

ADAPTIVE AND ENERGY EFFICIENT WALKING IN A HEXAPOD ROBOT UNDER NEUROMECHANICAL CONTROL AND SENSORIMOTOR LEARNING

E.Carolin¹,R.Kasilakshmi²,M.Suganya³

Mrs.I.Juliet AnbuMalar⁴ M.Tech

Veronikacarolin7@gmail.com¹, kasilkshm@gmail.com², suganya13195@gmail.com³

anbumalarjuliet@gmail.com⁴

1,2, 3 –Final year student (ECE)

4 –Assistant professor of ECE department

1, 2, 3, 4 - Dr. G. U. Pope College of Engineering, Sawyerpuram, Thoothukudi (Dt).

Abstract—

The control of multilegged animal walking is a neuromechanical process, and to achieve this in an adaptive and energy efficient way is a difficult and challenging problem. This is due to the fact that this process needs in real time: 1) to coordinate very many degrees of freedom of jointed legs; 2) to generate the proper leg stiffness (i.e., compliance); and 3) to determine joint angles that give rise to particular positions at the endpoints of the legs. To tackle this problem for a robotic application, here we present a neuromechanical controller coupled with sensorimotor learning. The controller consists of a modular neural network for coordinating 18 joints and several virtual agonist–antagonist muscle mechanisms (VAAMs) for variable compliant joint motions. In

addition, sensorimotor learning, including forward models and dual-rate learning processes, is introduced for predicting foot force feedback and for online tuning the VAAMs' stiffness parameters. The control and learning mechanisms enable the hexapod robot advanced mobility sensor driven-walking device (AMOS) to achieve variable compliant walking that accommodates different gaits and surfaces. As a consequence, AMOS can perform more energy efficient walking, compared to other small legged robots. In addition, this paper also shows that the tight combination of neural control with tunable muscle-like functions, guided by sensory feedback and coupled with sensorimotor learning, is a way forward to better understand and solve adaptive coordination problems in multilegged locomotion.

Index Terms—

Bio-inspired robot control, legged locomotion, muscle model, variable impedance control.

I. INTRODUCTION

LEGGED animals are capable of adjusting their leg stiffness to accommodate surfaces of variable structural properties, thereby leading to adaptive and energy efficient locomotion. They also tune their leg stiffness to accommodate different gaits based on energetic cost. Neurophysiological studies have revealed that these behaviors arise from the interplays between the nervous systems and the musculoskeletal structures (i.e., muscles and body) of legged animals. These neuromechanical interactions govern how legged animals achieve adaptive locomotion on different surfaces. For example, cockroaches rely more on their musculoskeletal structures to move over a regular surface. But moving over a more difficult one, they need to resort to the integrations of their nervous systems and musculoskeletal structures. As Bernstein pointed out, the need to control many degrees of freedom (DOFs) is a characteristics of neuromechanical systems. In a cockroach (e.g., *Blaberus discoidalis*), for instance, there are 220 muscles controlling legs with at least 19 DOFs that contribute to its locomotion. Owing to this, modeling the cooperations within and between different functional components of neuromechanical systems in legged locomotion is a very challenging task. Along this paradigm, Full and Koditschek proposed a specific solution where two types of dynamic models (i.e., template and anchor) are used to model legged locomotion with many DOFs. An anchor is a representative model with detailed descriptions of neural circuits, muscles, and joints. Whereas, a template represents the simplest model of

locomotion by trimming away the detailed descriptions (e.g., muscles and joints) of the DOFs. Referring to the template, hexapod robots (i.e., robot hexapod (RHex) robots) were designed by Saranli et al. Each RHex robot having only six DOFs showed unprecedented mobility over different surfaces. Besides, they can also achieve energy efficient locomotion by exploiting passive variable compliant legs. For example, leg compliance of an RHex robot was manually tuned to accommodate its running speeds based on energetic cost. The RHex robot is the best example for a coordination architecture controlling faster movement (e.g., running) where mechanical properties (e.g., leg compliance) must be increasingly well tuned to adapt to different environments. In such a case, more feed-forward and decentralized control can suffice, since feedback control may not be effective due to noisy sensing. By contrast, slower movement (e.g., walking) can heavily count on sensing which allows for more adaptive movement. Similarly, here more feedback and a centralized coordination architecture will be utilized to control our hexapod robot advanced mobility sensor driven-walking device (AMOS) in a neuromechanical manner. Moreover, the modeling of RHex robots is no more than a template, since this template behavior was not embedded within a very detailed model (i.e., anchor). The anchor model is a representative model describing a nervous system, muscles, joints, and legs with many DOFs like in insects. Templates and anchors are more than “simple models” and “complex models.” There should be a natural embedding of the template behavior within the anchor. Therefore, an anchor (i.e., detailed) controller for adaptive and energy efficient physical robot locomotion that accommodates different gaits and surfaces remains an important and unresolved problem in a neuromechanical

context. To solve this problem, we propose a neuromechanical controller coupled with sensorimotor learning for active tuning of passive properties (e.g., stiffness parameters) of the muscle-like components driving the joints during locomotion. Classical neural control and variable compliance control are generalized and integrated into our neuromechanical controller consisting of a modular neural network (MNN) and several virtual agonist-antagonist mechanisms (VAAMs). The proposed neuromechanical controller simplifies and integrates neural control and Hill's muscle model typically adopted in theoretical neuromechanical models, such that the controller is computationally implemented on physical legged robots with many DOFs. Such integration facilitates more adaptive and energy efficient walking on challenging surfaces. For instance, the neuromechanical controller enables AMOS to achieve more energy efficient walking on the challenging surfaces, compared to the adaptive neural controller. Such energy efficient walking can be also achieved by passive or active compliance control. Passive compliance control is typically regarded as the integration of actuators and visco elastic mechanics. Such control, however, leads to structural and sensory complexities that cause bulky and energy-inefficient legged robots with many DOFs. By contrast, our neuromechanical controller solves these problems by using virtual muscle-like mechanisms (i.e., VAAMs), which can be applied to variable compliance control of small legged robots with many DOFs. Moreover, the integration of the VAAMs and a proximo-distal gradient results in more stable compliant locomotion, compared to classical active compliance control. Active compliance control typically requires force/torque sensing at each joint of legged robots and its control parameters are

often adjusted by hand or offline learning. The parameters have to be relearned or manually readjusted when walking on different surfaces. Therefore, self-adjusting compliance control on a physical legged robot with many DOFs remains an important and unsolved problem in a context of energy efficient walking on different surfaces, which this paper tries to address in a more efficient way. As a result, the work enables our hexapod robot AMOS to achieve more energy efficient walking [i.e., lower costs of transport (COTs), than other small legged robots (less than 8 kg), when proper gaits are chosen for walking over different surfaces. Christo Ananth et al. [3] discussed about a system, GSM based AMR has low infrastructure cost and it reduces man power. The system is fully automatic, hence the probability of error is reduced. The data is highly secured and it not only solve the problem of traditional meter reading system but also provides additional features such as power disconnection, reconnection and the concept of power management. The database stores the current month and also all the previous month data for the future use. Hence the system saves a lot amount of time and energy. Due to the power fluctuations, there might be a damage in the home appliances. Hence to avoid such damages and to protect the appliances, the voltage controlling method can be implemented. These surfaces include loose surfaces (e.g., fine gravel and coarse gravel), an elastic surface (e.g., sponge), and a muddy surface (e.g., grassland). This paper is an extension. The MNN was developed to only generate reactive behaviors and omni directional walking. We proposed the muscle like mechanisms (i.e., VAAMs) for robotic compliant joint control. There, we also investigated muscle-like functions and how to vary compliant joint motions via the manual adjustment of the stiffness parameters of the passive elements of the VAAMs, but excluded

walking. We integrated the MNN and VAAMs into neuromechanical control of energy efficient hexapedal walking, in which there is only one specific gait with the manual adjustment of the stiffness parameters of the VAAMs for walking on different surfaces and surface classification. The new contributions of the work, thus, include the following.

1) Development of sensorimotor learning for self-adjusting the stiffness parameters of the VAAMs that adapts hexapedal walking to nine insect-like gaits and four challenging surfaces.

2) Combination of the muscle-like mechanisms (VAAMs) and sensorimotor learning showing a simple but effective way to achieve adaptive variable compliant joint motions without complex sensory systems and (physical) compliant components.

3) Integration of VAAMs with a proximo-distal gradient to remove locomotor instabilities under active compliance control.

4) Neuromechanical control coupled with sensorimotor learning providing a way forward to model and control adaptive and energy efficient legged locomotion with many DOFs.

5) Investigations on adaptive leg compliance for different gaits and energy efficient walking on different surfaces.

6) A better understanding of interactions between neuromechanical control, sensorimotor learning, sensory force feedback, and the environment under adaptive locomotion.

II. NEUROMECHANICAL CONTROLLER COUPLED WITH SENSORIMOTOR LEARNING

We include the feed-forward and feedback pathways into our neuromechanical controller. For the feed forward pathways, the controller not only consists of feed forward control via descending commands (i.e., S , N_i , and O_i) from a neural circuit to muscle-like components and body

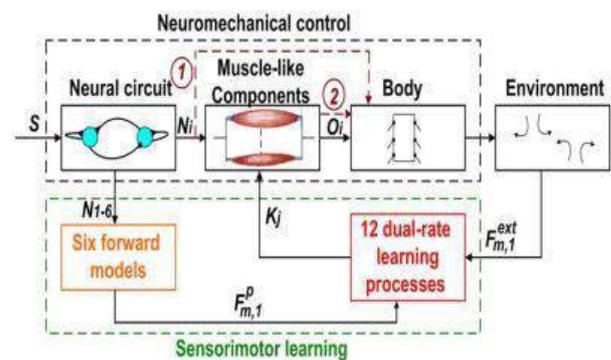


Fig. 1. Neuromechanical control coupled with sensorimotor learning applied to the hexapod robot AMOS.

Via neural outputs N_i ($i=1,2,\dots,18$), a neural circuit activates the muscle-like components that generate position commands (i.e., O_i) to move the leg joints of AMOS. The legs then interact with the environment, which produces force feedback (i.e., $F_{m,1}^{ext}$) ($m = 1,2,\dots,6$). Besides, six forward models predict expected force feedback (i.e., $F_{m,1}^p$) of the legs based on the outputs of the neural network. Using $F_{m,1}^{ext}$ and $F_{m,1}^p$ as the inputs, 12 dual-rate learning processes actively tune 12 stiffness parameters (i.e., K_j , $j = 7,8,\dots,17,18$) of muscle-like components driving 12 joints. There are three ways of generating position commands O_i driving the joints: feed-forward neural control for proximal joints combining feed-forward neural control and tendon-like compliance for intermediate joints, and tendon-like compliance for distal joints. Interestingly, these three ways are comparable to a proximo-distal gradient mechanics, but also includes six forward models for predicting force sensing (i.e., $F_{m,1}^p$) of the

six legs. In the feedback pathway, there is force sensing (i.e., Fext m,1) at the end effectors of the legs. Using Fp m,1 and Fext m,1 as the inputs, 12 dual-rate learning processes can actively tune the stiffness parameters (i.e., K_j) of the muscle-like components driving 12 joints of the legs. This leads to variable compliant leg motions over different surfaces. Actively tuning mechanical properties (e.g., joint stiffness) is an important characteristic of animal locomotion.

For example, the tunable mechanical properties of insect legs can help its locomotion over rough terrain. In addition to neuro mechanical interactions, studies of leg muscle architecture and function suggest that a proximo-distal gradient of muscle function and neural control exists, which reflects different control strategies for the joint. Following the gradient, proximal joints are under feed-forward neural control, and are rarely sensitive to changes in loading during stance. By contrast, distal joints are more sensitive to loading, and are basically driven by tendons. This proximo-distal gradient enhances locomotor stability of legged animals on rough terrain. Based on the gradient, the contractile elements (CEs) and passive elements of the VAAMs emulate feed-forward neural control and compliance of tendons respectively.

The proximal joints [i.e., thoraco coxal (TC) joints] of the hexapod robot are coordinated only by neural outputs. Whereas its distal joints [i.e., femur tibia (FTi) joints] are driven only by the passive elements emulating the compliance of tendons. The experimental results show that such a setup enables the hexapod robot to achieve more stable walking on rough surfaces (e.g., gravels). The setup enhances stability of legged robot locomotion under active compliance control which generally leads to locomotor instabilities.

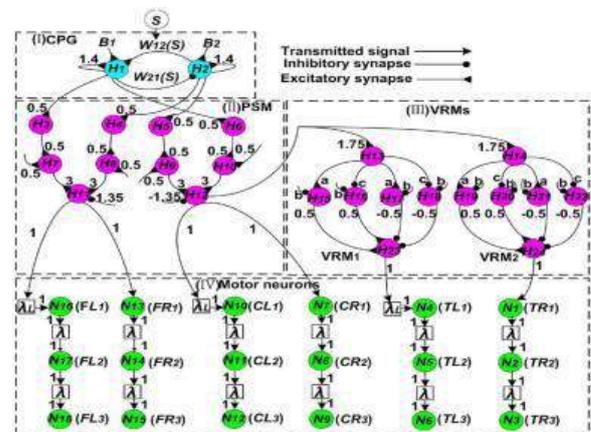


Fig. 2. MNN. There are three different neuron groups: input neuron (S), hidden neurons (H1–24), and output neurons (N1–18). The input neuron is used to control walking patterns of the hexapod robot AMOS. The hidden neurons are divided into three modules: CPG, PSM, and VRMs, which have different functionalities (see texts for details). All connection strengths together with bias terms are indicated by the small numbers except some parameters of the VRMs ($a = 1.7246$, $b = -2.48285$, and $c = -1.7246$). Delays λL and λ between output neurons are set to 48 and 16 time steps, respectively. Abbreviations are: TR(L)1,2,3 = TC joints of the right(left) front, middle, hind legs, CR(L)1,2,3 = CTr joints of the right(left) front, middle, hind legs, FR(L)1,2,3 = FTi joints of the right(left) front, middle, hind legs. Abbreviations are: R(F,M,H) = right (front, middle, hind) leg, L(F,M,H) = left (front, middle, hind) leg. In the following, we describe three above introduced components of our system. 1) A neural circuit which produces the commands to coordinate joint motions and to change gaits based on energetic cost. 2) Biomechanical components consisting of muscle-like components and a bio-inspired body. Walking systems particularly require an adaptive muscle model where its parameters can be easily and quickly tuned to achieve proper compliant joint motions. 3) Sensorimotor

learning which can predict sensory consequences of actions and actively tune compliance of joint motions; thereby enabling walking systems to accommodate different gaits and deal with different surfaces. The details of each component are described below.

Neural Circuit: Modular Neural Network

Our MNN is a biologically-inspired hierarchical neural controller. The MNN generates signals for interleg and intraleg coordination of the six-legged robot AMOS. Each leg has a TC joint allowing forward and backward motions, a coxa trochanteral (CTr) joint allowing elevation and depression motions, and an FTi joint allowing extension and flexion motions. The MNN consists of a central pattern generator (CPG) a phase switch module (PSM) and two velocity regulating modules (VRMs). All neurons of the MNN are modeled as discrete-time, nonspiking neurons. The activation H_i of each neuron develops according to

$$H_i(t) = \sum_{j=1}^m W_{ij} o_j(t-1) + B_i, \quad i = 1, \dots, m \quad (1)$$

where m denotes the number of units, B_i is an internal bias term (i.e., stationary input) to neuron i , and W_{ij} is the synaptic strength of the connection from neuron j to neuron i . The output o_i of every neuron of the MNN is calculated using a hyperbolic tangent (tanh) transfer function, i.e., $o_i = \tanh(H_i) \in [-1, 1]$. The weights W_{ij} are manually designed, except weights a , b , and c which are obtained by back-propagation learning [see Fig. 2(III)]. More details of determining the weights W_{ij} , The CPG consists of only two neurons with full connectivity. where B_1 and B_2 are set to 0.01. The weights W_{12} and W_{21} are given by

$$W_{12}(S) = 0.18+S, \quad W_{21}(S) = -0.18-S \quad (2)$$

where $S \in [0.01, 0.18]$ is the modulatory input determining the speed of the legs, which increases with increasing S . The PSM is a generic feed-forward network consisting of three hierarchical layers with ten hidden neurons (i.e., $H_3 - H_{12}$). The outputs of the PSM are projected to the FTi [i.e., $F(R,L)(1,2,3)$] and CTr [i.e., $C(R,L)(1,2,3)$] motor neurons as well as to the neurons H_{13} and H_{14} of the two VRMs. The VRMs are feed-forward networks projecting their outputs to the TC motor neurons $T(R,L)(1,2,3)$. Delays λ_L and λ between the motor neurons are fixed. The outputs $N_1 - N_{18}$ of the motor neurons are used to activate the muscle-like components to drive AMOS's legs. Here, we show how $N_1 - N_{18}$ enable the legs to perform a fast wave gait. In addition, nine gaits are achieved by changing the modulatory input S of the MNN. More details of the MNN.

Biomechanical Components

1) Muscle-Like Component—Virtual Agonist–Antagonist Mechanism:

The VAAM consists of a pair of agonist and antagonist mechanisms. It produces active and passive forces using its CEs and parallel elements (PEs). A physical joint is driven by a VAAM (i.e., M_1 and M_2). Virtual means that the joint, physically driven by a standard servo motor, imitates muscle-like behaviors as if it were driven by a pair of physical agonist and antagonist muscles. The joint actuation relies on the CEs, while the PEs govern joint compliance. The parallel elements are modeled as spring-damper systems in terms of a Voigt muscle mode.

The active forces produced by the CEs are approximated by the product of the neural activity N_j and the activity strengths $i(1,2)$. More details of mathematically modeling the

PEs and CEs. We apply Euler’s law to the rotation of the joint P. The motion equation of the joint P is given by

$$I\ddot{\theta} = \underbrace{f^{ext} \sin(\theta)L}_{\text{torque by } f^{ext}} + \left[\underbrace{rN_j}_{\text{torque by CE}_{(1,2)}} - \underbrace{r(2K\dot{\theta}r + 2D\ddot{\theta}r)}_{\text{torque by PE}_{(1,2)}} \right] \quad (3)$$

FIG. 3. VAAM for joint control interacting with the ground surface. (a) Physical joint P is driven by a VAAM (i.e., M1 and M2) with the lengths L1 and L2. The interaction results in an external force f_{ext} , which drives the joint P with radius r via the shank with length L . f_{ext} is sensed by a force sensor (i.e., O), and f_{\perp} is the amount of f_{ext} directly perpendicular to the position of the joint P. θ is the rotational angle of the joint P relative to the absolute frame Z. (b) Agonist and antagonist mechanisms consist of contractile and parallel elements (CE(1,2) and PE(1,2)). PE(1,2) are spring-damper systems producing passive forces. CE(1,2) generate active forces depending on the neural activity N_j and the activity strengths $i(1,2)$ (i.e., $i(1,2) \in [-1,1]$). The neural activity N_j is one of the outputs N_1-18 of the MNN. Equation (3) governs the angle θ of a physical joint driven by the VAAM that is activated by the output N_j ($j \in Z[1,18]$) of the MNN. The joint angle θ and joint velocity $\dot{\theta}$ in (3) are not from sensory

feedback but calculated using fourthorder Runge–Kutta. In principle, this bio-inspired compliant joint control approach (i.e., the VAAM) shares a connection to classical impedance control approaches in terms of spring-damper based compliance. However, it is a biological model where biological muscle functions (e.g., brakes) can be easily emulated by changing stiffness and damper parameters [i.e., K and D in (3)]. Here, through using sensorimotor learning (see Section II-E for details), K will be adjusted in an online manner while D will be fixed during walking. More advantages of the VAAM model are described.

2) Bio-Inspired Body—Hexapod Robot AMOS:

Here we use a hexapod robot as our experimental platform. It has six three-jointed legs and each leg emulates the morphology of a cockroach leg. Every leg has a TC joint allowing forward and backward motions, a CTr joint allowing elevation and depression motions, and an FTi joint allowing extension and flexion motions. Each joint is physically driven by a standard servo motor (i.e., HSR-5990TG). There is a force sensor (i.e., FS Series Force Sensor) used for detecting an analog force signal at each leg. A current sensor, installed inside the body of the hexapod robot, is used to measure the electrical current supplied to all motors of the robot. Here, the current sensor signal is used to calculate power consumption during walking. The sensory data are transmitted via an RS232 serial connection to an external PC on which the controller is

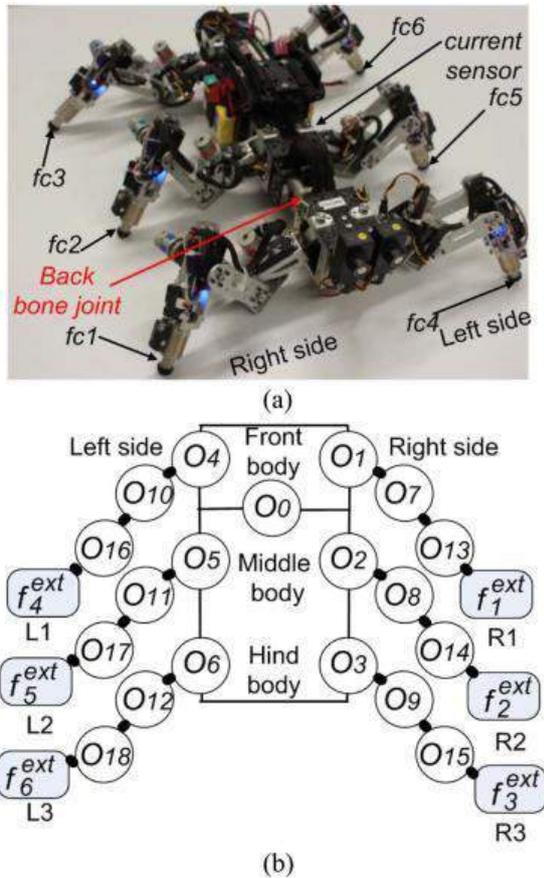


Fig. 4. Hexapod robot AMOS. Its three-jointed legs mimic leg morphology of an insect. (a) AMOS and its sensors. fc(1–6) are force sensors. (b) Outputs O0–18 controlling the 19 joints of AMOS when receiving analog signals $f_{ext} 1-6$, which are detected by the force sensors at the legs. Abbreviations are: L(1,2,3) = Left (Front, Middle, Hind)leg. R(1,2,3) = Right (Front, Middle, Hind) leg. implemented. The final motor commands of the controller are sent to the robot also via the serial connection.

Neuromechanical Control: Combining Neural Circuit and Biomechanical Components

The outputs $O1-18 \in [-1,1]$ of the neuromechanical controller are linearly scaled and transmitted to control the positions of the standard servo motors driving the 18 joints of

the hexapod robot in supplementary material]. Note that the command O0 here is set to a constant value (i.e., $O0 = 0$) for controlling the backbone joint to the middle position. For joint control (i.e., O1–18), different control strategies are applied to swing and stance phases, like virtual model controllers.

Swing Phase: When a leg is in swing phases (i.e., $f_{ext} i = 0, i = 1,2,\dots,5,6$), the outputs $N(i,i+6,i+12)$ of the MNN are linearly transformed into the outputs $O(i,i+6,i+12)$ controlling the TC, CTr, and FTi joints. $O(i,i+6,i+12)$ satisfy

$$[O_i, O_{i+6}, O_{i+12}]^T = [0.4N_i, 0.15N_{i+6}, -0.02N_{i+12}]^T - [0.05, -0.86, 0.43]^T, i \in \mathbb{Z}_{[1,6]} \quad (4)$$

Note that the last values of the outputs $O(i,i+6,i+12)$ of the swing phase are kept and transferred to the initial joint angles of the following stance phase. This leads to smooth switches from swing to stance phases.

Stance Phase: The TC joint of the leg allowing only horizontal motion is not affected by the PEs of the VAAM since there is only detection of vertical foot force at the end effector of the leg. As a consequence, the TC joint is driven by the CEs of the VAAM that simulate feed-forward neural control. By contrast, the CTr and FTi joints, contributing to vertical motion of the leg, can be influenced by vertical foot force. Based on the VAAMs, we test nine possible setups (see Table I in supplementary material) to control the CTr and FTi joints in a physical simulator (i.e., LPZROBOTS simulator). The simulation results show that the setup S2 leads to coordinated movement and stable locomotion with the smallest body oscillation. The setup S2 is as follows: each TC joint (i.e., proximal joint) is purely controlled by the CEs of the VAAM (i.e., pure actuation), each CTr joint (i.e., intermediate joint) is governed by the CEs and PEs of the

VAAM (i.e., combination of actuation and compliance), and each FTi joint (i.e., distal joint) is driven by the PEs (i.e., PE1 and PE2) of the VAAM (i.e., pure compliance) (see more details in Fig. 5 of supplementary material). Interestingly, this setup also complies with a proximo-distal gradient revealed by biological studies on three-jointed leg locomotion. These studies show that proximal joints mainly act as actuation while distal joints serve as compliance in legged animal locomotion. Such passive compliance and active actuation make the VAAM different from virtual model control (VMC), which only contains a virtual passive elements (e.g., spring) attached to the robot. In contrast to VMC controllers. The VAAM not only includes virtual passive elements to produce passive forces, but also integrates virtual CEs that generate active forces driven by neural control. The VAAM control is, thus, more strongly bio-inspired by integrating neural control with muscle-like functions, compared to VMC controllers. As a result, the VAAM control enables AMOS to not only achieve more stable walking under active compliance control, but also easily emulate muscle-like functions (e.g., brakes and springs). The outputs O_{1-18} of the proposed neuromechanical controller are calculated as follows. All TC joints are controlled only by CE(1,2) of the VAAM. The matrix of the outputs of the TC motor neurons is $T_{6 \times 1} = [N_1, N_2, \dots, N_6]$. O_j is given by ($j \in Z[1,6]$)

$$O_j = 0.4T_{j,1} - 0.05. \quad (5)$$

The details of (5) can be seen in [21, Eq. (A.4)]. Each CTr joint is driven by PE(1,2) and CE(1,2) of the VAAM. The matrix $\theta_{26 \times 1}$ of the CTr angles is the sum of the Hadamard products

$$\begin{aligned} \dot{\theta}^{26 \times 1} = & F_{6 \times 1}^{act} \circ \left(L_2 \cos(\theta^{26 \times 1}) + \dot{V}_{16 \times 1} \right) \\ & + \left[\gamma C_{6 \times 1} - 2\gamma^2 (K^{26 \times 1} \circ \theta^{26 \times 1} \right. \\ & \left. + D^{26 \times 1} \circ \dot{\theta}^{26 \times 1}) \right]. \end{aligned} \quad (6)$$

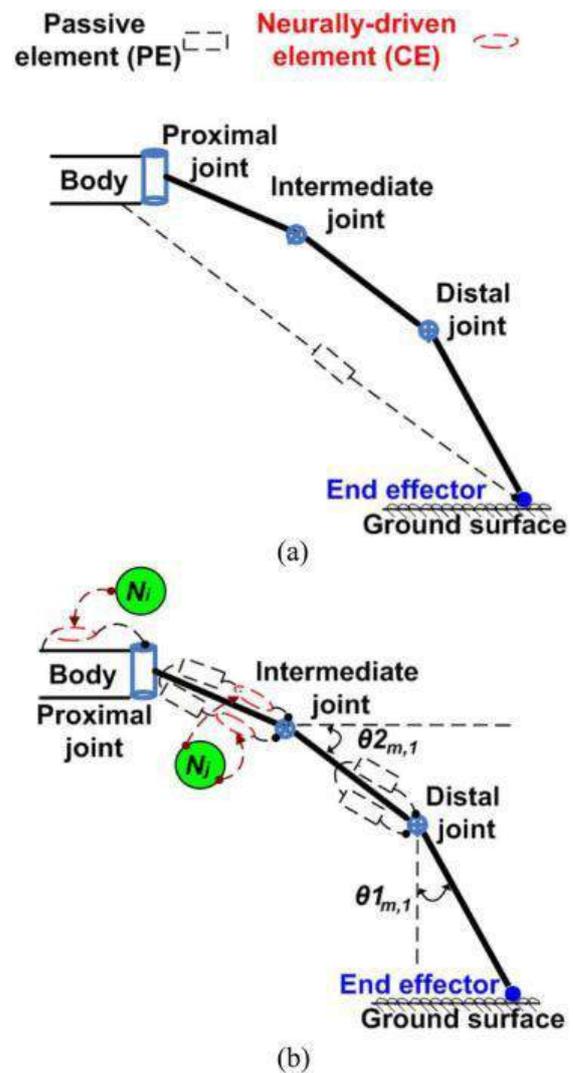


Fig. 5. Schematic of VMC and VAAM control. (a) Virtual model controller that only exploits a virtual passive element (e.g., spring) attaching the body to the end effector. (b) VAAM controller that uses the virtual CEs and passive elements. The controller is based on a proximo-distal gradient. The angles $\theta_{2m,1}$ of the CTr joints are linearly transformed into their output. O_j is given by ($j \in$)

$$O_j = -0.89 \theta_{2m,1} - 0.38, m = j - 6. \quad (7)$$

The details of (7) can be seen in [21, Eq. (A.5)]. Each FTi joint is driven only by PE(1,2) of the VAAM. The FTi angle matrix $\theta_{16 \times 1}$ is the sum of the Hadamard products (see [63, Eqs. (13)–(15)])

$$\theta_{16 \times 1} = F_{6 \times 1}^{act} \circ \sin(\theta_{16 \times 1}) L_1 - 2r^2 (K_{16 \times 1} \circ \theta_{16 \times 1} + D_{16 \times 1} \circ \dot{\theta}_{16 \times 1}). \quad (8)$$

The angles $\theta_{1m,1}$ [$m \in Z[1,6]$, see $\theta_{16 \times 1}$ in (8)] of the FTi joints can be linearly transformed into their outputs O_j (see more details in Fig. 5 of supplementary material). O_j is given by ($j \in Z$)

$$O_j = 0.928\theta_{1m,1} + 0.12, m = j - 12. \quad (9)$$

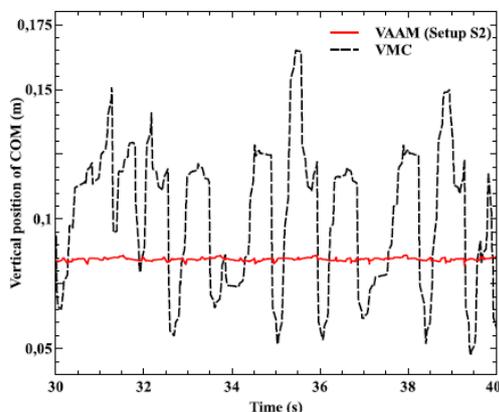


Fig. 6. Vertical positions of center of mass of the hexapod robot AMOS. The experiments are conducted in the physical simulator LPZROBOTS. The VAAM control (setup S2, see Table I in supplementary material) enables AMOS to walk stably (smaller body oscillations), compared to VMC.

Sensorimotor Learning for Adaptive Compliant Joint Motions

The adaptive compliant joint motions of AMOS are achieved by actively adjusting the stiffness parameters $K_{16 \times 1}$ and $K_{26 \times 1}$ [see

(6), (8)] of the passive elements of the VAAMs driving the FTi and CTr joints. Here, we apply sensorimotor learning for online adjusting $K_{16 \times 1}$ and $K_{26 \times 1}$ at every time step Δt (i.e., $\Delta t = 0.019$ s). For each leg, there are two dual-rate learning processes and a forward model for the CTr and FTi joints.

The forward models use the outputs i.e., $O_m(t)$ controlling the TC joints to predict foot force signals [i.e., $F_{pm,1}(t)$, $m=1,2,\dots,5,6$]. Specifically, $F_{pm,1}(t)$ will gradually increase to 1 when $O_m(t)$ is decreasing [see $F_{p4,1}(t)$ and $O_4(t)$]. $F_{pm,1}(t)$ is given by

$$F_{m,1}^p(t + \Delta t) = 0.2G_{m,1}(t) + 0.8F_{m,1}^p(t) \\ G_{m,1}(t) = \begin{cases} 1, & O_m(t + \Delta t) < O_m(t) \\ 0, & O_m(t + \Delta t) > O_m(t). \end{cases} \quad (10)$$

The matrix $e_{6 \times 1}(t)$ of errors between real and predicted foot force signals is

$$e_{6 \times 1}(t) = F_{6 \times 1}^{act}(t) - F_{6 \times 1}^p(t) \\ e_{6 \times 1}(t) = [e_1(t), e_2(t), \dots, e_5(t), e_6(t)]^T \quad (11)$$

where $F_{ext} 6 \times 1(t)$ is the matrix of the real foot force signals, i.e., $F_{ext} 6 \times 1(t) = f_{ext} 1-6(t)$ (see Fig. 4). $F_p 6 \times 1(t)$ is the matrix of the predicted foot force signals, i.e., $F_p 6 \times 1(t) = f_p 1-6(t)$. For reducing the errors, the processes adjust the stiffness parameters [e.g., $K_{14,1}(t)$] of the PEs driving the FTi and CTr joints in each leg [see Fig. 7(a)]. Each learning process consists of a fast learner and of a slow learner. Both learners are modeled as linear systems acting in parallel. The fast one learns compensating the error more quickly, is indicated by a higher learning rate, i.e., $B_{1f} > B_{1s}$. Whereas, the slow one retains previous states much better, is indicated by a high retention factor, i.e., $A_{1f} < A_{1s}$. Therefore, the

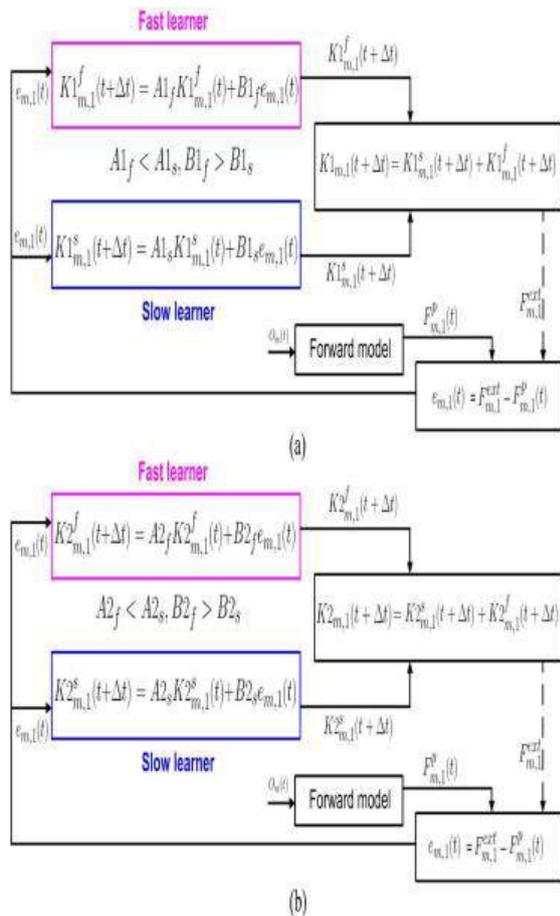


FIG.7. Sensorimotor learning for stiffness of the passive elements driving the FTi and CTr joints of the AMOSs legs. For each leg, there are two dual-rate learning processes for adjusting stiffness parameters (e.g., K14,1 and K24,1) by using expected and real foot forcesignals (e.g., Fp 4,1 and Fext 4,1). The expected foot force signal (e.g., Fp 4,1) is predicted by a forward model based on an output (e.g., O4) controlling the TC joint. Each dual-rate learning process consists of a fast learner and of a slower learner acting in parallel. (a) Dual-rate learning process for parameters K1m,1 and K2m,1. K1m,1 and K2m,1 (m = 1,2,...,6) are stiffness parameters stiffness parameters K1m,1. The parameters of the two learners are set as: A1f =0.59, A1s =0.992, B1f =0.378, and B1s =0.036. (b) Dual-rate learning for stiffness parameters K2m,1. The parameters of the two learners are set as:

$$A2f = 0.59, A2s = 0.992, B2f = 0.882, \text{ and } B2s = 0.084.$$

matrix $K16 \times 1(t)$ of stiffness parameters for the FTi joints is given by

$$\begin{aligned} K1_{6 \times 1}^f(t + \Delta t) &= A1_f K1_{6 \times 1}^f(t) + B1_f e_{6 \times 1}(t) \\ K1_{6 \times 1}^s(t + \Delta t) &= A1_s K1_{6 \times 1}^s(t) + B1_s e_{6 \times 1}(t) \\ K1_{6 \times 1}(t + \Delta t) &= K1_{6 \times 1}^f(t + \Delta t) + K1_{6 \times 1}^s(t + \Delta t) \end{aligned} \quad (12)$$

where $K1_{m,1}^f(t + \Delta t)$ are the outputs of fast learners, and $K1_{m,1}^s(t + \Delta t)$ are the outputs of slow learners. Note that the value of A1f and A1s and B1f and B1s are empirically chosen [see all values. Similarly, the matrix $K26 \times 1(t)$ of stiffness parameters for the CTr joints is given by

$$\begin{aligned} K2_{6 \times 1}^f(t + \Delta t) &= A2_f K2_{6 \times 1}^f(t) + B2_f e_{6 \times 1}(t) \\ K2_{6 \times 1}^s(t + \Delta t) &= A2_s K2_{6 \times 1}^s(t) + B2_s e_{6 \times 1}(t) \\ K2_{6 \times 1}(t + \Delta t) &= K2_{6 \times 1}^f(t + \Delta t) + K2_{6 \times 1}^s(t + \Delta t) \end{aligned} \quad (13)$$

where $K2_{m,1}^f(t + \Delta t)$ are the outputs of fast learners, and $K2_{m,1}^s(t + \Delta t)$ are the outputs of slow learners. Note that the value of A2f and A2s, and B2f and B2s are empirically chosen. Equations (12) and (13) are written in terms of time t different from formulated according to trial number n.

III. EXPERIMENTS

Sensorimotor Learning for Self-Adjusting Stiffness Parameters

For each leg, there are two learning processes coupled with a forward model for adjusting the stiffness parameters (e.g., K14,1 and K24,1). At the left front leg, for example, there are two outputs (i.e., K2f 4,1 and K2s 4,1) of fast and slow learners acting in parallel, which contribute to the stiffness parameter K24,1. One can see that the fast one learns K2f 4,1 more rapidly, which leads to smaller oscillations. By contrast, the slow one

retains $K_{2s\ 4,1}$ better, thereby leading to convergence. This is because the retention factor $A_{2f} = 0.59$ of the fast learner is lower than $A_{2s} = 0.992$ of the slow learner. Moreover, the fast learner is more sensitive to perturbations (i.e., stance phases) after learning compared to the slow learner. This is because the learning rate $B_{2f} = 0.882$ of the fast learner is higher than $B_{2s} = 0.084$ of the slow learner. The combination of the slow and fast learners enables the stiffness parameters (e.g., $K_{24,1}$) to achieve global convergence and local oscillatory stiffness response, which lead to stable and adaptive compliant hexapedal walking on challenging surfaces. Furthermore, the stiffness parameters (e.g., $K_{24,1}$) during swing phases

controlling the TC joint is applied to predict the foot force signal $F_{p\ 4,1}(t)$ [see (10)]. (b) Contact forces. $F_{ext\ 4,1}(t)$ and $F_{p\ 4,1}$ are the real and predicted contact forces. (c) Learning the stiffness parameter $K_{24,1}$. $K_{24,1}$ is the sum of the outputs (i.e., $K_{2f\ 4,1}$ and $K_{2s\ 4,1}$) of a fast learner and a slow learner using the error $e_{4,1}$ between $F_{ext\ 4,1}$ and $F_{p\ 4,1}$. The adjustment of stiffness parameter $K_{14,1}$ driving the FTi joint in the left front leg (d) O_{10} , and O_{16} are the outputs controlling the positions of the CTr and FTi joints in the left front leg. Swing phase are higher than the ones during stance phases. Since they (during the swing phases) are kept as the stiffness parameters from the previous stance phases. Note that sensorimotor learning is not applied to adaptively control the joints during swing phases, because only feedforward neural control [i.e., no stiffness parameters $K(1,2)m,1$] is used for joint control during swing phases. During stance phases, the stiffness parameters (e.g., $K_{24,1}$) initially decrease and only later increase. This is because the muscle-like mechanisms (i.e., VAAMs) initially soften the joints to absorb the impact of external loads, and later stiffen them to obtain more force for foothold and moving forward. Similarly, the PEs of the VAAMs also soften and stiffen the FTi joints during stances phases. In other words, the VAAMs stiffen joints when the external load increases (i.e., stance phases). This property of the VAAMs is comparable to that of biological muscles, which become stiff when the external load increase. Note that AMOS had difficulties to walk on all experimental surfaces when only fast or slow learners were used to tune stiffness parameters $K_{16 \times 1}$ and $K_{26 \times 1}$. This is because the slow or fast learners allow only for global convergences or local oscillatory stiffness responses. Whereas combining the slow and fast learners, the dual-rate learners enable $K_{1m,1}$ and $K_{2m,1}$ to achieve global

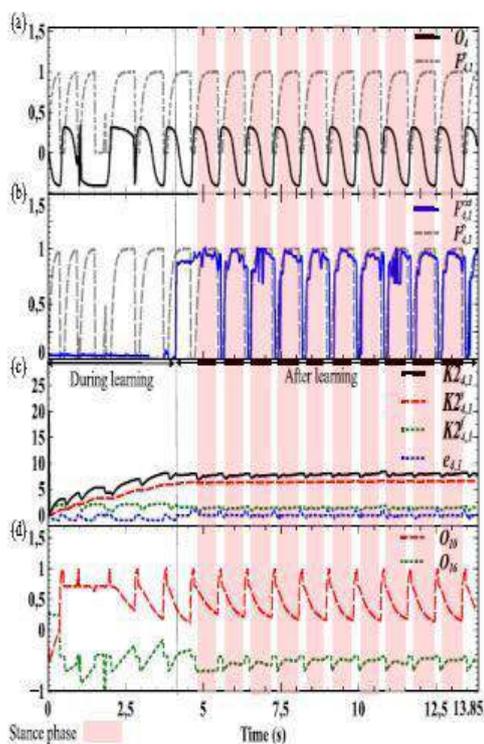


Fig. 8. Sensorimotor learning for adjusting stiffness parameter $K_{24,1}$. Here the gait is fast caterpillar (i.e., the modulatory input $S = 0.10$). (a) Forward model. The output $O_4(t)$

convergences and local oscillatory stiffness responses thereby leading to stable and adaptive walking on different surfaces. Moreover, the ranges of the stiffness parameters $K_{1m,1}$ and $K_{2m,1}$ vary between hind and nonhind legs. Lower $K_{1(3,6),1}$ and higher $K_{2(3,6),1}$ press the hind legs more down, which enhance locomotion stability, compared to the front and middle legs. This is because the mass of AMOS mainly concentrates on its hind part. Furthermore, the values of $B(1,2)_f$ and $B(1,2)_s$ are empirically chosen to produce proper stiffness parameters $K_{1m,1}$ and $K_{2m,1}$ which lead to appropriate (e.g., smooth) compliant joint motions of AMOS. For example, the compliant CTr joint motions are smoother when the parameters $K_{2m,1}$ of their driving VAAMs are self-adjusted.

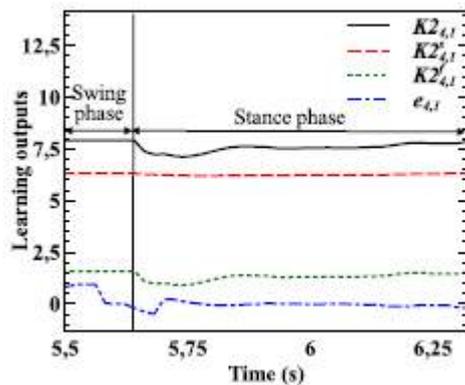


Fig.9. Sensorimotor learning for adjusting stiffness parameter $K_{24,1}$ during a swing and stance phases.

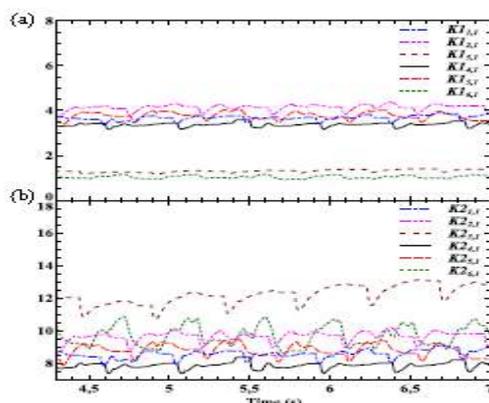


Fig. 10. Stiffness parameters $K_{11-6,1}$ and $K_{21-6,1}$ after learning. Here the gait is fast caterpillar (i.e., the modulatory input $S=0.10$). Stiffness parameters (a) $K_{11-6,1}$ of the VAAMs that drive the FTi joints and (b) $K_{21-6,1}$ of the VAAMs that drive the CTr joints.

Adaptive Leg Compliance for Different Gaits

Actively adjusting stiffness parameters $K_{16 \times 1}$ and $K_{26 \times 1}$ allows AMOS to accommodate different gaits. AMOS, for instance, walked on fine gravel when slow wave (i.e., $S = 0.02$) and fast caterpillar (i.e., $S = 0.10$) gaits were chosen. One can see that

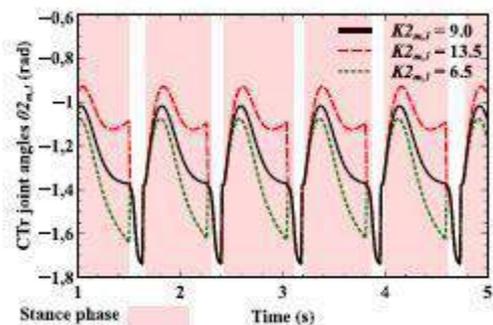


Fig. 11. Smoothness of the compliant CTr joint motions that varies with the stiffness parameters $K_{2m,1}$. Note that changing initial joint angles $\theta_{2m,1}$ does not affect the smoothness of the compliant CTr joint motions.

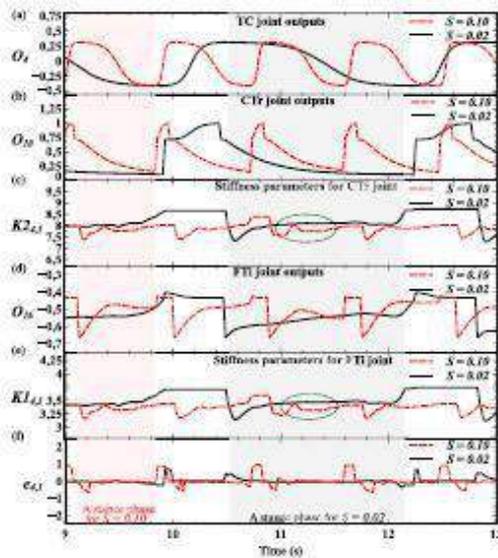


Fig. 12. Adjustments of $K_{14,1}$ and $K_{24,1}$ for different gaits. AMOS walked on fine gravel where its gait was chosen as slow wave (i.e., $S = 0.02$) and fast caterpillar (i.e., $S=0.10$) gait, respectively. Sensorimotor learning enables AMOS to self-adjust stiffness parameters $K_{14,1}$ and $K_{24,1}$ for the left front leg. (a) TC joint outputs O_4 . (b) CTr joint outputs O_{10} . (c) Stiffness parameters $K_{24,1}$ determine the compliance of CTr joint motions of the left front leg. (d) FTi joint outputs O_{16} . (e) Stiffness parameters $K_{14,1}$ determine the compliance of FTi joint motions of the left front leg. (f) Foot contact force errors $e_{4,1}$.

AMOS softens and stiffens its CTr and FTi joints during stance phases, no matter which gait is chosen. Moreover, the slow wave gait enables CTr and FTi joints to achieve stiffer motions that result from larger $K_{1m,1}$ and $K_{2m,1}$ compared to the fast caterpillar gait. That is, AMOS stiffens the legs during stance phases when the speed of its leg motion is reduced from the fast gait to the slow gait. This result is comparable to the finding of physiological experiments.

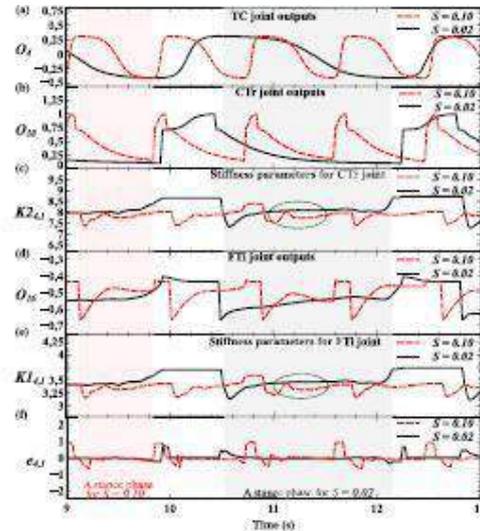


Fig. 13. Adjustments of $K_{14,1}$ and $K_{24,1}$ for different surfaces. An intermixed gait (i.e., modulatory input $S = 0.12$) was chosen for AMOS to walk on fine and coarse gravel, respectively. Sensorimotor learning enables AMOS to self-adjust stiffness parameters $K_{14,1}$ and $K_{24,1}$ for the left front leg. (a) TC joint outputs O_4 . (b) CTr joint outputs O_{10} . (c) Stiffness parameters $K_{24,1}$ determine the compliance of the CTr joint motions of the left front leg. (d) FTi joint outputs O_{16} . (e) Stiffness parameters $K_{14,1}$ determine the compliance of the FTi joint motions of the left front leg. (f) Foot contact force errors $e_{4,1}$. Low speed animals walk by vaulting stiffer legs. Conversely, AMOS softens its legs when the speed of its leg motion is increased from the slow gait to the fast one. This finding may reflect a control strategy of polyped (i.e., > two legs) locomotion where polyped systems soften the legs owing to energy efficiency requirements. Our experimental results also show that the fast caterpillar gait (i.e., $S = 0.10$) allows AMOS to achieve softer leg motions which lead to more energy-efficient locomotion on all experimental surfaces compared to the slow wave gait (i.e., $S = 0.02$). Note that errors during swing phases result from delayed feedback, which makes the

phase differences between the real and predicted forces. However, in this paper, the stiffness parameters are adjusted only to reduce the errors during stance phases. The large errors during swing phases are currently ignored for adjusting stiffness parameters. During the swing phases, the stiffness parameters are kept fixed as the stiffness parameters from the previous stance phases.

Adaptive Leg Compliance for Walking on Different surfaces

Actively adjusting stiffness parameters $K_{16 \times 1}$ and $K_{26 \times 1}$ also leads to adaptive locomotion on different surfaces, for example, when an intermixed gait (i.e., modulatory input $S=0.12$) was chosen for AMOS to walk on fine and coarse gravel, respectively. On these two surfaces, AMOS joints receive the same outputs of the MNN. One can see that the TC joint motions of the left front leg are the same because they are controlled only by feed-forward neural control (i.e., without passive elements). By contrast, CTr and FTi joint motions are different during stance phases when AMOS walks on fine and coarse gravel, respectively. This is because TC, CTr, and FTi joints act with different roles (i.e., compliance or actuation) for controlling leg motions in stance phases. Moreover, we can see that the CTr and FTi joints are stiffer [i.e., higher $K_{14,1}$ and $K_{24,1}$ values]. when AMOS walked on coarse gravel, compared to fine gravel. This makes the legs penetrate more deeply, but also extend more widely into the coarse gravel.

Energy Efficient Walking

In the previous sections, we show that the proposed neuromechanical controller coupled with sensorimotor learning enables AMOS to produce coordinated and variable compliant joint motions that accommodate different gaits

and surfaces. For each surface, nine gaits were chosen by changing the modulatory input S of the MNN. The variable compliant joint motions lead to different energy efficiencies of AMOS walking on fine gravel, coarse gravel, elastic sponge (stiffness 0.523 kN/m), and grass land. Typically, the energy efficiency is measured by COT (i.e., specific resistance) as

$$COT = \frac{P_{avg}}{mgv_{avg}}, v_{avg} = \frac{d}{t} \quad (14)$$

where P_{avg} is average power consumption. mg is the weight of AMOS, i.e., $mg = 52.974$ N. v_{avg} is its average forward speed when AMOS walks a distance d using time t . For each gait, we repeatedly ran the hexapod robot on each surface until ten successful runs were obtained. For each successful run, the average power consumption P_{avg} is calculated based on the electrical current supplied to all motors of AMOS, which is measured by a current sensor. Low COT corresponds to more energy efficient walking. COTs when AMOS walked on the four VVC surfaces using the nine gaits. One can see that AMOS achieves more energy efficient walking by using gaits with intermediate leg speeds, compared to a slower leg speed (i.e., modulatory input $S = 0.02$, the slow wave gait) or a faster leg speed (i.e., $S = 0.18$, the fast tripod gait). Moreover, different gaits let AMOS consume different energetic costs. For instance, the slow intermixed gait (i.e., $S=0.12$) enables AMOS to achieve more energy efficient walking on fine gravel while the fast intermixed gait (i.e., $S = 0.14$) is an efficient gait for AMOS walking on coarse gravel.

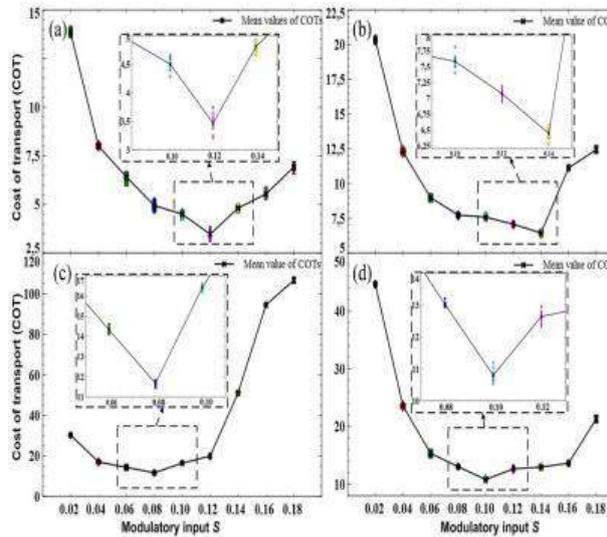


Fig. 14. Energy efficiencies of AMOS walking on different surfaces using different gaits. The energy efficiency is measured by COT [i.e., specific resistances]. Lower COT corresponds to more energy efficient locomotion. Nine gaits were chosen for AMOS walking over each experimental surface. (a) COTs on fine gravel. The slow intermixed gait (i.e., $S=0.12$) enables AMOS to achieve more energy efficient walking. (b) COTs on coarse gravel. The fast intermixed gait (i.e., $S=0.14$) is more energy-efficient for its walking on this surface. (c) COTs on elastic sponge. The slow caterpillar gait (i.e., $S = 0.08$) is the optimizer gait. (d) COTs on grass land. The fast caterpillar gait (i.e., $S = 0.10$) allows AMOS to achieve more energy efficient walking.

circuits. (b) RHex. (c) Harvard ambulatory microrobot2. d) Dynamic autonomous sprawled hexapod (e) Gregor I. (f) AMOS.

The slow (i.e., $S = 0.08$) and fast (i.e., $S=0.10$) caterpillar gaits make AMOS achieve more energy efficient walking on elastic sponge and grass land, respectively. Integrating neuromechanical control and sensorimotor learning, the adaptive neuromechanical controller enables AMOS to achieve adaptive compliant walking, which effectively accommodates different gaits and surfaces. Such walking is achieved by online adjusting stiffness parameters $K_{16 \times 1}$ and $K_{26 \times 1}$ of the passive elements driving the FTi and CTr joints. Note that all damper parameters $D(1,2)m,1$ were set to 1.0 in all experiments chosen by trial and error. As a result, the adaptive neuromechanical controller reduces COT of AMOSs walking to between 3.4 and 11.7s. Similarly, the adaptive neuromechanical controller allows for lower COT that corresponds to more energy efficient walking.

TABLE I

TIMES OF STANCE PHASES AND DELAYS VARY WITH S

Modulatory input S	Time of stance phases (s)	Time of delays (s)
0.02	1.82	0.07
0.10	0.77	0.08
0.18	0.56	0.10

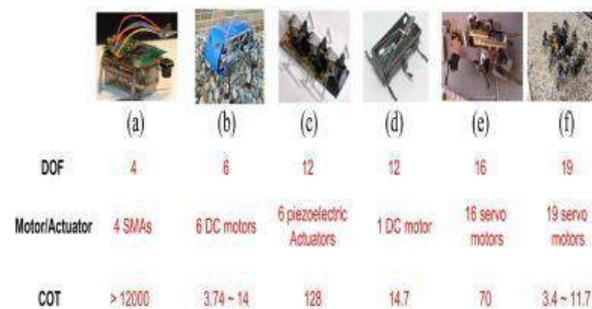


Fig. 15. COTs of small legged robots. (a) Millirobot enabled diagnostic of integrated

compared to the adaptive neural controller from. This is because the adjustable VAAMs of the adaptive neuromechanical controller produce high amplitude and smooth joint outputs during the stance phase which basically stiffen the legs and allow them to penetrate deeply into challenging surfaces (e.g., coarse gravel). By contrast, other neural controllers like the adaptive neural controller cannot achieve this due to the lack of muscle-

like mechanisms (e.g., VAAMs). Moreover, the adaptive neuromechanical controller makes AMOS achieve more energy efficient walking compared to other small legged robots (less than 8 kg).

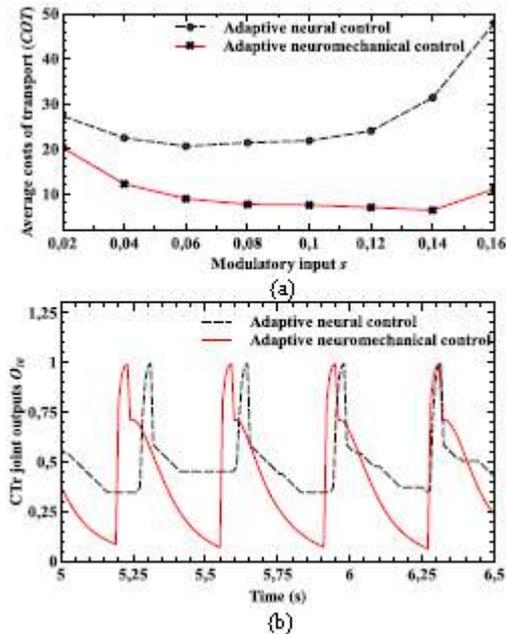


Fig. 16. COTs and CTr joint outputs under the adaptive neuromechanical and neural controllers. The experimental surface is the coarse gravel. (a) COTs. (b) CTr joint outputs O_{10} (with the fast intermixed gait, modulatory input $S=0.14$).

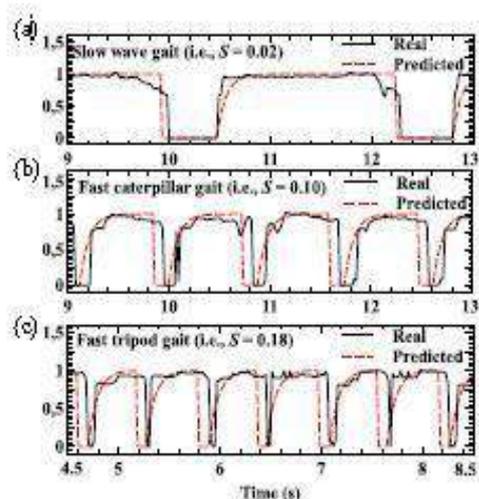


Fig. 17. Real and predicted contact forces at the left front leg. Here the slow wave (modulatory input $S = 0.02$), fast caterpillar ($S = 0.10$), and fast tripod ($S = 0.18$) gaits were chosen on fine gravel, respectively. All delay times are not larger than 0.10 s, although they increase with the increasing S . See the delay times between the real and predicted contact forces at Table I.

CONCLUSION

The proposed method enables the six-legged robot AMOS to achieve variable compliant joint motions with self-adjustments that accommodate different gaits and surfaces. These motions are generated by online tuning 12 stiffness parameters (i.e., $K_{1m,1}$ and $K_{2m,1}$, $m = 1,2,\dots,6$) of the muscle-like mechanisms (i.e., the VAAMs) driving 12 joints. This online tuning is achieved by sensorimotor learning only with force feedback at the end effectors of the legs. It is distinct from active compliance/impedance control which is achieved by using force/torque feedback at each joint of robotic system. Moreover, active compliance/impedance control often gives rise to unstable locomotion on tough terrain. Whereas our method utilizes the proximo-distal gradient to enhance locomotor stability on tough terrain (e.g., gravel). Our method also differs from passive compliance, which is characterized by physical passive components (e.g., springs and dampers). In addition, the proposed VAAM is a computational muscle model which can be easily applied to control physical legged robots. Thereby, the VAAM is also different from the Hill's muscle model where there are typically 16 parameters to be tuned, usually used in computer simulations. In conclusion, the main contribution of the work introduced here is that we present a way forward to understand and solve Bernstein's problem of how to efficiently control many

DOFs in multilegged locomotion tasks. This allows the six-legged robot AMOS to achieve adaptive and energy efficient walking without complex passive components or force/torque sensing systems. Due to the simple forward models of the proposed sensorimotor learning, the real contact forces lag (e.g., delay) behind the predicted contact forces. One can see that such delays slightly increase (see Table I) when the modulatory input S of the MNN increases. Thus, for future work, we will replace the simple forward models with advanced ones, like reservoir based online adaptive forward models. It has been shown that such reservoir-based forward models can accurately predict sensory feedback and are robust to variation of delayed feedback. Another option is to use another learning method, like deep learning.

REFERENCES

- [1] D. P. Ferris, M. Louie, and C. T. Farley, "Running in the real world: Adjusting leg stiffness for different surfaces," *Proc. Roy. Soc. London B Biol. Sci.*, vol. 265, no. 1400, pp. 989–994, Jun. 1998.
- [2] C. T. Farley, H. H. P. Houdijk, C. Van Strien, and M. Louie, "Mechanism of leg stiffness adjustment for hopping on surfaces of different stiffnesses," *J. Appl. Physiol.*, vol. 85, no. 3, pp. 1044–1055, Sep. 1998.
- [3] Christo Ananth, G. Poncelina, M. Poolammal, S. Priyanka, M. Rakshana, Praghash.K., "GSM Based AMR", *International Journal of Advanced Research in Biology, Ecology, Science and Technology (IJARBEST)*, Volume 1, Issue 4, July 2015, pp:26-28
- [4] A. E. Kerdok, A. A. Biewener, T. A. McMahon, P. G. Weyand, and H. M. Herr, "Energetics and mechanics of human running on surfaces of different stiffnesses," *J. Appl. Physiol.*, vol. 92, no. 2, pp. 469–478, Feb. 2002.
- [5] R. Full, C. Farley, and J. Winters, "Musculoskeletal dynamics in rhythmic systems: A comparative approach to legged locomotion," in *Biomechanics and Neural Control of Posture and Movement*, J. Winters and P. Crago, Eds. New York, NY, USA: Springer, 2000, pp. 192–205.
- [6] T. A. McMahon and G. C. Cheng, "The mechanics of running: How does stiffness couple with speed?" *J. Biomech.*, vol. 23, pp. 65–78, Feb. 1990.
- [7] J. Nishii, "Legged insects select the optimal locomotor pattern based on the energetic cost," *Biol. Cybern.*, vol. 83, no. 5, pp. 435–442, Nov. 2000.