MantisBot: A Platform for Investigating Mantis Behavior via Real-Time Neural Control

Nicholas S. Szczecinski(✉), David M. Chrzanowski, David W. Cofer, David R. Moore, Andrea S. Terrasi, Joshua P. Martin, Roy E. Ritzmann, and Roger D. Quinn

Case Western Reserve University, 10900 Euclid Ave., Cleveland, Ohio 44106, USA
nss36@case.edu

Abstract. We present Mantisbot, a 28 degree of freedom robot controlled in real-time by a neural simulation. MantisBot was designed as a 13.3:1 model of a male Tenodera sinensis with the animal's predominant degrees of freedom. The purpose of this robot is to investigate two main topics: 1. the control of targeted motion, such as prey-directed pivots and striking, and 2. the role of descending commands in transitioning between behaviors, such as standing, prey stalking, and walking. In order to more directly use data from the animal, the robot mimics its kinematics and range of motion as closely as possible, uses strain gages on its legs to measure femoral strain like insects, and is controlled by a realistic neural simulation of networks in the thoracic ganglia. This paper summarizes the mechanical, electrical, and software design of the robot, and how its neural control system generates reflexes observed in insects. It also presents preliminary results; the robot is capable of supporting its weight on four or six legs, and using sensory information for adaptive and corrective reflexes.

Keywords: Real-time neural control · Robot · Mantis

1 Introduction

Praying mantises make visually-guided posture adjustments to align themselves with prey [24], [4]. These adjustments require the animal to process visual information in the brain [23] to produce descending commands to low-level systems that control its body and legs, which execute these translations and rotations. We are especially interested in the central complex (CX), a midline neuropil in the arthropod brain, and its role in controlling behavior. The CX receives multimodal sensory information and communicates directly with premotor centers, suggesting that it plays a key role in controlling orientation and locomotion. Work with

N.S. Szczecinski—This work was supported by a NASA Office of the Chief Technologists Space Technology Research Fellowship (Grant Number NNX12AN24H).

S.P. Wilson et al. (Eds.): Living Machines 2015, LNAI 9222, pp. 175–186, 2015.
DOI: 10.1007/978-3-319-22979-9_18

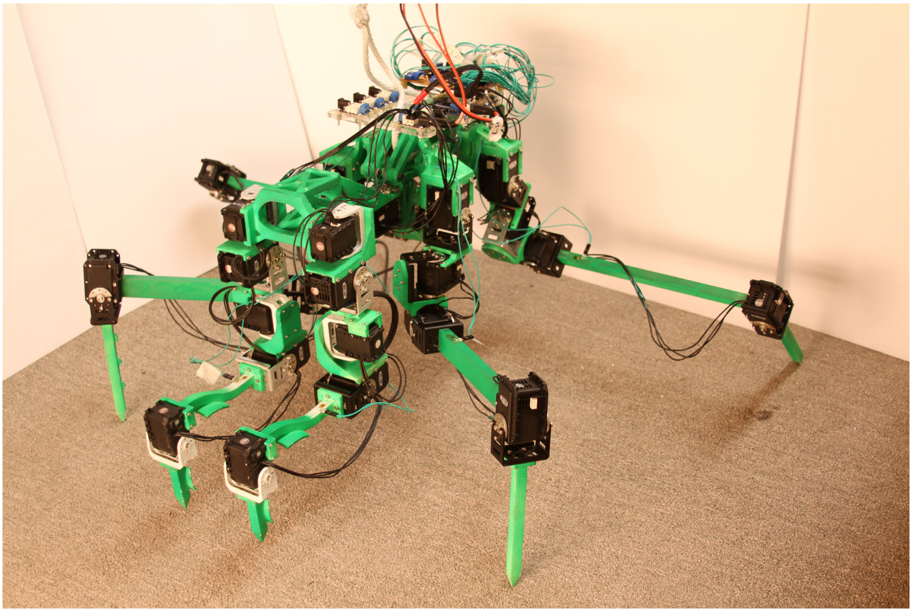

Fig. 1. MantisBot can support its own weight on four legs, with the front legs off the ground for striking. Communication with the robot is performed by the controller on the rear of the robot.

cockroaches suggests that activity in the CX precedes directed motion, specifically linear and angular velocity in the frontal and sagittal planes [9]. Mantises' directed motion facilitates studying the CX; therefore we have begun to investigate these questions in the praying mantis, one of the cockroach's closest living relatives [18], and an animal that exhibits deliberate, targeted motion as a predator.

In our previous work, we have constructed detailed neuromechanical models of insects to investigate questions about motor control [19] [20]. Building realistic models of animal locomotion requires the scientist to confront the details of network parameters and dynamics, an approach that has led to improved understanding of how animals generate rhythms for walking [17] [6] and modify them to change direction [13]. While neuromechanical software simulations are useful investigative tools, robots offer a more physically realistic way to explore animal control strategies [14]. Phenomena like ground contact and body strain are important details of controlling motion, yet they are difficult to model. In addition, the real world is noisy, and while noise can be modeled in simulation, a real-world environment is an excellent test for an experimental controller. Therefore, MantisBot was developed to be controlled by the AnimatLab Robotics Toolkit. This makes the transition from simulation to robot straightforward.

Many other robots have served as models by which to explore animal control systems, and only a few are mentioned here for brevity. ROBOT II, modeled after the stick insect and controlled by a finite state machine, was one of the first to implement leg reflexes based on insect behavior [8]. One of the most complete robotic models of insect behavior is WALKNET, which is a heterarchical artificial neural network that replicates behavioral data from stick insects [15]. WALKNET is used to control HECTOR, perhaps the most sophisticated biologically-inspired hexapod robot [16]. The robot SCORPION explores rhythm generation and reflexes through a neural system [7], including abstracted CPGs. Our goals for MantisBot are parallel to those for these other robots, except that we seek to explore how neural dynamics themselves affect the control of posture, reflexes, and rhythm. As such, we model the animal's control system as one hierarchical network of nonspiking neurons and synapses.

MantisBot is a robotic test platform for neural controllers that model insects' nervous systems, like those in our previous work [19] [20]. It mimics the anatomical proportions and kinematics of a male praying mantis *Tenodera sinensis*, and is controlled by a real-time neural simulation implemented with the AnimatLab Robotics Toolkit. This paper describes the robot, explores design decisions that make it like the animal, and explains the control system. Results are also presented from preliminary experiments, in which MantisBot uses sensory signals to coordinate its joints and exploits CPG dynamics to produce reflexive correction steps.

2 Robot Hardware

2.1 Mechanical and Electrical Design

MantisBot is actuated by Robotis Dynamixel MX-64T and smaller AX-12 smart servo motors (Robotis, South Korea). Each unit can measure position, mechanical load and temperature, and possesses its own microcontroller for communication. Our experiments revealed that MX-64Ts can output sufficient torque at stall, while only weighing 1.24 N.

Motors are controlled by an Arbotix-M (Vanadium Labs LLC, New York), an Arduino-compatible microcontroller based on the ATMEGA644p. Conveniently, chains of Dynamixels can be plugged into the TTL connectors built into the board, providing power and communication. In addition, this board is supported by the AnimatLab Robotics Toolkit, which provides low-level controls for MantisBot. We power both the servos and the Arbotix-M with a 12 VDC 83 A power supply. To avoid running all motors' current through the Arbotix-M, some motors plug directly into the microcontroller while others plug into a power hub, which when interfaced with the Arbotix-M via a TTL cable functions as a signal repeater.

The Arbotix-M also has eight analog inputs, allowing us to use strain gages for continuous load detection on each leg. Each strain gage is mounted on the proximal dorsal surface of the femur, providing the same kind of information as the femoral campaniform sensilla (fCS), which are crucial to timing stance and swing motions [25] [1]. A 5V rail is used to power an LM324 op-amp and Wheatstone quarter-bridge for each strain gage.

MantisBot's structural components are all made from polycarbonate (PC). We chose PC over aluminum because PC is sufficiently strong for a robot of this size and offers a better strength to weight ratio. Using PC also allowed most of the components to be 3D printed, allowing for complex geometries. For example, MantisBot's body segments are each a 3D truss, a very strong and light shape that would be difficult to produce with subtractive manufacturing. PC also is flexible enough that the amplifier gain for the strain gages can be set low (200), producing a very clean signal.

The microcontroller communicates with a desktop computer (i7 2770K 3.5 GHz, 32 GB RAM) at 256 kbps over a virtual serial connection (USB) using a modified version of the Firmata protocol. The Arbotix-M collects inputs from the robot and writes them to a buffer that AnimatLab uses to update the neural control system. MantisBot's inputs are the position of all 28 servos, as well as femoral strain, one gage for each of the six legs. The strain is used as a 10-bit analog signal to provide the network with continuous (i.e. not discrete) load signals. The network then writes new motor position commands to the buffer for the Arbotix-M to read.

2.2 Mantis Kinematics

Mantises are highly flexible insects with many degrees of freedom (DOF). Fig. 2 shows a to-scale schematic of the animal with segments and joints labeled.

The prothorax and mesothorax are connected by a multi-DOF joint which allows the mantis to rear and pivot the prothorax and the attached forelegs and head. Each thoracic segment (prothorax or T1, mesothorax or T2, and metathorax or T3) has a pair of multi-jointed legs.

Each leg has four main segments, moving distally: the coxa, trochanter, femur, and tibia, terminating in a tarsus (foot) for gripping the substrate. The T1 legs are highly mobile, possessing three thorax-coxa (ThC) joints, which together function like a ball-and-socket joint. The trochanter and femur are fused, keeping the coxa-trochanter (CTr) and femur-tibia (FTi) joints parallel. CTr extension lowers the tarsus toward the ground, and FTi flexion pulls the tarsus backward toward the body. The T2 and T3 legs each has the same DOF as those in T1, with the addition of a mobile trochanter-femur (TrF) joint.

Because the cockroach has been studied more thoroughly and is closely related [18], it may be helpful to contrast mantis leg anatomy to that of the cockroach. Unlike the cockroach, the mantis's T2 legs possess inwardly-mobile ThC3 joints, which rotate the leg ventrally about the coxae's long axis. The ThC3 joint is used to maintain the mantis's upright hunting posture. The T3 legs are nearly identical to the T2, except that the ThC3 moves the leg dorsally, and the TrF joint is less mobile. Fig. 2 shows a schematic of each leg and the DOF it possesses.

The T2 and T3 legs also differ from the T1 in that the segments are proportioned differently. The raptorial T1 legs are specialized for grasping, while T2 and T3 provide a wide, stable base for four-legged posture. Table 1 shows the measurements of leg segments from a male *Tenodera* on which MantisBot is

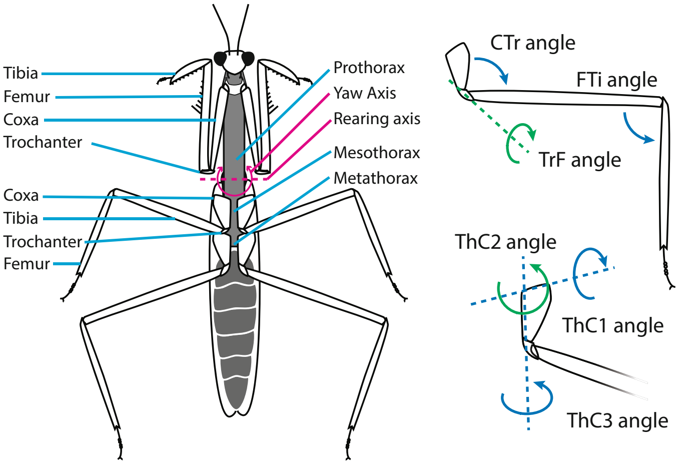

Fig. 2. Scaled schematic of a male *Tenodera sinensis*, with segments and degrees of freedom labeled. Joints with green arrows are not included on MantisBot. To see how the robot captures these proportions and DOF, see Fig. 3.

based. On the front leg, the femur is roughly 150% the length of the coxa, and 250% the length of the tibia. On the middle leg, the coxa is much shorter and the tibia is much longer, making the femur about 350% the length of the coxa, and 130% the length of the tibia.

2.3 Robot Kinematics

MantisBot has two body segments, a prothorax and mesothorax/metathorax, which are connected by a two-DOF joint. This enables the prothorax to rear and yaw with respect to the main body segment. The yawing motion is directly driven by an MX-64T. The rearing is driven by a four-bar mechanism underneath the thorax, with an MX-64T on the rear of the robot. The four-bar mechanism both provides additional mechanical advantage required to lift the prothorax and moves the center of mass of the robot rearward, which is beneficial for quadrupedal posture.

MantisBot's T1 legs include all of the degrees of freedom of the animal. This is important because the front legs are the most mobile and volitional [4], and will be necessary for studying directed motions such as striking at prey, or mobile locomotive tasks such as climbing. MantisBot's T2 and T3 legs possess ThC1, ThC3, CTr and FTi joints, which our previous works suggests are the most crucial for postural tasks [20]. Fig. 3 shows photos of each of the robot's legs overlaid with a scale schematic of the corresponding leg from *Tenodera*. The most noticeable discrepancy is that MantisBot's T2 and T3 coxae are longer than the animal's. This is necessary because placing the motors as close together as possible would establish a 23.1:1 scale, which would make the legs very long and reduce the mechanical advantage of the proximal leg motors so much that the robot would be unable to support itself. However, a 13.3:1 scale for the femora and tibia would mitigate this problem, establishing the scale used for most of the robot. These proportions are within the range of variation of other mantid species (G. Svenson, per. comm.). In total, MantisBot weighs 63.2 N, and when all motors are zeroed, has an envelope of 90 cm wide, 60 cm long, and 50 cm tall.

3 Robot Control Architecture

3.1 AnimatLab-MantisBot Interface

MantisBot is the first mobile robot designed to be controlled with the AnimatLab Robotics Toolkit (ART). MantisBot was designed with AnimatLab 2 by first constructing a virtual model of its body, servos, sensors, and nervous system. The ART lets us assemble a model of the robot in a graphical user interface (GUI),

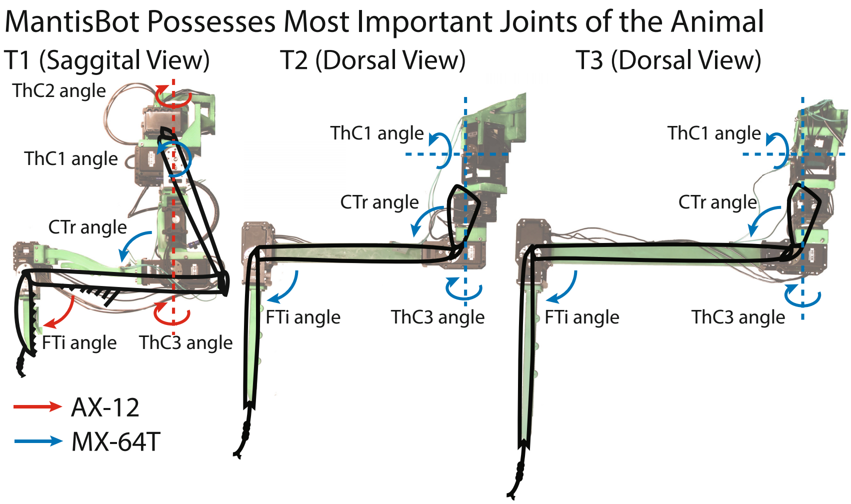

Fig. 3. Photos of each leg of the robot with joints labeled. 13.3:1 leg schematics from Fig. 2 are overlaid to compare the robot's proportions to the animal's. This comparison is quantified in Table 1.

Table 1. Segment lengths for the animal (left) and robot (center) in mm. The ratios between the segments is also shown for comparison (right). Weight is also included in the top row. Compare the lengths to the overlays in Fig. 3.

Animal	0.108	N		Robot	63.2	N		Ratio	585:1		
Segment	T1	T2	T3	Segment	T1	T2	T3	Segment	T1	T2	T3
Coxa	13.3	5.50	5.60	Coxa	135	127	127	Coxa	10.2:1	23.1:1	22.7:1
Femur	18.2	18.6	24.0	Femur	175	249	320	Femur	9.62:1	13.4:1	13.3:1
Tibia	7.3	14.2	18.0	Tibia	100	190	240	Tibia	13.7:1	13.4:1	13.3:1

and fully simulate the rigid body model and servos via the open source Bullet physics engine (bulletphysics.org). In simulation, the nervous system interfaces with the virtual body, producing a neuromechanical simulation used for initial testing of the control system. The ART is designed to allow the same nervous system to control both the simulation and the real robot by swapping out the underlying physics engine for a "robotic engine", which handles communication with the robot.

A link to the robot control hardware is added within the GUI by defining a hardware interface, which contains one or more I/O control modules that interact with a specific type of microcontroller. Part interfaces can be added to an I/O controller to link specific sensors or motors to their counterparts within the simulation.

MantisBot uses the Firmata I/O protocol to communicate with an Arbotix-M. Firmata allows the robotics engine to interface with most servos and sensors without requiring any new programming. The robotics engine runs on the master computer, and configures the I/O of the Arbotix-M slave. For a Dynamixel servo, the membrane voltage from a motor neuron is converted to a position command to control torque output (see Section 3.2). Motor commands that have changed are sent to the Arbotix-M, which updates the servos simultaneously. Servos are read round-robin with the data from one servo being read and sent back during each update. Digital and analog signals are sent back each time they change. The engine converts the incoming sensor values into currents that are injected into sensory neurons, completing the sensory/motor feedback loop. The robotics engine ensures that neural processing is kept in synchrony with the real-time I/O of the hardware, but on a per time step basis it easily simulates MantisBots control system of 775 neurons and 1258 synapses 150 times faster than real time.

3.2 Neural Controller - Single Joint Control

In order to apply animal data as directly as possible, MantisBot's controller is composed of conductance-based nonspiking neuron models. Neural dynamics are simulated as

$$\frac{dV}{dt} = G_{mem}(E_{rest} - V) + \sum g_{syn}(E_{syn} - V) + G_{Na} m_{\infty} h (E_{Na} - V) \tag{1}$$

$$\frac{dh}{dt} = \frac{(h_{\infty} - h)}{\tau(V)} \tag{2}$$

in which V is the membrane voltage, G are constant conductance values, g are instantaneous conductance values, and m and h are the sodium channel activation deactivation, respectively. The subscript mem stands for membrane, syn stands for synaptic, Na stands for sodium, and ∞ stands for steady state. The summation is over all incoming synapses to the neuron. Modeling neurons this way is appropriate because nonspiking neurons are known to exist throughout motor control systems in insects [3], and a single nonspiking model approximates the mean activity of a population of coupled spiking neurons. In addition, we make use of persistent sodium channels to build nonspiking central pattern generators (CPGs) like those in [11] [17] [6] [19]. MantisBot's controller is hierarchical and distributed, mimicking that in insects [2]. Each of MantisBot's joints has the same controller topology, shown in Fig. 4, tuned to the range of motion of that joint. For preliminary controller verification, most joints possess a CPG (in red). Future locomotion work will require that every joint has its own CPG. The CPG is based on persistent sodium models in other modeling studies [6] [17], and its parameters are designed to operate close to the oscillatory regime, such that it does not oscillate without descending excitation, but sensory information may cause a single flexion/extension transition. Phase space and phase response analysis have revealed that an inhibitory input to the CPG's interneurons (INs) will cause a single transition when it is applied, but an excitatory input will cause two transitions, one when it is applied and one when it is removed [12]. These

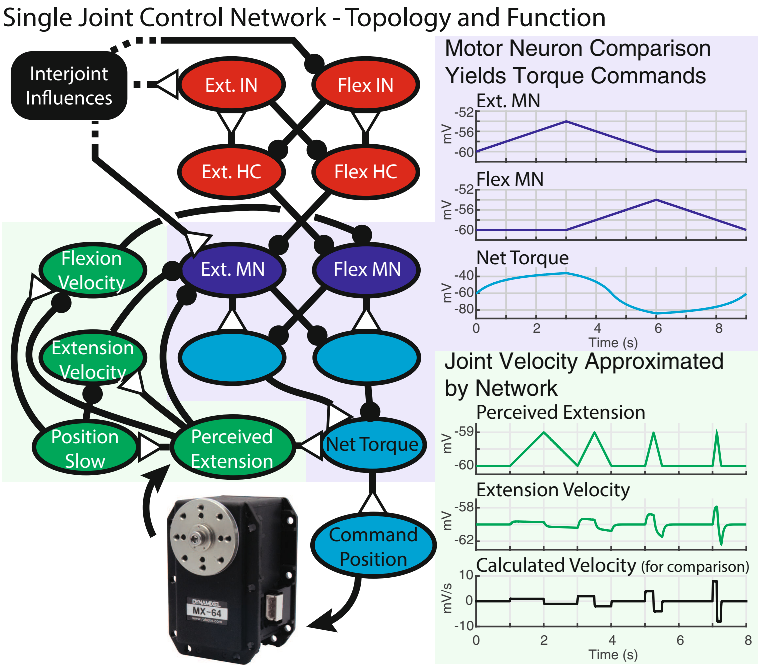

Fig. 4. Diagram showing the joint controller network implemented on MantisBot, as well as plots demonstrating how the structures produce relevant signals. CPGs (red, interneurons "IN" and half-centers "HC") inhibit motor neurons (dark blue, "MN"), the comparison of which yields a torque command (light blue). The first data set (violet shading) shows how this structure functions. The servo returns a perceived position (green), which provides position and velocity feedback to the MNs. The second data set (green shading) shows that this network reproduces a smoothed version of the time derivative of the perceived position. Interjoint influences, shown as a black box, may affect the CPGs or MNs.

transitions can be exploited to implement reflexes seen in insects (described in Section 3.3).

Each joint's torque output is commanded by an antagonistic pair of motor neurons (MNs). As an abstraction of MN activation causing muscle force, the motor neuron voltages are compared and the difference is added (for extension) or subtracted (for flexion) from the current servo position. This signal is then sent as a position command to the servo. But since the MNs' activation is in addition to the current position, this is actually a torque command in the form of the MN activation times the proportional gain of the servo. Data in Fig. 4 shows how the network converts MN activations to a desired torque.

To mimic passive forces that dominate joint control in small animals [10], each joint's MNs receive persistent position and velocity feedback, seeking a constant flexed position and zero velocity. Velocity signals from servos are noisy at low speeds, so our controller approximates velocity with a simple network based on vision filtering in *Drosophila* [22]. The position signal is passed through an interneuron, whose time constant is an order of magnitude larger than the sensory neuron. When the sensory neuron's membrane voltage fluctuates, the slower neuron's response lags behind, and taking the difference between the two yields an approximate velocity calculation. Data in Fig. 4 (green) compares this network's output with the actual, calculated differential of a position signal.

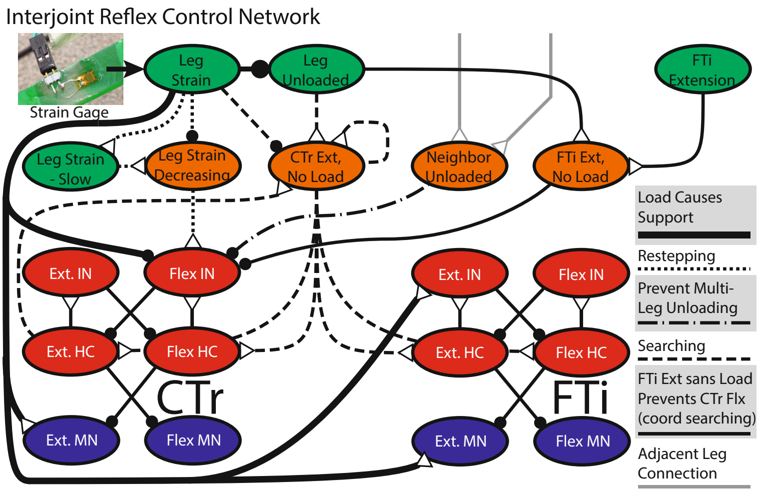

Fig. 5. A network diagram that shows interjoint reflexes based on insect neurobiology. Connections related to a particular reflex are drawn with the same line type. Load causes support [25], decreasing load causes corrective resteps [26], a leg may not unload when the posterior or contralateral leg is unloaded [5], leg extension without ground contact initiates searching, and FTi extension without leg load causes CTr extension [21].

3.3 Neural Controller - Intraleg Control

Posture is generated by coordinating the joints within a leg (Fig. 5). The mechanisms discussed in Section 3.2 provide the resistance reflexes that insects are known to exhibit when standing still [2], but insects are also known to use loading information to control both the timing and magnitude of leg extension [25]. Therefore, the FTi and CTr extensor MNs receive excitatory input directly from the strain gage on each leg. This lets each leg produce support forces proportional to the load acting on it. Loading information also feeds into the FTi CPG, even when it is inactive. It is known that load information from the fCS entrains the motion of the FTi [1], therefore in this control network load information excites the extension IN in the CPG, such that cycling the load causes the joint to actively oscillate. Adding interjoint pathways enables MantisBot to exhibit several intraleg reflexes observed in insects, which can be used to improve postural stability without volitional command of the each leg. For instance, if the load on the leg decreases rapidly while the leg is in support, the CTr will temporarily flex, as observed in crickets [26]. This lifts the leg, unloading it. Because the FTi CPG is excited by load, unloading the leg will cause the FTi to flex. When

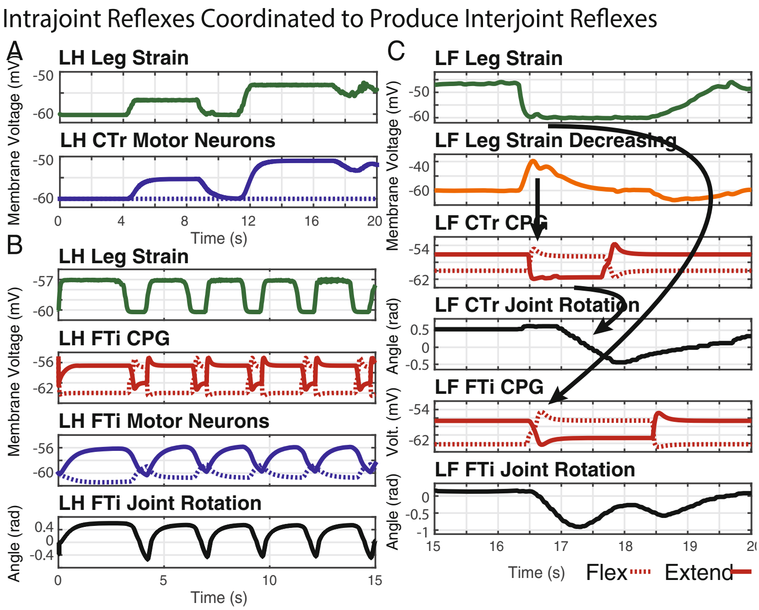

Fig. 6. MantisBot's low level reflexes can be coordinated through the interjoint pathways in Fig. 5 to produce coordinate reflexive posture adjustments. (A) The MNs of support joints receive direct inputs from strain sensors within that leg, to control motor output amplitude. (B) CPGs for the FTi joints can be entrained by strain sensors within that leg, to control motor output timing. (C) These and other joint level reflexes can be coordinated to produce leg level motions, such as a corrective restep when the leg senses it is slipping. The flow of reflexes is described in the text.

the CTr extends and loads the leg again, the FTi is more flexed, moving the foot closer to the body and stable ground. All of these reflexes are illustrated with data from MantisBot in Fig. 6, and additional reflexes are illustrated and explained in Fig. 5.

4 Conclusions

MantisBot is a research robot designed after the praying mantis *Tenodera sinensis* and controlled with the AnimatLab Robotics Toolkit to explore the neural control of motion by descending commands from the central complex. It is capable of emulating reflexes seen in the animal via biological neural networks, including central pattern generators. These basic capabilities provide a basis for future behavioral development and show the feasibility of controlling a robot in this manner.

References

1. Akay, T., Bässler, U., Gerharz, P., Büschges, A.: The role of sensory signals from the insect coxa-trochanteral joint in controlling motor activity of the femur-tibia joint. Journal of Neurophysiology **85**(2), 594 (2001)
2. Büschges, A., Gruhn, M.: Mechanosensory Feedback in Walking: From Joint Control to Locomotor Patterns. Advances In Insect Physiology **34**(07), 193–230 (2007)
3. Büschges, A., Wolf, H.: Nonspiking local interneurons in insect leg motor control. I. Common layout and species-specific response properties of femur-tibia joint control pathways in stick insect and locust. Journal of Neurophysiology **73**(5), 1843–1860 (1995)
4. Cleal, K.S., Prete, F.R.: The Predatory Strike of Free Ranging Praying Mantises, Sphodromantis lineola (Burmeister). II: Strikes in the Horizontal Plane. Brain Behavior and Evolution **48**, 191–204 (1996)
5. Cruse, H.: What mechanisms coordinate leg movement in walking arthropods? Trends in Neurosciences **13**(1990), 15–21 (1990)
6. Daun-Gruhn, S.: A mathematical modeling study of inter-segmental coordination during stick insect walking. Journal of Computational Neuroscience, 255–278, June 2010
7. Dirk, S., Frank, K.: The bio-inspired SCORPION robot: design, control & lessons learned. In: Climbing and Walking Robots, Towards New Applications, pp. 197–218, October 2007
8. Espenschied, K.S., Quinn, R.D., Beer, R., Chiel, H.J.: Biologically based distributed control and local reflexes improve rough terrain locomotion in a hexapod robot. Robotics and Autonomous Systems **18**(1–2), 59–64 (1996)
9. Guo, P., Ritzmann, R.E.: Neural activity in the central complex of the cockroach brain is linked to turning behaviors. The Journal of Experimental Biology **216**(Pt 6), 992–1002 (2013)
10. Hooper, S.L., Guschlbauer, C., Blümel, M., Rosenbaum, P., Gruhn, M., Akay, T., Büschges, A.: Neural control of unloaded leg posture and of leg swing in stick insect, cockroach, and mouse differs from that in larger animals. The Journal of Neuroscience : The Official Journal of the Society for Neuroscience **29**(13), 4109–4119 (2009)

11. Hunt, A., Schmidt, M., Fischer, M., Quinn, R.D.: Neuromechanical simulation of an inter-leg controller for tetrapod coordination. In: Duff, A., Lepora, N.F., Mura, A., Prescott, T.J., Verschure, P.F.M.J. (eds.) Living Machines 2014. LNCS, vol. 8608, pp. 142–153. Springer, Heidelberg (2014)
12. Hunt, A.J., Szczecinski, N.S., Andrada, E., Fischer, M.S., Quinn, R.D.: Using data and neural dynamics to design and control a neuromechanical rat model. In: Living Machines (accepted 2015)
13. Knops, S.A., Tóth, T.I., Guschlbauer, C., Gruhn, M., Daun-Gruhn, S.: A neuromechanical model for the neural basis of curve walking in the stick insect. Journal of Neurophysiology, 679–691, November 2012
14. Ritzmann, R.E., Quinn, R.D., Watson, J.T., Zill, S.N.: Insect walking and biorobotics: A relationship with mutual benefits. Bioscience **50**(1), 23–33 (2000)
15. Schilling, M., Hoinville, T., Schmitz, J., Cruse, H.: Walknet, a bio-inspired controller for hexapod walking. Biological Cybernetics **107**(4), 397–419 (2013)
16. Schneider, A., Paskarbeit, J., Schaeffersmann, M., Schmitz, J.: HECTOR, a new hexapod robot platform with increased mobility - control approach, design and communication. In: Advances in Autonomous Mini Robots, pp. 249–264 (2012)
17. Spardy, L.E., Markin, S.N., Shevtsova, N.A., Prilutsky, B.I., Rybak, I.A., Rubin, J.E.: A dynamical systems analysis of afferent control in a neuromechanical model of locomotion: I. Rhythm generation. Journal of Neural Engineering **8**(6), 65003 (2011)
18. Svenson, G.J., Whiting, M.F.: Phylogeny of Mantodea based on molecular data : evolution of a charismatic predator. Systematic Entomology **29**, 359–370 (2004)
19. Szczecinski, N.S., Brown, A.E., Bender, J.A., Quinn, R.D., Ritzmann, R.E.: A Neuromechanical Simulation of Insect Walking and Transition to Turning of the Cockroach Blaberus discoidalis. Biological Cybernetics (2013)
20. Szczecinski, N.S., Martin, J.P., Ritzmann, R.E., Quinn, R.D.: Neuromechanical mantis model replicates animal postures via biological neural models. In: Duff, A., Lepora, N.F., Mura, A., Prescott, T.J., Verschure, P.F.M.J. (eds.) Living Machines 2014. LNCS, vol. 8608, pp. 296–307. Springer, Heidelberg (2014)
21. Tryba, A.K., Ritzmann, R.E.: Multi-joint coordination during walking and foothold searching in the Blaberus cockroach. I. Kinematics and Electromyograms, June 2000
22. Tuthill, J.C., Nern, A., Rubin, G.M., Reiser, M.B.: Wide-field feedback neurons dynamically tune early visual processing. Neuron **82**(4), 887–895 (2014)
23. Yamawaki, Y., Toh, Y.: Response Properties of Visual Interneurons to Motion Stimuli in the Praying Response Properties of Visual Interneurons to Motion Stimuli in the Praying Mantis, Tenodera aridifolia. Zoological Science **20**(7), 819–832 (2003)
24. Yamawaki, Y., Uno, K., Ikeda, R., Toh, Y.: Coordinated movements of the head and body during orienting behaviour in the praying mantis Tenodera aridifolia. Journal of Insect Physiology **57**(7), 1010–1016 (2011)
25. Zill, S.N., Schmitz, J., Büschges, A.: Load sensing and control of posture and locomotion. Arthropod Structure & Development **33**(3), 273–286 (2004)
26. Zill, S., Frazier, S.: Characteristics of dynamic postural reactions in the locust hindleg. Journal of Comparative Physiology A **170**, 761–772 (1992)