highest cocontraction value (*x10*) to all the lower cocontraction values revealed precise motor control for cocontraction values *x9* and *x7* (again, the first 50 trials were performed with cocontraction *x10*, and then cocontraction was modified). However, for cocontraction *x5* and *x3*, the tracking error substantially increased. Again, to prioritize the integrity of the robot, the cerebellar solution acquired for cocontraction level *x10* was not further tested to control cocontraction levels *x1* and *x0* for the circular trajectory, and *x0* for the figure eight trajectory (Fig. 6C and fig. S6 C).

To understand why cerebellar solutions learned at low cocontraction are effective with higher cocontraction levels, but the opposite does not hold, we measured the differences between the learning of these low and high cocontraction cerebellar solutions. After cerebellar motor adaptation to each cocontraction value, we evaluated the complexity of the neural solution at the granule cells – Purkinje cells (GC–PC) layer; the only cerebellar layer endowed with synaptic plasticity. This assessment involved computing the entropy of the synaptic weight distribution at the GC–PC layer, providing a value that represents the degree of complexity of the acquired cerebellar solution (see Materials and Methods). The synaptic entropy measurements exhibited a decreasing pattern as cocontraction increased (Fig. 6D and fig. S6 D), and also showed a high correlation with the position MAE tracking provided by the corresponding cocontraction value. When considering only the synaptic entropy measurements corresponding to cocontraction values that significantly reduced the tracking error (statistical significance shown in Fig. 2G and H), the entropy measurements followed a linear regression with negative slope (Fig. 6E and fig. S6 E). This implies that the higher the cocontraction, the less complex the required cerebellar solution, suggesting that cocontraction simplified cerebellar motor learning.

Active modulation of cocontraction across different terrains

Results presented so far used externally provided cocontraction, homogeneous to all joints. However, to enable a continuous spectrum ranging from soft to robust motor behavior, active modulation of muscle cocontraction is required. Hence, we developed a strategy to dynamically adjust cocontraction depending on the environment and task requirements, based on a weighted trade-off between compliance (soft behavior) and accuracy (robust behavior).

To actively modulate cocontraction, a seamless transition between cocontraction levels must be guaranteed. As observed in the previous section, cerebellar adaptation to low cocontraction enables a seamless transition to higher cocontraction levels; this finding provides the foundation for our cocontraction adjustment strategy. Initially, cerebellar adaptation was set to operate with cocontraction *x0*; thus mimicking the biologically natural motion with low cocontraction in free-space (*68, 69*), and enabling the motor repertoire to effectively range from low to high cocontraction. Then, for a given motor task, priority toward compliant or robust behavior was specified. If compliance was prioritized, cocontraction *x0* was maintained regardless of interactions or perturbations. Conversely, if robustness was prioritized, the cocontraction level increased in response to deviations from the desired trajectory; the stronger the perturbation, the greater the increase in cocontraction. The priority between compliance and robustness was weighted as $W_{C\text{-}R} \in [0.0, 1.0]$, which could range within the full spectrum: from $W_{C\text{-}R} = 0.0$ to establish full compliance priority, to $W_{C\text{-}R} = 1.0$ to establish full robustness priority. The weighted trade-off and the cocontraction level were joint-specific, thus allowing for the specific adjustment of the softness/robustness of each joint (see Materials and Methods).

To test active modulation of cocontraction, we expanded the motor benchmark by making the robot perform in different terrains. We designed a T-shaped trajectory (see fig. S1 for the trajectory description) to which the cerebellum adapted by operating in free-space with cocontraction *x0*. Then, while performing the trajectory, we submerged the robot end-effector in fluids of different viscosity (η), consisting of aqueous solutions of xanthan gum (XG) at different concentrations. Specifically, we used 25 L of water and XG in concentrations of 1%, 2%, 3%, 4%, 5%, and 6% (see Supplementary Materials for the viscosity values of the different XG concentrations). The properties of XG allowed us to modify the viscosity of the fluid (*71*); the higher the viscosity, the greater the constraint for the end-effector motion while submerged in the fluid. For each terrain, we conducted tests with the robot in compliance priority mode ($W_{C\text{-}R} = 0.0$) and robustness priority mode ($W_{C\text{-}R} = 1.0$) (Fig. 7A-D, and fig. S7). When compliance was prioritized, the cocontraction level was maintained at *x0* as the end-effector was submerged in the fluid (Fig. 7C), resulting in the robot avoiding strong reactions to the perturbations, limiting the end-effector motion and increasing the overall trajectory deviation (Fig. 7D). Conversely, when robustness was prioritized, the cocontraction level increased as the end-effector was submerged in the fluid, resulting in higher cocontraction levels corresponding to increased fluid viscosity (Fig. 7C). The robustness enabled by the increased cocontracion allowed the end-effector to cover a wider range of the trajectory despite the fluid viscosity, resulting in a reduced overall trajectory deviation (Fig. 7D). For visualization of cocontraction active modulation across different terrains, please refer to movie s2. To focus on the contribution of cocontraction to adapt the motor behavior to different terrains, cerebellar adaptation was disabled in these experiments. With cerebellar adaptation enabled, the cerebellum would gradually absorb the cocontraction contribution, resulting in a progressive reduction of the cocontraction level and error convergence to lower values (fig. S8).

Finally, we tested the differentiation of compliant and robust behavior depending on the direction of motion. This was allowed by joint-specific cocontraction; perturbations in one direction were allowed by maintaining low cocontraction in specific joints, whereas perturbations in another direction were rejected by triggering a cocontraction increase in other joints. We differentiated between frontal and lateral motion of the end-effector. The T-shaped trajectory and the robot arm configuration determined that frontal motion resulted from the action of joints *j2*, *j4*, and *j6*; and lateral motion resulted from the action of joint *j1*. For each terrain, we tested two behavior configurations (Fig. 7E-G, and fig. S9): frontal robustness and lateral compliance ($W_{C\text{-}R}$ = 1.0 for robust behavior at joints *j2*, *j4*, *j6*, and $W_{C\text{-}R}$ = 0.0 for compliant behavior at joint *j1*); frontal compliance and lateral robustness ($W_{C\text{-}R}$ = 0.0 for compliant behavior at joints *j2*, *j4*, *j6*, and $W_{C\text{-}R}$ = 1.0 for robust behavior at joint *j1*). In both cases, joints *j3* and *j5* were set to robust behavior. In the first configuration, the robustness of joints *j2*, *j4*, and *j6* allowed the end-effector to cover a larger distance in the frontal direction, and lateral motion was more constrained. In the second configuration, the robustness of joint *j1* allowed the end-effector to cover a larger distance in the lateral direction, and frontal motion was more constrained (Fig. 7G).

DISCUSSION

Robots operating in unstructured scenarios face time-changing physical interactions with the environment, other robots, and/or humans (physical HRI). Fine interactions with the physical world require adjustable robot motor behavior to ensure stability; a trade-off between performance and safety requirements must be reached (*72*). For instance, contact with an unknown surface requires the robot to adjust its stiffness to maintain stability. Similarly, interactions with another agent, such as during a handover, require the robot to adapt to the other's actions and consequences. During physical HRI, low robot stiffness facilitates the human to freely drive the robot; high stiffness is beneficial when the robot is demanded to perform with high accuracy in free-space or a soft, controlled environment. Such adaptive motor behavior has been mastered in nature through the combination of the CNS and body biomechanics (*12*), providing the right balance between precision and flexibility for fine interactions. Biological motor behavior must be understood from a neuromechanics integrative view that can lead to advanced control solutions for robotics. Within this integrative scope, we mimicked this evolutionary solution by combining a cerebellar SNN with synaptic plasticity, and a muscle model that includes an SC-reflex mechanism, cocontraction and mechanical viscoelasticity. This integration built our cerebello-muscular torque controller, enabling adjustable robot behavior to meet task requirements: regulation of performance accuracy and robustness/softness to external perturbations. The muscle model tunable cocontraction allowed to modify the muscle dynamics and, consequently, change the robot stiffness and adjust its motor behavior. The cerebellar SNN provided the required motor adaptation to command the muscles and drive the robot toward its motor goal despite dynamic changes. We validated the cerebello-muscular torque controller through a motor control benchmark comprising various trajectories at different speeds, external perturbations, and operation across unknown terrains.

The CNS can facilitate movement accuracy by increasing muscle cocontraction, as demonstrated by previous studies wherein cocontraction reduced trajectory variability and tracking error in human arm goal-directed movements (*28-30*). In our robotic setup, the increase in cocontraction reduced the trajectory variability and tracking error, thus validating the application of muscle cocontraction to enhance robot performance accuracy. However, there is a limit to this improvement of performance accuracy: there exists a finite range beyond which increasing cocontraction does not improve the performance accuracy. The effective cocontraction range depends on the nature of the movement (trajectory and speed of motion); cocontraction requires proper modulation rather than reckless increase.

Another feature of biological cocontraction is that it strengthens joint stiffness (*29, 30*). It is a means to modulate the mechanical impedance of the body and adjust motor behavior to the environment (*38*), facilitating dynamic stability (*73*) and calibrating the kinematic deviations induced by external perturbations as higher cocontraction allows smaller deviations under external forces (*61*). In robotics, this adjustable behavior is addressed by variable impedance control. Research efforts have been devoted to vary the robot impedance during task execution and enable operation across unknown environments. However, the proposed solutions usually face the following limitations (*42*): traditional variable impedance control relies on accurate dynamic models, which are often nontrivial; unknown environment properties can be compensated for using complex sensing capabilities, but at the cost of reducing the generalization of these solutions to diverse robots; learning-based strategies can circumvent the need for analytical modeling but usually depend on specific cost functions and policies for each desired motor behavior.

Apart from variable impedance control, other approaches address a continuous spectrum of motor behavior. That is the case of unified admittance-impedance control, in which the causality of the controller varies depending on the circumstances: impedance causality (motion inputs and force outputs) is generally better suited for interactions with stiff environments, and admittance causality (force inputs and motion outputs) is better suited for interactions with soft environments (*40, 74, 75*). Analytical constraints have usually limited the proposed solutions, and future work is expected to expand these approaches to nonlinear, coupled, multi-DOF robots (*74-76*). Here, we adopt impedance causality from a neuromechanics perspective, and experimentally validate our cerebello-muscular torque controller using a six DOF robot with nonlinear joints.

The original impedance control method already proposed biological muscle redundancies as a solution to modulate mechanical impedance without requiring feedback (*40*). Indeed, to modulate physical interactions, humans regulate limb stiffness to reduce the dependency on reliable feedback (*45*). Following that approach, we used muscle cocontraction to regulate the robot stiffness and enable a continuous spectrum of adjustable robot motor behavior: low and high cocontraction allowed for soft and robust behavior against external perturbations, respectively. Besides, by including the cerebellar SNN, we can also address the non-ideal effects of nonlinear dynamics and communication delays, a challenge for the unified admittance-impedance control (*77*), which cerebellar SNN robot control inherently copes with (*9, 10*). Our neuromechanics approach presents a model-free solution, in which impedance is adjusted through muscle activity and integrated as part of the whole system dynamics; cerebellar adaptation acquires the robot and muscle dynamics as a whole. Our control approach does not require separate control laws for robot dynamics and desired impedance, unlike other solutions that implement individual control laws for each (*78*).

Importantly, our cerebello-muscular torque controller improved the performance accuracy and modulated the motor behavior through subtle changes in the output commands, and it demonstrated a suitable complementarity between the cerebellar slower motor control and the muscular faster motor behavior. Thus, our controller draws another parallel with biological motor control, in which the mechanical properties of muscles and cocontraction allow to induce substantial changes in behavior by small shifts in the motor commands arising from neural circuitry (*12*).

Studying the synaptic weight distributions of the cerebellar learned solutions elucidated the cocontraction effects on cerebellar learning. Results showed that the higher the cocontraction, the lower the entropy of the cerebellar GC–PC synaptic weight distribution; high cocontraction levels lead to simpler learning computational requirements. This finding is aligned with biological evidence that demonstrates that cocontraction facilitates acquisition of internal models, and that it is observed in early stages of learning and then gradually reduced as learning progresses and stabilizes (*68*, *70*). In a previous study we showed that the SC facilitates cerebellar motor learning (*56*), echoed in the present work where the cocontraction increase implied higher SC contribution through stretch-reflex. Notably, the cerebellar solution acquired for a given cocontraction level could be effectively applied to control the robot under higher cocontraction levels. Thus, motor adaptation to low cocontraction allowed our cerebello-muscular torque controller to seamlessly transition to higher cocontraction levels. This transition enables a continuous spectrum of robot motor behavior, with soft and robust motor behaviors at opposing ends of the spectrum. Importantly, this finding also implies a reduction in the operation time required for training: motor learning can be conducted only at low cocontraction, which will also allow performance at high cocontraction without requiring specific training. This is a desirable feature for learning-based methods, which can compensate for inaccurate modeling but usually at the cost of extensive training and data inefficiency (*42*), and typically relying on specific cost functions to achieve different motor behaviors (*79*).

Furthermore, the seamless transition between cocontraction levels enabled the active modulation of cocontraction based on the terrain and task requirements. Following natural biological motor behavior (low cocontraction motion in free-space, and cocontraction increase to counteract perturbations), our neuromechanics approach modulates cocontraction to adapt the robot behavior to unknown terrains; the greater the motion constraint imposed by the terrain, the higher the cocontraction increase and subsequent robot stiffness. Besides, our cocontraction adjustment strategy strictly adhered to the definition of impedance itself (acceptance of motion inputs and response as force outputs), and relied solely on position feedback. This contrasts with variable impedance solutions that rely on contact force sensing capabilities or prior knowledge of the environment characteristics (*78*, *80*). Other variable impedance solutions mimic human stiffness regulation through more complex and specific hardware such as leader-follower motion tracking systems or electromyography (EMG) sensors (*81*, *82*). Our solution directly equips the robot with muscle dynamics to mimic the human behavior, thus rendering robot control independent of external operation or devices. The implemented strategy allowed for a continuous spectrum of robot motor behavior with low technical requirements, simplifying its implementation and application to other robots.

Lastly, cocontraction modulation was joint-specific, enabling more versatile motor behaviors. We differentiated motor behaviors depending on the direction of motion, allowing the robot to reject or accept perturbations depending on their direction. Such targeted, specific behavior is one of the goals of impedance control (*41*). An ideal impedance controller shall provide the accuracy and robustness of well-designed controllers, along with the flexibility and generalization capabilities of learning-based controllers (*42*). Our neuromechanics approach combines cerebellar adaptability and muscle dynamics to provide a model-free solution for adjustable accuracy, generalization across a continuous spectrum from soft to robust motor behavior, and operation in unknown environments.

Muscle actuation poses some challenges for motor control (nonlinearities, higher dimensionality of the control signals) but also presents many benefits (self-stability, computational load reduction, stiffness modulation through cocontraction) (*22-24*, *26*, *27*), and it has inspired hardware-based solutions for robot control (*32-34*). Here, we have enlarged the family of muscle-inspired solutions by integrating a software-based muscle model and a cerebellar SNN, which tackles the challenges and harnesses the benefits of muscle-like actuation. Importantly, the software nature and low technical requirements of our approach facilitate its integration in other robotic platforms, in contrast to other solutions that rely on specific equipment. Here, the robot was driven through torque commands, however, deployment of our solution might as well be feasible using other approaches such as torque estimation from current sensing. Further improvement of the model could be also achieved by incorporating more intricate SC circuits demonstrated to facilitate cerebellar learning (*56*), beyond the SC-based reflex present in the current model.

Our benchmark demonstrated the core features of our neuromechanics approach, with potential applications in physical HRI scenarios. For example, rehabilitation assistance robots should provide motor guidance and movement resistance tailored to the patient's recovery needs (*83*). During initial recovery stages, the robot could prioritize accuracy to assume higher motor responsibilities and guide the patient's movement, and then shift to compliance priority to promote the patient to lead the motion as recovery progresses. For movement resistance, the robot stiffness could be adjusted depending on the therapy demands. Additionally, feedback from the neuromechanics controller could inform the therapist about the patient's progress. Another example is household robots which must prioritize human safety when interactions occur (*84*) but, at the same time, be able to perform tasks that require accuracy and/or varying stiffness, for instance, different stiffness for cleaning or polishing different surfaces. Beyond HRI, neuromechanics approaches could also facilitate locomotion across unstructured terrains; adjustable stiffness is key for successful animal locomotion across various terrains (*85*). The easy implementation of our approach could allow robots to directly benefit from this biological feature and enable stable locomotion in varying environments (*86, 87*).

Our work embraces previous studies that have elucidated CNS operation and musculoskeletal biomechanics. Continued progress in neuromechanics research will keep pushing forward the development of more advanced robots.

MATERIALS AND METHODS

Objective and study design

We explored the applicability to robotics of a fundamental feature of biological motor control: variable muscle cocontraction to adjust motor behavior in terms of performance accuracy and compliance. To that end, we merged a cerebellar SNN model equipped with synaptic plasticity, a muscle model incorporating both active and passive muscular properties, and a torque-controlled six DOF robot (Baxter). The combination of these three components constituted our cerebello-muscular robot control loop. The integration used the Robot Operating System (ROS), which facilitated modularity (*88*). The following sections provide further details into the specifics of these elements.

The muscle model

The muscle model derived from Ekeberg's original muscle implementation, which modeled the mechanical forces induced by the muscles of swimming fish (*46*), and later adapted to robot control for undulatory swimming research (*48*). These previous implementations modeled an antagonistic muscle pair (the model comprised a flexor and an extensor muscle), and accounted for the viscoelastic properties of muscles, approximated as linear spring-dampers. We extended Ekeberg's model to introduce an adjustable cocontraction term and an SC-based reflex component. In our muscle model, the resulting output torque (τ) was given by:

$$\tau_j = \alpha_j (A_{F,j} - A_{E,j}) + R_{SC,j} + \beta_j (\mathrm{A}_{F,j} + A_{E,j} + 2c_j + \gamma_j)(\theta_{r,j} - \theta_{a,j}) + \delta_j \theta'_{a,j} \tag{1}$$

$$R_{SC,j} = \begin{cases} \alpha_j c_j \left(\dfrac{\theta_{d,j} - \theta_{a,j}}{\Delta\theta_{\max}} \right) & \text{if } (\theta_{d,j} - \theta_{a,j}) < \Delta\theta_{\max} \\ \alpha_j c_j & \text{if } (\theta_{d,j} - \theta_{a,j}) \geq \Delta\theta_{\max} \end{cases} \tag{2}$$

where $j \in [1, 6]$ stands for the joint index; α_j corresponds to the muscle activation gain of each joint; A_F and A_E are the cerebellar muscle activation commands for flexion and extension, respectively; R_{SC} is the SC-based reflex component; β_j is the muscle stiffness gain; c stands for the cocontraction level; γ_j is the muscle tonic stiffness; $\theta_{r,j}$ is the joint resting position; θ_a is the joint actual position; δ_j stands for the damping coefficient; θ'_a is the joint velocity. Please refer to Supplementary Materials (table s1) for the values of each muscle parameter. The SC-based reflex component was determined by the cocontraction level (c) and the mismatch between the desired and actual joint positions (θ_d and θ_a). It saturated when the mismatch exceeded the maximum allowed deviation ($\Delta\theta_{max}$=0.35 rad for all joints). The adjustable cocontraction level established a baseline muscle activation in both flexion and extension direction of movement; A_F and A_E became ($A_F + c$) and ($A_E + c$). Varying the c values across a wide range adjusted the muscle dynamics, resulting in different motor behaviors.

The output muscle torque (τ) resulted from the combination of the different muscle components (Fig. 4): active component; SC-based reflex component; elastic component; viscous component. The cerebellum directly controlled the muscle active component by adapting the descending activation commands to drive the movement in either the flexion or extension direction. From Eq. 1, the muscle active component was characterized by:

$$\alpha_j (A_{F,j} - A_{E,j}) \tag{3}$$

The reflex component (R_{SC}), defined by Eq. 2, operated as a spinal stretch reflex based on the equilibrium point (EP) hypothesis (*89, 90*). The stretch reflex responds to changes in muscle length detected by muscle spindles. In our model, the reflex responded to changes in joint position, which can be derived from muscle length (*91, 92*). Our EP shifted around the desired position (*62*), thus enabling goal-directed movements and reproducing SC functionality (*91, 93*). In the EP hypothesis, the CNS specifies the threshold of the spinal stretch reflex (*61*), found to be modulated by descending signals during voluntary movements (*94-96*). In our model, the reflex was modulated by the cocontraction level, which has been modeled together with the EP hypothesis to compensate for loads during arm movements (*62*).

The muscle elastic component drove each joint toward its resting position. The arm resting position was defined as the joint configuration at which, once settled, zero torque did not induce any robot motion (see fig. s12 for an illustration of the robot resting position). From Eq. 1, the elastic component was characterized by:

$$\beta_j(A_{F,j}+A_{E,j}+2c_j+\gamma_j)(\theta_{r,j}-\theta_{a,j}) \tag{4}$$

Finally, the viscous component, was determined by the joint velocity and the muscle damping coefficient. From Eq. 1, it was characterized by:

$$\delta_j\theta'_{a,j} \tag{5}$$

Further analysis of the muscle components is illustrated in fig. S10, highlighting the contribution of each muscle component through ablation experiments. Ablation of the SC-reflex component deteriorates performance accuracy and facilitates output torque convergence, as the controller lacks the reflex fast corrective term, which helps improve accuracy but can cause early-stage oscillations. Ablation of the elastic component slightly improves performance accuracy, as the controller no longer needs to compensate for the elasticity pulling the arm to its resting position. Ablation of the viscous component leads to oscillations and deteriorates performance accuracy due to the lack of damping response.

Active modulation of cocontraction

Active modulation of the cocontraction level was joint-specific, it depended on deviations from the desired joint trajectory, and it was based on a weighted trade-off between compliance and accuracy. Cocontraction was dynamically adjusted as follows:

$$c_j(t)=20(W_{C-R,j})^3\cdot\frac{|\theta_{d,j}(t)-\theta_{a,j}(t)|}{e^{-2W_{C-R,j}}} \tag{6}$$

where $j \in [1, 6]$ stands for the joint index; c_j stands for the joint cocontraction; $W_{C\text{-}R,j} \in [0.0, 1.0]$ stands for the joint compliance-robustness weighted priority, with $W_{C\text{-}R,j} = 0$ defining the compliance priority mode, and $W_{C\text{-}R,j} = 1$ defining the robustness priority mode; $\theta_{d,j}$ and $\theta_{a,j}$ are the desired and actual joint position, respectively. The resulting cocontraction value was constrained to the range [0.0, 10.0], and then normalized within the range [0.0, 1.0] before entering the muscle model equation (cocontraction ranging from *x0* to *x10*).

The cerebellar model

The cerebellar SNN, built upon previous work (*9, 10, 56*), is derived from the Marr-Albus-Ito cerebellar theory (*50-52*), which established the foundation for supervised motor learning in the cerebellum (*53*). The cerebellar network comprised 16374 neurons distributed across five neural layers, that were divided into six microcomplexes each (*97*). These microcomplexes processed the sensorimotor data related to each of the six DOFs of the robot. The intricate cerebellar network orchestrated, via neural adaptation, the translation of sensorimotor information into coordinated muscle activation commands for the robot movement.

The implemented cerebellar neural layers were the following: mossy fibers (MFs), granule cells (GCs), Purkinje cells (PCs), climbing fibers (CFs), and deep cerebellar nuclei (DCN) (refer to Supplementary Materials for the cerebellar neural network topology). The MFs

(168 neurons) encoded the robot sensory information (desired and actual state of the joints) as afferent projections to the cerebellum, which formed excitatory synapses with the GCs. The GCs (14406 neurons) univocally recoded the received sensory input (*98*), which was then transmitted through excitatory connections to the PCs (600 neurons). The PC layer also received excitatory inputs from the CFs (600 neurons), which conveyed an instructive signal per joint that represented the mismatch between the desired and actual joint state. The PC layer, in turn, projected inhibitory connections to the DCN layer (600 neurons), which also received excitatory synapses from MFs and CFs. Finally, the DCN activity provided the cerebellar efferent projections, delivering the agonist-antagonist activation commands (A_F and A_E) to the muscle model for flexion-extension control. To differentiate between flexion and extension control, the DCN layer maintained separate agonist-antagonist subpopulations, thus allowing each microcomplex to generate the two output signals for the corresponding joint: the agonist DCN subpopulation generated the flexion activation command (A_F); the antagonist subpopulation generated the extension activation command (A_E).

Cerebellar adaptation was facilitated via a spike-timing-dependent plasticity (STDP) mechanism, which correlated input sensory information with the instructive signal, and adjusted accordingly the synaptic weights in the GC–PC connections. This STDP mechanism was ruled by the balance between two opposed processes: long-term potentiation (LTP) and long-term depression (LTD). LTP induced a fixed synaptic weight increment for every spike that reached a PC through a GC, as follows:

$$\text{LTP } \Delta w_{GC_j - PC_i}(t) = \alpha \cdot \delta_{GC}(t) \cdot dt \tag{7}$$

where $\Delta w_{\text{GCj-PCi}}$ denotes the synaptic weight change between the j^{th} GC and i^{th} PC; $\alpha = 0.002$ nS is the synaptic weight increment; and δ_{GC} is the Dirac delta function of a GC spike. The LTD process, which depressed the GC–PC synaptic weights, correlated the GC and CF activity as follows:

$$\text{LTD } \Delta w_{GC_j - PC_i}(t) = \beta \cdot \int_{-\infty}^{t_{CF}} k(t - t_{CF}) \cdot \delta_{GC}(t) \cdot dt \tag{8}$$

$$k(x) = \begin{cases} \dfrac{-(x + d_k)}{\tau_{\text{LTD}} - d_k} \cdot e^{\frac{x + d_k}{\tau_{\text{LTD}} - d_k} + 1} & \text{if } x < -d_k \\ 0 & \text{if } x \geq -d_k \end{cases} \tag{9}$$

where $\beta = -0.0008$ nS denotes the synaptic weight decrement; t_{CF} is the arrival time of a CF spike; $k(\text{x})$ defines an integrative kernel where $d_{\text{k}} = 120$ ms established the kernel width; and $\tau_{\text{LTD}} = 150$ ms is the kernel eligibility trace peak, which establishes that the synaptic weight decrement is maximum for the GC spikes received τ_{LTD} ms before the arrival of a CF spike (for $\text{x} = \tau_{\text{LTD}}$; $k(\text{x}) = 1$). This interplay between LTP and LTD temporally correlated the instructive signal (CF activity) with the previous sensory state (GC activity).

The regulation of the GC–PC synaptic distribution, in turn, modulated the cerebellar efferent projections; the inhibitory action of PCs on DCN neurons shaped the cerebellar output commands. The cerebellar output commands arose from DCN activity: each DCN microcomplex generated the joint agonist-antagonist pair of output activation commands (A_F and A_E). These commands were computed from the DCN activity as follows:

$$\mathrm{DCN}_{j,i}(t) = \int_{t-T_{step}}^{t} \delta_{\mathrm{DCN}\,j,i}(t) \cdot dt \tag{10}$$

$$\begin{aligned}\mathrm{DCN}_{AG,j}(t) &= \frac{1}{M_j} \cdot \sum_{i=1}^{i=50} [\alpha_j \cdot \mathrm{DCN}_{j,i}(t)] \\ \mathrm{DCN}_{ANT,j}(t) &= \frac{1}{M_j} \cdot \sum_{i=51}^{i=100} [\alpha_j \cdot \mathrm{DCN}_{j,i}(t)]\end{aligned} \tag{11}$$

$$\begin{aligned}A_{F,j}(t) &= \frac{1}{21} \cdot \sum_{x=0}^{20} DCN_{AG,j}(t - x \cdot Tstep) \\ A_{E,j}(t) &= \frac{1}{21} \cdot \sum_{x=0}^{20} DCN_{ANT,j}(t - x \cdot Tstep)\end{aligned} \tag{12}$$

where $j \in [1, 6]$ stands for the joint index; i is the DCN neuron index within the microcomplex, which was halved in agonist ($i \in [1, 50]$) and antagonist subpopulations ($i \in [51, 100]$); T_{step} is the 2 ms duration of a time step; δ_{DCN} is the Dirac delta function of a DCN spike; DCN_{AG} and DCN_{ANT} are the agonist and antagonist output of the microcomplex; $\alpha_j = [0.75, 1.0, 0.375, 0.5, 0.05, 0.05]$ defines an output gain to weigh each joint relative position and mass; and $M_j = [15.0, 25.0, 9.0, 12.5, 1.8, 1.0]$ is a normalization factor used to normalize DCN_{AG} and DCN_{ANT} within the range [0, 1] (DCN_{AG} and DCN_{ANT} saturated to 1.0). The DCN agonist and antagonist output were then filtered (mean filter of size 21), providing A_F and A_E per joint to command the muscle model.

Please refer to Supplementary Materials for a comprehensive understanding of the implemented leaky integrate-and-fire (LIF) neuron model (*99*), the translation process from the robot analog domain to spiking neural activity, and the computation of the cerebellar instructive signal. Further details can also be found in (*9, 10, 56*), from which this model was built upon.

The robot

We used the Baxter robot (*49*) as the front-end body of our cerebello-muscular control loop. Baxter is a collaborative robot equipped with two arms, series elastic actuators, and allows for torque control (*100*). Our benchmark involved six DOF of Baxter's right arm (fig. S12). Baxter is known for its accuracy limitations, which are further highlighted when working at high speed (*101, 102*). We used the factory-default position controller as a baseline to define appropriate performance for the robot in use. The factory-default position controller is the position control method provided by the manufacturer. In position control mode, a position command (desired position coordinate for each joint) is directly sent to the robot, which internally handles the application of the desired position through the factory-default position controller. Parametrization of this internal controller is not provided by the manufacturer. For a computational complexity analysis, please refer to Supplementary Materials.

The motor control benchmark

The motor control benchmark aimed to capture the complex dynamics of the six DOF robot arm, including nonlinearities and joint interaction forces. We designed a set of sinusoidal-like and bell-shaped joint trajectories (position and velocity coordinates) performed at different speeds (*103*, *104*). This set of trajectories provided three different desired paths for the end-effector to follow: a circular trajectory performed in the vertical plane; a figure eight trajectory performed in the horizontal plane; a T-shaped trajectory performed in the horizontal plane. Please refer to the Supplementary Materials for detailed descriptions of the joint coordinates corresponding to each motor task. To evaluate the response to external perturbations, we attached a 0.5 and 1.0 kg payload to the end-effector during the circular and figure eight trajectory, respectively. To test cocontraction across different terrains and to assess the active modulation of cocontraction, we submerged the end-effector in fluids of different viscosity while performing the T-shaped trajectory. The terrains consisted of an aqueous solution of XG at different concentrations. The rheological properties of XG allowed us to cover a wide viscosity range (*71*).

The fluid viscosity was measured using a Haake MARS III controlled-stress rheometer (Thermo Fisher Scientific, Waltham, MA, USA) equipped with a concentric cylinder geometry. For each terrain, three repetitions of the viscosity measurement were taken to ensure the accuracy and reliability of the results (fig. S11 and table s9).

Performance accuracy metric

The performance accuracy was determined by the difference between the desired and actual trajectory. At each time step we measured the difference between desired and actual joint position (θ_d and θ_a), and computed the average difference through the duration of the motor task. The accuracy of a motor task trial, measured as the position MAE, was given by the joint average error, as follows:

$$\text{MAE}_j = \frac{T_{step}}{T} \sum_{t=0s}^{T} |\, \theta_{d,j}(t) - \theta_{a,j}(t) \,| \tag{13}$$

$$\text{MAE} = \frac{1}{N} \sum_{j=1}^{N} \text{MAE}_j \tag{14}$$

where T_{step} is the 2 ms time step; T denotes the motor task duration; $j \in [1, 6]$ is the joint index; and N = 6 is the total number of joints. Our final metric was defined by the mean and standard deviation (SD) of the position MAE observed over the final 100 trials of the motor adaptation process.

Torque time-integral metric

To quantify the amount of joint torque used during the execution of the motor tasks, we calculated the joint torque time-integral (TTI) (*64-67*) throughout the motor task duration. The TTI for each trial of the motor task was calculated as follows:

$$\text{TTI} = \int_{t=0s}^{T} \tau_j(t) \cdot dt \tag{15}$$

where τ_j defines the joint torque commanded to the robot at each time step. The final TTI metric was determined by the mean and SD over the final 100 trials of the motor adaptation process. This method was also used to compute the time-integral measurements for the different muscle components.

Robot physical agency metric

The assessment of robot physical agency was based on the different behaviors resulting from the varying level of cocontraction. These differences were observed through the robot response when attaching a payload to the end-effector (0.5 kg for the circular trajectory, 1 kg for the figure eight trajectory). We measured the difference in robot performance before and after attaching the payload for each cocontraction value. This difference was defined as the payload-induced deviation, Δ_{MAE}, calculated as follows:

$$\Delta_{\text{MAE}} = \text{MAE}_{post-payload} - \text{MAE}_{pre-payload} \tag{16}$$

where $MAE_{post\text{-}payload}$ and $MAE_{pre\text{-}payload}$ were given by the mean position MAE obtained from 100 trials performed with and without the payload, respectively. Based on the deviations induced by the payload in each motor task scenario, we defined our robot physical agency metric. A hypothetical deviation of 0.0 rad (Δ_{MAEmin}) would indicate the most robust robot response to the perturbation, thus correlating to the highest robot physical agency (RPA = 1.0). Conversely, the greatest deviation (Δ_{MAEmax}) found experimentally within the cocontraction range indicated the softest robot reaction to the perturbation, thus correlating to the lowest robot physical agency (RPA = 0.0). We obtained a normalized RPA value from the deviation allowed by each cocontraction value, Δ_{MAE}, as follows:

$$\text{RPA} = 1.0 - \frac{\Delta_{\text{MAE}}}{\Delta_{\text{MAE}\max}} \tag{17}$$

PD control of the muscle dynamics

To contextualize the complementarity of the cerebellum and muscle dynamics, and the spectrum of motor behavior enabled by our cerebello-muscular controller, we also tested the muscle dynamics commanded by a PD controller. The implemented muscle dynamics differentiate between flexion and extension activation commands (A_F and A_E), which could be provided by the cerebellum thanks to the division of the DCN layer into agonist and antagonist zones. Since a PD controller lacks this division, the muscle model under PD control had to be modified as follows:

$$\tau_j = \alpha_j (PD_{output,j}) + R_{SC,j} - \beta_j (2c + \gamma_j)(\theta_{r,j} - \theta_{a,j}) + \delta_j \theta'_{a,j} \tag{18}$$

$$PD_{output,j} = k_{p,j}(\theta_{d,j} - \theta_{a,j}) + k_{d,j}(\theta'_{d,j} - \theta'_{a,j}) \tag{19}$$

where $k_{p,j}$ and $k_{d,j}$ are the proportional and derivative gain of each joint, respectively; and the rest of the muscle components remain the same as for the cerebello-muscular controller. Please see table s5 for the PD gains for each trajectory. When testing the reaction to the 0.5 kg payload while performing the circular trajectory, we also tested a PD with higher gains governing the muscle dynamics (fig. S5). Please see table s6 for the higher PD gains for the circular trajectory. To further contextualize our solution, we also tested direct PD control without muscle dynamics (fig. S5) using three different PD controllers (v1, v2, v3) performing the circular trajectory. Please see table s7 for the PD gains of the three different direct PD controllers.

Evaluation of the cerebellar learned solutions

To evaluate the cerebellar learned solutions, we measured the complexity of the GC–PC synaptic weight distribution acquired after motor adaptation for each cocontraction value. The 14406 GC neurons innervated the 600 PC neurons in an all-to-one fashion; each PC received an excitatory synapse from every GC. Therefore, the synaptic weight distribution of the GC–PC layer was represented by a matrix ($W_{GC\text{-}PC}$) of size I x J, with I=14406 and J=600 for the total number of GC and PC neurons, respectively. Each cell in the matrix stored the synaptic weight (w) between GC_i and PC_j, as follows:

$$W_{GC-PC} = \begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,J} \\ w_{2,1} & w_{2,2} & \dots & w_{2,J} \\ \dots & \dots & \dots & \dots \\ w_{I,1} & w_{I,2} & \dots & w_{I,J} \end{bmatrix} \tag{20}$$

Once obtained the synaptic weight distribution matrix for each cocontraction value, we applied Shannon's entropy using a Python library (*105*). The entropy value provided a quantifiable metric of the complexity found in each GC–PC synaptic weight distribution (*56*). Higher entropy values indicated greater complexity within the obtained cerebellar solution.

Statistical analysis

Statistical significance presented in Figs. 2G, H; 3; 4B, D, F, H, J; and 7C, D was obtained by comparing the samples of consecutive cocontraction values using Welch's t-test (*106*) implementation from SciPy open-source software (function *scipy.stats.ttest_ind*) (*107*). The figures display the *p-values* (*) below the threshold indicated in the figure legend, except for Fig. 7C, D, in which the *p-values* greater than the threshold are displayed for visual simplicity of the figure. The boxplots in Fig. 7C cover from the first to the third quartile, with a line drawn at the median (cocontraction from all joints was considered); whiskers extend from the box limits to the farthest data point within 1.5 times the interquartile range (IQR).

Supplementary Materials

Supplementary Methods
Supplementary Results
Figs. S1 to S12
Tables S1 to S9
Movies S1 and S2

References and Notes

1. G.-Z. Yang, J. Bellingham, P. E. Dupont, P. Fischer, L. Floridi, R. Full, N. Jacobstein, V. Kumar, M. McNutt, R. Merrifield, The grand challenges of science robotics. *Science Robotics* **3**, eaar7650 (2018).
2. M. Kennedy III, The role of collaborative robotics in assistive and rehabilitation applications. *Science Robotics* **8**, eadk6743 (2023).

3. V. Ortenzi, A. Cosgun, T. Pardi, W. P. Chan, E. Croft, D. Kulić, Object handovers: a review for robotics. *IEEE Transactions on Robotics* **37**, 1855-1873 (2021).
4. R. Pfeifer, M. Lungarella, F. Iida, Self-organization, embodiment, and biologically inspired robotics. *Science* **318**, 1088-1093 (2007).
5. D. Floreano, A. J. Ijspeert, S. Schaal, Robotics and neuroscience. *Current Biology* **24**, R910-R920 (2014).
6. M. Chahine, R. Hasani, P. Kao, A. Ray, R. Shubert, M. Lechner, A. Amini, D. Rus, Robust flight navigation out of distribution with liquid neural networks. *Science Robotics* **8**, eadc8892 (2023).
7. J. Dupeyroux, J. R. Serres, S. Viollet, AntBot: A six-legged walking robot able to home like desert ants in outdoor environments. *Science Robotics* **4**, eaau0307 (2019).
8. S. Ma, J. Pei, W. Zhang, G. Wang, D. Feng, F. Yu, C. Song, H. Qu, C. Ma, M. Lu, Neuromorphic computing chip with spatiotemporal elasticity for multi-intelligent-tasking robots. *Science Robotics* **7**, eabk2948 (2022).
9. I. Abadía, F. Naveros, J. A. Garrido, E. Ros, N. R. Luque, On robot compliance: A cerebellar control approach. *IEEE transactions on cybernetics* **51**, 2476-2489 (2019).
10. I. Abadía, F. Naveros, E. Ros, R. R. Carrillo, N. R. Luque, A cerebellar-based solution to the nondeterministic time delay problem in robotic control. *Science Robotics* **6**, eabf2756 (2021).
11. R. Pfeifer, J. Bongard, *How the body shapes the way we think: a new view of intelligence*. (MIT press, 2006).
12. E. Tytell, P. Holmes, A. H. Cohen, Spikes alone do not behavior make: why neuroscience needs biomechanics. *Current opinion in neurobiology* **21**, 816-822 (2011).
13. S. Collins, A. Ruina, R. Tedrake, M. Wisse, Efficient bipedal robots based on passive-dynamic walkers. *Science* **307**, 1082-1085 (2005).
14. A. J. Ijspeert, Biorobotics: Using robots to emulate and investigate agile locomotion. *Science* **346**, 196-203 (2014).
15. S. Wolf, G. Grioli, O. Eiberger, W. Friedl, M. Grebenstein, H. Höppner, E. Burdet, D. G. Caldwell, R. Carloni, M. G. Catalano, Variable stiffness actuators: Review on design and components. *IEEE/ASME transactions on mechatronics* **21**, 2418-2430 (2015).
16. Y. Wang, X. Yang, Y. Chen, D. K. Wainwright, C. P. Kenaley, Z. Gong, Z. Liu, H. Liu, J. Guan, T. Wang, A biorobotic adhesive disc for underwater hitchhiking inspired by the remora suckerfish. *Science Robotics* **2**, eaan8072 (2017).
17. C. Laschi, B. Mazzolai, M. Cianchetti, Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Science Robotics* **1**, eaah3690 (2016).
18. A. R. Sobinov, S. J. Bensmaia, The neural mechanisms of manual dexterity. *Nature Reviews Neuroscience* **22**, 741-757 (2021).
19. M. W. Marzke, Upper-limb evolution and development. *JBJS* **91**, 26-30 (2009).
20. R. Gopura, K. Kiguchi, E. Horikawa, A study on human upper-limb muscles activities during daily upper-limb motions. *International Journal of Bioelectromagnetism* **12**, 54-61 (2010).
21. D. C. Taylor, J. D. Dalton JR, A. V. Seaber, W. E. Garrett JR, Viscoelastic properties of muscle-tendon units: the biomechanical effects of stretching. *The American journal of sports medicine* **18**, 300-309 (1990).
22. H. Wagner, R. Blickhan, Stabilizing function of antagonistic neuromusculoskeletal systems: an analytical investigation. *Biological cybernetics* **89**, 71-79 (2003).
23. R. Blickhan, A. Seyfarth, H. Geyer, S. Grimmer, H. Wagner, M. Günther, Intelligence by mechanics. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **365**, 199-220 (2007).
24. I. E. Brown, G. E. Loeb, A reductionist approach to creating and using neuromusculoskeletal models in *Biomechanics and neural control of posture and movement*. (Springer, 2000), pp. 148-163.
25. H. Hauser, A. J. Ijspeert, R. M. Füchslin, R. Pfeifer, W. Maass, Towards a theoretical foundation for morphological computation with compliant bodies. *Biological cybernetics* **105**, 355-370 (2011).
26. D. Haeufle, M. Günther, G. Wunner, S. Schmitt, Quantifying control effort of biological and technical movements: an information-entropy-based approach. *Physical Review E* **89**, 012716 (2014).
27. D. F. Haeufle, I. Wochner, D. Holzmüller, D. Driess, M. Günther, S. Schmitt, Muscles reduce neuronal information load: quantification of control effort in biological vs. robotic pointing and walking. *Frontiers in Robotics and AI* **7**, 511258 (2020).
28. P. L. Gribble, L. I. Mullin, N. Cothros, A. Mattar, Role of cocontraction in arm movement accuracy. *Journal of neurophysiology* **89**, 2396-2405 (2003).
29. R. Osu, N. Kamimura, H. Iwasaki, E. Nakano, C. M. Harris, Y. Wada, M. Kawato, Optimal impedance control for task achievement in the presence of signal-dependent noise. *Journal of Neurophysiology* **92**, 1199-1215 (2004).
30. L. P. Selen, P. J. Beek, J. H. Van Dieën, Impedance is modulated to meet accuracy demands during goal-directed arm movements. *Experimental Brain Research* **172**, 129-138 (2006).

31. W. Liang, H. Liu, K. Wang, Z. Qian, L. Ren, L. Ren, Comparative study of robotic artificial actuators and biological muscle. *Advances in Mechanical Engineering* **12**, 1687814020933409 (2020).
32. Y. Morimoto, H. Onoe, S. Takeuchi, Biohybrid robot powered by an antagonistic pair of skeletal muscle tissues. *Science Robotics* **3**, eaat4440 (2018).
33. X. Ji, X. Liu, V. Cacucciolo, M. Imboden, Y. Civet, A. El Haitami, S. Cantin, Y. Perriard, H. Shea, An autonomous untethered fast soft robotic insect driven by low-voltage dielectric elastomer actuators. *Science Robotics* **4**, eaaz6451 (2019).
34. D. Büchler, R. Calandra, J. Peters, Learning to control highly accelerated ballistic movements on muscular robots. *Robotics and Autonomous Systems* **159**, 104230 (2023).
35. Y. Mengüç, N. Correll, R. Kramer, J. Paik, Will robots be bodies with brains or brains with bodies? *Science Robotics* **2**, eaar4527 (2017).
36. R. Raman, R. Bashir, Biomimicry, biofabrication, and biohybrid systems: The emergence and evolution of biological design. *Advanced healthcare materials* **6**, 1700496 (2017).
37. H. Chaoui, P. Sicard, W. Gueaieb, ANN-based adaptive control of robotic manipulators with friction and joint elasticity. *IEEE Transactions on Industrial Electronics* **56**, 3174-3187 (2009).
38. N. Hogan, Adaptive control of mechanical impedance by coactivation of antagonist muscles. *IEEE Transactions on automatic control* **29**, 681-690 (1984).
39. N. Hogan, Impedance control: An approach to manipulation in *1984 American control conference*. (IEEE, 1984), pp. 304-313.
40. N. Hogan, Impedance control: An approach to manipulation: Part II—Implementation. (1985).
41. P. Song, Y. Yu, X. Zhang, A tutorial survey and comparison of impedance control on robotic manipulation. *Robotica* **37**, 801-836 (2019).
42. F. J. Abu-Dakka, M. Saveriano, Variable impedance control and learning—a review. *Frontiers in Robotics and AI* **7**, 590681 (2020).
43. S. Jung, T. C. Hsia, R. G. Bonitz, Force tracking impedance control of robot manipulators under unknown environment. *IEEE Transactions on Control Systems Technology* **12**, 474-483 (2004).
44. N. Hogan, Stable execution of contact tasks using impedance control in *Proceedings. 1987 IEEE International Conference on Robotics and Automation*. (IEEE, 1987), vol. 4, pp. 1047-1054.
45. J. F. Soechting, M. Flanders, Sensorimotor control of contact force. *Current opinion in neurobiology* **18**, 565-572 (2008).
46. Ö. Ekeberg, A combined neuronal and mechanical model of fish swimming. *Biological cybernetics* **69**, 363-374 (1993).
47. A. J. Ijspeert, J. Hallam, D. Willshaw, From lampreys to salamanders: evolving neural controllers for swimming and walking in *From Animals to Animats, Proceedings of the Fifth International Conference of The Society for Adaptive Behavior (SAB98)*. (MIT Press, 1998), pp. 390-399.
48. R. Thandiackal, K. Melo, L. Paez, J. Herault, T. Kano, K. Akiyama, F. Boyer, D. Ryczko, A. Ishiguro, A. J. Ijspeert, Emergence of robust self-organized undulatory swimming based on local hydrodynamic force sensing. *Science Robotics* **6**, eabf6354 (2021).
49. C. Fitzgerald, Developing baxter in *2013 IEEE conference on technologies for practical robot applications (TePRA)*. (IEEE, 2013), pp. 1-6.
50. D. Marr, W. T. Thach, A theory of cerebellar cortex. *From the Retina to the Neocortex: Selected Papers of David Marr*, 11-50 (1991).
51. J. S. Albus, A theory of cerebellar function. *Mathematical biosciences* **10**, 25-61 (1971).
52. M. Ito, Neurophysiological aspects of the cerebellar motor control system. *International journal of neurology* **7**, 162-176 (1970).
53. J. L. Raymond, J. F. Medina, Computational principles of supervised learning in the cerebellum. *Annual review of neuroscience* **41**, 233-253 (2018).
54. J. F. Medina, Teaching the cerebellum about reward. *Nature neuroscience* **22**, 846-848 (2019).
55. J. Knüsel, A. Crespi, J.-M. Cabelguen, A. J. Ijspeert, D. Ryczko, Reproducing five motor behaviors in a salamander robot with virtual muscles and a distributed CPG controller regulated by drive signals and proprioceptive feedback. *Frontiers in neurorobotics* **14**, 604426 (2020).
56. A. Bruel, I. Abadía, T. Collin, I. Sakr, H. Lorach, N. R. Luque, E. Ros, A. Ijspeert, The spinal cord facilitates cerebellar upper limb motor learning and control; inputs from neuromusculoskeletal simulation. *PLOS Computational Biology* **20**, e1011008 (2024).
57. W. Thach, Correlation of neural discharge with pattern and force of muscular activity, joint position, and direction of intended next movement in motor cortex and cerebellum. *Journal of neurophysiology* **41**, 654-676 (1978).
58. S. H. Scott, J. F. Kalaska, Reaching movements with similar hand paths but different arm orientations. I. Activity of individual cells in motor cortex. *Journal of neurophysiology* **77**, 826-852 (1997).

59. D. Humphrey, Separate cell systems in the motor cortex of the monkey for the control of joint movement and of joint stiffness. *Electroencephalography and Clinical neurophysiology. Supplement* **36**, 393-408 (1982).

60. D. Humphrey, Separate cortical systems for the control of joint movement and joint stiffness: Reciprocal activation and coactivation of antagonist muscles. *Advances in neurology* **39**, 347-372 (1983).

61. M. L. Latash, Muscle coactivation: definitions, mechanisms, and functions. *Journal of neurophysiology* **120**, 88-104 (2018).

62. P. L. Gribble, D. J. Ostry, Compensation for loads during arm movements using equilibrium-point control. *Experimental Brain Research* **135**, 474-482 (2000).

63. C. Disselhorst-Klug, T. Schmitz-Rode, G. Rau, Surface electromyography and muscle force: Limits in sEMG–force relationship and new approaches for applications. *Clinical biomechanics* **24**, 225-235 (2009).

64. B. Beihoff, A survey of torque transduction methodologies for industrial applications in *Conference Record of 1996 Annual Pulp and Paper Industry Technical Conference*. (IEEE, 1996), pp. 220-229.

65. A. J. Del-Ama, Á. Gil-Agudo, J. L. Pons, J. C. Moreno, Hybrid FES-robot cooperative control of ambulatory gait rehabilitation exoskeleton. *Journal of neuroengineering and rehabilitation* **11**, 1-15 (2014).

66. J. A. Mettler, D. M. Magee, B. M. Doucet, Low-frequency electrical stimulation with variable intensity preserves torque. *Journal of Electromyography and Kinesiology* **42**, 49-56 (2018).

67. N. Royer, K. Nosaka, V. Doguet, M. Jubeau, Neuromuscular responses to isometric, concentric and eccentric contractions of the knee extensors at the same torque-time integral. *European Journal of Applied Physiology*, 1-13 (2022).

68. D. W. Franklin, R. Osu, E. Burdet, M. Kawato, T. E. Milner, Adaptation to stable and unstable dynamics achieved by combined impedance control and inverse dynamics model. *Journal of neurophysiology* **90**, 3270-3282 (2003).

69. R. Osu, D. W. Franklin, H. Kato, H. Gomi, K. Domen, T. Yoshioka, M. Kawato, Short-and long-term changes in joint co-contraction associated with motor learning as revealed from surface EMG. *Journal of neurophysiology* **88**, 991-1004 (2002).

70. J. B. Heald, D. W. Franklin, D. M. Wolpert, Increasing muscle co-contraction speeds up internal model acquisition during dynamic motor learning. *Scientific reports* **8**, 16355 (2018).

71. N. B. Wyatt, M. W. Liberatore, Rheology and viscosity scaling of the polyelectrolyte xanthan gum. *Journal of Applied Polymer Science* **114**, 4076-4084 (2009).

72. A. De Santis, B. Siciliano, A. De Luca, A. Bicchi, An atlas of physical human–robot interaction. *Mechanism and Machine Theory* **43**, 253-270 (2008).

73. T. E. Milner, Adaptation to destabilizing dynamics by means of muscle cocontraction. *Experimental brain research* **143**, 406-416 (2002).

74. C. Ott, R. Mukherjee, Y. Nakamura, A hybrid system framework for unified impedance and admittance control. *Journal of Intelligent & Robotic Systems* **78**, 359-375 (2015).

75. N. Berezny, M. Ahmadi, Interpolating across the impedance/admittance spectrum with Unified Interaction Control. *Discover Mechanical Engineering* **2**, 16 (2023).

76. F. Cavenago, L. Voli, M. Massari, Adaptive hybrid system framework for unified impedance and admittance control. *Journal of Intelligent & Robotic Systems* **91**, 569-581 (2018).

77. D. A. Lawrence, Impedance control stability properties in common implementations in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*. (IEEE, 1988), pp. 1185-1190.

78. H. N. Rahimi, I. Howard, L. Cui, Neural impedance adaption for assistive human–robot interaction. *Neurocomputing* **290**, 50-59 (2018).

79. J. Buchli, F. Stulp, E. Theodorou, S. Schaal, Learning variable impedance control. *The International Journal of Robotics Research* **30**, 820-833 (2011).

80. K.-k. Lee, M. Buss, Force tracking impedance control with variable target stiffness. *IFAC Proceedings Volumes* **41**, 6751-6756 (2008).

81. L. Muratore, A. Laurenzi, N. G. Tsagarakis, A self-modulated impedance multimodal interaction framework for human-robot collaboration in *2019 International Conference on Robotics and Automation (ICRA)*. (IEEE, 2019), pp. 4998-5004.

82. A. Ajoudani, N. Tsagarakis, A. Bicchi, Tele-impedance: Teleoperation with impedance regulation using a body–machine interface. *The International Journal of Robotics Research* **31**, 1642-1656 (2012).

83. E. L. Miller, L. Murray, L. Richards, R. D. Zorowitz, T. Bakas, P. Clark, S. A. Billinger, Comprehensive overview of nursing and interdisciplinary rehabilitation care of the stroke patient: a scientific statement from the American Heart Association. *Stroke* **41**, 2402-2448 (2010).

84. S. Haddadin, E. Croft, Physical human–robot interaction. *Springer handbook of robotics*, 1835-1874 (2016).

85. D. P. Ferris, M. Louie, C. T. Farley, Running in the real world: adjusting leg stiffness for different surfaces. *Proceedings of the Royal Society of London. Series B: Biological Sciences* **265**, 989-994 (1998).

86. K. C. Galloway, J. E. Clark, M. Yim, D. E. Koditschek, Experimental investigations into the role of passive variable compliant legs for dynamic robotic locomotion in *2011 IEEE International Conference on Robotics and Automation*. (IEEE, 2011), pp. 1243-1249.

87. P. Ramdya, A. J. Ijspeert, The neuromechanics of animal locomotion: From biology to robotics and back. *Science Robotics* **8**, eadg0279 (2023).

88. M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, A. Y. Ng, ROS: an open-source Robot Operating System in *ICRA workshop on open source software*. (Kobe, Japan, 2009), vol. 3, pp. 5.

89. S. F. Giszter, F. A. Mussa-Ivaldi, E. Bizzi, Convergent force fields organized in the frog's spinal cord. *Journal of neuroscience* **13**, 467-491 (1993).

90. A. G. Feldman, Once more on the equilibrium-point hypothesis (λ model) for motor control. *Journal of motor behavior* **18**, 17-54 (1986).

91. D. A. Kistemaker, A. J. Van Soest, M. F. Bobbert, Is equilibrium point control feasible for fast goal-directed single-joint movements? *Journal of Neurophysiology* **95**, 2898-2912 (2006).

92. E. Burdet, K. P. Tee, I. Mareels, T. E. Milner, C.-M. Chew, D. W. Franklin, R. Osu, M. Kawato, Stability and motor adaptation in human arm movements. *Biological cybernetics* **94**, 20-32 (2006).

93. T. Buhrmann, E. A. Di Paolo, Spinal circuits can accommodate interaction torques during multijoint limb movements. *Frontiers in computational neuroscience* **8**, 144 (2014).

94. A. J. Fink, K. R. Croce, Z. J. Huang, L. Abbott, T. M. Jessell, E. Azim, Presynaptic inhibition of spinal sensory feedback ensures smooth movement. *Nature* **509**, 43-48 (2014).

95. D. J. Bennett, Stretch reflex responses in the human elbow joint during a voluntary movement. *The Journal of physiology* **474**, 339-351 (1994).

96. J. Shemmell, M. A. Krutky, E. J. Perreault, Stretch sensitive reflexes as an adaptive mechanism for maintaining limb stability. *Clinical Neurophysiology* **121**, 1680-1689 (2010).

97. M. Ito, Cerebellar microcomplexes. *International review of neurobiology* **41**, 475-487 (1997).

98. R. R. Carrillo, F. Naveros, E. Ros, N. R. Luque, A metric for evaluating neural input representation in supervised learning networks. *Frontiers in neuroscience* **12**, 913 (2018).

99. W. Gerstner, W. M. Kistler, *Spiking neuron models: Single neurons, populations, plasticity*. (Cambridge university press, 2002).

100. G. A. Pratt, M. M. Williamson, Series elastic actuators in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*. (IEEE, 1995), vol. 1, pp. 399-406.

101. S. Cremer, L. Mastromoro, D. O. Popa, On the performance of the Baxter research robot in *2016 IEEE international symposium on assembly and manufacturing (ISAM)*. (IEEE, 2016), pp. 106-111.

102. M. Alqatamin, N. Taghavi, S. K. Das, D. O. Popa, Observer-Free Output Feedback Tracking Control for Collaborative Robotics in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*. (IEEE, 2022), pp. 1017-1022.

103. P. van der Smagt, Benchmarking cerebellar control. *Robotics and Autonomous Systems* **32**, 237-251 (2000).

104. H. Hoffmann, G. Petkos, S. Bitzer, S. Vijayakumar, Sensor-assisted adaptive motor control under continuously varying context in *ICINCO-ICSO*. (2007), pp. 262-269.

105. S. Van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, scikit-image: image processing in Python. *PeerJ* **2**, e453 (2014).

106. B. L. Welch, The generalization of ‘STUDENT'S’problem when several different population varlances are involved. *Biometrika* **34**, 28-35 (1947).

107. P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature methods* **17**, 261-272 (2020).

108. E. Ros, R. Carrillo, E. M. Ortigosa, B. Barbour, R. Agís, Event-driven simulation scheme for spiking neural networks using lookup tables to characterize neuronal dynamics. *Neural computation* **18**, 2959-2993 (2006).

109. N. R. Luque, F. Naveros, R. R. Carrillo, E. Ros, A. Arleo, Spike burst-pause dynamics of Purkinje cells regulate sensorimotor adaptation. *PLoS computational biology* **15**, e1006298 (2019).

110. S. Kuroda, K. Yamamoto, H. Miyamoto, K. Doya, M. Kawato, Statistical characteristics of climbing fiber spikes necessary for efficient cerebellar learning. *Biological cybernetics* **84**, 183-192 (2001).

111. J. Keating, W. Thach, Nonclock behavior of inferior olive neurons: interspike interval of Purkinje cell complex spike discharge in the awake behaving monkey is random. *Journal of neurophysiology* **73**, 1329-1340 (1995).

112. N. R. Luque, J. A. Garrido, R. R. Carrillo, J.-M. C. Olivier, E. Ros, Cerebellarlike corrective model inference engine for manipulation tasks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **41**, 1299-1312 (2011).

113. C. Boucheny, R. Carrillo, E. Ros, O. J.-M. Coenen, Real-time spiking neural network: An adaptive cerebellar model in *International Work-Conference on Artificial Neural Networks*. (Springer, 2005), pp. 136-144.

Acknowledgments: We thank Dr. L. Paez for the valuable discussions regarding the muscle model. We thank Prof. M.T. López-López and Dr. F.J. Vázquez-Pérez for the help and assistance on viscosity measurements. **Funding:** This work was supported by MUSCLEBOT (CNS2022-135243) funded by MCIN/AEI/10.13039/501100011033/ and by NextGenerationEU/PRTR; TREMBLE-ICED (PID2023-146392NB-I00) funded by MCIN/AEI/10.13039/; DLROB (TED2021-131294B-I00) funded by MCIN/AEI/10.13039/501100011033/ and by NextGenerationEU/PRTR awarded to N.R.L.. SENSCOMP (PID2022-140095NB-I00) funded by MCIN/AEI/10.13039/501100011033/ and EU FEDER; INTARE (TED2021-131466B-I00) funded by MCIN/AEI/10.13039/501100011033/ and NextGenerationEU/PRTR awarded to E.R.. EU Human Brain Project Specific Grant Agreement 3 (H2020-RIA. 945539) awarded to A.I. and E.R.. **Author contributions:** I.A., A.B., A.I., E.R., and N.R.L. conceived the working hypothesis, the proposed approach and the experimental setup; I.A, A.B. and N.R.L. implemented the proposed approach; I.A. and N.R.L. conducted and extracted experimental results; A.I., E.R., and N.R.L. provided funding; G.C., A.I., E.R., and N.R.L. supervised this work; all authors contributed to the writing of the manuscript. **Competing interests:** The authors declare that they have no competing interests. **Data and materials availability:** All data needed to evaluate the conclusions in the paper are present in the paper or the Supplementary Materials. Dataset (https://doi.org/10.5281/zenodo.14283696) and source code (https://doi.org/10.5281/zenodo.14283645) available at Zenodo.

Figures:

Fig. 1. Neuromechanics robot control for adjustable motor behavior. (A) Different control approaches. (i) Encoder-based position control: uses position data from encoders and requires kinematic models; suitable for position tracking but limited for dynamic interactions. (ii) Torque transducer-based impedance control: uses torque transducers or contact force sensors to measure and control interaction force/torque assisted by dynamic models; effective for adaptive behavior but requires complex modeling and control. (iii) Neuromechanics encoder-based torque control: uses muscular and cerebellar mechanisms to control torque using position feedback; adjustable compliance and accuracy without kinematic or dynamic models (model-free), effective in unknown terrains. **(B)** Cerebello-muscular control loop. Motor task defined as desired joint coordinates (position θ_d, and velocity θ'_d) to be tracked. A module representing the motor cortex provides the desired coordinates and the cocontraction level (c). The cerebellar SNN receives through MFs the sensory input comprising desired and actual (θ_a, θ'_a) joint coordinates, and the instructive signal (ε) through the CFs. Activity from DCN layer delivers flexion-extension activation commands (A_F and A_E) to the muscle model, which also receives θ_d, c, θ_a and θ'_a as sensory input and generates joint torque commands (τ) to drive the robot. Control loop operating at 500 Hz. **(C)** Schematic of the muscle model. Active and passive components are integrated: the cerebellar A_F and A_E commands, and a spinal cord (SC) reflex, actively drive the robot behavior; passive mechanical viscoelasticity adds to the output torque. Muscle viscoelasticity schematic inspired from (*46*).

Fig. 2. Cocontraction effect on performance accuracy. (A) Position MAE evolution during cerebellar motor adaptation for the circular and **(B)** figure eight trajectories with the lowest and highest cocontraction values. **(C)** Tracking of desired joint position coordinates for the circular and **(D)** figure eight trajectories with the lowest and highest cocontraction values. Lines show the mean and shading shows the SD per time step of the last 100 trials of the motor adaptation (n = 100). **(E)** Cartesian space representation of the robot end-effector trajectory with the lowest and highest

cocontraction values for the circular and **(F)** figure eight trajectories. The density function over the last 100 trials of the motor adaptation process is shown. **(G)** Position MAE per trial, data shows the mean and error bars show the SD of the final 100 trials (n = 100) performed by the cerebello-muscular controller under each cocontraction value for the circular and **(H)** figure eight trajectories. Consecutive cocontraction values were compared using Welch's t-test. The performance of the factory-default position control is included for reference. Dashed lines in (G) and (H) added to cerebello-muscular data as guides to the eye.

Fig. 3. Cocontraction effect on performance accuracy at different motion speed. (A) Position MAE per trial for the cerebello-muscular controller under each cocontraction value performing the circular trajectory at different speeds: trajectory duration of 2.5, 2.0, 1.5, 1.0; and 0.8 s, and **(B)** figure eight trajectory at different speeds: trajectory duration of 2.5 and 2.0 s. Data shows the mean and error bars show the SD over the final 100 trials (n = 100) of the motor adaptation process. Consecutive cocontraction values were compared using Welch's t-test. Dashed lines added to cerebello-muscular data as guides to the eye.

Fig. 4. Muscle torque components. (A) Muscle torque profiles of the cerebello-muscular controller performing the circular trajectory with the lowest and highest cocontraction values. **(B)** Torque time-integral (TTI) per joint, across different cocontraction values, performing the circular and figure eight trajectories. The following muscle torque components are shown, together with their time-integral measurements: **(C)** active component at the lowest and highest cocontraction values for the circular trajectory; **(D)** time-integral of the active component for each cocontraction value, for the circular and figure eight trajectories; **(E)** reflex component at the lowest and highest cocontraction values for the circular trajectory; **(F)** time-integral of the reflex component for each cocontraction value, for the circular and figure eight trajectories; **(G)** passive elastic component at the lowest and highest cocontraction values for the circular trajectory; **(H)** time-integral of the passive elastic component for each cocontraction value, for the circular and figure eight trajectories; **(I)** passive viscous component at the lowest and highest cocontraction values for the circular trajectory; **(J)** time-integral of the passive viscous component for each cocontraction value, for the circular and figure eight trajectories. Data in (A), (C), (E), (G) and (I) shows the mean and shading shows the SD per time step along the trajectory duration, using the last 100 trials of the motor adaptation process (n = 100). Data in (B), (D), (F), (H) and (J) shows the mean and error bars show the SD per trial using the last 100 trials of the motor adaptation process (n = 100). Consecutive cocontraction values were compared using Welch's t-test.

Fig. 5. Cocontraction and response against external perturbations. (A) Position MAE before and after attachment of a 0.5 kg payload for the cerebello-muscular controller and PD-muscular controller at varying cocontraction values performing the 2.5 s circular trajectory. **(B)** Spectrum of robot behavior allowed by the cerebello-muscular and PD-muscular controllers, quantified as the robot physical agency (RPA). Dashed lines included as guides to the eye. **(C)** Position MAE evolution during the motor adaptation process of the cerebello-muscular controller after attaching the 0.5 kg payload while performing the circular trajectory. **(D)** Gradual reduction of the cocontraction level following the attachment of the 0.5 kg payload while performing the circular trajectory.

Fig. 6. Application of the cerebellar solution learned for a specific cocontraction value to other cocontraction values while performing the circular trajectory. (A)

Application of the cerebellar solution learned for cocontraction *x0* to higher cocontraction values. From trial 0 to 50 the cerebello-muscular controller performed with *c* = *x0*, then the cocontraction value was modified. **(B)** Application of the cerebellar solution learned for cocontraction *x5* to both lower and higher cocontraction values (excluding *c* = *x0* for safety reasons). From trial 0 to 50 the cerebello-muscular controller performed with *c* = *x5*, then the cocontraction value was modified. **(C)** Application of the cerebellar solution learned for cocontraction *x10* to lower cocontraction values (excluding *c* = *x1* and *c* = *x0* for safety reasons). From trial 0 to 50 the cerebello-muscular controller performed with *c* = *x10*, then the cocontraction value was modified. In (A), (B), and (C), cerebellar learning was disabled. **(D)** Entropy of the cerebellar GC–PC synaptic weights distribution (shown in black) after motor adaptation for each cocontraction value, correlated with the corresponding position MAE (shown in blue). MAE data (taken from Fig. 2G) shows the mean and error bars show the SD of the last 100 trials of the motor adaptation process (n = 100). **(E)** Linear regression of synaptic entropy measurements for those cocontraction values that showed a significant improvement in position MAE (statistical significance of position MAE taken from Fig. 2G). Dashed lines in (D) and (E) included as guides to the eye.

Fig. 7. Active modulation of cocontraction across unknown terrains. (A) Compliance priority and robustness priority modes performing in 2% concentration of xanthan gum (XG), and **(B)** 6% concentration of XG. (i) Position MAE as the robot end-effector is submerged in the viscous fluid. (ii) Cartesian path described by the submerged end-effector (density of 50 trials shown). The T-shaped trajectory follows the point sequence A, B, C, B, D, B, A. (iii) Mean cocontraction per trial as the end-effector is submerged in the viscous fluid. (iv) Joint cocontraction per trial with submerged end-effector (line shows the mean and shading the SD during the first 50 s, sampling every 2 ms). **(C)** Joint cocontraction across terrains in compliance priority and robustness priority modes. Data corresponds to the first 50 s with the end-effector submerged, sampling every 2 ms (cocontraction from all joints is considered). Boxplots cover from the first to the third quartile, line drawn at the median; whiskers extend from the box limits to the farthest data point within 1.5 times the interquartile range. **(D)** Position MAE in compliance priority and robustness priority modes across all terrains. Data shows mean and error bars show SD of 50 trials (n = 50) with the end-effector submerged. **(E)** Differentiated motor behaviors depending on the direction of motion in 2% XG, and **(F)** 6% XG. (i) Cartesian path described by the submerged end-effector (density of 50 trials shown). (ii) Joint cocontraction per trial with the end-effector submerged (line shows the mean and shading the SD during the first 50 s, sampling every 2 ms). Two motor behaviors were differentiated: frontal robustness and lateral compliance; lateral robustness and frontal compliance. **(G)** Distance covered by the end-effector in the frontal and lateral directions across terrains for both motor behaviors. Data shows mean and error bars the SD of distance covered per trial by the submerged end-effector (n = 50 trials). **(H)** End-effector submerged in 2% XG and 6% XG concentrations.

Movies:

Movie 1. Overview of the neuromechanics approach.

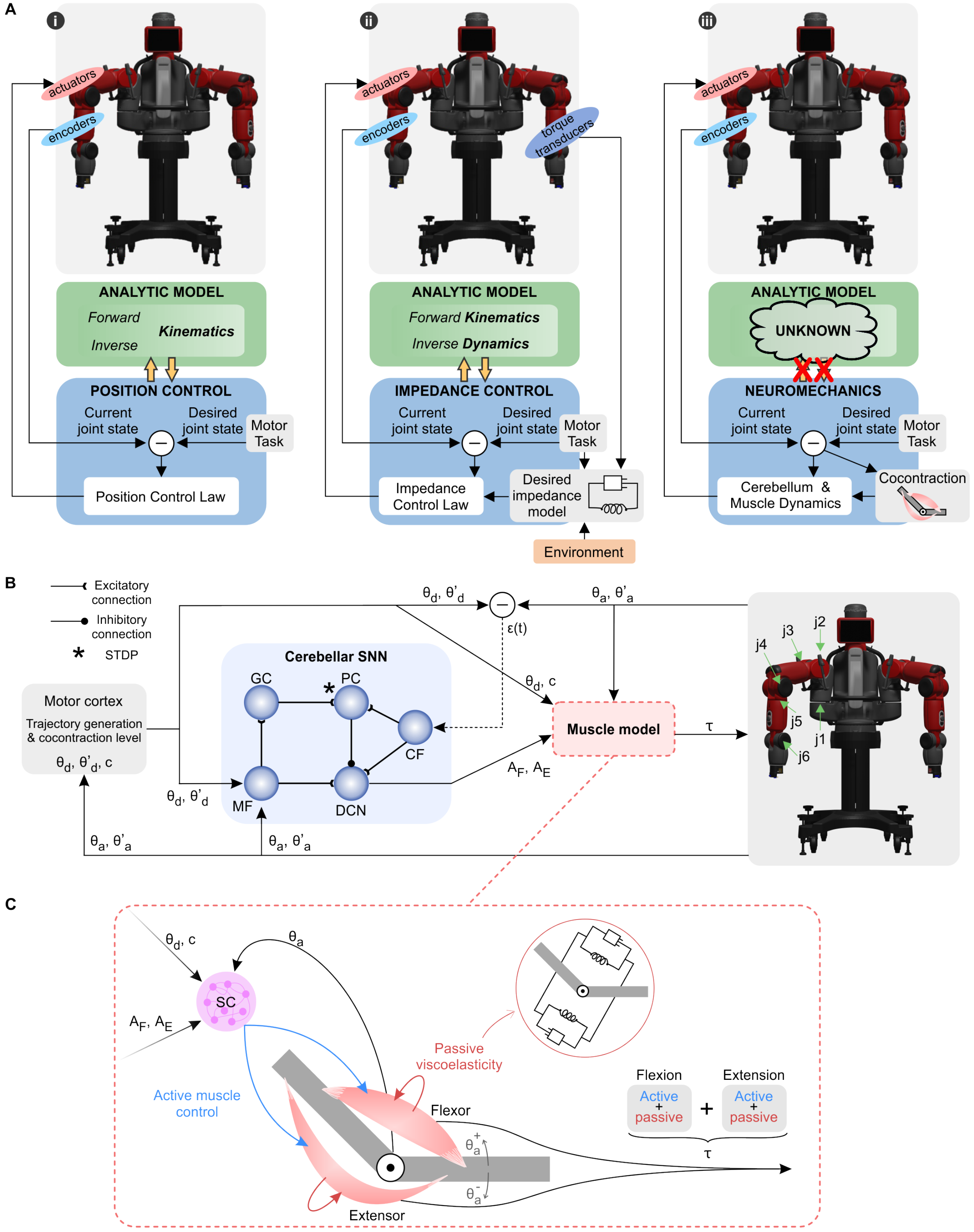

A
i
actuators
encoders
ANALYTIC MODEL
Forward
Inverse
Kinematics
POSITION CONTROL
Current joint state
Desired joint state
Motor Task
Position Control Law
ii
actuators
encoders
torque transducers
ANALYTIC MODEL
Forward Kinematics
Inverse Dynamics
IMPEDANCE CONTROL
Current joint state
Desired joint state
Motor Task
Impedance Control Law
Desired impedance model
Environment
iii
actuators
encoders
ANALYTIC MODEL
UNKNOWN
NEUROMECHANICS
Current joint state
Desired joint state
Motor Task
Cerebellum & Muscle Dynamics
Cocontraction
B
Excitatory connection
Inhibitory connection
* STDP
Motor cortex
Trajectory generation & cocontraction level
θd, θ'd, c
Cerebellar SNN
GC
PC
CF
MF
DCN
θd, θ'd
θa, θ'a
ε(t)
θd, c
Muscle model
AF, AE
τ
j1
j2
j3
j4
j5
j6
C
θd, c
θa
SC
AF, AE
Active muscle control
Passive viscoelasticity
Flexor
Extensor
θa+
θa-
Flexion
Extension
Active + passive
Active + passive
τ

Figure 1

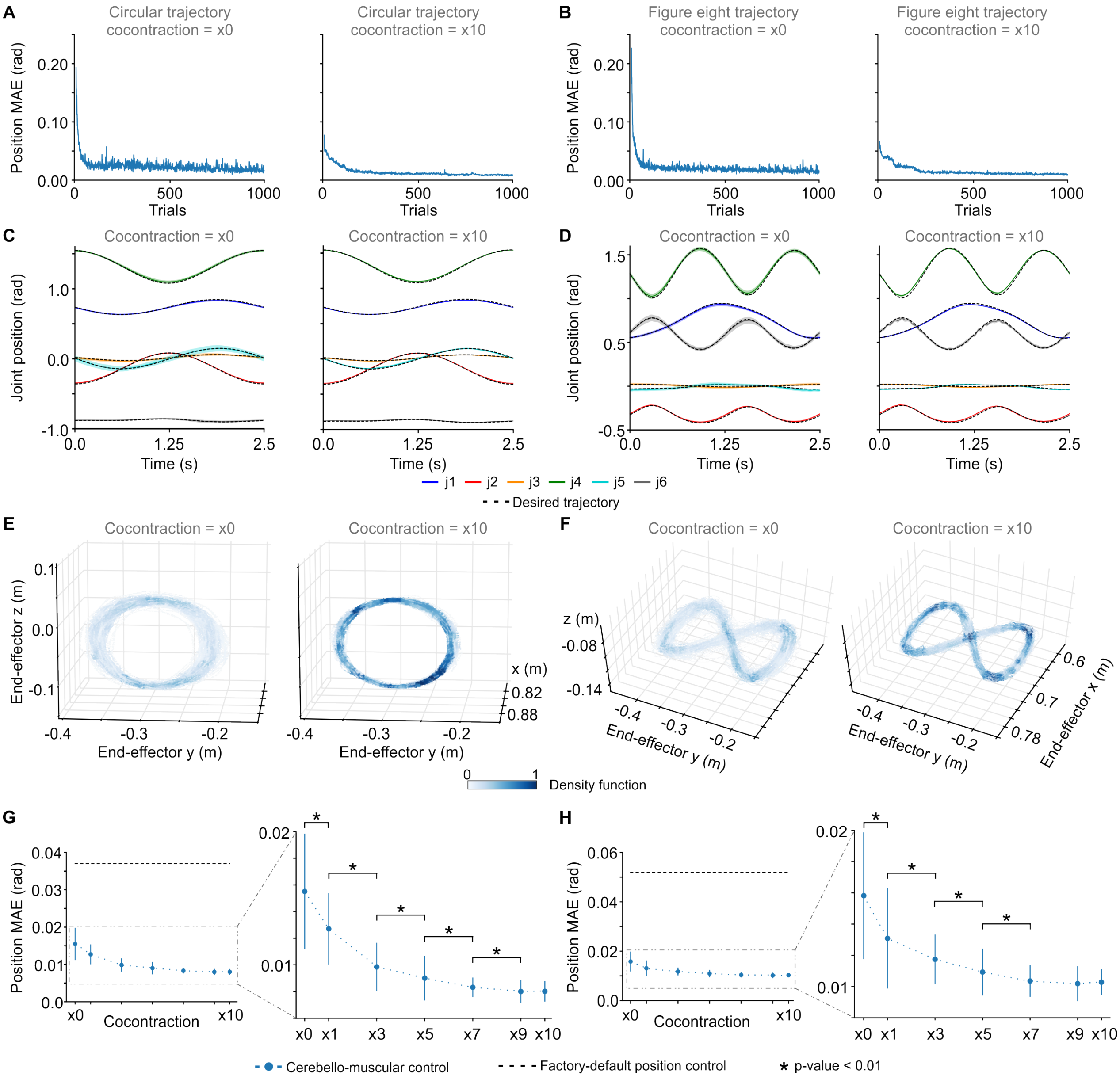

A
Circular trajectory cocontraction = x0
Circular trajectory cocontraction = x10
B
Figure eight trajectory cocontraction = x0
Figure eight trajectory cocontraction = x10
Position MAE (rad)
0.20
0.10
0.00
0
500
1000
Trials
C
Cocontraction = x0
Cocontraction = x10
D
Joint position (rad)
1.0
0.0
-1.0
1.5
0.5
-0.5
0.0
1.25
2.5
Time (s)
j1
j2
j3
j4
j5
j6
Desired trajectory
E
F
End-effector z (m)
0.1
0.0
-0.1
x (m)
0.82
0.88
-0.4
-0.3
-0.2
End-effector y (m)
z (m)
-0.08
-0.14
0.6
0.7
0.78
End-effector x (m)
0
1
Density function
G
H
0.04
0.03
0.02
0.01
0.0
0.06
x0
x1
x3
x5
x7
x9
x10
Cocontraction
*
Cerebello-muscular control
Factory-default position control
* p-value < 0.01

Figure 2

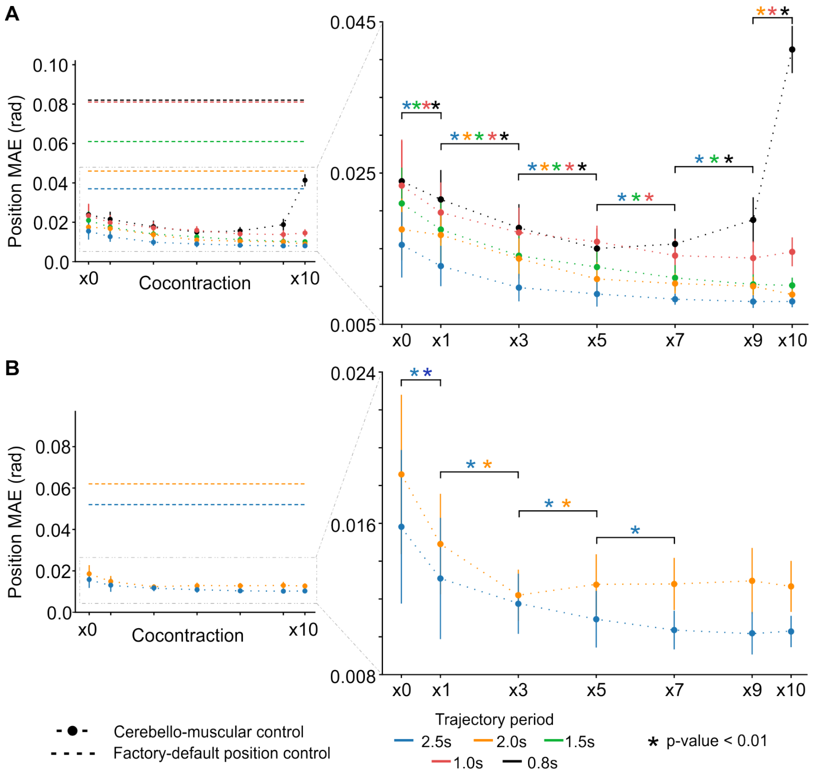
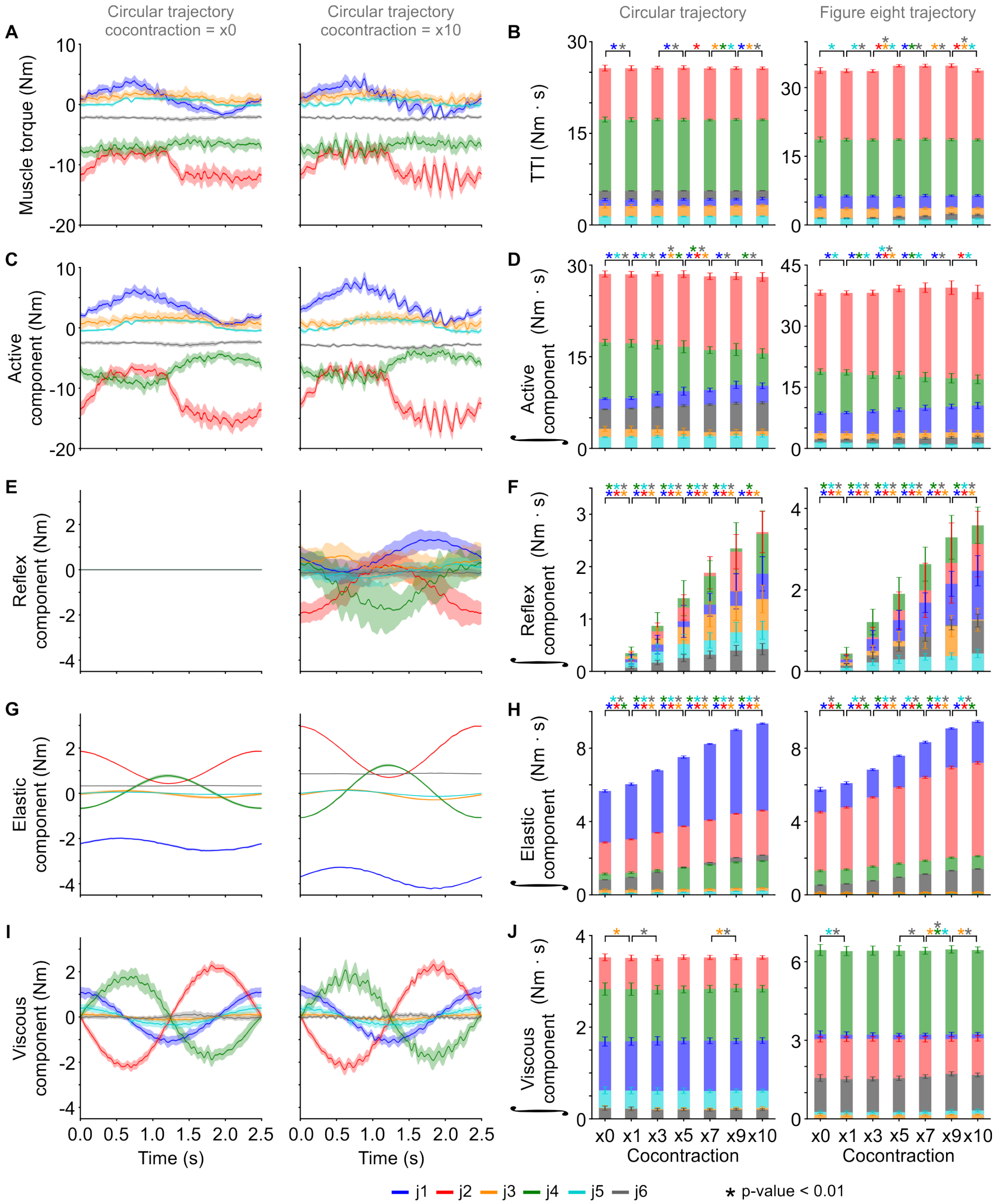

A
Position MAE (rad)
0.10
0.08
0.06
0.04
0.02
0.0
x0
Cocontraction
x10
0.045
0.025
0.005
x0
x1
x3
x5
x7
x9
x10
B
Position MAE (rad)
0.08
0.06
0.04
0.02
0.0
x0
Cocontraction
x10
0.024
0.016
0.008
x0
x1
x3
x5
x7
x9
x10
Cerebello-muscular control
Factory-default position control
Trajectory period
2.5s
2.0s
1.5s
1.0s
0.8s
p-value < 0.01

Figure 3



1296 Figure 4

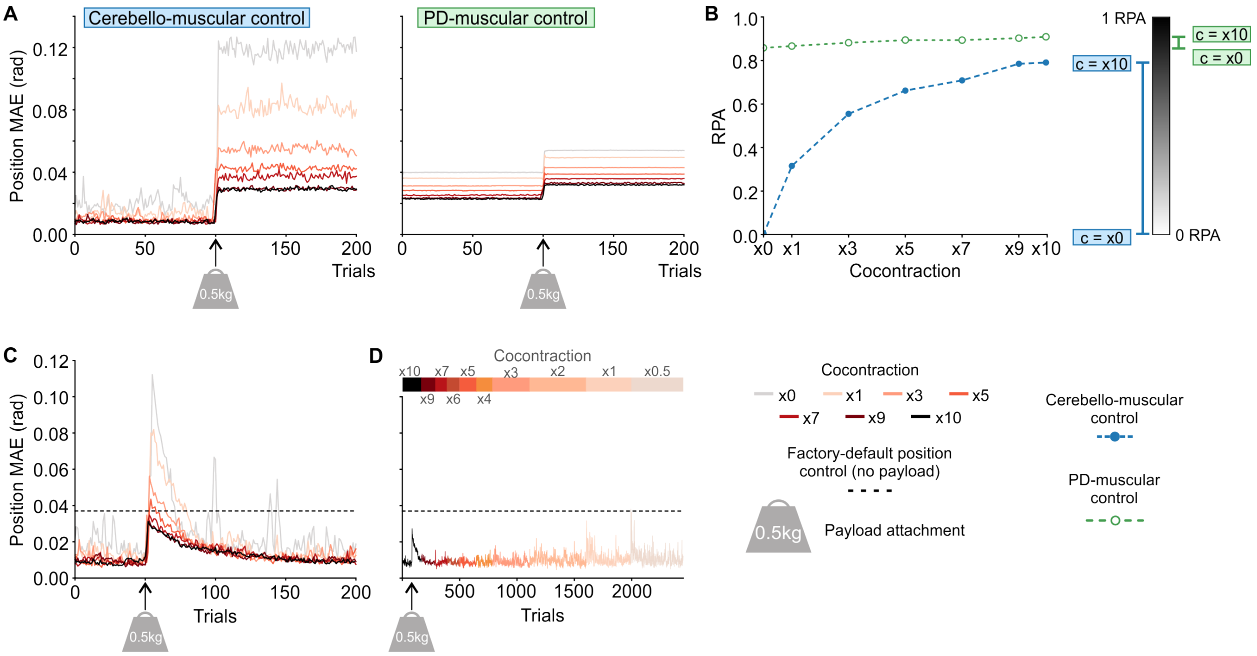

A
Cerebello-muscular control
PD-muscular control
Position MAE (rad)
0.00
0.04
0.08
0.12
0
50
150
200
Trials
0.5kg
B
RPA
0.0
0.2
0.4
0.6
0.8
1.0
x0
x1
x3
x5
x7
x9
x10
Cocontraction
1 RPA
0 RPA
c = x10
c = x0
C
0.00
0.02
0.04
0.06
0.08
0.10
0.12
100
D
Cocontraction
x10
x9
x7
x6
x5
x4
x3
x2
x1
x0.5
500
1000
1500
2000
Cerebello-muscular control
PD-muscular control
Factory-default position control (no payload)
Payload attachment

Figure 5

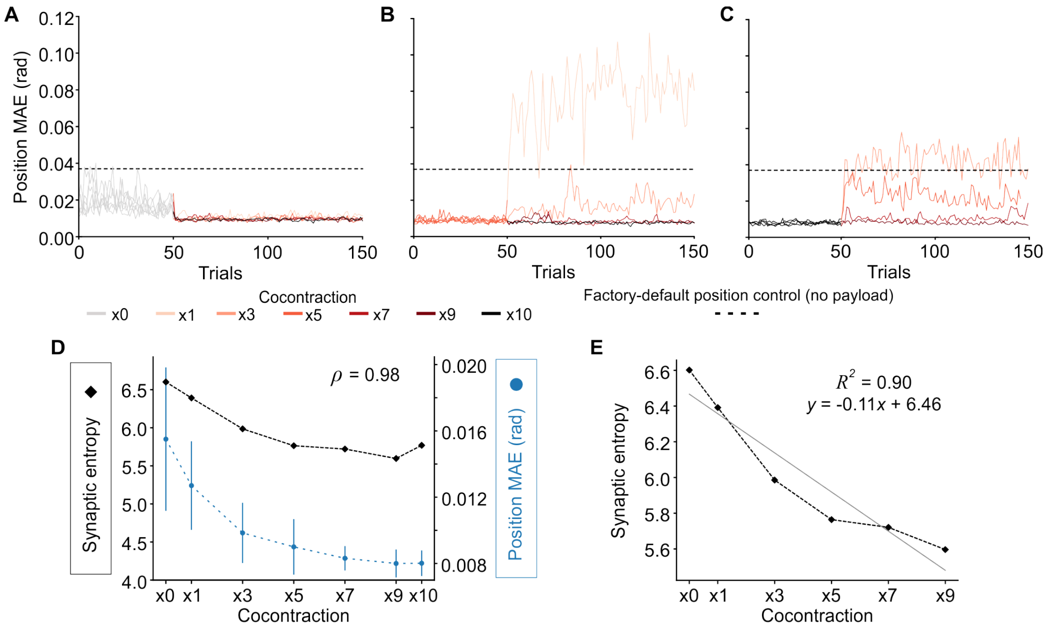

A
B
C
Position MAE (rad)
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0
50
100
150
Trials
Cocontraction
x0
x1
x3
x5
x7
x9
x10
Factory-default position control (no payload)
D
Synaptic entropy
6.5
6.0
5.5
5.0
4.5
4.0
ρ = 0.98
0.020
0.016
0.012
0.008
Position MAE (rad)
x0 x1 x3 x5 x7 x9 x10
Cocontraction
E
Synaptic entropy
6.6
6.4
6.2
6.0
5.8
5.6
R^2 = 0.90
y = -0.11x + 6.46
x0 x1 x3 x5 x7 x9
Cocontraction

Figure 6

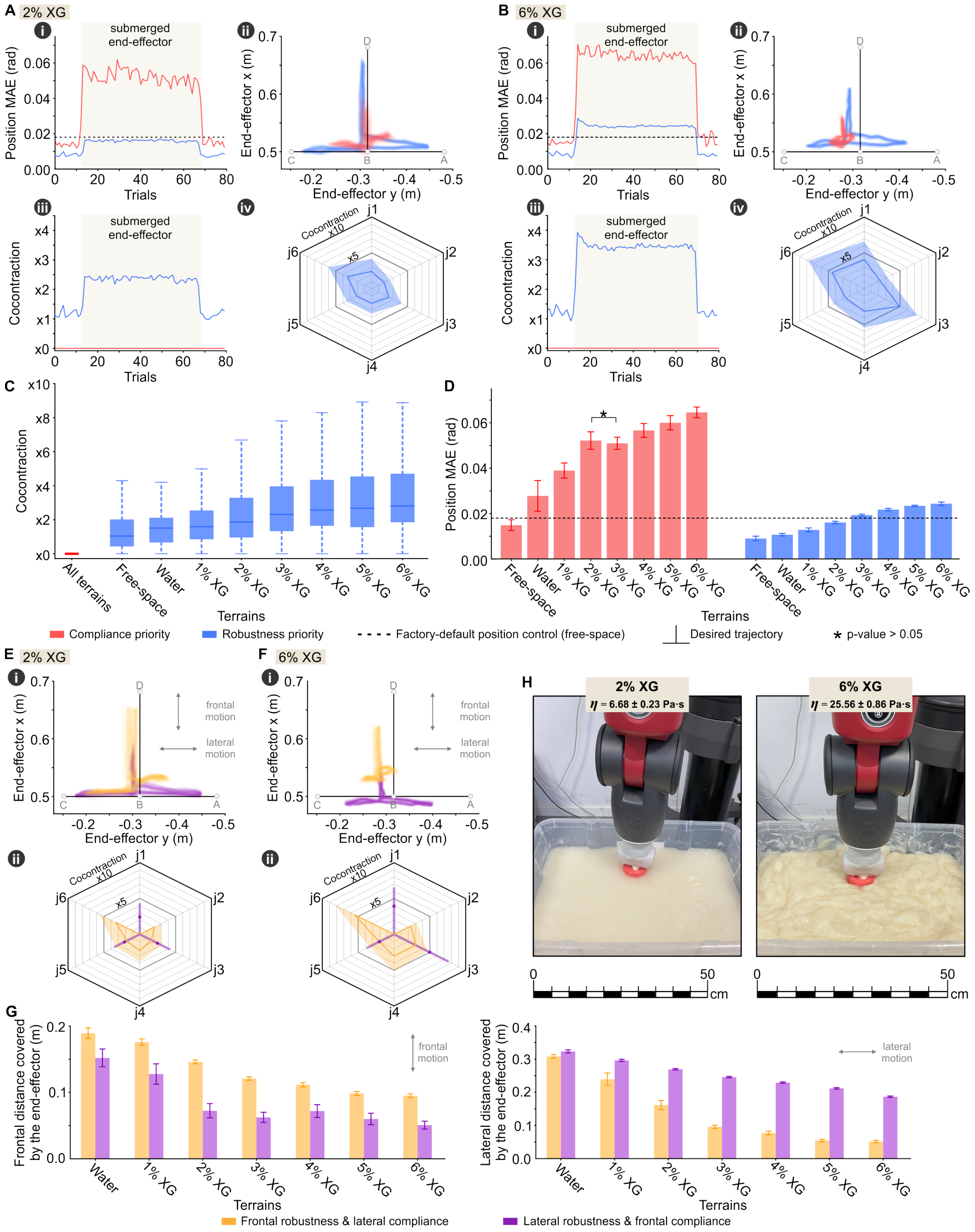

A 2% XG
i
submerged end-effector
Position MAE (rad)
0.00
0.02
0.04
0.06
Trials
0 20 40 60 80
ii
End-effector x (m)
0.5 0.6 0.7
End-effector y (m)
-0.2 -0.3 -0.4 -0.5
C B A D
iii
Cocontraction
x0 x1 x2 x3 x4
iv
Cocontraction x10
x5
j1 j2 j3 j4 j5 j6
B 6% XG
C
Cocontraction
x0 x2 x4 x6 x8 x10
All terrains
Free-space
Water
1% XG
2% XG
3% XG
4% XG
5% XG
6% XG
Terrains
D
Position MAE (rad)
0.00 0.02 0.04 0.06
*
Compliance priority
Robustness priority
Factory-default position control (free-space)
Desired trajectory
* p-value > 0.05
E 2% XG
F 6% XG
frontal motion
lateral motion
H
2% XG
η = 6.68 ± 0.23 Pa·s
6% XG
η = 25.56 ± 0.86 Pa·s
0 50 cm
G
Frontal distance covered by the end-effector (m)
0.0 0.1 0.2
Lateral distance covered by the end-effector (m)
0.0 0.1 0.2 0.3 0.4
Frontal robustness & lateral compliance
Lateral robustness & frontal compliance

Figure 7

Movie 1 still image.

Suppplementary Materials for

A neuromechanics solution for adjustable compliance and accuracy

Ignacio Abadía, Alice Bruel, Grégoire Courtine, Auke Ijspeert, Eduardo Ros, Niceto R. Luque

The supplementary materials include:

Supplementary Methods
Supplementary Results
Figs. S1 to S12
Tables S1 to S9
Movies S1 and S2

Supplementary Methods

Muscle model parameters

The configuration values of the muscle parameters of each joint are described in table S1.

Table S1. Muscle model parameters.

Joint	Parameters				
	Muscle gain (α)	Muscle stiffness gain (β)	Muscle tonic stiffness (γ)	Muscle damping (δ)	Joint resting position (θ_r) (rad)
j1	35.0	1.0	3.0	-4.0	0.0
j2	45.0	1.0	3.0	-4.0	0.2155
j3	35.0	1.0	3.0	-1.0	0.0
j4	35.0	1.0	3.0	-3.0	1.3349
j5	10.5	0.3	1.0	-1.0	0.0
j6	10.5	0.3	1.0	-1.0	0.0

Leaky integrate-and-fire neuron model

The cerebellar model was implemented using leaky integrate-and-fire (LIF) neurons (*99*), due to their efficient computational cost for spike generation and processing, facilitating the real time requirements of the robot control loop. LIF neurons fired a spike when the membrane potential reached the firing threshold, after which the membrane potential was reset to its resting value. The neural dynamics was defined by the membrane potential and the excitatory (AMPA for the α-amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid receptor; and NMDA for the N-methyl-D-aspartate receptor) and inhibitory (GABA for the γ-aminobutyric acid receptor) chemical conductances, as defined by:

$$C \cdot \frac{dV}{dt} = I_{int} + I_{ext} \tag{1}$$

$$I_{int} = -g_l \cdot \left(V + E_L\right) \tag{2}$$

$$I_{ext} = -\left(g_{AMPA}(t) + g_{NMDA}(t) \cdot g_{NMDA\_inf}\right) \cdot \left(V - E_{AMPA}\right) - g_{GABA}(t) \cdot \left(V - E_{GABA}\right) \tag{3}$$

$$g_{AMPA}(t) = g_{AMPA}(t_0) \cdot e^{\frac{t-t_0}{\tau_{AMPA}}} + \sum_{i=1}^{N} \delta_{AMPA_i}(t) \cdot w_i \tag{4}$$

$$g_{NMDA}(t) = g_{NMDA}(t_0) \cdot e^{\frac{t-t_0}{\tau_{NMDA}}} + \sum_{i=1}^{N} \delta_{NMDA_i}(t) \cdot w_i \tag{5}$$

$$g_{GABA}(t) = g_{GABA}(t_0) \cdot e^{\frac{t-t_0}{\tau_{GABA}}} + \sum_{i=1}^{N} \delta_{GABA_i}(t) \cdot w_i \tag{6}$$

$$g_{NMDA\_inf} = 1 \Big/ \left(1 + exp\left(62 \cdot V\right) \cdot \frac{1.2}{3.57}\right) \tag{7}$$

where C is the membrane capacitance; V is the membrane potential; I_{int} is the internal current; I_{ext} is the external current; E_L is the resting potential and g_L the conductance responsible for the passive decay term towards E_L. The conductances g_{AMPA}, g_{NMDA} and g_{GABA} integrate all the contributions received by each receptor type (AMPA, NMDA, GABA) through individual synapses, being g_{NMDA_inf} the NMDA activation channel. These conductances were defined as decaying exponential functions (*99*, *108*, *109*), and their values were incremented proportionally to the synaptic weights (w_i) upon each presynaptic spike arrival (δ, Dirac delta functions). When the membrane potential reached the firing threshold (V_{thr}) it was then reset to the resting potential (E_L) during the refractory period (T_{ref}).

The configuration parameters for the neuron models are described in table S2. Cerebellar neural network topology is described in table S3.

Table S2. Neuron model parameters. Dash entries indicate not applicable.

Parameter	**GC**	**PC**	**DCN**
C (pF)	2.0	100.0	2.0
g_l (nS)	1.0	6.0	0.2
E_L (mV)	-65.0	-70.0	-70.0
E_{AMPA} (mV)	0.0	0.0	0.0
E_{GABA} (mV)	-	-	-80.0
τ_{AMPA} (ms)	1.0	1.2	0.5
τ_{NMDA} (ms)	-	-	14.0
τ_{GABA} (ms)	-	-	10.0
V_{thr} (mV)	-50.0	-52.0	-40.0
T_{ref} (ms)	1.0	2.0	1.0

Table S3. Cerebellar network topology. Dash entries indicate not applicable.

Neurons		**Synapses**				
Presynaptic	Postsynaptic	Connectivity	Number	Type	Initial weight (nS)	Weight range (nS)
168 MFs	14406 GCs	Four to one	57624	AMPA	0.18	-
168 MFs	600 DCN	All to one	100800	AMPA	0.1	-
14406 GCs	600 PCs	All to one	8643600	AMPA	1.6	[0.0, 5.0]
600 PCs	600 DCNs	One to one	600	GABA	1.0	-
600 CFs	600 PCs	One to one	600	AMPA	0.0	-
600 CFs	600 DCN	One to one	600	AMPA	0.5	-
600 CFs	600 DCN	One to one	600	NMDA	0.25	-

Conversion from analog to spike domain

The cerebellar SNN sensory input information had to be translated from its original analog domain (joint desired and actual coordinates: θ_d, θ'_d, θ_a, θ'_a) into the spike domain (MFs activity). The 168 MFs were divided into six microcomplexes (one per DOF) of 28 neurons each. Each microcomplex was then divided into four subgroups of seven neurons each, and each subgroup was devoted to coding one of the input analog signals (θ_d, θ'_d, θ_a, θ'_a). Each of the seven neurons within the subgroup acted as a sensory receptor for a specific analog range of its input signal: at time *t*, a MF fired a spike (δ_{MF}) if the analog value of its corresponding input signal (θ_x) was within its receptor interval (θ_{MF}), as described by:

$$\delta_{MF,i}(t) \leftrightarrow \theta_{\mathrm{x}}(t) \in \theta_{\mathrm{MF,i}} \tag{8}$$

$$\theta_{\mathrm{MF,i}} = [c_i - w, c_i + w] \tag{9}$$

$$c_i = \theta_{\min} + \left(\frac{\theta_{\max} - \theta_{\min}}{N-1} \right) \cdot i \tag{10}$$

$$w = \frac{1}{2} \cdot \left(\frac{\theta_{\max} - \theta_{\min}}{N-1} \right) \tag{11}$$

where *i* is the MF neuron index within the subgroup ($i \in [0,6]$); c_i and *w* are the center and width of the MF_i specific analog range, respectively; θ_{max} and θ_{min} are the analog limits of the input joint signal; and N=7 is the number of MFs within the subgroup. The receptor intervals within the subgroup were non-overlapping, therefore only one MF within the subgroup was active at each time step; four MFs per microcomplex were active at each time step to code the input sensory state (θ_d, θ'_d, θ_a, θ'_a) in a univocal spiking representation.

Cerebellar instructive signal

The cerebellar instructive signal (ε) was computed by comparing the desired (θ_d, θ'_d) and actual joint state (θ_a, θ'_a), as follows:

$$\varepsilon_j(t) = k_{p,j}[\theta_{d,j}(t) - \theta_{a,j}(t)] + k_{v,j}[\theta'_{d,j}(t) - \theta'_{a,j}(t)] \tag{12}$$

where $k_{p,j}$=[1.5, 2.0, 3.0, 2.0, 3.0, 3.0] and $k_{v,j}$=[1.5, 1.0, 3.0, 1.0, 3.0, 0.5] are the position and velocity error gains for each joint (*j*).

The instructive signal had to be translated from its original analog domain into spiking domain (CFs activity) to enter the cerebellar SNN. The 600 CFs were divided into six microcomplexes (one per DOF) of 100 neurons each. Within the microcomplex, the first/last 50 neurons were devoted to the agonist/antagonist sensed error (positive/negative joint error). Electrophysiological recordings of CF activity show a chaotic and low firing rate, between 1 and 10 Hz per neuron (*110*), which could hamper capturing the high-frequency information of the instructive signal. However, the chaotic firing does allow for a statistical sampling of the entire signal range over multiple trials (*110*, *111*). We replicated this behavior using a Poisson CF model: given the instructive signal $\varepsilon(t)$ and a random number $\eta(t) \in [0, 1]$, the given CF fired a spike $\delta_{CFspike}(t) \leftrightarrow \varepsilon(t) > \eta(t)$, otherwise remaining silent (*112*, *113*).

Computational complexity of PID controller and cerebellar SNN

We conducted a computational complexity analysis, comparing the megaflops (Mflops) required for a proportional-integral-derivative (PID) controller at each time step (assuming the factory-default position controller is implemented as a PID) with the estimated Mflops needed to generate the number of spikes per time step in our neuromechanics approach. Below is a detailed description of the computational complexity analysis. For the calculations, we considered two plausible architectures for the SNN: an event/spike-based (neuromorphic) architecture and a time-driven (von Neumann) architecture.

Given the traditional PID formulation:

$$u_i(t) = K_{p_i} e_i(t) + K_{i_i} \int e_i(t)\,dt + K_{d_i} \frac{de_i(t)}{dt} \tag{13}$$

where the proportional term, the integral term (using Forward Euler) and the derivative term (using finite difference approximation) are:

$$P_i[k] + I_i[k] + D_i[k] \tag{14}$$

The traditional PID formulation then discretized resulting the original formulation as:

$$u_i[k] = K_{p_i} e_i[k] + I_i[k-1] + K_{i_i} e_i[k]\Delta t + K_{d_i} \frac{e_i[k] - e_i[k-1]}{\Delta t} = P_i[k] + I_i[k] + D_i[k] \tag{15}$$

Considering the error at time step *k* as:

$$e_i[k] = q_d[k] - q[k] \tag{16}$$

and Δt the sampling time interval, we would obtain the final equation:

$$u_i[k] = K_{p_i}\left(q_{d_i}[k] - q_i[k]\right) + I_i[k-1] + K_{i_i}\left(q_{d_i}[k] - q_i[k]\right)\Delta t + K_{d_i} \frac{\left(q_{d_i}[k] - q_i[k]\right) - \left(q_{d_i}[k-1] - q_i[k-1]\right)}{\Delta t} \tag{17}$$

For each degree of freedom *i*, the PID control output $u_i[k]$ at time step *k* involves the following floating-point operations (flops): Error term: 1 flop (subtraction). Proportional term: 1 flop (multiplication) + 1 flop (error). Integral term: 2 flops (multiplication + addition) + 1 flop (error). Derivative term: 3 flops (subtraction, division, multiplication) + 1 flop (error). Summing the terms to compute the control output: 2 flops (additions).

Summing up all the operations for one degree of freedom, we would obtain 11 flops. Since the operations for each degree of freedom are independent, the total number of flops for six degrees of freedom is: 6×11 flops/time step= 66 flops/time step = 0.000066 Mflops /time step.

Regarding the computational complexity of the cerebellar SNN, the traditional LIF model equation is given by:

$$C \cdot \frac{dV(t)}{dt} = -g_l \left(V(t) + E_L\right) + I_{ext}(t)$$
$$being \quad \tau = \frac{C}{g_l} \tag{18}$$

For numerical simulations, this equation is often discretized using methods like the Forward Euler method. The update rule for the membrane potential at each time step *Δt* is given by:

$$V[k+1] = V[k] + \Delta t \left(\frac{-\left(V[k] + E_L\right)}{\tau} + \frac{I_{ext}[k]}{C_m} \right) \tag{19}$$

The operations involved are: 3 flops (division to compute *τ*, division by *τ*, and division by C_m); 3 flops (three additions); 1 flop (multiplication).

Summing up all the operations, we would have a total of 7 flops per spike. Our cerebellar SNN averaged 170 internal spikes per time step. Assuming an event-based architecture, the total Mflops would be: 170 spikes×7 flops/spike = 1190 flops = 0.00119 Mflops.

If we were to consider time-driven instead of event-driven architecture running our cerebellum model, we would have a total of 7 flops per neuron for a total of 15606 internal neurons. The total Mflops would be: 15606 internal neurons × 7 flops/time step = 109242 flops = 0.109 Mflops.

Therefore, the computational complexity of a PID controller and a cerebellar SNN is described as in Table S2.

Table S4. Computational complexity analysis.

PID controller	**Cerebellar SNN assuming event-based architecture**	**Gain**
0.000066 Mflops/time step	0.00119 Mflops/time step	x18
	Cerebellar SNN assuming event-based architecture	
	0.109Mflops/time step	x1655

In summary, while our cerebello-muscular control requires more computational resources, this is due to the increased complexity inherent in its design. Nonetheless, we believe that the advantages in performance would justify these computational demands.

Table S5. Gains of the PD-muscular controller.

Joint	Circular trajectory		Figure eight trajectory	
	Proportional gain (Kp)	Derivative gain (Kd)	Proportional gain (Kp)	Derivative gain (Kd)
j1	7.5	0.01	7.0	0.04
j2	8.0	0.01	8.0	0.04
j3	1.0	0.01	1.0	0.04
j4	6.0	0.005	4.5	0.02
j5	1.0	0.005	1.0	0.025
j6	4.0	0.005	4.5	0.02

Table S6. Gains of the PD-muscular controller with higher gains for the circular trajectory.

Joint	Circular trajectory	
	Proportional gain (Kp)	Derivative gain (Kd)
j1	8.0	0.05
j2	9.0	0.05
j3	1.0	0.05
j4	8.0	0.025
j5	1.5	0.025
j6	8.0	0.025

Table S7. Gains of the PD controllers without muscle dynamics for the circular trajectory.

Joint	PD v1		PD v2		PD v3	
	Proportional gain (Kp)	Derivative gain (Kd)	Proportional gain (Kp)	Derivative gain (Kd)	Proportional gain (Kp)	Derivative gain (Kd)
j1	39.6	18.0	99.0	20.0	132.0	20.0
j2	62.1	10.8	155.25	12.0	207.0	12.0
j3	15.9	2.7	39.75	3.0	53.0	3.0
j4	59.1	4.05	147.75	4.5	197.0	4.5
j5	9.0	2.223	22.5	2.47	30.0	2.47
j6	25.5	2.7	63.75	3.0	85.0	3.0

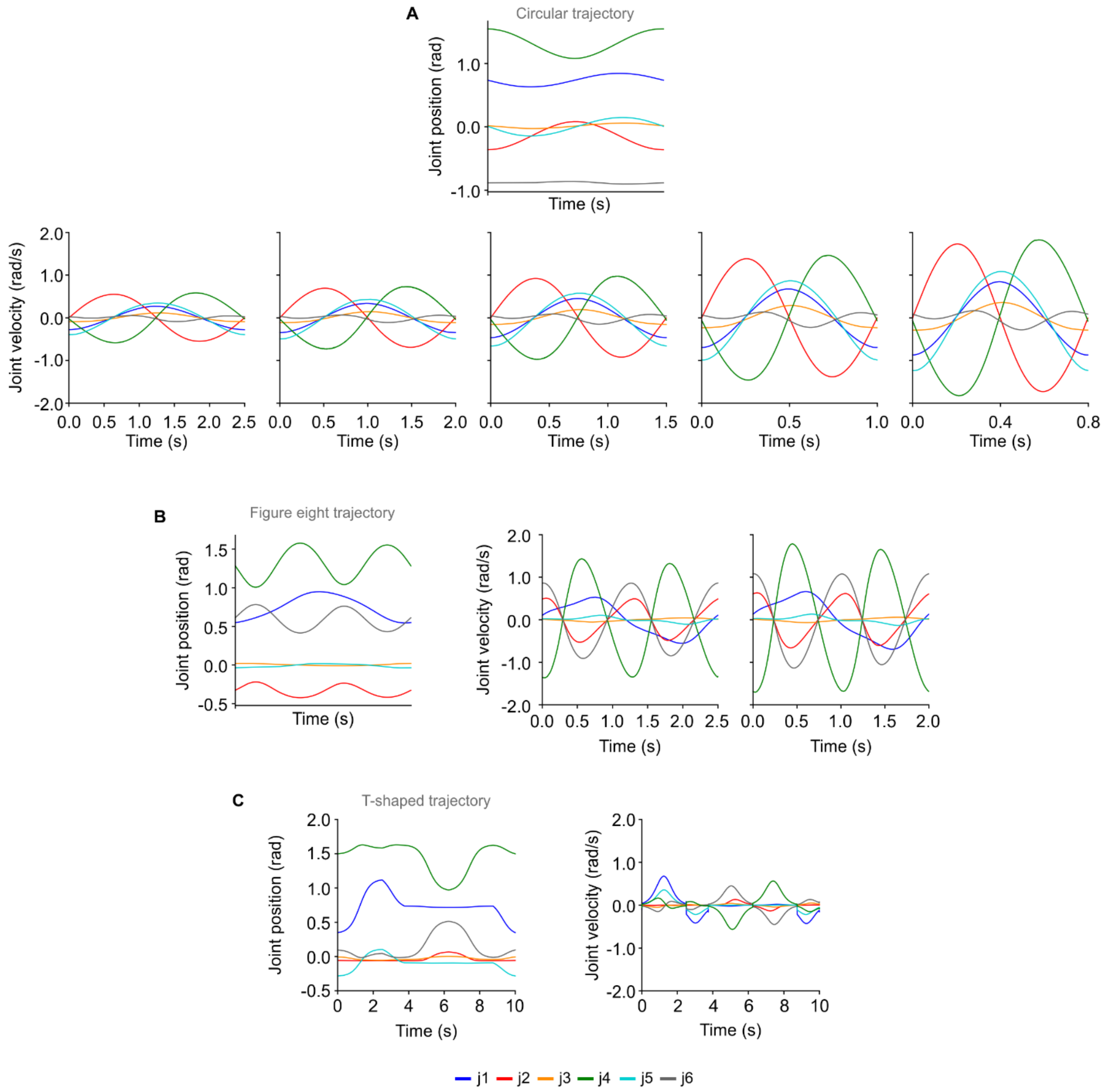


Fig. S1. Joint coordinates of the motor tasks. (A) Joint position coordinates and joint velocity coordinates of the circular trajectory performed at different speeds. **(B)** Joint position coordinates and joint velocity coordinates of the figure eight trajectory performed at different speeds. **(C)** Joint position coordinates and joint velocity coordinates of the T-shaped trajectory.

Supplementary Results

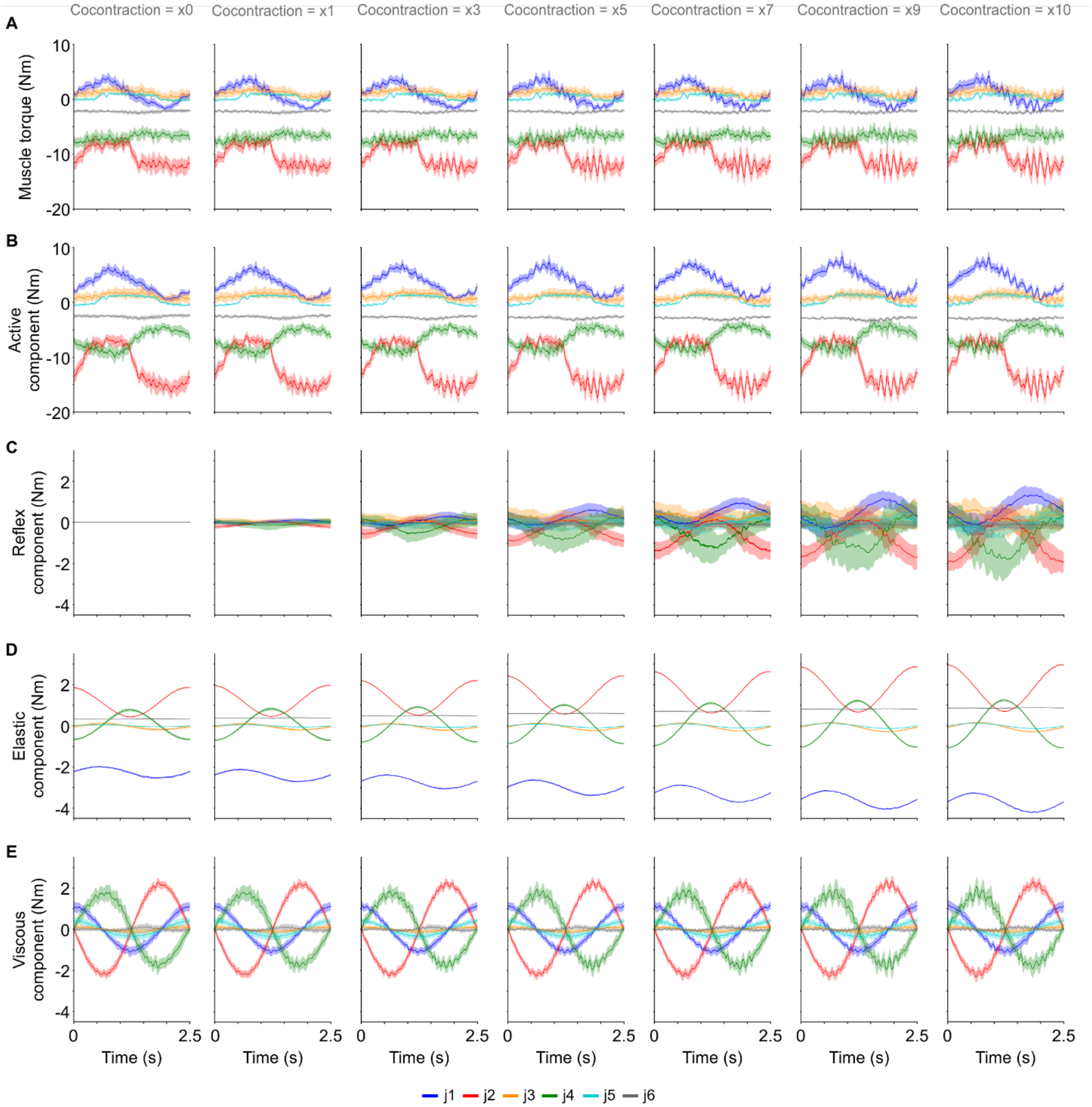


Fig. S2. Muscle torque components during the circular trajectory for all cocontraction values. (A) Resulting muscle torque. **(B)** Active component. **(C)** Reflex component. **(D)** Elastic component. **(E)** Viscous component. Data in (A), (B), (C), (D) and (E) shows the mean and shading shows the SD per time step along the trajectory duration, using the last 100 trials of the motor adaptation process (n = 100).

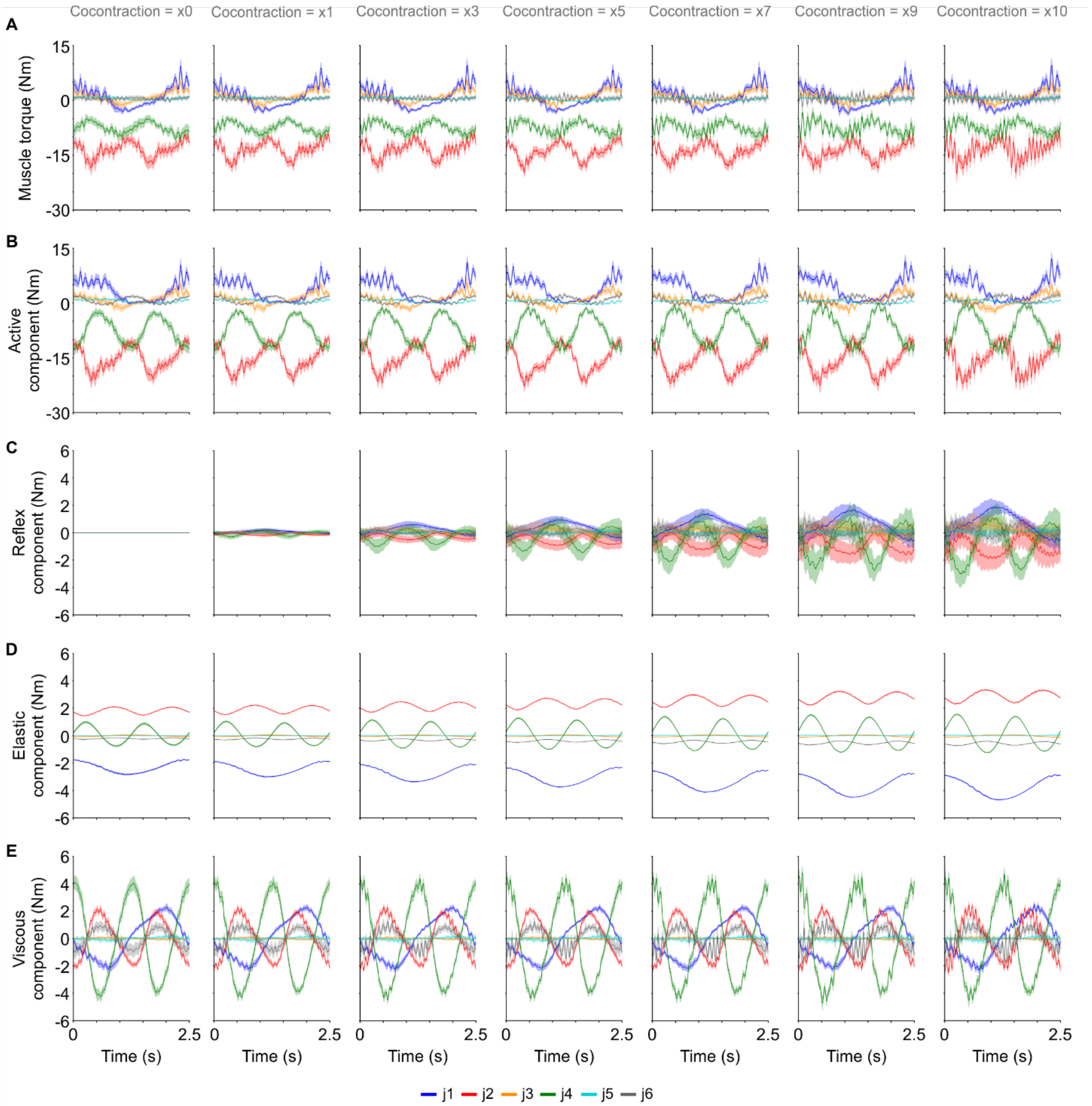


Fig. S3. Muscle torque components during the figure eight trajectory for all cocontraction values. (A) Resulting muscle torque. **(B)** Active component. **(C)** Reflex component. **(D)** Elastic component. **(E)** Viscous component. Data in (A), (B), (C), (D) and (E) shows the mean and shading shows the SD per time step along the trajectory duration, using the last 100 trials of the motor adaptation process (n = 100).

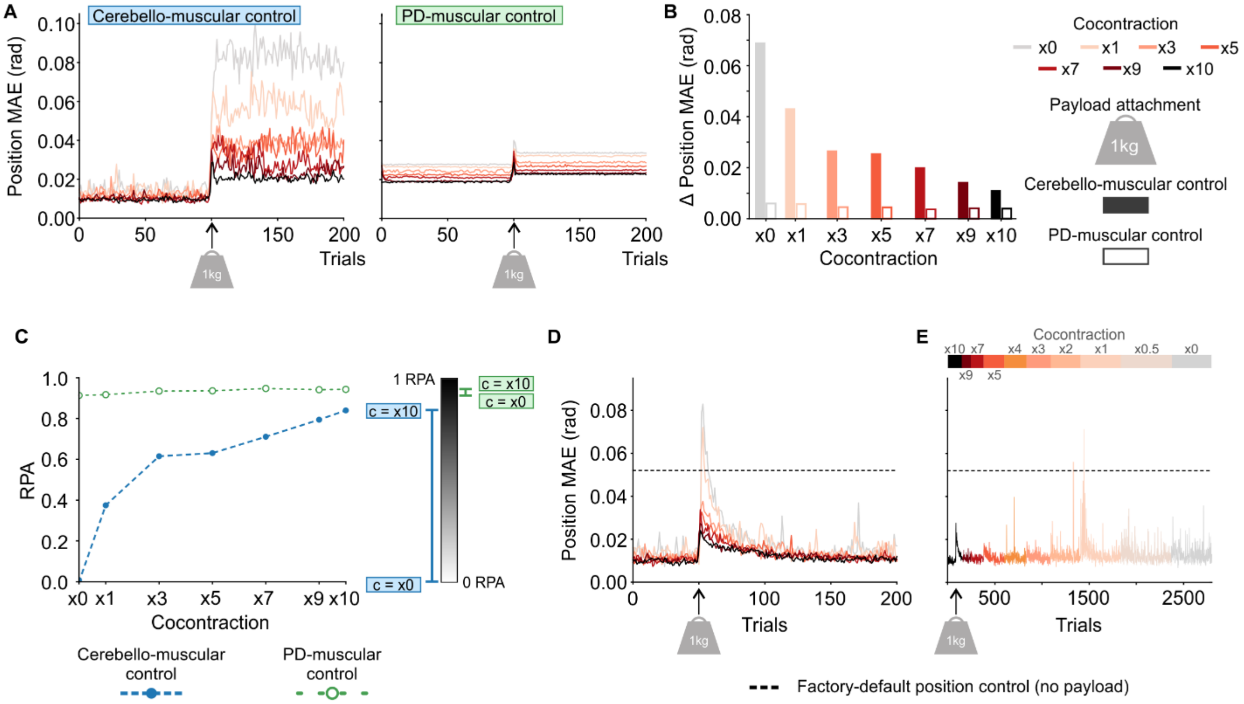


Fig. S4. Response against external perturbations during the figure eight trajectory with varying cocontraction levels. (A) Position MAE before and after attachment of a 1.0 kg payload for the cerebello-muscular and PD-muscular controller at varying cocontraction levels performing the 2.5 s figure eight trajectory. **(B)** Position MAE deviations induced by the attached payload. **(C)** Spectrum of robot motor behavior allowed by the cerebello-muscular and PD-muscular controllers, quantified as the robot physical agency (RPA). Dashed lines added as guides to the eye. **(D)** Position MAE evolution during the motor adaptation process of the cerebello-muscular controller after attaching the 1.0 kg payload while performing the figure eight trajectory. **(E)** Gradual reduction of the cocontraction level following the attachment of the 1.0 kg payload while performing the figure eight trajectory.

Table S8. Limb stiffness measurements for the cerebello-muscular torque controller.

Cocontraction	**Limb stiffness (N/m)**	
	Circle trajectory	Figure eight trajectory
x0	36.9	139.3
x1	46.6	227.7
x3	63.6	290.2
x5	78.9	378.6
x7	94.6	470.3
x9	112.1	464.7
x10	127.6	600.4

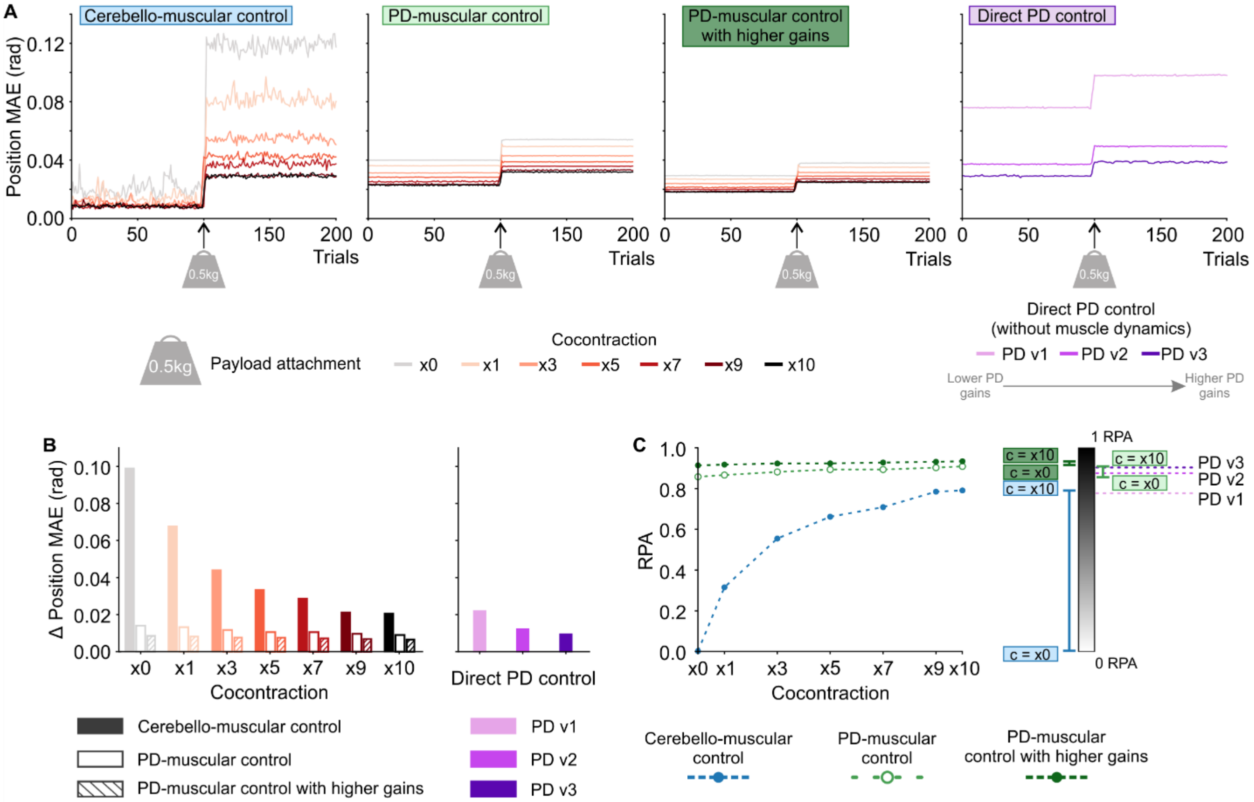


Fig. S5. Deviation induced when attaching a payload during the circular trajectory, for the cerebello-muscular controller, two PD-muscular controllers with different gains, and three PD controllers with different gains. (A) Position MAE before and after payload attachment (0.5 kg) for the cerebello-muscular torque controller, PD-muscular controller, PD-muscular controller with higher gains, and three PD controllers with different gains and without muscle dynamics. A 0.5 kg payload was attached while performing the 2.5 s circular trajectory. **(B)** Position MAE deviations induced by the payload attachment. **(C)** Spectrum of robot motor behavior allowed by the cerebello-muscular, PD-muscular controllers, and PD controllers, quantified as the robot physical agency (RPA). Dashed lines added as guides to the eye.

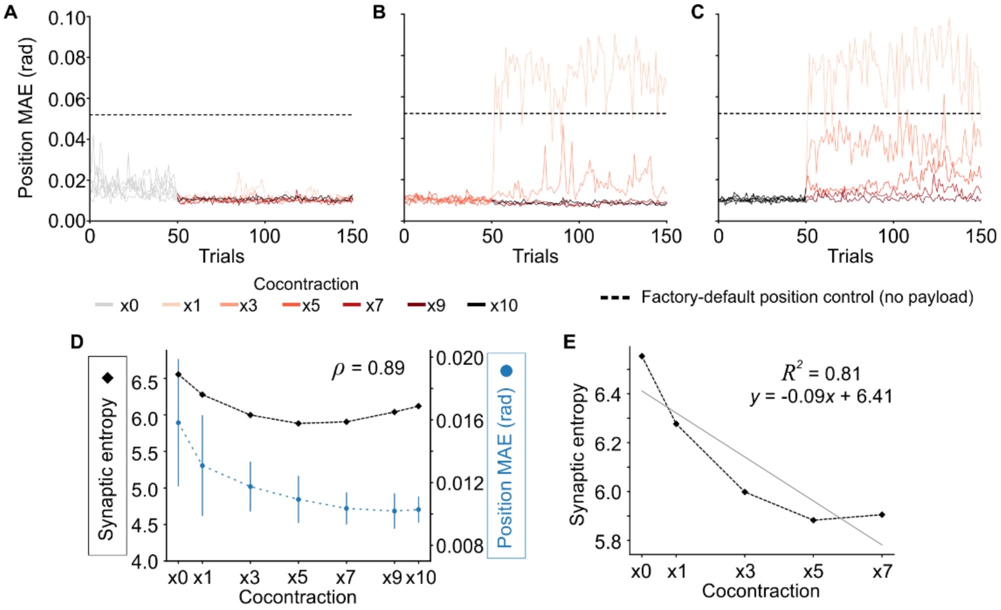


Fig. S6. Application of the cerebellar solution learned for a specific concontraction value to other cocontraction values while performing the figure eight trajectory. (A) Application of the cerebellar solution learned for cocontraction *x0* to higher cocontraction values. From trial 0 to 50 the cerebello-muscular controller performed with *c* = *x0*, then the cocontraction value was modified. **(B)** Application of the cerebellar solution learned for cocontraction *x5* to both lower and higher cocontraction values (excluding *c* = *x0* for safety reasons). From trial 0 to 50 the cerebello-muscular controller performed with *c* = *x5*, then the cocontraction value was modified. **(C)** Application of the cerebellar solution learned for cocontraction *x10* to lower cocontraction values (excluding *c* = *x0* for safety reasons). From trial 0 to 50 the cerebello-muscular controller performed with *c* = *x10*, then the cocontraction value was modified. In (A), (B), and (C), cerebellar learning was disabled. **(D)** Entropy of the cerebellar GC–PC synaptic weights distribution (shown in black) after motor adaptation for each cocontraction value, correlated with the corresponding position MAE (shown in blue). MAE data (taken from Fig. 2H) shows the mean and error bars show the SD of the last 100 trials of the motor adaptation process (n = 100). **(E)** Linear regression of synaptic entropy measurements for those cocontraction values that showed a significant improvement in position MAE (statistical significance of position MAE differences taken from Fig. 2H). Dashed lines in (D) and (E) included as guides to the eye.

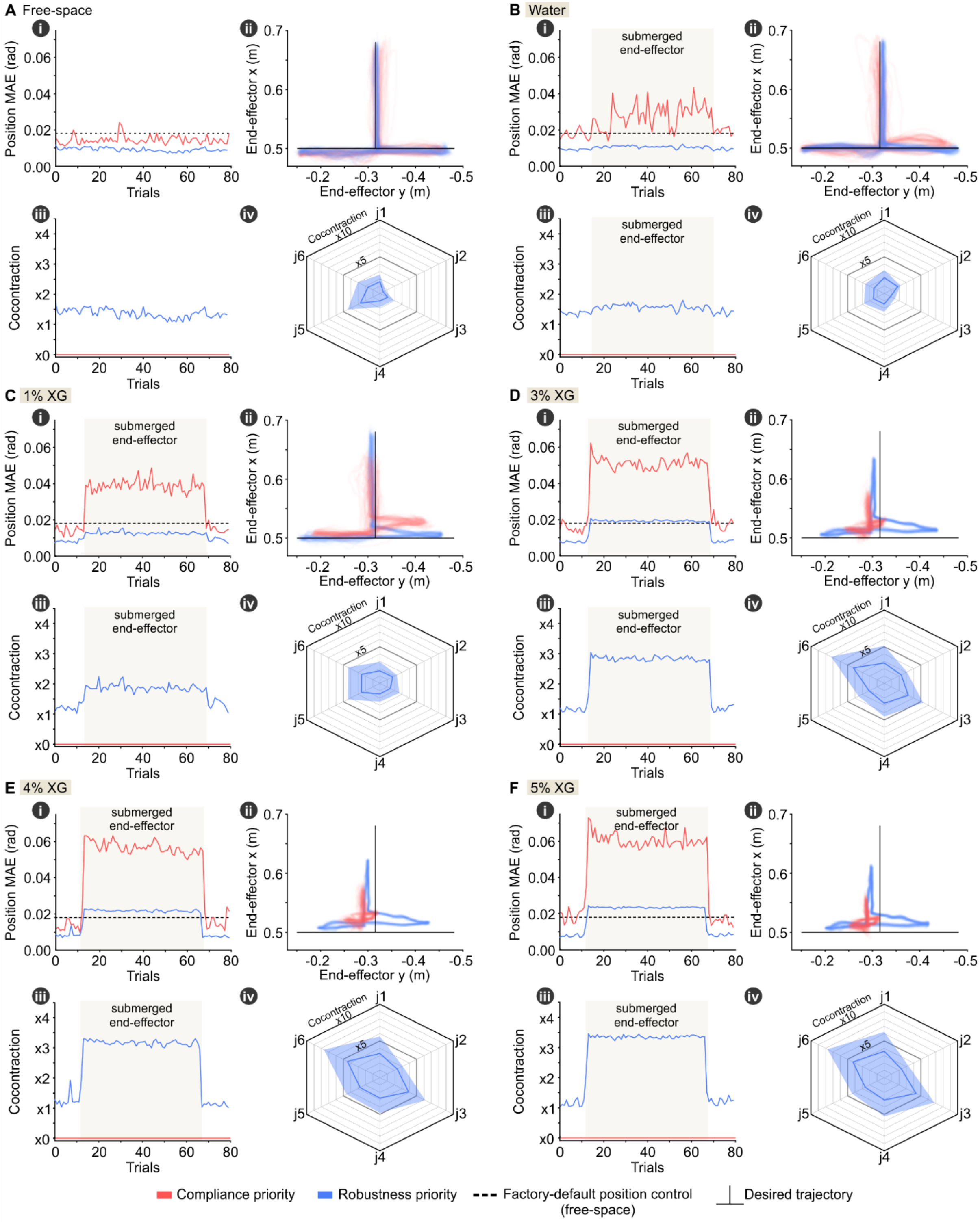


Fig. S7. Active modulation of cocontraction across different terrains. Performance in compliance priority mode and robustness priority mode operating in: **(A)** free-space, **(B)** water, **(C)** 1% concentration of xanthan gum (XG), **(D)** 3% concentration of XG, **(E)** 4% concentration of XG, and **(F)** 5% concentration of XG. (i) Position MAE as the robot end-effector is submerged in the viscous fluid. (ii) Cartesian path described by the submerged end-effector (density of 50 trials shown). (iii) Mean cocontraction per trial as the end-effector is submerged in the viscous fluid. (iv) Joint cocontraction per trial with the end-effector submerged in the fluid (line shows the mean and shading the SD during the first 50 s, sampling every 2 ms).

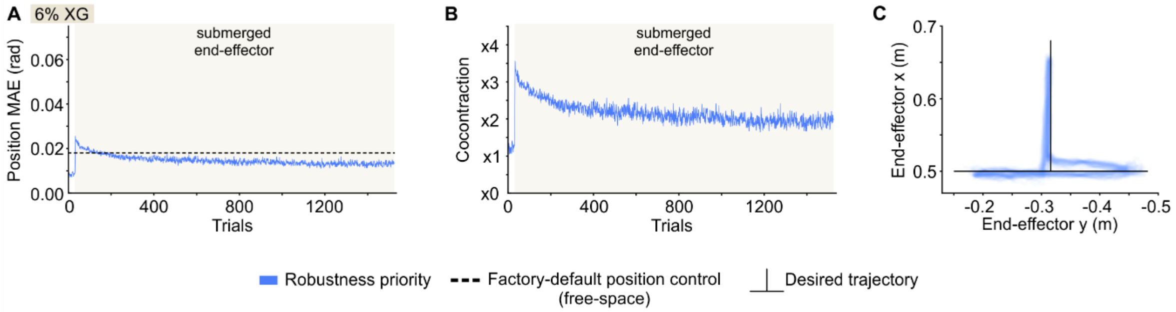


Fig. S8. Cerebellar adaptation to the robot performing in 6% concentration of xanthan gum (XG) operating in robustness priority mode. (A) Evolution of the position MAE as cerebellar adaptation progresses. **(B)** Evolution of the cocontraction level as cerebellar adaptation progresses. **(C)** Cartesian path described by the end-effector submerged in the viscous fluid (density of the last 50 trials shown).

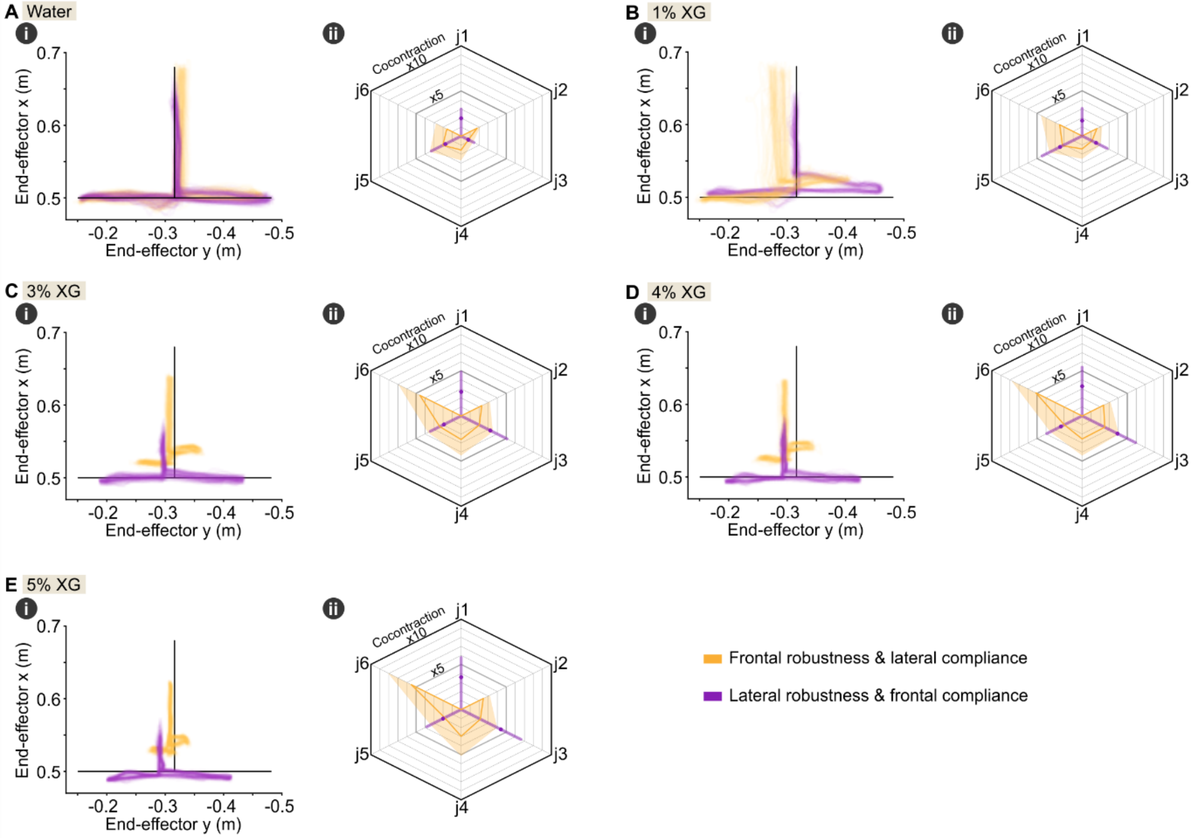


Fig. S9. Differentiated motor behaviors depending on the direction of motion across different terrains. Performance in **(A)** water**, (B)** 1% concentration of xanthan gum (XG), **(C)** 3% concentration of XG, **(D)** 4% concentration of XG, **(E)** 5 % concentration of XG. (i) Cartesian path described by the end-effector when submerged in the viscous fluid (density of 50 trials shown). (ii) Joint cocontraction per trial with the end-effector submerged in the fluid (line shows the mean and shading the SD during the first 50 s, sampling every 2 ms). Two motor behaviors were differentiated: frontal robustness and lateral compliance; lateral robustness and frontal compliance.

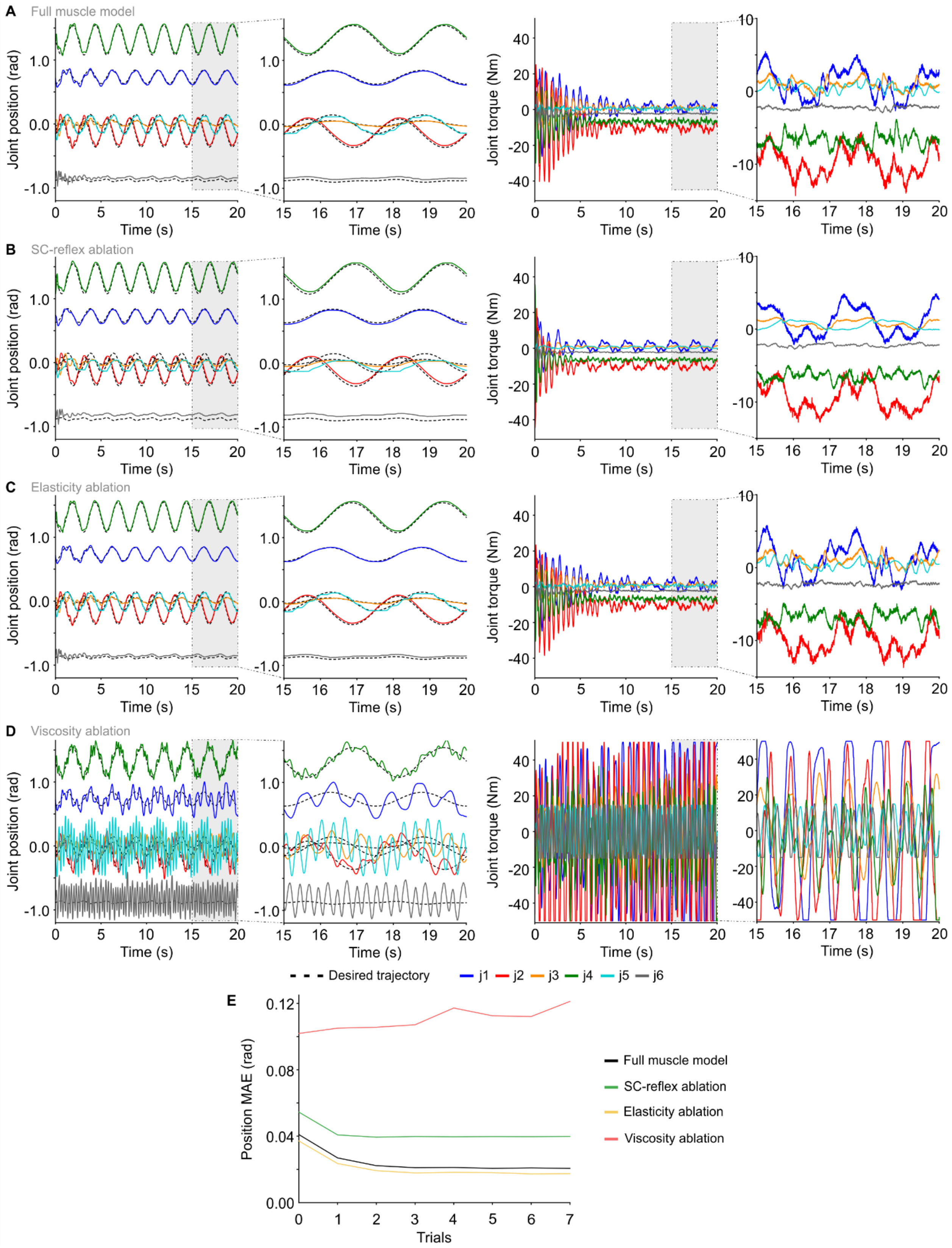


Fig. S10. Muscle components ablation experiments. (A) Complete muscle dynamics (full muscle model) used as reference behavior. **(B)** Muscle dynamics without the SC-reflex component: SC-reflex ablation. **(C)** Muscle dynamics without the elastic component: elasticity ablation. **(D)** Muscle dynamics without the viscous component: viscosity ablation. All scenarios using the PD controller configuration described in Methods. **(E)** Position MAE obtained for the four different scenarios performing the circular trajectory.

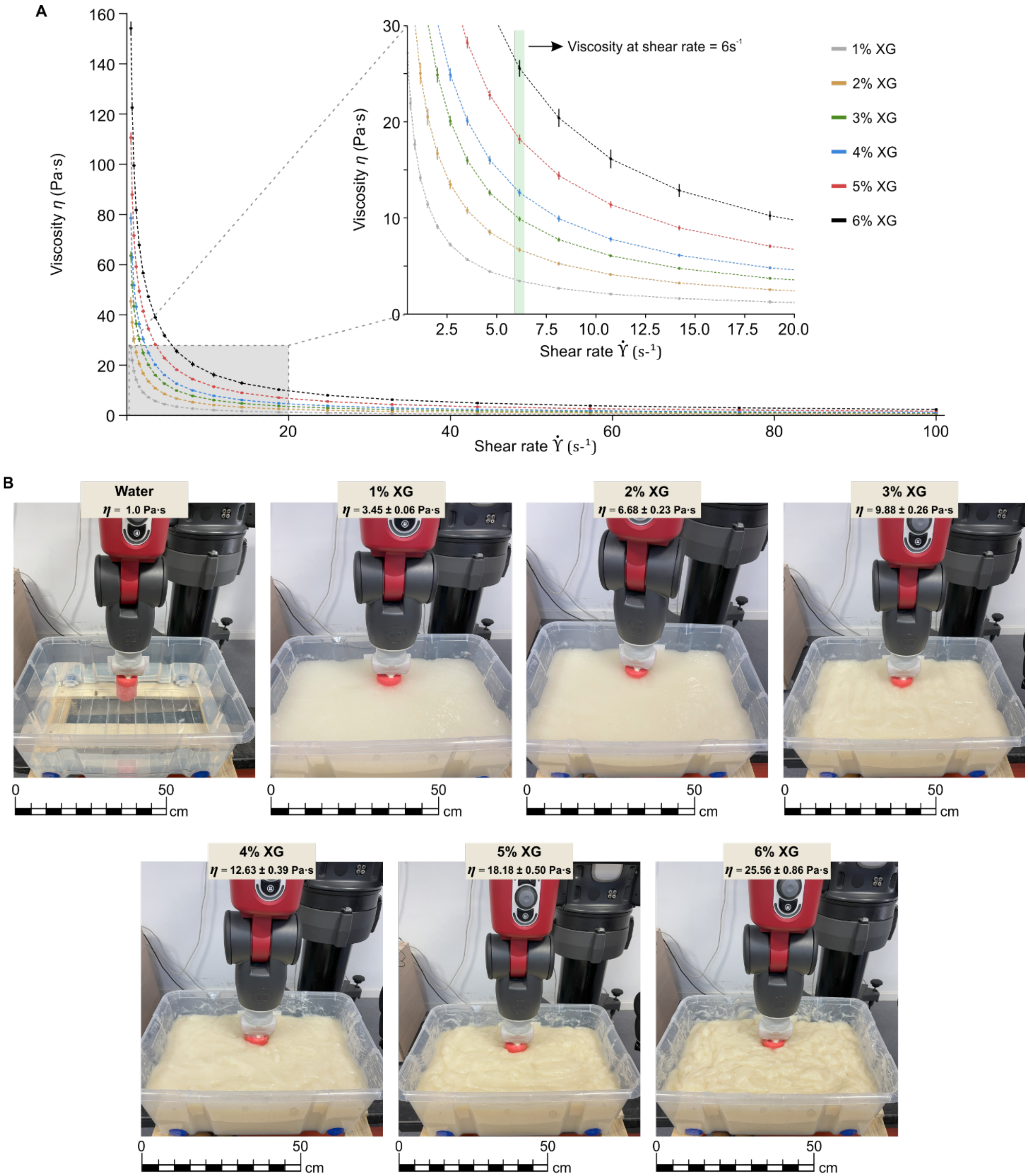
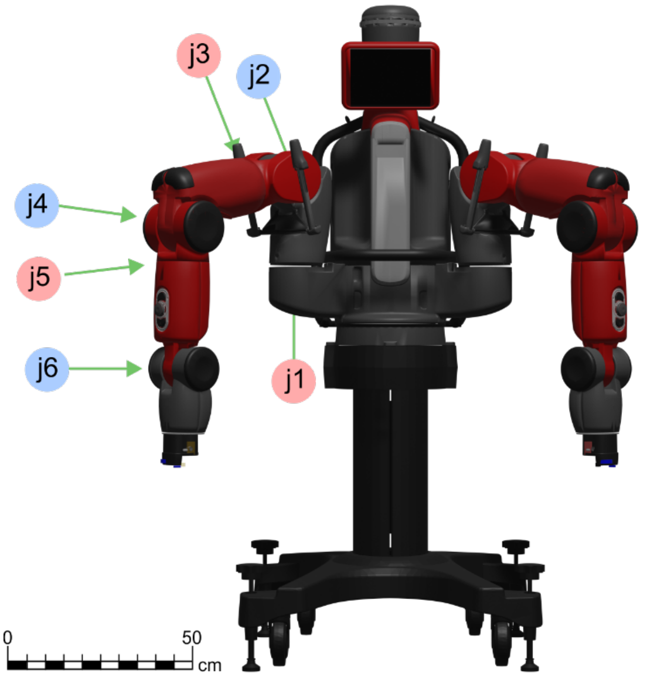


Fig. S11. Terrain viscosity. (A) Viscosity as a function of shear rate. Each terrain consisted of an aqueous solution of xanthan gum (XG) at different concentrations: from 1% to 6% XG concentration. Three repetitions ($n = 3$) of the viscosity (η) measurements were taken for each XG concentration, data shows the mean and error bars show SD. **(B)** Robot end-effector submerged in the different viscous fluids and their corresponding viscosity.

Table S9. Viscosity of the different xanthan gum (XG) solutions. Each specified quantity of XG was mixed with 25 L of water. The viscosity was measured at a shear rate of $6s^{-1}$.

XG concentration	XG mass	Viscosity η (Pa·s)
1%	0.25 kg	3.45 ± 0.06
2%	0.5 kg	6.68 ± 0.23
3%	0.75 kg	9.88 ± 0.26
4%	1.0 kg	12.63 ± 0.39
5%	1.25 kg	18.18 ± 0.50
6%	1.5 kg	25.56 ± 0.86



Joint	Resting position (rad)	Position range (rad)	Max speed (rad/s)
j1	0.0	[-1.7016, 1.7016]	2.0
j2	0.2155	[-2.147, 1.047]	2.0
j3	0.0	[-3.0541, 3.0541]	2.0
j4	1.3349	[-0.05, 2.618]	2.0
j5	0.0	[-3.059, 3.059]	4.0
j6	0.0	[-1.5707, 2.094]	4.0

● twisting joint ● rotational joint

Fig. S12. Robot arm at resting position. The motor tasks involved six DOFs (joints j1 to j6) of the robot right arm. The depicted arm position illustrates the robot resting position.

Movie S1. Robot motor behaviors enabled by the cocontraction range. The video includes the robot reacting to external perturbations in the form of: physical interactions with other robot; reaction to an attached payload; physical interactions with human operator.

Movie S2. Active modulation of cocontraction. The video includes the robot performing in different unknown terrains in compliance priority mode and robustness priority mode, and also the differentiation of behavior depending on the direction of motion. The video displays the end-effector Cartesian path and the joint cocontraction as the robot end-effector is submerged in the viscous fluids.