Sensor-Driven Neural Control for Omnidirectional Locomotion and Versatile Reactive Behaviors of Walking Machines

P. Manoonpong [a,*], F. Pasemann [b] F. Wörgötter [a]

[a] *Bernstein Center for Computational Neuroscience (BCCN), University of Göttingen, D-37073 Göttingen, Germany*

[b] *Fraunhofer-Institut für Intelligente Analyse- und Informationssysteme (IAIS), D-53754 Sankt Augustin, Germany*

Abstract

This article describes modular neural control structures for different walking machines utilizing discrete-time neurodynamics. A simple neural oscillator network serves as central pattern generator producing the basic rhythmic leg movements. Other modules, like the velocity regulating and the phase switching networks, enable the machines to perform omni-directional walking as well as reactive behaviors, like obstacle avoidance and different types of tropisms. These behaviors are generated in a sensori-motor loop with respect to appropriate sensor inputs, to which a neural preprocessing is applied. The presented neuromodules are small so that their structure-function relationship can be analysed. The complete controller is general in the sense that it can be easily adapted to different types of even-legged walking machines without changing its internal structure and parameters.

Key words: Omnidirectional walking; Walking machines; Neural control; Sensor-driven behavior; Central pattern generator

* Corresponding author. Tel.: +49 (0) 551 5176-530; fax: +49 (0) 551 5176-449.
Email addresses: poramate@bccn-goettingen.de (P. Manoonpong), frank.pasemann@iais.fraunhofer.de (F. Pasemann), worgott@bccn-goettingen.de (F. Wörgötter).

1 Introduction

Walking animals are able to move around not only on flat terrains but also on rough terrains and even to perform a variety of walking patterns [1] [6]. They can adapt themselves to environmental changes in order to survive. They even show several fascinating behaviors, like reflexes, fixed action patterns [2], and taxes [4]. Neurobiologists suggest that solving these tasks basically results from coupling the appropriate biomechanics [19,24] with neural control [17,71]. For instance, cockroaches and stick insects can walk forward, backward, and in lateral directions [33,56] because of their appropriate biomechanical legs [66]. Neural control [66], on the other hand, plays a role in generating different walking behaviors with respect to environmental stimuli and walking situations. Therefore, during the last few decades several roboticists have begun actively to look to the biological sciences for the constructions and the controller design of robotic systems in particular walking machines to approach the animals in their levels of performance.

From this point of view, most previous studies in the domain of walking machines have paid attention on the construction of such machines to have animal-like properties [5,40] and perform efficient locomotion [7,13,43]. Others have focused on the generation of locomotion based on engineering technologies [16,64] as well as biological principles [17,21,40]. In general, all these machines were solely designed for the purpose of motion without responding to environmental stimuli. That is, most of them can traverse over uneven terrain, walk with different gaits, or perform omnidirectional walking. However, in this research area, only a few works have shown physical walking machines which can react to an environmental stimulus with standard walking behavior using different approaches [3,16,26,35]. This shows that less attention has been paid to walking machines performing various reactive behaviors with a variety of walking patterns. In other words, contributions developing embodied control techniques for sensor-driven behaviors of many degrees-of-freedom systems are rare.

The goal of this article is to present a novel approach to tackle this challenging problem. We have developed our walking machines together with their neural control, in a stepwise manner during the last three years [28,49,50]. They were designed and constructed on the basis of biological investigations, e.g., leg and body structures [47], and can now perform various reactive behaviors including several kinds of walking patterns (omnidirectional walking)

[1] Here, we describe walking patterns as the direction of motions, e.g., forward, backward, turn left and right, and so on. It is sometimes referred to as walking behaviors or walking modes. All these terms mean exactly the same thing.
[2] The action perseveres for longer than the stimulus itself.

under a sensor-driven neural controller. However, the rationale behind this article is not only to demonstrate biologically inspired walking machines with sensor-driven omnidirectional locomotion and versatile behaviors but also to investigate the analyzable neural mechanisms underlying this approach in order to understand their inherent dynamical properties.

Adapting a modular approach, the sensor-driven neural controller was mainly built from a combination of neural control and preprocessing modules. Neural control here serves as a locomotion generator while preprocessing is used to trigger various reactive behaviors corresponding to sensory inputs. This modular architecture is then considered as a major advantage, compared to many other controllers which were developed for walking machines, for instance, using genetic algorithms [10,75] or evolutionary techniques [26,42,58]. In general, these controllers were too complex to be mathematically analyzed in detail, in particular if they consist of a massive recurrent connectivity structure. Furthermore, most of them [10,26,42,58,75] have been created for a specific type of walking machines. Applying such controllers to other machines may require the modification of the network's internal parameters or structure. In contrast, the neural control developed here can be successfully applied to a physical six- as well as to an eight-legged walking machine having different morphologies, and it is also able to generate different walking modes without altering its internal parameters or structure. The walking patterns are basically generated by using the discrete-time dynamics of a simple 2-neuron oscillator network. While, the steering is done via velocity regulating networks (VRNs) and a phase switching network (PSN). The controller has the capability to prevent the walking machines from getting stuck in corners or deadlock situations by applying hysteresis effects provided by the recurrent structure of the neural preprocessing modules. By a simple modification of the neural preprocessing, it is possible to manually operate the machines via a joystick or to generate another kind of a reactive behavior, e.g., sound tropism (positive tropism) [49].

The article is organized as follows. Fist we describe the technical specifications of the walking machines together with their physical simulator. Second, we present the modular neural controller together with its subnetworks (modules) for generating omnidirectional walking and controlling reflex behavior of the machines. Third, we illustrate the performance of the modular neural control. Then, we analyze the neural structures and finally we show the result of the versatile reactive behaviors, followed by discussion and conclusions.

2 Walking machine platforms: AMOS-WD06 and -WD08

In order to explore the performance of neural control in physical systems, two walking machines with different morphology are employed. Both walking machines consist of many active joints and different kinds of sensory systems.

The six-legged walking machine AMOS-WD06 consists of six identical legs. Each leg has three joints (three DOF): the thoraco-coxal (TC-) joint enables forward (+) and backward (−) movements, the coxa-trochanteral (CTr-) joint enables elevation (+) and depression (−) of the leg, and the femur-tibia (FTi-) joint enables extension (+) and flexion (−) of the tibia (see Fig. 1a). The morphology of this multi-jointed leg is modeled on the basis of a stick insect leg [17] but the tarsus segments are ignored. The length of the tibiae, which are attached to the FTi-joints, is proportional to the dimension of the machine. With a new design of the tibia part, each of them contains a spring damped compliant element to absorb impact force as well as to measure ground contact during walking.

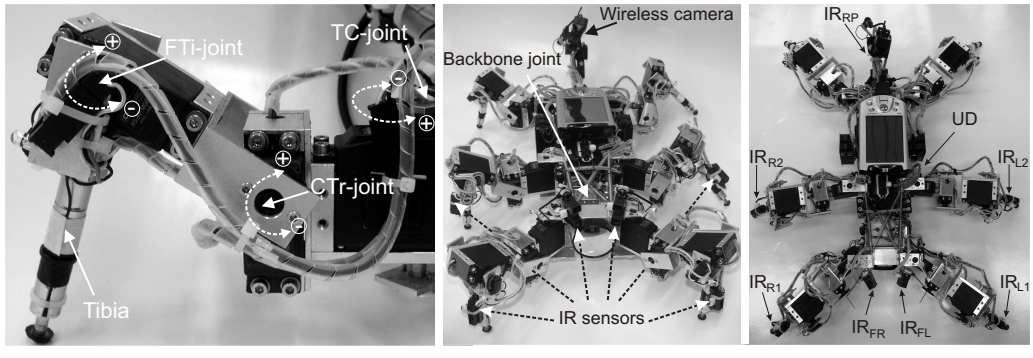

Fig. 1. (a) The physical leg with three DOF of the AMOS-WD06. (b) The physical six-legged walking machine AMOS-WD06. (c) Top view.

The body of the AMOS-WD06 consists of two parts: a front part where two forelegs are installed and a central body part where two middle legs and two hind legs are attached. The front and central parts are formed as narrow as possible with maximum symmetry to ensure optimal torque from the supporting legs to the center line of the trunk and to keep the machine balanced to ensure stability of walking. They are connected by one active backbone joint which can be activated to rotate around the lateral or transverse axis (pitch axis). This backbone joint is inspired by the invertebrate morphology of the American cockroach's trunk (see [47] for details). In addition, a tail with two DOF rotating in the horizontal and vertical axes is implemented on the back of the trunk. This actively moveable tail, which can be manually controlled, is used to install a mini wireless camera for monitoring the environment while the machine is walking. All leg and tail joints are driven by analog servo-

motors while the backbone joint is driven by a digital one. The size of the walking machine is 30 cm wide, 40 cm long, 12 cm high without its tail. The weight of the fully equipped robot (including 21 servomotors, all electronic components, sensors, and a mobile processor) is approximately 4.2 kg. This machine has six foot contact sensors for recording and analyzing the walking patterns and seven infrared (IR) sensors for eliciting reactive behaviors, e.g., obstacle avoidance and escape response. One pair of the IR sensors is located at the front part ($IR_{FR,FL}$), more two pairs are fixed at the tibiae of the two front ($IR_{R1,L1}$) and two middle legs ($IR_{R2,L2}$), and the rest of them is installed at the rear part (IR_{RP}) (see Figs. 1b and c). Additionally, one upside-down detector sensor (UD) is implemented beside the machine trunk (see Fig. 1c). It is applied to trigger a self-protective reflex behavior when the machine is turned into an upside-down position.

The eight-legged walking machine AMOS-WD08 consists of eight identical legs, each with three joints (3 DOF). They are built from a construction kit [3]. The configuration (compare Fig. 1a) and functionality of its legs are similar to those of the AMOS-WD06 except its tibiae where the spring element has not been yet implemented. The chassis of the AMOS-WD08 is designed following a scorpion body contour but the tail part is ignored (see [74] for details). It is constructed from only one part where all legs are orientedly attached (see Fig. 2). All leg joints are driven by micro digital servomotors. The size of the walking machine is 25 cm wide, 35 cm long, 10 cm high. The weight of the fully equipped robot (including 24 servomotors, all electronic components, sensors, and a mobile processor) is approximately 2.7 kg. Each of the AMOS-WD08 joints has a potentiometer sensor for detecting the actual angle position. In addition, it has a photoresistor sensor in each foot tip for measuring the ground contact.

The control of both walking machines is kept on a simple but powerful board, the Multi-Servo IO-Board (MBoard) [4], which is able to control up to 32 motors, and which has 36 analog sensor inputs and a size of 125 mm x 42 mm. The MBoard can be interfaced with a personal computer (PC) or a personal digital assistant (PDA) via an RS232 serial connection at 57.6 kbits per second.

The testing of the neural controllers however was first performed using a physical simulation environment "Yet Another Robot Simulator" (YARS) implemented on a 2-GHz PC with an update frequency of 25 Hz. The simulator is based on the Open Dynamics Engine (ODE) [72]. It provides a defined set of geometries, joints, motors and sensors which is adequate to create the AMOS-WD06 and -WD08 in a virtual environment. The basic features of the simulated walking machines are closely coupled to the physical ones, e.g., weight,

[3] http://www.ais.fraunhofer.de/~breitha/projects/RoboKit/RoboKit.html.
[4] http://www.ais.fraunhofer.de/BE/volksbot/mboard-content.html.

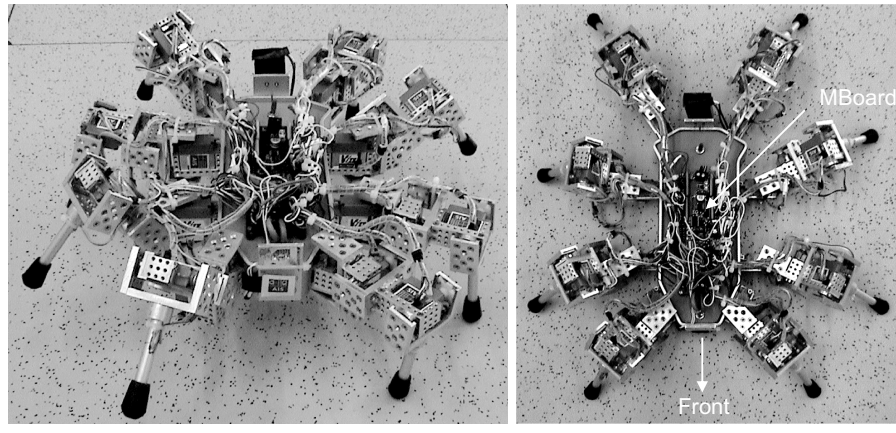

Fig. 2. The physical eight-legged walking machine AMOS-WD08. (a) Front view. (b) Top view.

dimension, motor torque and so on. The simulated walking machines with their virtual environment are shown in Fig. 3. The simulator is precise enough to reflect the corresponding behavior of the physical walking machines. This simulation environment is also connected to the Integrated Structure Evolution Environment (ISEE) [38,39] which is a software platform for evolving and developing neural controllers. After test on the simulator, a developed neural controller is applied to the physical walking machines to demonstrate walking in real environments.

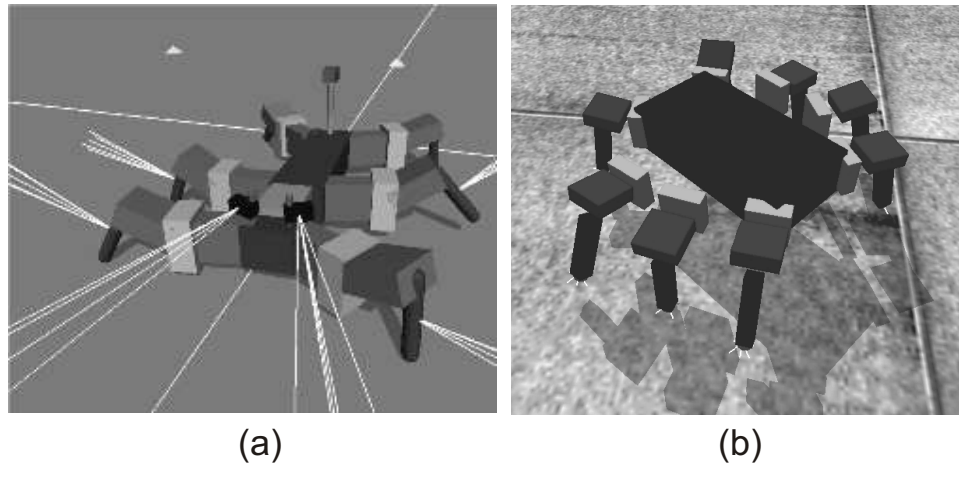

Fig. 3. The simulated walking machines in their environment. (a) Simulated AMOS-WD06. (b) Simulated AMOS-WD08.

3 Modular neural control

Modular neural control for generating omnidirectional walking of the machines and controlling the protection reflex (see Fig. 4) consists of three subordinate networks or modules (colored boxes in Fig. 4): a neural oscillator network, several velocity regulating networks (VRNs), and a phase switching network (PSN). The neural oscillator network, serving as a central pattern generator (CPG), generates periodic output signals. These signals are provided to all CTr-joints and FTi-joints only indirectly passing through all hidden neurons of the PSN. TC-joints are regulated via the VRNs. Thus, the basic rhythmic leg movement is generated by the neural oscillator network and the steering capability of the walking machines is realized by the PSN and the VRNs in accordance with the given input parameters I. All networks are described in detail in the following sections. The structure of this controller and the location of the corresponding motor neurons on the walking machines (AMOS-WD06 [5] and -WD08) are shown in Figs. 4 and 5. The developed controller is universal in the sense that it can be easily adapted to different types of even-legged walking machines with three-degree-of-freedom legs, e.g., four, six, and eight legs, without changing the internal structure and its parameters (dashed frame in Fig. 4) except only adding or reducing the number of output (motor) neurons (TR, TL, CR, CL, FR, FL (dashed arrows in Fig. 4)) depending on the number of legs. For instance, applying the controller for the six-legged walking machine we set $i = 1$ and 3, and $j = 2$ (compare Figs. 4 and 5); i.e., the controller drives 18 motor neurons which are TR_1, CR_1, FR_1 (TC-, CTr- and FTi-motor neurons of a right front leg), TR_2, CR_2, FR_2 (a right middle leg), TR_3, CR_3, FR_3 (a right hind leg), TL_1, CL_1, FL_1 (a left front leg), TL_2, CL_2, FL_2 (a left middle leg), and TL_3, CL_3, FL_3 (a left hind leg). The connection strengths together with the bias terms targeting to the adding motor neurons (dashed arrows in Fig. 4) are similar to depicted motor neurons ($TR_{i,j}, TL_{i,j}, CR_{i,j}, CL_{i,j}, FR_{i,j}, FL_{i,j}$, see Fig. 26 for the complete network). In case of the eight-legged walking machine, we set $i = 1$ and 3 and $j = 2$ and 4 (compare Fig. 5 and see Fig. 27 for the complete network) while setting $i = 1$ and $j = 2$ is for four legs. Note that applying the controller to four-legged walking machines may require additional control together with an appropriate sensor system, e.g., posture control [2,41,43], for balance and dynamically stable locomotion. Nonetheless, using this modular neural controller, only one gait can be obtained where the diagonal legs are paired and move together, i.e., a trot gait for four legs and a tripod gait for six legs.

[5] Describing the controller driving the machine also with the backbone joint will go beyond the scope of this article. Thus, the motor neuron controlling the backbone joint BJ is not activated; i.e., the backbone joint functions as a rigid connection. However, it can be modulated by the periodic signal via the PSN or VRNs to perform an appropriate motion, e.g., helping the machine during climbing over obstacles.

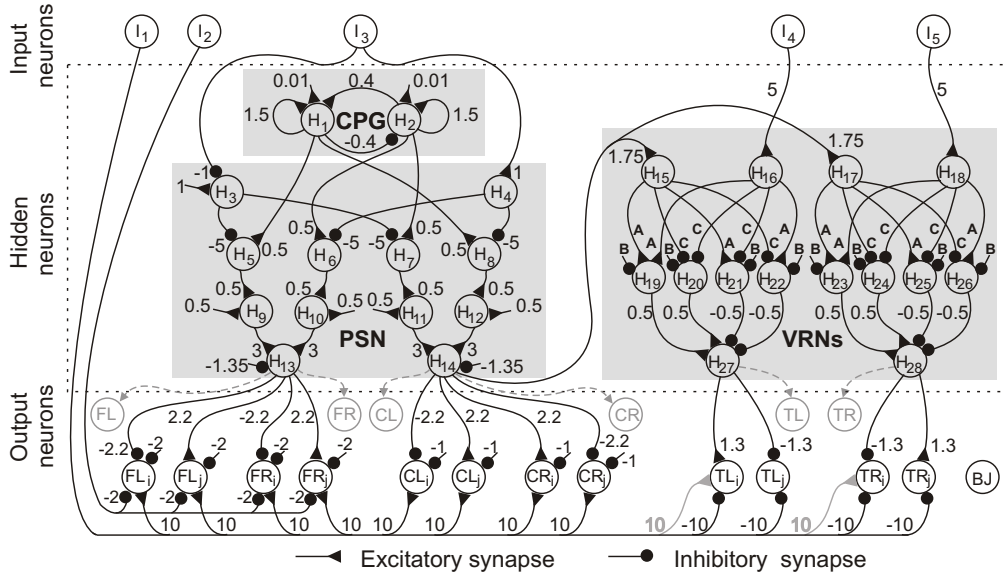

Fig. 4. Modular neural control for omnidirectional locomotion and the reflex behavior of walking machines. There are three different neuron groups: input, hidden and output. Input neurons I are the neurons used to control walking direction ($I_{2,3,4,5}$) and to trigger the protection reflex (I_1). Hidden neurons H are divided into three subgroups or modules (CPG, VRNs, and PSN) having different functionalities (see text for details). Output neurons, here called "motor neurons", ($TR_{i,j}, TL_{i,j}, CR_{i,j}, CL_{i,j}, FR_{i,j}, FL_{i,j}$) directly command the position of servo motors. Abbreviations are: BJ = a backbone joint, $TR(L)$ = TC-joints of right (left) legs, $CR(L)$ = CTr-joints of right (left) legs, $FR(L)$ = FTi-joints of right (left) legs. Indexing variables i and j are for odd and even numbers, respectively. They define the leg position on each side (see text for details and compare Fig. 5). All connection strengths together with bias terms are indicated by the small numbers except some parameters of the VRNs given by $A = 1.7246$, $B = -2.48285$, $C = -1.7246$. Note that dashed arrows indicate additional neurons which can be added depending on the number of legs.

All neurons of the network, updated with frequency of 25 Hz, are modeled as discrete-time non-spiking neurons. The state and output of each neuron are governed by Eqs. 1, 2 [47], respectively:

$$a_i(t+1) = \sum_{j=1}^{n} W_{ij}\, o_j(t) + B_i \quad i = 1, \ldots, n, \tag{1}$$

$$o_i = \tanh(a_i) = \frac{2}{1 + e^{-2a_i}} - 1, \tag{2}$$

where n denotes the number of units, a_i their activity, B_i represents a fixed internal bias term together with a stationary input to neuron i, W_{ij} the synaptic strength of the connection from neuron j to neuron i, and o_i the neuron output. Input units are configured as linear buffers.

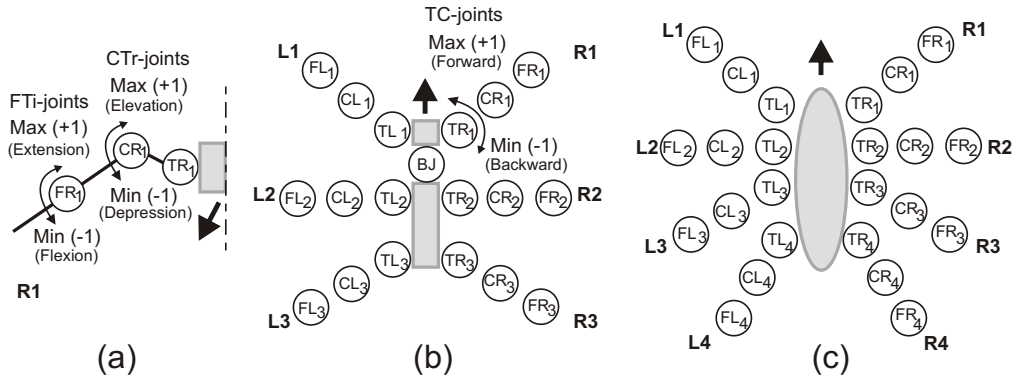

Fig. 5. (a) The movements of the CTr- and FTi-joints of the right front leg (R_1) with the remaining legs of both machines performing the same (front view). (b) The location of the motor neurons on the AMOS-WD06 and the movements of the TC-joints. (c) The location of the motor neurons on the AMOS-WD08 where the movements of the TC-joints are similar to those of the AMOS-WD06.

3.1 Neural oscillator network

The basic locomotion and rhythm of stepping in fast walking animals, e.g., cockroaches [22,23], mostly relies on a CPG [6], while their peripheral sensors are used to control walking behaviors [11,67]. By contrast, in slow walking animals (e.g., stick insects), sensory feedback plays a critical role in shaping the final motor pattern which is originally generated by CPGs located at each leg joint [17,66]. According to a principle of biological locomotion control, the basic rhythmic movements of the legs of our walking machines will be generated by a CPG.

The concept of CPG controllers for legged locomotion has been studied in various works [7,40,44,52]. For instance, Beer et al. [12] used a set of six coupled pacemaker (oscillator) neurons where each drives one of the six legs. Kimura et al. [43] used a CPG network constructed with four neural oscillators where each single oscillator consisting of two mutually inhibiting neurons based on a continuous-time oscillator, called Matsuoka Neural Oscillator [53]. His CPG network has been applied to control a four-legged walking machine where each leg of the machine is driven by one of the neurons. Others use one oscillator for each degree of freedom (DOF) of the leg joints [14,70].

Here the model of a CPG is realized by using the discrete-time dynamics of a simple 2-neuron network [61]. The network consists of two neurons with full connectivity (see Fig. 6).

Its parameters are selected in accordance with the dynamics of the 2-neuron

[6] A group of interconnected neurons that can be activated to generate a locomotor pattern without the requirement of sensory feedback [32,34,51].

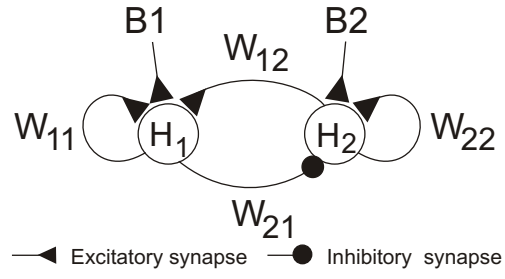

Fig. 6. The 2-neuron oscillator network.

system [61] staying near the Neimark-Sacker bifurcation where the quasi-periodic attractors occur. They are empirically adjusted through the Integrated Structure Evolution Environment (ISEE) to acquire the optimal periodic output signals for generating locomotion of the walking machines [28,50]. In Fig. 7, we show examples of periodic output signals having different frequencies resulting from different weights. Additional investigation of the network can be seen at [47].

Figure 7 shows that such a network has the capability to generate various sinusoidal outputs depending on the weights. Increasing W_{11}, W_{22} and W_{12} but decreasing W_{21} in a proportional way, the system dynamics still stays near or beyond the Neimark-Sacker bifurcation [61], resulting in an increased frequency of the sinusoidal outputs of the network. Correspondingly, the amplitude of the signals will also slightly increase.

One can utilize the modifiable frequency for walking speed control. To compare the effect of the different frequencies with the walking speed three parameter sets (A, F, K shown in Fig. 7) are chosen for the AMOS-WD06. In this situation, the machine will only walk forward with a tripod gait. The average speed values for the different frequencies are displayed in Fig. 7f.

For our purpose, the parameter set for the network controller is given by $B_1 = B_2 = 0.01$, $W_{21} = -0.4$, $W_{12} = 0.4$ and $W_{11} = W_{22} = 1.5$. It generates sinusoidal outputs that differ in phase by $\pi/2$ with a frequency of approximately 0.8 Hz (see Fig. 8). Note that this parameter set is used to acquire an optimal walking speed of the machines during test of omnidirectional locomotion control. However, we will make use of a sensory signal to modify all weights of the network in a similar way as shown in Fig. 7. As a consequence, walking speed of the machines can be varied by one sensor signal described in Sect. 6.1.

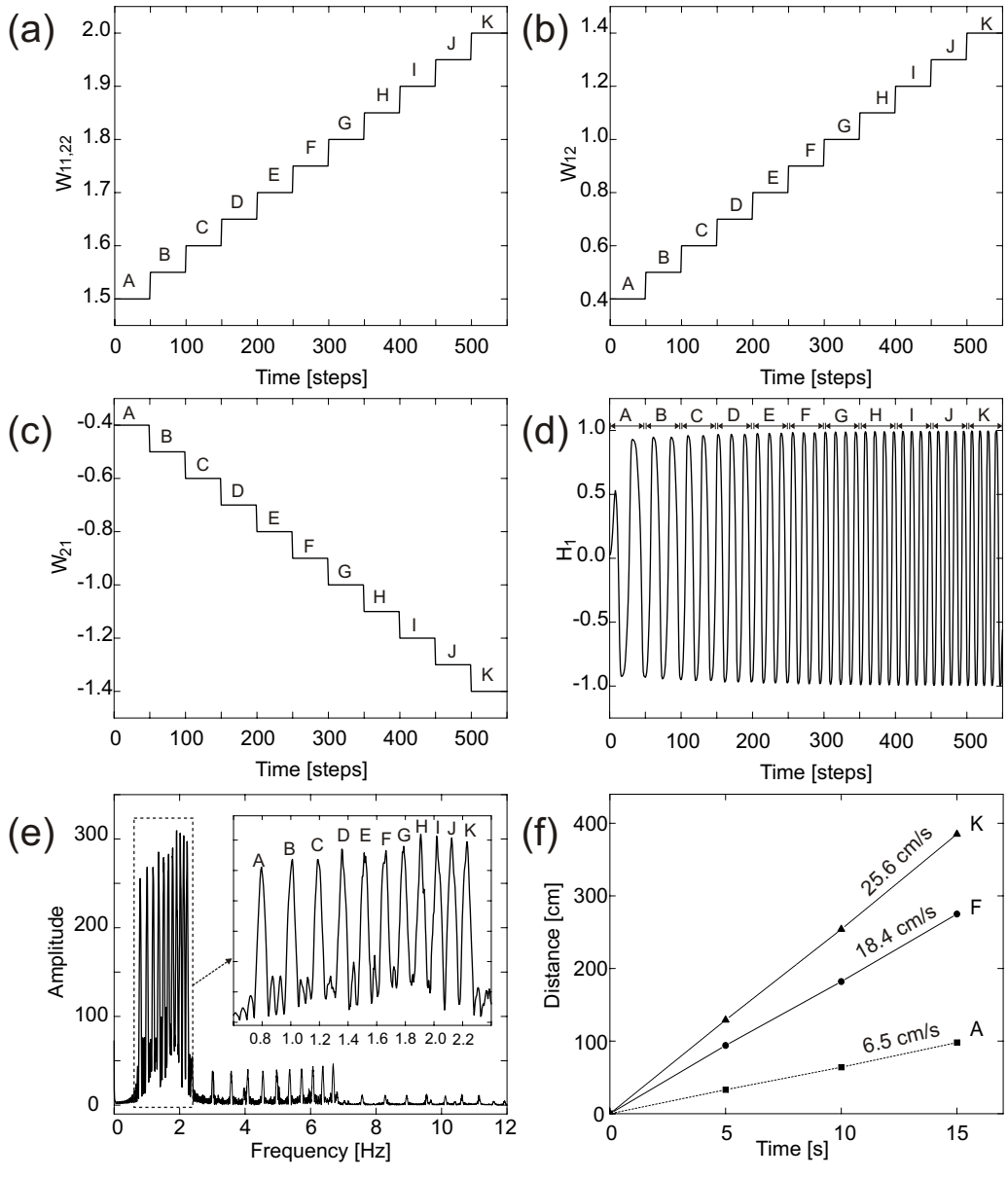

Fig. 7. (a) to (c) present the weight change of the network where W_{11} and W_{22} are equal. W_{11}, W_{22}, and W_{12} are increased every 50 time steps from A to K while W_{21} is decreased. Note that the bias terms B_1, B_2 are set to 0.01 in all cases. (d) The output signal H_1 of the network varies from low (A) to high (K) frequencies according to the weight changes. The other output H_2 having a similar pattern with the leading phase by $\pi/2$ is not shown. (e) Frequency analysis of the recorded sinusoidal signal H_1 shows that it consists of several frequencies where each period from A to K has different eigen frequencies (zoom panel). (f) Comparison of the walking speed with different frequencies of the sinusoidal signal.

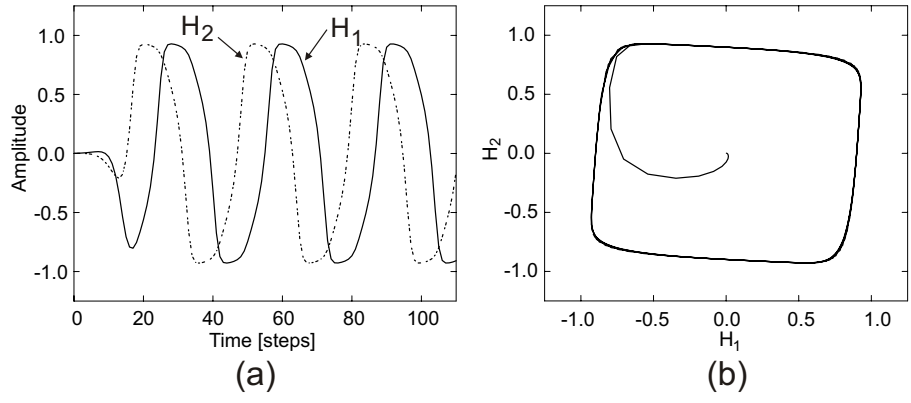

Fig. 8. (a) The sinusoidal output signals from the neural oscillator network where the parameter set is: $B_1 = B_2 = 0.01, W_{21} = -0.4, W_{12} = 0.4$ and $W_{11} = W_{22} = 1.5$. The update frequency of the network is approximately 25 Hz. (b) The phase space with the quasi-periodic attractor of the oscillator network.

3.2 Velocity regulating network

Having established that the neural oscillator with appropriate synaptic weights projecting to the motor neurons will generate one type of walking pattern, e.g., forward motion with a tripod gait [28], the next step is to change walking patterns, e.g., from walking forward to backward as well as spot turning to the left and right. The simplest way to achieve this is to generate 180-degree-out-of-phase sinusoidal signals which drive the TC-joints [1,76]. By doing so, we use a so-called velocity regulating network (VRN). The network is taken from [27] where it was partly constructed and partly trained by using the backpropagation rule in [69]. It approximates a multiplication function on two input values $x, y \in [-\mathbf{1}, +\mathbf{1}]$ (see [47] for details). For this purpose, the input x is the periodic signal, either coming directly from the neural oscillator network (see [50]) or through the PSN (see Fig. 4), to generate the rhythmic leg movement and the input y is the input parameter I or a sensory signal to control the corresponding walking behavior. The output signal of the VRN will be used to directly drive the TC-joints. Note that two VRNs are coupled with the neural controller network where each of them drives the TC-joints on each side (see Figs. 4 and 5). Figure 9 presents the network structure and its output signal which achieves 180 degree phase shift, when the input y changes from -1 to $+1$ and vice versa.

Because the VRN behaves qualitatively like a multiplication function, it has capability to increase or decrease the amplitude of the periodic signal by the magnitude of the input y. Consequently, it will affect the walking speed of the machines; i.e., the higher amplitude of the signal the faster they walk (not shown but see [47] for experiments).

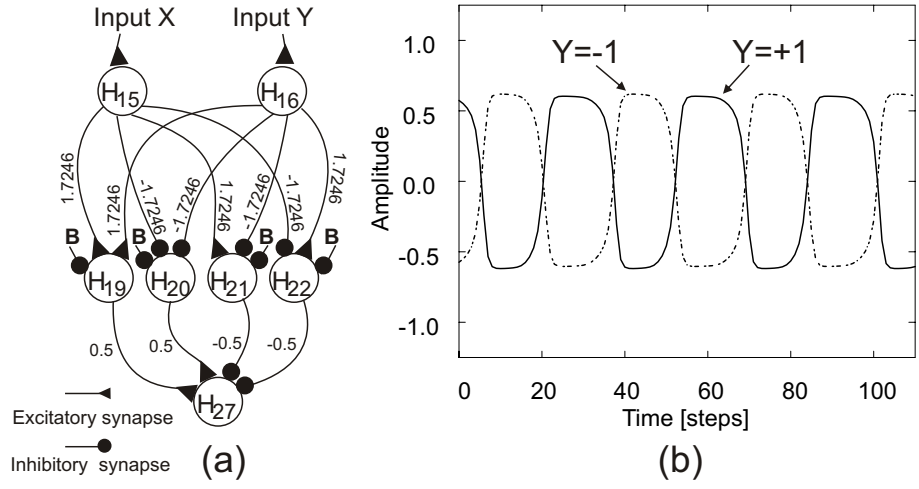

Fig. 9. (a) The VRN where bias terms B are all equal to -2.48285. (b) The output signal (H_{27}) when the input y is set to $+1$ (solid line) and -1 (dashed line).

3.3 Phase switching network

Combining the neural oscillator network and the VRNs presented above will lead to neural control [50] that is able to generate five different walking patterns (forward, backward, turning left and right, and marching). These walking patterns result from modifying the rhythmic signals at the TC-joints by the two input neurons $I_{4,5}$ (see Fig. 4) while the rhythmic signals at the CTr-joints remain unchanged. At the same time, all FTi-joints will be inhibited (flexed position) through I_2 (see Fig. 4). Activating all FTi-joints by periodic signals, the walking machines will perform at least three more walking modes (e.g., forward/backward diagonal and lateral motions to the right) according to the given inputs $I_{4,5}$ (see Table 1 and Fig. 11). To enhance the walking capability (e.g., also forward/backward diagonal and lateral motions to the left), one possibility is to reverse the phase of the periodic signals driving the CTr- and FTi- joints. That is, these periodic signals can be switched to lead or lag behind each other by $\pi/2$ in phase in accordance with the given input I_3 (see Fig. 4).

To this end, a hand-designed feedforward network, called phase switching network (PSN), is applied. The network consists of four hierarchical layers with 12 neurons. First, the periodic signals of the neural oscillator network are provided to the PSN through two pairs of hidden neurons ($H_{5,6}$ and $H_{7,8}$, see Fig. 10). The synaptic weights projecting to them are calculated in the way that they should not change the periodic form of their input signals and should keep the amplitude of the signals as high as possible. Thus, we set these synaptic weights to 0.5, which will convert the signals in the linear domain of the sigmoid transfer function. The activation of $H_{5,6,7,8}$ is controlled by higher layer neurons $H_{3,4}$ with large inhibitory connections (i.e., -5.0). H_3

(or H_4) will inhibit its target neurons (see Fig. 10) if it is activated, where its activation will be controlled by the binary value of I_3 (see Table 1). As a result, one neuron of each pair (H_5 or H_6 and H_7 or H_8) will be activated while the other will be inhibited. For instance, if H_5 and H_7 are activated, they will give periodic outputs while H_6 and H_8 will give a constant value of -1.0 and vice versa. Therefore, we have to shift the signals of the inhibited neuron, e.g., $H_{6,8}$, from -1.0 to 0.0 before summing them with the activated neuron, e.g., $H_{5,7}$. This is done by the hidden neurons $H_{9,10,11,12}$ of the lower layer. The synaptic weights together with the bias terms connected to them are set in a way that the signals will be again converted in the linear domain and the output signals of the inhibited neurons will be shifted to minimally 0.0. That is, we again choose them as 0.5. Finally, we amplify the output signals of $H_{9,10}$ and $H_{11,12}$ with larger synaptic weights, i.e., 3.0, and combine them via the output neurons $H_{13,14}$. Additionally, we set the bias terms of $H_{13,14}$ to -1.35 to shift an offset of the resulting output signals down. Note that one can optimize this network, for instance by using backpropagation, but for our purposes of controlling the machines, it is good enough. The resulting network together with its output signals with respect to the given input I_3 is shown in Fig. 10.

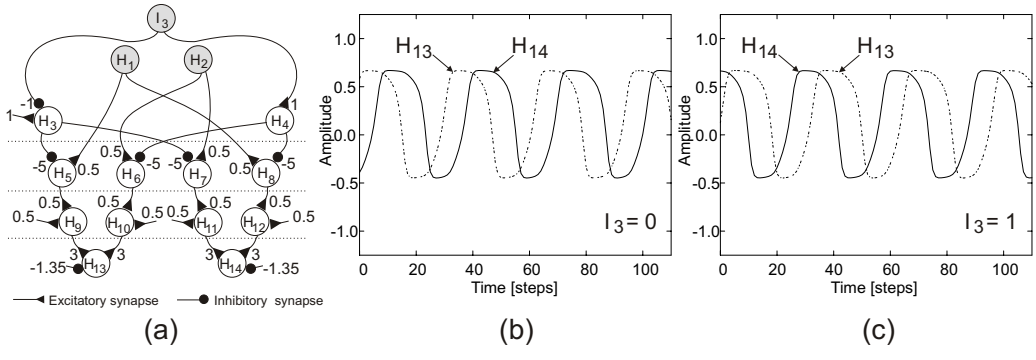

Fig. 10. (a) The PSN. (b), (c) Output signals ($H_{13,14}$) of the network when the input I_3 is set to 0 and 1. For instance, if I_3 is set to 0, H_3 will be activated because of its bias term while H_4 will be deactivated. Thus, H_3 will inhibit the activation of its targeting neurons $H_{5,7}$. As a result, $H_{13,14}$ of the network will generate the periodic signals originally coming from $H_{1,2}$ through $H_{6,10}$ and $H_{8,12}$. On the other hand, the periodic signals will go through other neuron paths if I_3 is set to 1.

It can be seen that the network switches the phase of the two sinusoidal signals originally coming from the neural oscillator network when I_3 is changed from 0 to 1 and vice versa. By applying this network property, the movements of the CTr- and FTi- joints will be reversed corresponding to the modification of I_3. Consequently, the walking machines will change their diagonal or lateral walking from the right to the left and vice versa.

3.4 Neural parameters for generating different walking patterns

The integration of three different functional neural modules described above gives the complete neural locomotion controller. The synaptic weights of the connections between the controller and the corresponding motor neurons together with the bias term of each motor neuron are chosen intuitively. They are adjusted through the YARS simulator to obtain an optimal gait; i.e., a static gait where the diagonal legs are paired and move synchronously. The actual parameters of the controller together with its structure are depicted in Fig. 4. However, all these synaptic weights and the bias terms can be changed depending on hardware configuration, e.g., the position of actuators.

This neural controller can generate different walking patterns which are controlled by the four input neurons $I_{2,\ldots,5}$. The connection weights between these input neurons and their targeting neurons are shown in Fig. 4. In addition, the reflex behavior [7] will be triggered via the input neuron I_1 which will excite TR_1 and TL_1 joints and all CTr- and FTi- joints and inhibit the remaining TC-joints (see Fig. 4).

Appropriate input parameter sets for the different walking patterns and the reflex behavior are presented in Table 1 where the first column describes the desired actions in accordance with five input parameters shown in the other columns.

As shown in Table 1, this neural controller can produce at least 12 different actions with respect to the given inputs. More actions, e.g., turning in different radians or curve walking in forward and backward directions, can be achieved by varying I_4 and I_5 between -1.0 and 1.0 (see Fig. 11) while other input parameters are fixed ($I_1 = 0$, $I_2 = 1.0$, and $I_3 = 1$ or 0). As a consequence, the amplitude of the periodic signals of the left and right TC-joints is regulated. For all cases, I_1 and I_3 are set as binary values (0 or 1) which affect all joints and switch the phase of the two periodic signals, respectively. On the other hand, I_2 can vary between 0.0 and 1.0 which suppresses the amplitude of the periodic signal of the FTi-joints; i.e., the larger the value of I_2 the lower the amplitude. As a result, the walking machines will perform a very small step in the lateral or diagonal direction or no step at all if I_2 is set to 1.0. Setting I_2 to negative values might cause unstable walking.

[7] Recall that if the walking machines are turned into an upside-down position, they will perform a stable standing in this position by rotating their CTr-, FTi-joints into the maximum elevation and extension positions. While, all TC-joints will be turned into the minimum backward position except those of the right R_1 and left L_1 front legs which will be turned into the maximum forward position.

Table 1
Input parameters for the different walking patterns and the reflex behavior

Actions	I_1	I_2	I_3	I_4	I_5
Forward	0	1.0	1, 0	-1.0	-1.0
Backward	0	1.0	1, 0	1.0	1.0
Turn right	0	1.0	1, 0	-1.0	1.0
Turn left	0	1.0	1, 0	1.0	-1.0
Marching	0	1.0	1, 0	0.0	0.0
FDiR	0	0.0	0	-1.0	-1.0
BDiR	0	0.0	0	1.0	1.0
LaR	0	0.0	0	0.0	0.0
FDiL	0	0.0	1	-1.0	-1.0
BDiL	0	0.0	1	1.0	1.0
LaL	0	0.0	1	0.0	0.0
Reflex	1	0.0 ...1.0	1, 0	-1.0 ...1.0	-1.0 ...1.0

Abbreviations are: $FDiR$ and $BDiR$ = forward and backward diagonal motion to the right, $FDiL$ and $BDiL$ = forward and backward diagonal motion to the left, LaR and LaL = lateral motion to the right and the left. Note that marching is an action where all the legs are positioned and held in a vertical position and support is switched between the two tripods (for six legs) or the two tetrapods (for eight legs).

4 Robot walking experiments with the modular neural controller

In this section several experiments demonstrating the performance of the modular neural network are described. The first attempt was to observe omnidirectional walking behavior of the machines in the physical simulator (YARS) where the results can be seen as a video clip at http://www.nld.ds.mpg.de/~poramate/RAS/OmniS.mpg. After the test on the simulator the controller has been applied to the physical walking machines (AMOS-WD06 and -WD08). It shows that the simulated walking machines and the physical walking machines behave similarly. We encourage readers to watch the video clips of the real robot walking experiments at http://www.nld.ds.mpg.de/~poramate/RAS/OmniR.mpg.

Here we report the data of one leg of the AMOS-WD08 during performing different walking behaviors. All types of walking behaviors have been carried out sequentially and with continuous transitions. Note that all different walking behaviors were generated through the input parameters $I_{2,\ldots,5}$ while I_1 was set to 0 in all cases (see Table 1 and Fig. 11).

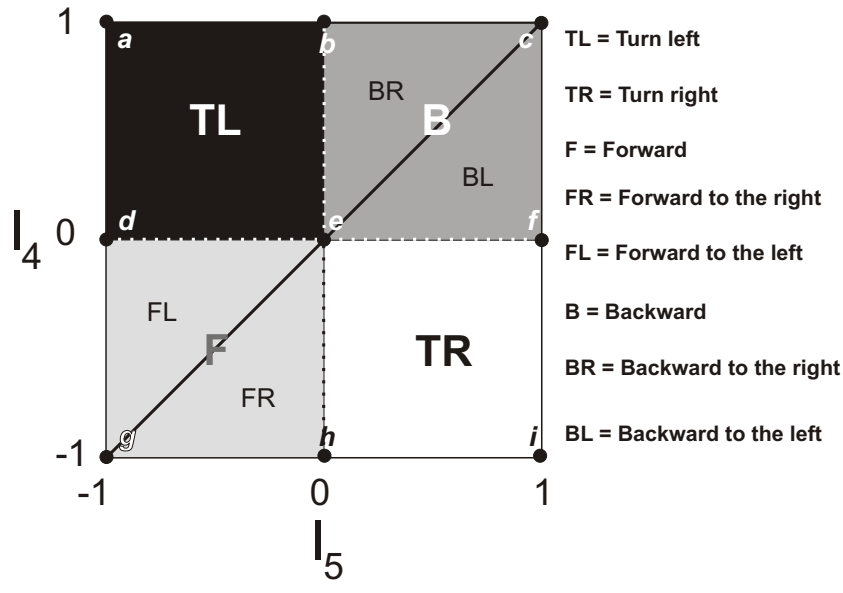

Fig. 11. Plot of the input space (I_4, I_5, compare Fig. 4) which is classified into four main areas. For input values in a dark square area ($\overline{dabe}$), the walking machines will perform spot turning to the left with different radians while a white square area ($\overline{efih}$) is for the right turn. In light grey triangle areas ($\overline{gde}$ and $\overline{geh}$), they will move forward in different curves to the left and the right and dark grey triangle areas ($\overline{ebc}$ and $\overline{ecf}$) are for backward to the right and the left, respectively. Additionally, if I_4 and I_5 are varied along the diagonal line ($\overline{gec}$), the machines will walk straight forward ($\overline{ge}$) and backward ($\overline{ec}$) with different walking speeds.

Figures 12 and 13 show real-time data of the movements of the TC-joint (TL_1), the CTr-joint (CL_1), and the FTi-joint (FL_1) of the left front leg and a gait diagram with respect to each walking behavior. Note that all these data were obtained from the angle and foot contact sensor neurons where their activations are between -1.0 and $+1.0$.

During marching in place the amplitudes of the TC- and FTi-joints are 0.0 while the CTr-joints oscillate with a very small amplitude (see TL_1, FL_1, and CL_1 in Figs. 12a and 13a). For forward and back walking, the CTr-joints and especially the TC-joints perform large amplitude oscillations while the FTi-joints were inhibited to stay in the flexed position (see FL_1 in Figs. 12a and 13a). The differences between forward and backward walking can be observed through the phase difference of the TC- and CTr-joints; i.e., they differ in phase by approximately $\pi/4$ during forward walking while they are almost in antiphase (the phase shift between them is 180 degrees) when the machine walks backward (see TL_1 and CL_1 in Figs. 12a and 13a). Activating the FTi- and CTr-joints but suppressing the TC-joints cause the lateral walking behavior. When the periodic signal (e.g., here the output of H_{13}) driving the FTi-joints leads another periodic one (e.g., here the output of H_{14}) driving the CTr-joints by $\pi/2$ in phase, the walking machine performs a lateral walking behavior to the right and vice versa for moving to the left (compare CL_1 and FL_1

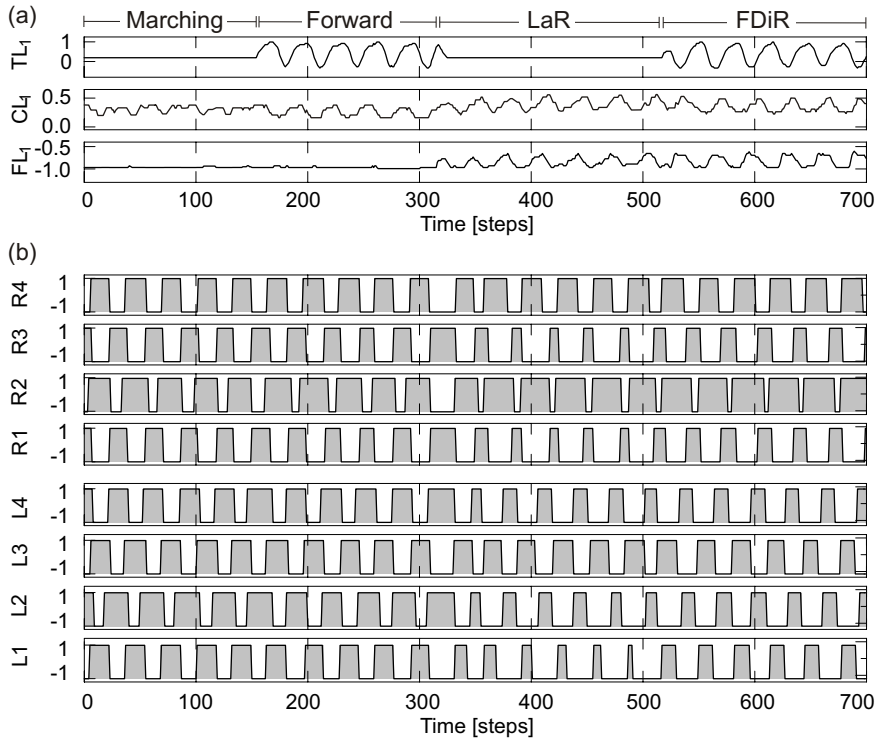

Fig. 12. (a) The angle sensor signals of the TC-joint (TL_1), CTr-joint (CL_1), and FTi-joint (FL_1) of the left front leg. Note that the signals of other joints having similar patterns to those of the left front leg are not shown. At the beginning, a marching behavior (M) was given; i.e., the machine stepped in place, then a forward behavior (F) was activated. After that we switched from a forward motion to a lateral motion to the right (LaR). Finally, the forward behavior and the lateral behavior were simultaneously stimulated resulting in a forward diagonal walking to the right ($FDiR$). (b) Gait diagram with respect to the stimulated walking behaviors. It shows two walking phases: the swing and stance phases. During the swing phase the foot has no ground contact and the ground contact sensor neuron has low activation (≈ -1.0). During the stance phase (gray blocks) the foot touches the ground and the neuron activation gets high (≈ 1.0).

in Figs. 12a and 13a). In addition, the combination of forward/backward and lateral motions enables the machine to walk in the diagonal directions where all joints were activated (see TL_1, CL_1 and FL_1 in Figs. 12a and 13a) [8].

In all walking behaviors, the machine walks with one gait type where the diagonal legs are paired and move together; e.g., $R4$, $R2$, $L3$ and $L1$ step in phase while the remaining legs step out of phase (see Figs. 12b and 13b). With this gait the eight-legged walking machine is always supported by at least four legs while the six-legged walking machine has at least three support legs called

[8] Note that the phase shifts of the right FTi- and CTr-joints during lateral and diagonal walking are inverse to the left FTi- and CTr-joints presented in Figs. 12a and 13a.

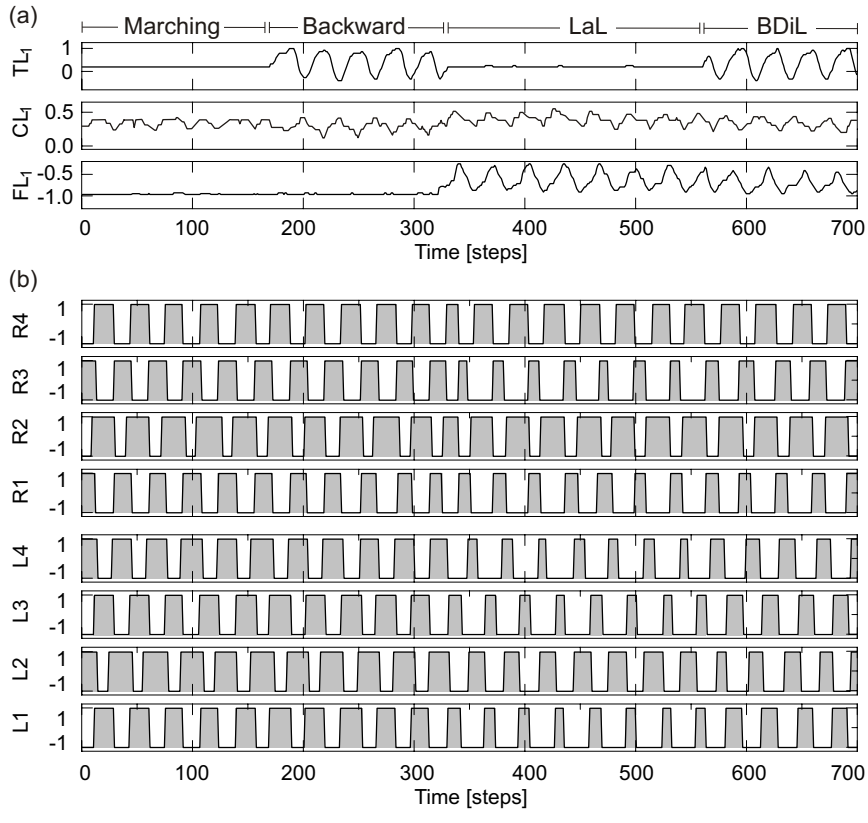

Fig. 13. (a) The angle sensor signals of the TC-joint (TL_1), CTr-joint (CL_1), and FTi-joint (FL_1) of the left front leg. Note that the signals of other joints having similar patterns to those of the left front leg are not shown. A marching behavior (M) was initially given, then a backward behavior (B) was activated. After that we switched from a backward motion to a lateral motion to the left (LaL). At the end, the backward behavior and the lateral behavior were simultaneously stimulated resulting in a backward diagonal walking to the left ($BDiL$). (b) Gait diagram with respect to the stimulated walking behaviors.

the tripod gait.

In Figs. 14 and 15, we show the experimental results when the movement of the TC-joints was modified while the movement of the CTr-joints remained unchanged and the FTi-joints were inhibited. Changing the movement of TC-joints of all left legs can switch between forward walking and spot turning to the left (see Fig. 14). On the other hand, reversing the movement of TC-joints of all right legs can change from forward walking to spot turning to the right and vice versa (see Fig. 14). Turning all TC-joints of both sides into the opposite direction while the machine walks forward enables it to walk backward. Additionally, decreasing or increasing the amplitude of the movement of the TC-joints on the left or the right side results in curve walking (see Fig. 15). Walking gaits during spot turning and curve walking are similar to the one observed in the forward/backward walking (see Figs. 12b and 13b).

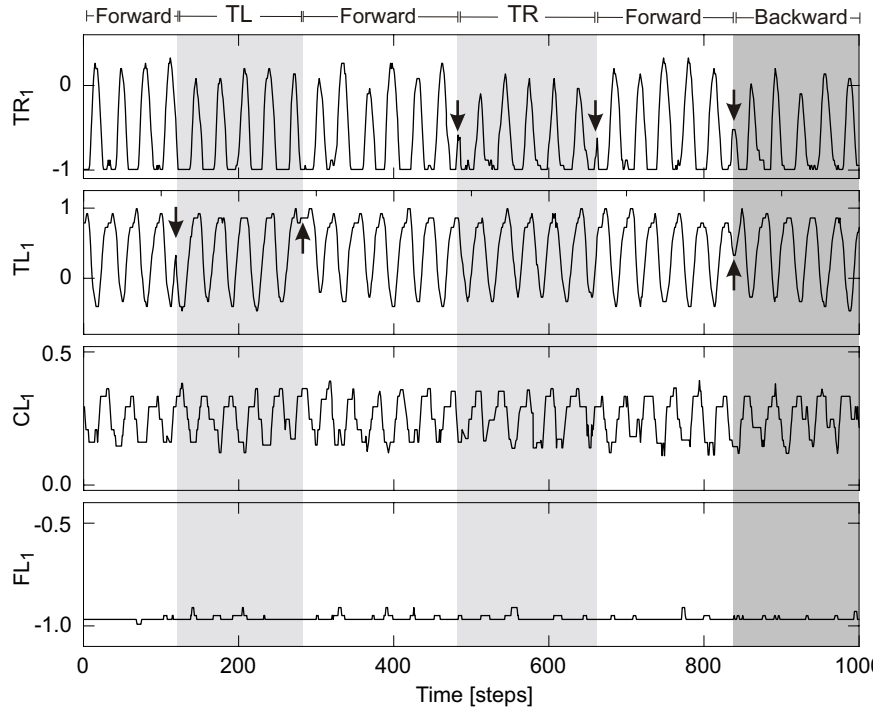

Fig. 14. The angle sensor signals of the TC-joint (TR_1) of the right front leg and the TC-joint (TL_1), CTr-joint (CL_1), and FTi-joint (FL_1) of the left front leg. Arrows indicate the modification of the movement of the TC-joints, which allows the walking machine to perform forward (white area) and backward (dark gray area) walking in a straight-line as well as spot turning to left (TL) and right (TR) direction (light gray area). Note that the input parameters controlling each walking behavior were given as shown in Table 1, e.g., TL was stimulated by setting $I_1 = 0$, $I_2 = 1.0$, $I_3 = 1$, $I_4 = 1.0$, $I_5 = -1.0$.

In order to see the reaction of the TC-joints, the CTr-joints, and the FTi-joints when the reflex behavior is triggered, we initially let the walking machine perform forward diagonal motion to the left ($FDiL$) where the oscillations of all joints can be observed, the result of which is shown in Fig. 16. After around 200 time steps, the reflex behavior was activated where I_1 was set to 1; i.e., the machine was turned in an upside-down position. Accordingly, all TC-joints rotated into the minimum position (compare Fig. 5) except TL_1 and TR_1 as well as all CTr- and FTi- joints, which turned into maximum position (compare Fig. 5). Note that we set TL_1 and TR_1 differently from other TC-joints because this configuration makes the walking machine stand stable in an upside-down position.

As demonstrated, the modular neural controller successfully accomplishes omnidirectional walking as well as the reflex behavior, which can be controlled through the five input neurons. Additionally, the controller is effective. That is, without any modifications of its internal parameters and structure, it can be applied to walking machines having different morphologies, e.g., different

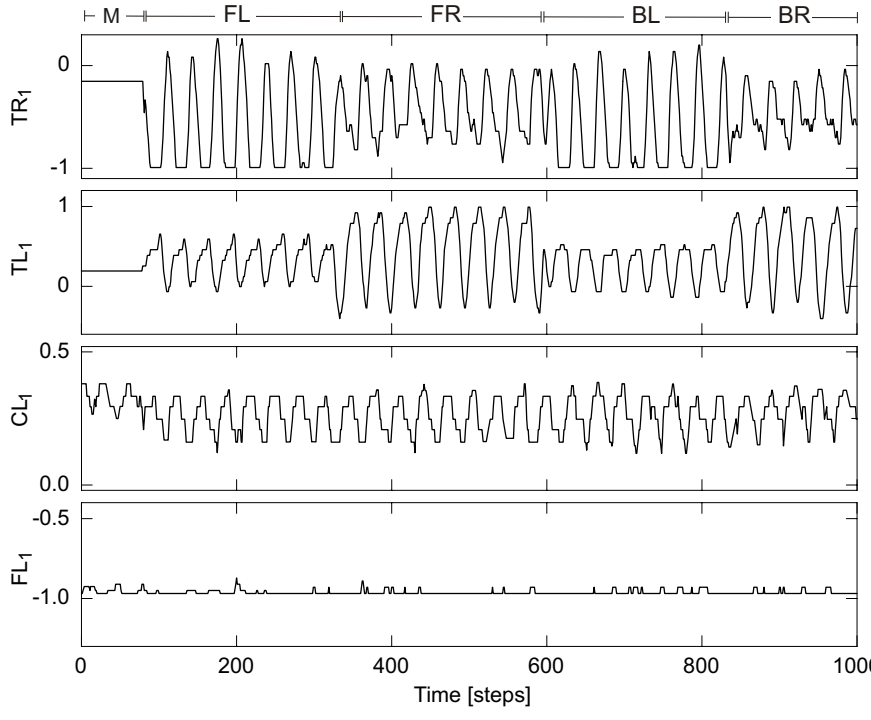

Fig. 15. The angle sensor signals of the TC-joint (TR_1) of the right front leg and the TC-joint (TL_1), CTr-joint (CL_1), and FTi-joint (FL_1) of the left front leg. Varying only amplitude of the TC-joints will provide different actions: marching in place (M), forward and backward curve walking to the left (FL, BL) and the right (FR, BR). Here, the input parameters ($I_1 = 0$, $I_2 = 1.0$, $I_3 = 1$) generating the different walking behaviors are fixed except I_4 and I_5. They were selected from the plot of the input space (see Fig. 11) according to each desired walking behavior. They were set to $I_4 = -0.3$ and $I_5 = -0.8$ for FL, $I_4 = -0.8$ and $I_5 = -0.3$ for FR, $I_4 = 0.3$ and $I_5 = 0.8$ for BL and $I_4 = 0.8$ and $I_5 = 0.3$ for BR.

body structure and the orientation of their legs (see Figs. 1 and 2). Inspite of such changes the controller still enables them to perform almost the same desired behaviors. For a demonstration of this, we refer the reader to the video clip at http://www.nld.ds.mpg.de/∼poramate/RAS/OmniR.mpg.

5 Sensor-driven neural control

The neural locomotion control described above utilizes a modular concept. It is constructed with different small neural modules, which generate various walking behaviors in accordance with the given input parameters. Furthermore, on the basis of this modular concept, the controller can be coupled with other neural modules, e.g., neural preprocessing of sensory signals. By coupling these two, we receive a so-called sensor-driven neural controller. The diagram of the controller is shown in Fig. 17 (see Fig. 28 for the complete

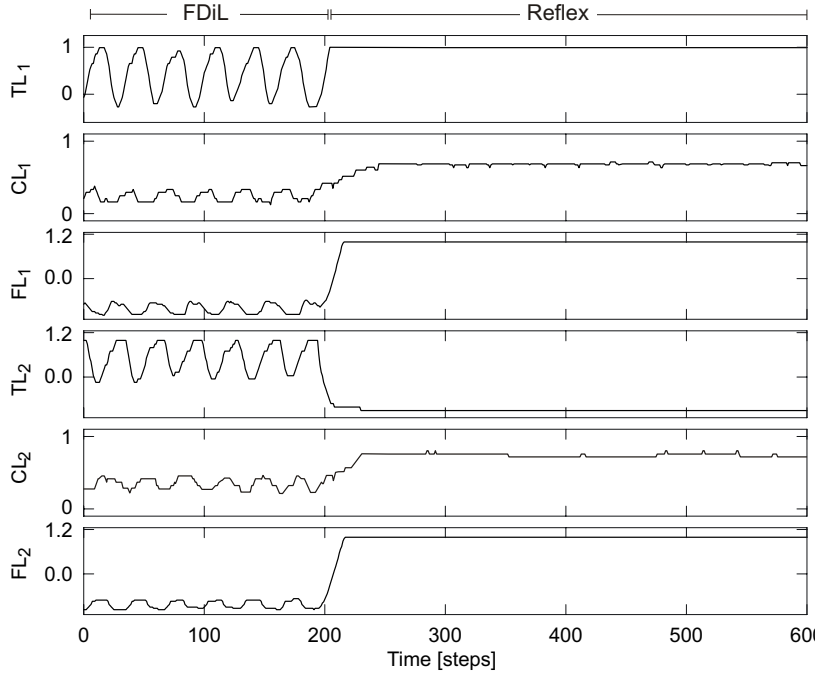

Fig. 16. The angle sensor signals of the TC-joints ($TL_{1,2}$), the CTr-joints ($CL_{1,2}$), and the FTi-joints ($FL_{1,2}$) of the left front L_1 and middle L_2 legs (compare Fig. 5). Other joints having similar patterns are not shown. During the first period, the machine walked diagonally in the forward direction to the left and after around 200 time steps the reflex behavior was activated which inhibited the oscillations of all joints. Note that the input parameters $I_{1,...,5}$ were set as described in Table 1 to produce the desired actions.

network structure).

In order to create an effective sensor-driven neural controller, we make use of the signals of the IR sensors and the UD sensor implemented on the AMOS-WD06 (see Fig. 1). These sensory data are used to provide environmental information for our sensor-driven robot system. Nonetheless, the raw sensory signals require preprocessors to eliminate the sensory noise as well as to shape the sensory data for activating the appropriate reactive behavior. Thus, in the following sections, we will describe the sensory preprocessing units together with their performance followed by a demonstration of sensor-driven behaviors.

6 Neural preprocessing

We will now present four neural preprocessing units which use the dynamic properties of recurrent neural networks. The first unit ($NP1$) is used to pre-

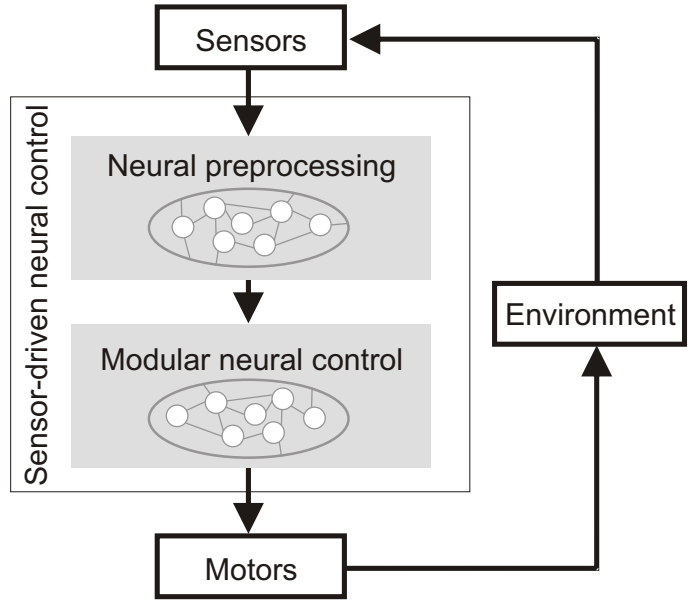

Fig. 17. Diagram of sensor-driven neural control. The controller acts as an artificial perception-action system, i.e., the sensor signals go through the neural preprocessing modules into the modular neural control module (compare Fig. 4) which commands the actuators. As a result, the robot's behavior is generated by interacting with its environment in a sensorimotor loop.

process the upside–down detector (UD) signal, the output of which triggers the reflex action. The second unit ($NP2$) does preprocessing of one IR sensor signal where this sensor is mounted at the rear part (IR_{RP}) of the walking machine. The output of this second unit regulates the walking speed of the machine. The third unit ($NP3$) is applied to the IR sensors implemented at the middle legs ($IR_{R2,L2}$) while the last unit ($NP4$) is for those installed on the front part ($IR_{FR,FL}$) and at the front legs ($IR_{R1,L1}$). The outputs of both units serve to drive a reactive obstacle avoidance behavior. The details of all these neural preprocessing units are described in the following.

6.1 Neural preprocessing of the reflex and speed control signals

We construct the neural preprocessing units of the UD and IR_{RP} signals in the same manner. Each of them is configured as a hysteresis element using a single neuron with a "super-critical" self-connection (> 4) [60]. The discrete dynamics of the single neuron with a self-connection is given by:

$$a_{NPi}(t+1) = w_{NPi}\sigma(a_{NPi}(t)) + \Theta_{NPi} + C_{NPi}V_{NPi} \quad i = 1, 2, \tag{3}$$

using a standard sigmoidal transfer function to yield the output:

$$o_{NPi} = \sigma(a_{NPi}) = \frac{1}{1 + e^{-a_{NPi}}}, \tag{4}$$

where $i = 1$ and 2 are applied for the neural preprocessing of the UD signal and the IR_{RP} signal, respectively. V_{NPi} is the output voltage signal of the sensor which is linearly mapped onto the interval $[0, 1]$. Θ_{NPi} stands for the threshold (bias term), w_{NPi} is a self-connection weight, and C_{NPi} represents a positive amplification factor of the input signal.

On the basis of its well-understood functionality [37,47,60], the neural parameters (Θ_{NPi}, w_{NPi}, C_{NPi}) were manually designed to obtain an appropriate hysteresis of each sensory signal. First we set $C_{NP1,NP2}$ to a high value, i.e., 6.0, to amplify the raw sensory signals. Then we adjusted the amplified signals such that they will cross forward and backward through the hysteresis domain [60]. To do so, we chose $\Theta_{NP1,NP2} = -6.0$. Consequently, the amplified signals sweep over the input interval between -6.0 and 0.0. Finally, we tuned the self-connection weight of each neural preprocessor to derive a reasonable hysteresis interval on the input space; i.e., $w_{NP1} = 7.2$, $w_{NP2} = 8.6$. The width of the hysteresis is proportional to the strength of the self-connections (compare Figs. 18e and 18f). This effect determines the duration of the response to an environmental stimulus; i.e., the wider the hysteresis, the longer the action perseveres (here, the reflex and fast walking behaviors). The resulting neural preprocessors of both sensory signals together with their hysteresis effect are presented in Fig. 18.

We directly feed the output of the UD preprocessor o_{NP1} to its target neuron I_1 in the neural control module (Fig. 18a). As a result, the walking machine will perform the reflex action when o_{NP1} shows high activation (≈ 1.0; meaning that the walking machine is turned into the upside-down position). This reflex action will be deactivated when o_{NP1} shows low activation (≈ 0.0; meaning that the walking machine is returned into its normal walking position).

On the other hand, the output of the IR_{RP} preprocessor o_{NP2} is used to modify all synapses of the neural oscillator network following Eqs. 5, 6, 7:

$$w_{11,22} = 1.5 + 0.5 o_{NP2}, \tag{5}$$
$$w_{12} = 0.4 + o_{NP2}, \tag{6}$$
$$w_{21} = -0.4 - o_{NP2}, \tag{7}$$

where $w_{11,22}$, w_{12}, and w_{21} are the connection weights between the neurons of the oscillator network (Fig. 18b). Using this sensor-modified synaptic mechanism, all connection weights will be changed in accordance with the activation

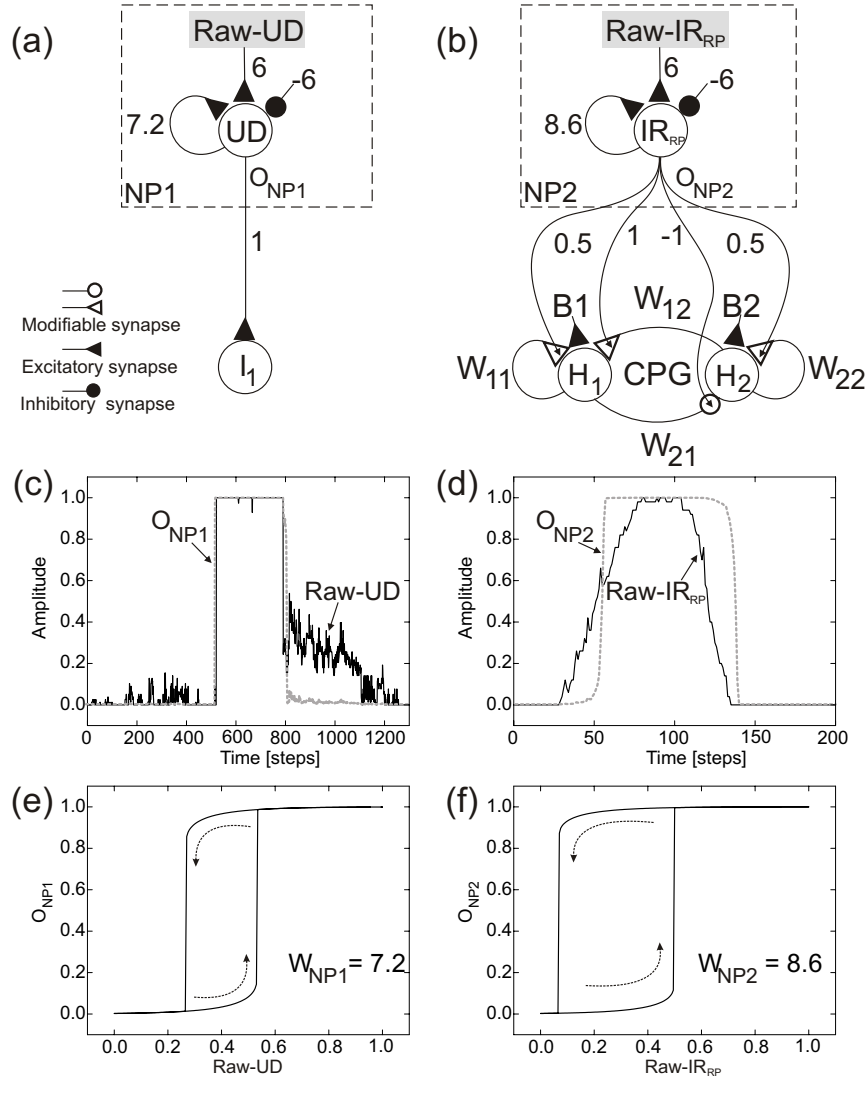

Fig. 18. (a) and (b) show recurrent neural preprocessing of the UD signal (dashed frame) and the IR_{RP} signal (dashed frame), respectively. The output of the UD signal preprocessor is fed to I_1 while the output of the IR_{RP} signal preprocessor is used to change the synaptic strength of the neural oscillator. (c) and (d) show the input signals (Raw-UD, $Raw - IR_{RP}$ (solid line)) before preprocessing and the output signals (o_{NP1}, o_{NP2} (dashed line)) after preprocessing. (e) and (f) show the hysteresis effect between the input and output of the UD and IR_{RP} signal preprocessors, respectively. The raw input of both signal preprocessors varies between 0.0 and 1.0 while their output gets low (≈ 0.0) and high (≈ 1.0) activations at different points. The UD output will show high activation when the input increases to values above 0.54. On the other hand, it will show low activation when the input decreases below 0.26. For the IR_{RP} output, it will show high activation when the input increases to values above 0.49 while it will show low activation when the input decreases below 0.06.

of the IR_{RP} signal. As a consequence, the walking speed of the machine will be increased when o_{NP2} gets high activation (≈ 1.0; meaning that there is

an object approaching from behind the walking machine) and vice versa. Due to the large hysteresis interval (Fig. 18f), once the walking machine has been stimulated to perform fast walking, it will keep to this action for a few steps even if the activating stimulus is removed (e.g., the raw-IR signal gradually decreases, see Fig. 18d). This way, predatory escape behavior can be simulated. That is, without stimulus the machine walks with normal speed (6.5 cm/s) but if a stimulus is generated by the attack of a predator (e.g., humans or other robots) from the rear, the machine will increase its speed (up to 25.6 cm/s) to escape.

6.2 Neural preprocessing of the walking pattern control signals

Neural preprocessing of the six IR sensor signals ($IR_{FR,FL,R1,R2,L1,L2}$), controlling walking patterns, consists of two independent modules. One module ($NP3$) preprocesses the data of $IR_{R2,L2}$. The output signals of this module will activate lateral walking and also control the direction to the left or the right with respect to the sensory signals. The other module ($NP4$) is for the $IR_{FR,FL,R1,L1}$ signals. Its outputs will make the walking machine turn to avoid obstacles and prevent it from getting stuck in a corner or a deadlock situation.

The neural preprocessing unit $NP3$ is designed as an XNOR network [9]. This neural network has two input neurons, two hidden neurons, and one output neuron. All neurons are modeled as a standard additive neuron (Eq. 1) with the standard sigmoidal transfer function (Eq. 4). The network was trained for 2000 time steps by using the backpropagation algorithm where the learning rate was set to 0.7 (for more details of this algorithm, see [54,68]). Here, the IR signals ($IR_{R2,L2}$) are linearly mapped onto the interval $[0, 1]$ and provided to the input neurons of the network. These sensory signals have to be first filtered before feeding them to the XNOR network. Therefore we again apply the hysteresis effect of the recurrent neural network to eliminate sensory noise. Thus, the input neurons of the XNOR network are configured as the hysteresis elements which are similar to those shown in Fig. 18b. The modified XNOR network together with its weights is shown in Fig. 19 and its desired output (o_{NP3}) corresponding to the given inputs (the filtered IR signals; $I_{NP3_{L2}}$, $I_{NP3_{R2}}$) is presented in Table 2.

We directly feed the output of this preprocessor (o_{NP3}) to its target neuron I_2 in the neural control module (Fig. 19). Consequently, the lateral walking pattern will be activated when o_{NP3} gets low activation ≈ 0.0 meaning that one of the IR_{R2} and IR_{L2} sensors gives a high output signal ≈ 1.0 (compare Table 2). Furthermore, the output signal of the hysteresis element $I_{NP3_{R2}}$ serves to control the lateral direction through I_3 of the neural controller. That

[9] It has function as an exclusive NOR gate.

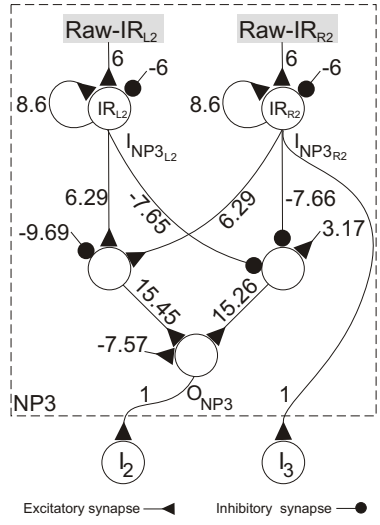

Fig. 19. Neural preprocessing unit $NP3$ of the IR signals $IR_{R2,L2}$ (dashed frame). It is created as an XNOR network with a self-connection at its input neurons. As a result, the input neurons function as hysteresis elements, the effect of which is similar to that shown in Figs. 18d and 18f. The output of the network o_{NP3} is fed to I_2 and the output of one hysteresis element $I_{NP3_{R2}}$ is connected to I_3 (see text for details).

Table 2
The input-output characteristic for the XNOR network.

$I_{NP3_{L2}}$	$I_{NP3_{R2}}$	o_{NP3}
≈ 1.0	≈ 1.0	≈ 1.0
≈ 1.0	≈ 0.0	≈ 0.0
≈ 0.0	≈ 1.0	≈ 0.0
≈ 0.0	≈ 0.0	≈ 1.0

is, once the lateral walking pattern is activated, the machine will walk laterally to the right as long as the $I_{NP3_{R2}}$ signal shows low activation ≈ 0.0; otherwise it will walk laterally to the left (compare Table 1). Hence, the machine will perform lateral walking to the right if there is an obstacle at its left middle leg and vice versa. In special conditions, e.g., detecting obstacles on both lateral sides during walking forward, the IR_{R2} and IR_{L2} sensors will give high output activations at the same time resulting in the inhibition of lateral motions. The walking machine then continues to walk forward.

The last neural preprocessing unit $NP4$, which will complete the reactive obstacle avoidance behavior, is built based on the minimal recurrent controller (MRC) [37,62]. It consists of two mutually inhibiting neurons with self-connection (see Fig. 20a). This controller has been originally developed for controlling a miniature Khepera robot [55], which is a two wheeled platform.

Here, it serves as a neural preprocessor for controlling the turning directions of the machine to avoid obstacles and to escape from a corner and even a deadlock situation.

All neurons of the preprocessing unit are modeled as the standard additive neuron (Eq. 1) with the hyperbolic tangent transfer function (Eq. 2). The connection weights of the network were manually adjusted on the basis of their well understood functionalities [37,47]. First, the weights from the input to the output units were set to a high value to amplify the sensory signals, i.e., 7.0. Then the self-connection weights of the output neurons were manually tuned to derive a reasonable hysteresis interval on the input space. Recall that the width of the hysteresis is proportional to the strength of the self-connections. In this case, the hysteresis effect determines the turning angle in front of the obstacles for avoiding them, i.e., the wider the hysteresis, the larger the turning angle. Both self-connections are set to 5.4 to obtain a suitable turning angle of the AMOS-WD06. Finally, the recurrent connections between output neurons were symmetrized and manually adjusted to -3.55. Such inhibitory recurrent connections are formed as a so-called *even loop* [59], which also shows hysteresis phenomenon (see Fig. 20c). In general conditions, only one neuron at a time is able to produce a positive output, while the other one has a negative output, and vice versa. This guarantees the optimal functionality for avoiding obstacles and escaping from sharp corners. The resulting network and its hysteresis effect are shown in Fig. 20.

The four IR sensor signals used as the inputs to the network are mapped onto the interval $[-1, +1]$, with -1 representing "no obstacles", and $+1$ "an obstacle is near". Nonetheless, the network has only two inputs ($I1_{NP4}$, $I2_{NP4}$) and no internal neurons. Thus, one input $I1_{NP4}$ corresponds to the mean value of the two left IR sensors (IR_{FL}, IR_{L1}) and the second input $I2_{NP4}$ to that of the two right IR sensors (IR_{FR}, IR_{R1}). The set-up parameters cause that the network can eliminate the noise of the sensory signals. It can even determine the turning angle as well as the turning direction. As a consequence, the walking machine is able to avoid the obstacles and escape from corners as well as deadlock situations. By applying $o1_{NP4}$ and $o2_{NP4}$ of the preprocessing network to their target neurons I_5, I_4 in the neural control module, the behavior of the walking machine can be autonomously switched; for instance, switching from walking forward to turning left when there are obstacles on the right, or vice versa. The network output also determines in which direction the walking machine should turn in corners or deadlock situations depending on which sensor side has been previously active. In special situations, like walking toward a wall, both sides (right and left) of IR sensors might get positive outputs at the same time resulting in the reversion of the motor signals of all TC-joints. The machine then walks backward. During walking backward, the activation of the sensory signal of one side might be still active while the other might be inactive. Correspondingly, the walking machine will turn into the opposite

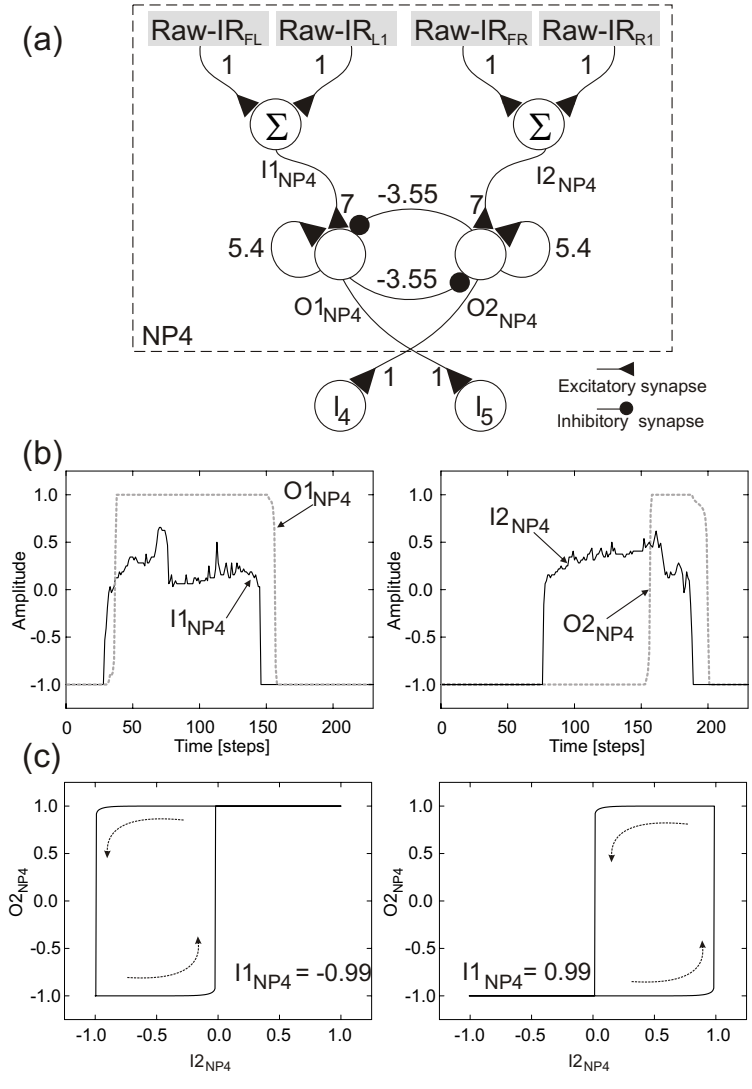

Fig. 20. (a) Neural preprocessing unit $NP4$ of the IR signals $IR_{FR,FL,R1,L1}$ (dashed frame). It receives the mean values of the left ($I1_{NP4}$) and right ($I2_{NP4}$) IR signals and provides outputs ($o1_{NP4}$ and $o2_{NP4}$) which control the spot turning direction of the walking machine through the corresponding neuron $I_{4,5}$ of the neural control. (b) Mean sensory signals ($I1_{NP4}$ and $I2_{NP4}$) before preprocessing and output signals ($o1_{NP4}$ and $o2_{NP4}$) after preprocessing. Due to the inhibitory synapses between two output neurons and the high activity of $o1_{NP4}$, $o2_{NP4}$ is still inactive although $I2_{NP4}$ becomes activated at around 75 time steps. At around 150 time steps, the switching condition between $o1_{NP4}$ and $o2_{NP4}$ occurs because $I1_{NP4}$ becomes inactivated, meaning "no obstacles detected" while $I2_{NP4}$ is still active, meaning "obstacles detected". (c) Different hysteresis domains of the input $I2_{NP4}$ for the output neuron $o2_{NP4}$ of the network with $I1_{NP4}$ fixed. On the other hand, if such a condition occurs for $o1_{NP4}$, $I1_{NP4}$ will show the same hysteresis effect as $I2_{NP4}$ does.

direction of the active signal and it can finally leave from the wall.

7 Sensor-driven behaviors

In the previous section, the neural control and preprocessing of walking machines have been reported. To demonstrate their capabilities for controlling the reactive behaviors of the walking machine in a real environment, several experiments were carried out and performed on the six-legged walking machine AMOS-WD06 with its mobile system [10] .

The first experiment was to show the reflex behavior together with sensory-motor data. After that an escape behavior is presented and followed by obstacle avoidance in different situations. Note that, here, the limitation of the IR sensor system in detecting objects with respect to different distances between the sensors and an object as well as different object dimensions is not explicitly described because it has been reported in our previous work [50].

7.1 Reflex behavior

Reflex behavior [4] is rapid, stereotyped response which is generally found in an animal. Such behavior is triggered by a certain environmental stimulus and it remains as long as the stimulus is given. Reflexes enable an animal to rapidly adapt its behavior to unforeseen environmental changes. Reflexes are usually applied for tasks such as postural control, withdrawal from painful stimuli, and the adaptation of gait to rough terrain.

Inspired by this concept, we have implemented a protection reflex on the AMOS-WD06 in a way that it will be triggered as soon as the machine is turned into an upside-down position. As a consequence, it stands still in this position as long as the stimulus (UD signal) is presented. This reflex response together with real time sensory-motor data is shown in Fig. 21.

As shown in Fig. 21, the AMOS-WD06 walked forward at the beginning (photo (1)). During walking forward, the CTr-joints and the TC-joints performed periodic movements while the FTi-joints were inhibited to stay in the flexed position. After around 280 time steps, the walking machine was turned into an upside-down position (photos (2) and (3)) resulting in the inhibition of all

[10] The sensor-driven neural controller (see Fig. 28) was implemented on the mobile processor unit (the PDA communicating with the MBoard via RS232 interface). During experiments, we use battery packs for powering the robot system which can run up to 35 minutes.

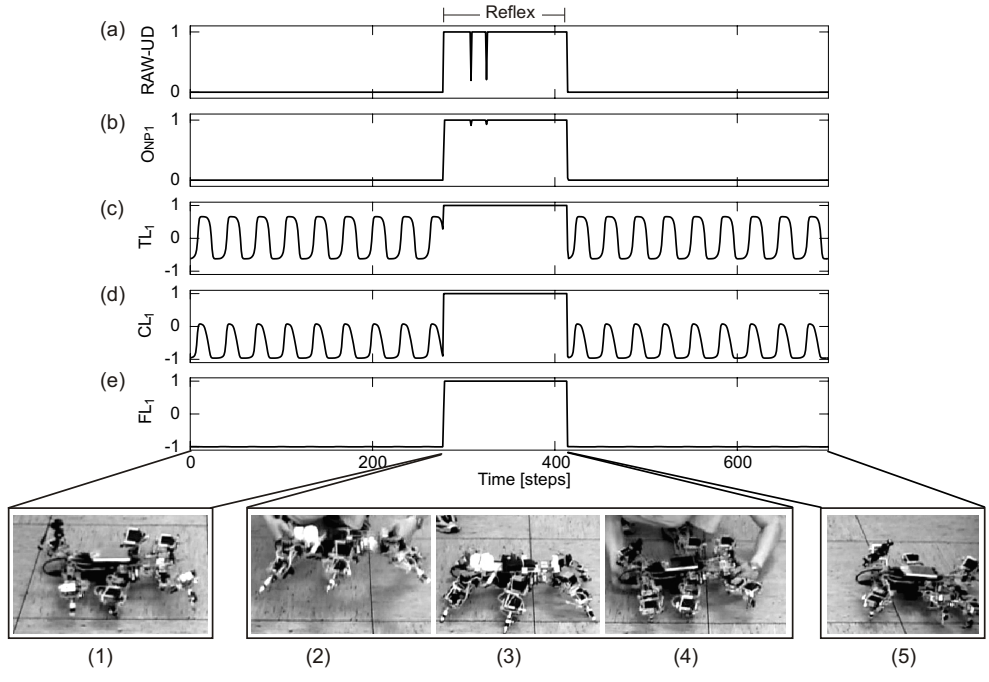

Fig. 21. (a) Raw-UD signal. (b) Preprocessed UD signal. (c)–(e) Motor neuron signals of the TC-joint (TL_1), CTr-joint (CL_1), and FTi-joint (FL_1) of the left front leg, respectively. Signals of other joints having similar patterns to those of the left front leg are not shown. A series of photos of the reflex behavior is shown below. The video clip of this experiment can be seen at http://www.nld.ds.mpg.de/∼poramate/RAS/Reflex.mpg.

joints. Then it was returned into its normal walking position at around 420 time steps (photo (4)) and it continued walking afterwards (photo (5)).

From this experimental result, one can see that such a reflex behavior can be simply activated by the UD signal through the sensor-driven neural controller (see Fig. 28) as follows. At first the raw-UD signal is filtered via the neural preprocessing unit $NP1$ (see Fig. 18a) and then the filtered signal drives the reflex behavior by inhibiting all motor neurons. The reflex behavior presented here is used to protect the walking machine from damage when it is turned into the upside-down position.

7.2 Escape behavior

Escape behavior can be described as a fixed-action pattern [4], which is a time-extended response pattern activated by a stimulus. That is, the action perseveres for longer than the stimulus itself. In contrast to the reflex behavior, such fixed-action patterns (here, the escape behavior) are usually more complex and specific than reflexes. The intensity and duration of the response are not controlled by the strength and duration of the stimulus. In fact, once

a fixed-action pattern has been activated, it will be performed even if the activating stimulus is removed.

Here, we reproduce such a behavior on the AMOS-WD06. The escape behavior will be activated as soon as the IR_{RP} sensor detects an object. As a consequence, the AMOS-WD06 increases its walking speed, as if it escapes from an attack. This action will be preserved for a few steps even if the activating stimulus has already been removed. The simulated escape behavior together with real time sensory-motor data is shown in Fig. 22.

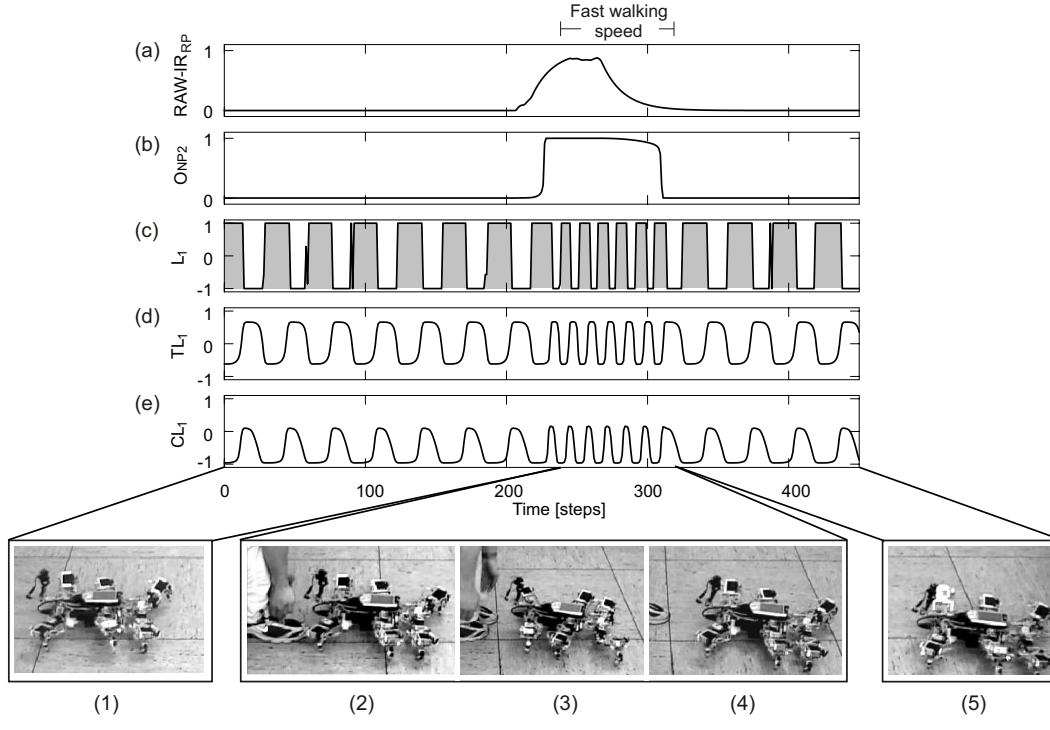

Fig. 22. (a) Raw-IR_{RP} signal. (b) Preprocessed IR_{RP} signal. (c) Foot contact sensor signal of the front left leg. (d), (e) Motor neuron signals of the TC-joint (TL_1) and the CTr-joint (CL_1) of the left front leg, respectively. The signals of the FTi-joint (FL_1) similar to the one shown in Fig. 21e and also of the other joints having similar patterns to those of the left front leg are not shown. A series of photos of the escape behavior is presented below. The video clip of this experiment can be seen at http://www.nld.ds.mpg.de/∼poramate/RAS/Escape.mpg.

As shown in Fig. 22, the AMOS-WD06 walked forward with its normal speed (6.5 cm/s) at the beginning (photo (1)). During walking forward, the CTr-joints and the TC-joints performed periodic movements while the FTi-joints were inhibited to stay in the flexed position. After around 230 time steps, there was an attack by its predator, i.e., a human (photo (2)), leading to high output activation of the IR_{RP} sensor. As a consequence, the AMOS-WD06 performed the escape behavior (photos (3) and (4)) by increasing its walking speed up to 25.6 cm/s where the periodic signals of the motor neurons as well as the foot contact sensor signals oscillated at a higher frequency. Then it

returned to its normal walking speed at around 300 time steps meaning that it was far enough from its predator (photo (5)).

From this experimental result, one can see that such an escape behavior can be induced by the sensor-driven neural controller (see Fig. 28) in the way that the raw-IR_{RP} signal is first preprocessed via the neural preprocessing unit $NP2$ (see Fig. 18b). Then the preprocessed signal triggers the escape behavior [29,31] by simply increasing the step frequency of the walking machine where the step amplitude is also slightly increased (compare Fig. 7d).

7.3 Obstacle avoidance behavior

Obstacle avoidance behavior is realized in most animals because they need to do this in cluttered real environments for example during foraging [47]. This behavior can be classified as orientational responses [4] driving an animal away from a stimulus (negative tropism).

Similarly, such a behavior has been implemented on the AMOS-WD06 in order to enable it to avoid obstacles as well as to protect it from getting stuck in corners or deadlock situations during walking in an unknown environment. The AMOS-WD06 will avoid obstacles as soon as one of the IR sensors installed at its legs ($IR_{R1,R2,L1,L2}$) and its front part ($IR_{FR,FL}$) detects them. As a consequence, it will actively perform spot turning, walking away in a diagonal direction or even walking backward in response to the sensory inputs. Here three different situations showing different obstacle avoidance behaviors are illustrated in Figs. 23–25.

As shown in Fig. 23, the AMOS-WD06 walked forward at the beginning (photo (1)). During walking forward, the CTr-joints and the TC-joints performed periodic movements while the FTi-joints were inhibited to stay in the flexed position. After around 230 time steps, the obstacles were placed beside the machine (photo (2)). Consequently, the activation of the middle left sensor (IR_{L2}) got high ≈ 1.0 causing a reduction in the activation of the neural preprocessing output (o_{NP3}, cf. Sect. 6.2). As a result, the FTi-joints became activated making the AMOS-WD06 perform forward diagonal walking to the right (photos (3) and (4)). Finally, after avoiding the obstacle it returned to normal forward walking at around 350 time steps.

Figure 24 shows another kind of obstacle avoidance behavior where the obstacles were placed on the left front at around 220 time steps (photo (2)). Consequently, the front left sensor (IR_{L1}) was active which made the TC-joints of all right legs turn to the opposite direction leading to spot turning to the right (photo (3)). During making a right turn, the IR_{L1} became inactive for a few steps and then it was active again causing the AMOS-WD06 to walk

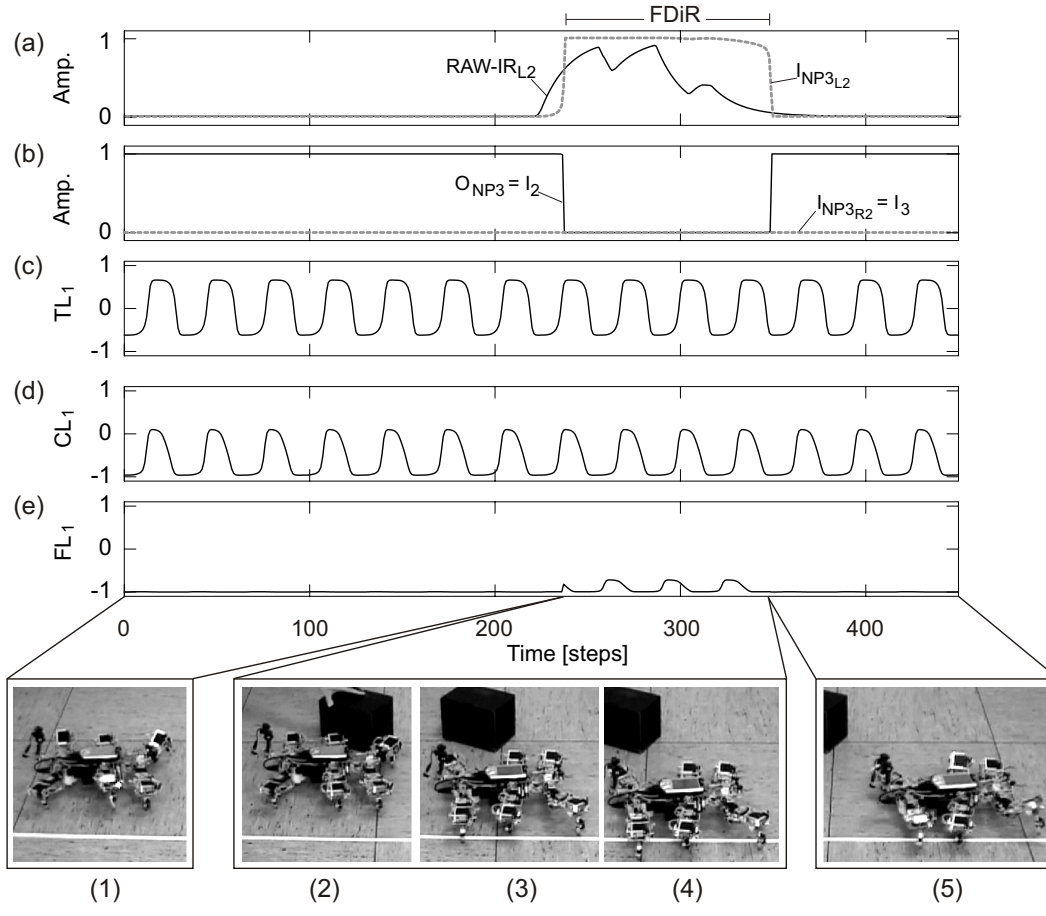

Fig. 23. (a) Raw-IR_{L2} signal (solid line) before preprocessing and the output signal ($I_{NP3_{L2}}$, dashed line) after preprocessing (compare Fig. 19). (b) The signals controlling the lateral motions (cf. Sect. 6.2 and compare Fig. 19). (c)–(d) The motor neuron signals of the TC-joint (TL_1), the CTr-joint (CL_1), and the FTi-joint (FL_1) of the left front leg, respectively. The signals of other joints having similar patterns to those of the left front leg are not shown. A series of photos of the obstacle avoidance behavior is shown below. Note that *Amp.* means the amplitude of neuron activation and $FDiR$ stands for a forward diagonal walking to the right. The video clip of this experiment can be seen at http://www.nld.ds.mpg.de/∼poramate/RAS/ObstacleAvoidance1.mpg.

forward for a few steps and then to turn right again until around 310 time steps. After that it performed a forward diagonal walking to the right (photo (4)) until around 380 time steps due to the middle left sensor (IR_{L2}) being high ≈ 1.0. Eventually, the AMOS-WD06 was able to avoid the obstacles and continued to walk forward (photo (5)).

In a special situation shown in Fig. 25, e.g., the obstacles were placed in front of the AMOS-WD06 (photo (2)), both front sensors ($IR_{FR,FL}$) were almost simultaneously active. Thus the TC-joints of all legs moved in the reverse direction which caused the AMOS-WD06 to walk backward at around 150 time steps. While walking backward the IR_{FL} was still active but the other

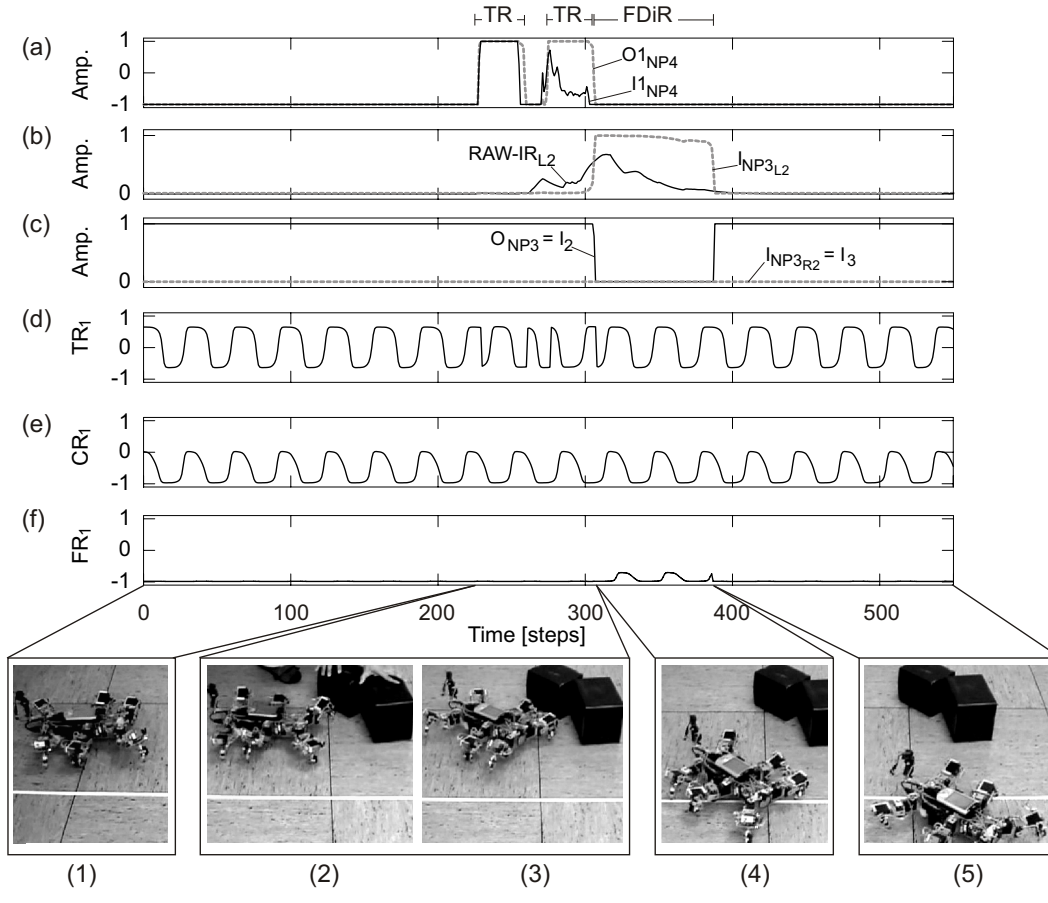

Fig. 24. (a) Mean sensory signal on the left side ($I1_{NP4}$, solid line) before preprocessing and output signal ($o1_{NP4}$, dashed line) after preprocessing (compare Fig. 20). (b) Raw-IR_{L2} signal (solid line) before preprocessing and output signal ($I_{NP3_{L2}}$, dashed line) after preprocessing (compare Fig. 19). (c) Signals controlling the lateral motions (cf. Sect. 6.2 and compare Fig. 19). (d)–(f) Motor neuron signals of the TC-joint (TL_1), the CTr-joint (CL_1), and the FTi-joint (FL_1) of the left front leg, respectively. The signals of other joints having similar patterns to those of the left front leg are not shown. A series of photos of the obstacle avoidance behavior is shown below. Abbreviations are: $Amp.$ = the amplitude of neuron activation, TR = turn right, $FDiR$ = a forward diagonal walking to the right. The video clip of this experiment can be seen at http://www.nld.ds.mpg.de/∼poramate/RAS/ObstacleAvoidance2.mpg.

was already inactive forcing the machine to turn to the right until, eventually, it was able to avoid the obstacles (photos (3)–(5)).

As demonstrated, the sensor-driven neural controller (see Fig. 28) is suitable to successfully solve the obstacle avoidance task. Additionally, the controller can even protect the machines from getting stuck in corners or deadlock situations. This is demonstrated in a video clip at http://www.nld.ds.mpg.de/ ∼poramate/RAS/AllObstacleAvoidance.mpg. Thus, due to this functionality, the walking machines can autonomously perform exploration. We en-

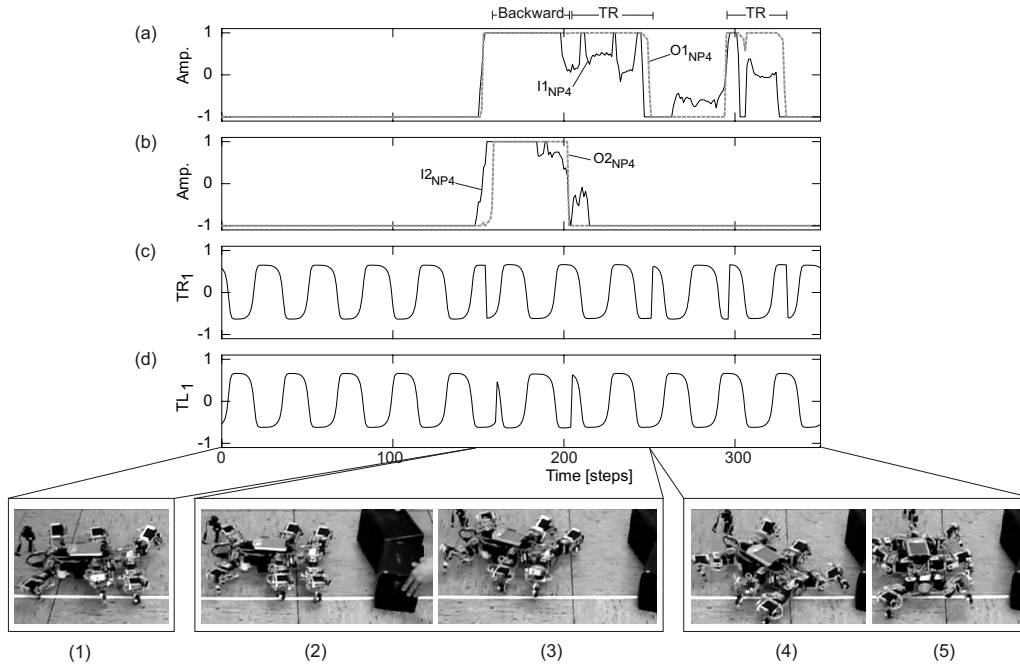

Fig. 25. (a) Mean sensory signals on the left side ($I1_{NP4}$, solid line) before preprocessing and output signal ($o1_{NP4}$, dashed line) after preprocessing (compare Fig. 20). (b) Mean sensory signals before ($I2_{NP4}$, solid line) and after ($o2_{NP4}$, dashed line) preprocessing on the right side (compare Fig. 20). (c), (d) Motor neuron signals of the TC-joints of the right (TR_1) and left (TL_1) front legs, respectively. Abbreviations are: *Amp.* = amplitude of neuron activation, TR = turn right. The video clip of this experiment can be seen at http://www.nld.ds.mpg.de/∼poramate/RAS/ObstacleAvoidance3.mpg.

courage readers to see more demonstration showing all reactive behaviors at http://www.nld.ds.mpg.de/∼poramate/RAS/AMOSWD06MedleyDemo.mpg.

8 Discussion and comparison with other walking control techniques

Here, we briefly discuss some remaining issues concerning the walking machines and their controller, because most of the relevant discussion points have been treated in the above sections. The proposed walking machines were designed with different morphologies analogous to walking animals. They were constructed in a straightforward way as mechatronic systems consisting of several sensors and actuators. They were also simulated in a physical simulation environment with the intention to develop and test neural controllers before implementation in the physical walking machines.

The complete sensor-driven neural controller of the walking machines was modeled with an artificial neural network using discrete-time dynamics. Part of it was developed by realizing dynamical properties of recurrent neural net-

works. The controller was designed as a modular structure composed of two main modules: the neural locomotion control and the neural sensory preprocessing networks. The neural locomotion control, based on a CPG, generates omnidirectional walking and drives the reflex behavior while the neural preprocessing networks filter sensory noise as well as manage the sensory data for activating an appropriate reactive behavior. This kind of the sensor-driven neural controller is different from many others which have been developed for walking machines.

Several successful physical walking machines employ classical control, like posture control [46,64], subsumption architecture [18,16], and other AI control algorithms [8,9,36]. The mammal-like machine BISAM [41] uses a method combining aspects of classical- (joint trajectory control) and biologically inspired reflex-based control for locomotion behavior robust against disturbances and applicable on rough terrain. Similarly, such reflex-based techniques have been also applied to other robots, e.g., the TEKKEN robot [43], the RHex robot [45], the Scorpion robot [44], and the TITAN-VIII robot [73], in order to give the controller the flexibility to handle also rough terrain and even a slip area [73]. In contrast to BISAM, their basic walking patterns were generated by a kind of central pattern-generating system in analogy to the CPGs found in animal locomotion. One impressive work on biologically inspired locomotion control has been recently reported by Ijspeert et al. [40]. There, a CPG model consisting of several nonlinear oscillators together with classical PD control was employed to modulate three locomotion modes (swimming, serpentine crawling, and walking) in a salamander-like robot, Salamandra robotica. On the other hand, the Terry I and II hexapod robots have no central oscillator controlling their leg movement. Instead, they use a combination of classical control and distributed artificial neural networks called "Walknet" [21] inspired by stick insect locomotion. For the generation of advanced locomotion control, Bongard et al. [15] presented an active process that allows a robot to generate successful motor patterns for locomotion, before and after damage, through autonomous and continuous self-modeling (internal model). This algorithm was tested on a Starfish-like walking machine where the machine was able to perform forward motion adaptively changing its gait to compensate for simulated injuries. In other words, if the machine has been damaged, it could sense the problem and attempt to compensate for this by generating a new walking behavior.

Although these described controllers are mainly designed for locomotion control, most of them do not serve for generating omnidirectional walking or even for producing various reactive behaviors, e.g., escape and obstacle avoidance behaviors. However, if omnidirectional walking behavior is taken into account for locomotion controller design, as presented in [20,25,44,63], then obtaining different walking patterns, such as forward, backward and diagonal motions, is achieved by changing the control from outside rather than autonomously

using the corresponding sensory inputs. Furthermore these omnidirectional locomotion controllers rely on non-neural implementations.

Compared to many of these approaches, we emphasize here the embeddedness and generalization abilities of our sensor-driven neural controller. That is, it can be implemented on the mobile processor (PDA or microcontroller) of the walking machines and it can be applied to a physical six- as well as to an eight-legged walking machine generating omnidirectional walking and a variety of reactive behaviors without altering internal parameters or the structure of the locomotion controller (compare Figs. 26 and 27). Because the design comprises independent modules one can simply replace the neural preprocessing modules of the UD and IR signals with other types of signal processing units to acquire different reactive behaviors, e.g., auditory signal processing for generating sound tropism [49]. Such a sensor-driven neural controller can even be modified for manual operation, e.g., by receiving input signals from a joystick. In addition, the speed control mechanism for escape behavior is also comparable with those found for walking animals where the step amplitude is slightly increased and the step frequency is varied ([30,57,65]). However, there is one restriction of using this controller which is that only one gait can be generated where the diagonal legs are paired and move together.

9 Conclusions

Two different types of physical walking machines were presented together with their physical simulations. The design of their morphologies was based on biological principles. Such robot systems can serve as hardware platforms for studying the coordination of many degrees-of-freedom, for performing experiments with neural controllers, and for the development for artificial perception-action systems.

In this study, (omnidirectional) locomotion of the walking machines is generated by modular neural controllers consisting of three different functional modules: the neural oscillator network, the velocity regulating network, and the phase switching network. The neural oscillator network acts as a CPG for basic rhythmic leg movements and is able to change also the walking speed. The generation of different walking patterns is done by the velocity regulating and the phase switching networks. As a result, this modular neural control can produce at least 11 different walking patterns and a self-protective reflex by using five input neurons. The proposed modular neural control can easily be adapted to control other kinds of walking machines *without changing the internal network structure and its parameters*. We want to emphasize that the controller not only works for the six- and eight-legged walking machines presented here but it can be applied equally efficient to other even-legged robots.

Furthermore, integrating the neural preprocessing of sensor signals provides an effective sensor-driven behavior control based completely on neural network techniques. The preprocessing obtained by a small recurrent neural network, which is robust against sensory noise by utilizing hysteresis phenomena. In addition, the sensor-driven neural controller has been implemented on the embedded system (mobile processor) of the six-legged walking machine. Thus, the machine can autonomously perform the desired behaviors, reflex, escape and obstacle avoidance, with respect to corresponding sensory inputs. More demanding tasks will be related to the use of additional sensors, like photoresistor sensors, to enable the machine to perform also other kinds of positive or negative (photo-)tropisms. We also aim to implement a reflexive neural mechanism [48] to obtain more robust locomotion which enables also walking on rough terrain.

Appendix: The modular neural network of the six- and eight-legged walking machines

The modular neural network has been described in the sections above. Here we present the complete network structure together with its parameters. The network shown in Fig. 26 is used to control omnidirectional locomotion and the reflex behavior of the six-legged walking machine AMOS-WD06 while the one shown in Fig. 27 is constructed for the eight-legged walking machine AMOS-WD08. In addition, the complete structure of the sensor-driven neural control of the AMOS-WD06 is exemplified in Fig. 28.

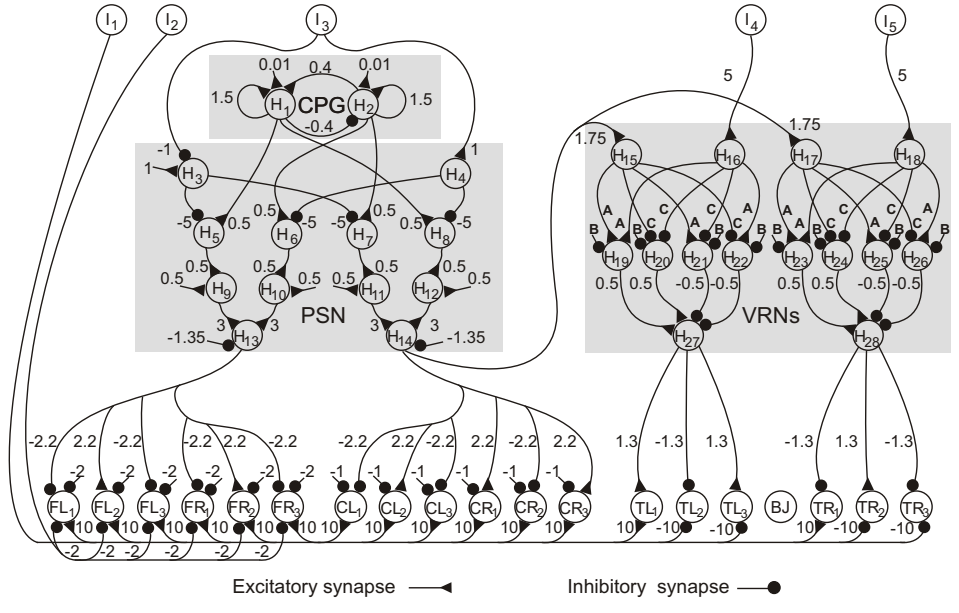

Fig. 26. The modular neural control of the six-legged walking machine AMOS-WD06. Parameters are $A = 1.7246$, $B = -2.48285$, $C = -1.7246$.

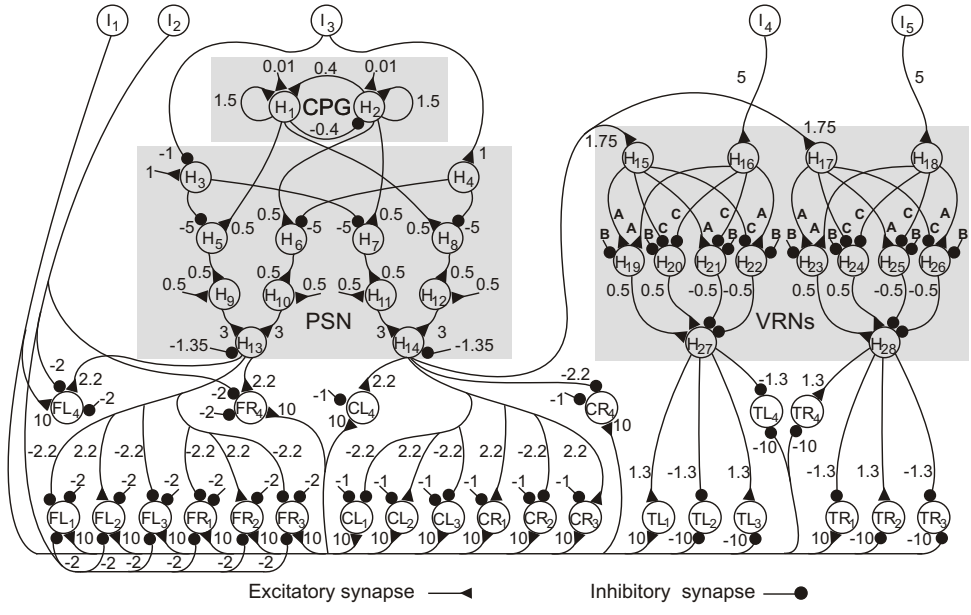

Fig. 27. The modular neural control of the eight-legged walking machine AMOS-WD08. Parameters are $A = 1.7246$, $B = -2.48285$, $C = -1.7246$.

Acknowledgments

The Integrated Structure Evolution Environment (ISEE) software platform for the evolution of recurrent neural networks and Yet Another Robot Simulator (YARS) were kindly provided by Keyan Zahedi of Fraunhofer IAIS. We thank Martin Hülse, and Steffen Wischmann for technique advises.

References

[1] T. Akay, B. C. Ludwar, M. L. Goeritz, J. Schmitz, A. Bueschges, Segment Specificity of Load Signal Processing Depends on Walking Direction in the Stick Insect Leg Muscle Control System, Journal of Neuroscience 27 (12) (2007) 3285–3294.

[2] M. Albrecht, T. Backhaus, S. Planthaber, H. Stoeppeler, D. Spenneberg, F. Kirchner, Aimee: A four legged robot for robocup rescue, in: Proceedings of the 8th International Conference on Climbing and Walking Robots, 2005.

[3] K. S. Ali, R. C. Arkin, Implementing schema-theoretic models of animal behavior in robotic systems, in: Proceedings of the Fifth International Workshop on Advanced Motion Control, 1998.

[4] R. C. Arkin, Behavior-Based Robotics, MIT Press, 1998.

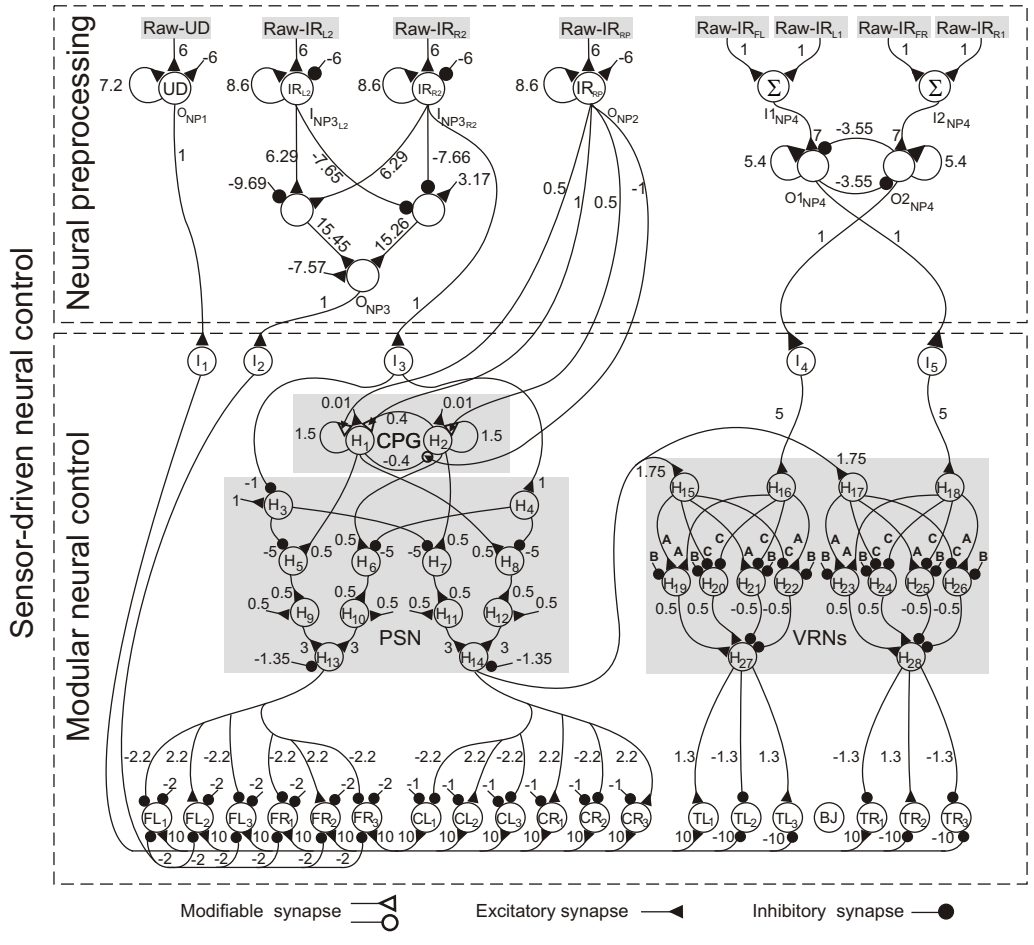

Fig. 28. The sensor-driven neural control of the six-legged walking machine AMOS-WD06. Parameters are $A = 1.7246$, $B = -2.48285$, $C = -1.7246$.

[5] K. S. Aschenbeck, N. I. Kern, R. J. Bachmann, R. D. Quinn, Design of a quadruped robot driven by air muscles, in: Proceedings of the IEEE/RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, 2006.

[6] J. Ayers, J. Crisman, The lobster as a model for an omnidirectional robotic ambulation control architecture, in: R. Beer, R. Ritzmann, T. McKenna (eds.), Biological neural networks in invertebrate neuroethology and robotics, Academic Press, 1992.

[7] J. Ayers, J. Witting, Biomimetic approaches to the control of underwater walking machines, Philosophical Transactions of the Royal Society, A 365 (2007) 273–295.

[8] K. R. Barai, K. Nonami, Experiments in learning distributed control for a hexapod robot, Robotics and Autonomous Systems 54 (10) (2006) 864–872.

[9] T. D. Barfoot, E. J. P. Earon, G. M. T. D'Eleuterio, Optimal two-degree-of-freedom fuzzy control for locomotion control of a hydraulically actuated hexapod robot, Information Sciences 177 (2007) 1892–1915.

[10] R. D. Beer, J. C. Gallagher, Evolving dynamic neural networks for adaptive behavior, Adaptive Behavior 1 (1992) 91–122.

[11] R. D. Beer, G. J. Kacmarcik, R. E. Ritzmann, H. J. Chiel, A model of distributed sensorimotor control in the cockroach escape turn, in: R. Lippman, J. Moody, D. Touretzky (eds.), Advances in Neural Information Processing Systems 3, Morgan Kaufmann, 1990.

[12] R. D. Beer, R. D. Quinn, H. J. Chiel, R. E. Ritzmann, Biologically inspired approaches to robotics: what can we learn from insects?, Communications of the ACM 40 (3) (1997) 30–38.

[13] K. Berns, W. Ilg, M. Deck, J. Albiez, R. Dillmann, Mechanical construction and computer architecture of the four-legged walking machine BISAM, IEEE Transactions on Mechatronics (1999) 1–7.

[14] A. Billard, A. J. Ijspeert, Biologically inspired neural controllers for motor control in a quadruped robot, in: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks - IJCNN2000, vol. 6, 2000.

[15] J. Bongard, V. Zykov, H. Lipson, Resilient machines through continuous self-modeling, Science 314 (5802) (2006) 1118–1121.

[16] R. A. Brooks, A robot that walks: emergent behaviors from a carefully evolved network, Neural Computation 12 (1989) 253–262.

[17] A. Bueschges, Sensory control and organization of neural networks mediating coordination of multisegmental organs for locomotion, Journal of Neurophysiology 93 (2005) 1127–1135.

[18] E. Celaya, J. L. Albarral, Implementation of a hierarchical walk controller for the Lauron III hexapot robot, in: Proceedings of the 6th International Conference on Climbing and Walking robots, 2003.

[19] H. J. Chiel, R. D. Beer, The brain has a body: adaptive behavior emerges from interactions of nervous system, body and environment, Trends in neurosciences 20 (12) (1997) 553–557.

[20] S. Chitta, J. P. Ostrowski, New insights into quasi-static and dynamic omnidirectional quadrupedal walking, in: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 4, 2001.

[21] H. Cruse, T. H. Kindermann, M. Schumm, J. Dean, J. Schmitz, Walknet–a biologically inspired network to control six-legged walking, Neural Networks 11 (1998) 1435–1447.

[22] F. Delcomyn, The walking of cockroaches – deceptive simplicity, in: R. Beer, R. Ritzmann, T. McKenna (eds.), Biological neural networks in invertebrate neuroethology and robotics, Academic Press, 1992.

[23] F. Delcomyn, Walking robots and the central and peripheral control of locomotion in insects, Autonomous Robots 7 (1999) 259–270.

[24] M. H. Dickinson, C. T. Farley, R. J. Full, M. A. R. Koehl, R. Kram, S. Lehman, How animals move: An integrative view, Science 288 (2000) 100–106.

[25] M. R. Fielding, G. R. Dunlop, Omnidirection hexapod walking and efficient gaits using restrictedness, International Journal of Robotics Research 23 (2004) 1105–1110.

[26] D. Filliat, J. Kodjabachian, J. A. Meyer, Incremental evolution of neural controllers for navigation in a 6 legged robot, in: Proceedings of the 4th International Symposium on Artificial Life and Robotics, 1999.

[27] J. Fischer, A modulatory learning rule for neural learning and metalearning in real world robots with many degrees of freedom, Ph.D. thesis, University of Muenster, Germany (2004).

[28] J. Fischer, F. Pasemann, P. Manoonpong, Neuro-controllers for walking machines - an evolutionary approach to robust behavior, in: M. Armada, P. Gonzalez de Santos (eds.), Proceedings of the 7th International Conference on Climbing and Walking Robots, Springer Berlin, 2004.

[29] K. Fouad, F. Libersat, W. Rathmayer, Neuromodulation of the escape behavior in the cockroach Periplaneta americana by the venom of the parasitic wasp Ampulex compressa, Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology 178 (1) (1996) 91–100.

[30] J. P. Gabriel, A. Bueschges, Control of stepping velocity in a single insect leg during walking, Philosophical transactions. Series A, Mathematical, physical, and engineering sciences 365 (1850) (2007) 251–271.

[31] H. Gras, M. Hoerner, Wind-evoked escape running of the cricket Gryllus bimaculatus. I. behavioural analysis, Journal of Experimental Biology 171 (1992) 189–214.

[32] S. Grillner, Biological pattern generation: The cellular and computational logic of networks in motion, Neuron 52 (2006) 751–766.

[33] M. Gruhn, O. Hoffmann, M. Duebbert, H. Scharstein, A. Bueschges, Tethered stick insect walking: A modified slippery surface setup with optomotor stimulation and electrical monitoring of tarsal contact, Journal of Neuroscience Methods 158 (2006) 195–206.

[34] S. L. Hooper, Central pattern generators, Current Biology 10 (5) (2000) R176–R177.

[35] A. D. Horchler, R. E. Reeve, B. Webb, R. D. Quinn, Robot phonotaxis in the wild: A biologically inspired approach to outdoor sound localization, Advanced Robotics 18 (8) (2004) 801–816.

[36] H. Hu, D. Gu, Hybrid learning architecture for fuzzy control of quadruped walking robots, International Journal of Intelligent Systems 20 (2) (2005) 131–152.

[37] M. Huelse, F. Pasemann, Dynamical neural schmitt trigger for robot control, in: D. JR (ed.), Proceedings of the International Conference on Artificial Neural Networks, vol. 2415, Springer Verlag, 2002.

[38] M. Huelse, S. Wischmann, F. Pasemann, Structure and function of evolved neuro-controllers for autonomous robots, Connection Science 16 (4) (2004) 249–266.

[39] M. Huelse, S. Wischmann, K. Zahedi, ISEE-a framework for the evolution and analysis of recurrent neural networks for embodied agents, ERCIM News 64 (2006) 33–34.

[40] A. J. Ijspeert, A. Crespi, D. Ryczko, J. M. Cabelguen, From swimming to walking with a salamander robot driven by a spinal cord model, Science 315 (5817) (2007) 1416–1420.

[41] W. Ilg, J. Albiez, R. Dillmann, Adaptive posture-control for a four-legged walking machine using some principles of mammalian locomotion, in: Proceedings of the International Symposium on Adaptive Motion of Animals and Machines, 2006.

[42] N. Jacobi, Running across the reality gap: Octopod locomotion evolved in a minimal simulation, in: Proceedings of the First European Workshop on Evolutionary Robotics, 1998.

[43] H. Kimura, Y. Fukuoka, A. H. Cohen, Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts, International Journal of Robotics Research 26 (5) (2007) 475–490.

[44] F. Kirchner, D. Spenneberg, R. Linnemann, A biologically inspired approach toward robust real-world locomotion in legged robots, in: J. Ayers, J. Davis, A. Rudolph (eds.), Neurotechnology for Biomimetic Robots, MIT Press, 2002.

[45] E. Klavins, H. Komsuoglu, R. Full, D. E. Koditschek, The role of reflexes versus central pattern generators in dynamical legged locomotion, in: J. Ayers, J. Davis, A. Rudolph (eds.), Neurotechnology for Biomimetic Robots, MIT Press, 2002.

[46] R. Kurazume, K. Yoneda, S. Hirose, Feedforward and feedback dynamic trot gait control for quadruped walking vehicle, Autonomous Robots 12 (2) (2002) 157–172.

[47] P. Manoonpong, Neural Preprocessing and Control of Reactive Walking Machines: Towards Versatile Artificial Perception-Action Systems, Cognitive Technologies, Springer, 2007.

[48] P. Manoonpong, T. Geng, T. Kulvicius, B. Porr, F. Woergoetter, Adaptive, fast walking in a biped robot under neuronal control and learning, PLoS Computational Biology 3 (7) (2007) e134.

[49] P. Manoonpong, F. Pasemann, J. Fischer, H. Roth, Neural processing of auditory signals and modular neural control for sound tropism of walking

machines, International Journal of Advanced Robotic Systems 2 (3) (2005) 223–234.

[50] P. Manoonpong, F. Pasemann, H. Roth, Modular reactive neurocontrol for biologically-inspired walking machines, International Journal of Robotics Research 26 (3) (2007) 301–331.

[51] E. Marder, D. Bucher, Central pattern generators and the control of rhythmic movements, Current biology 11 (2001) R986–R996.

[52] I. Markelic, K. Zahedi, An evolved neural network for fast quadrupedal locomotion, in: Proceedings of the 10th International Conference on Climbing and Walking Robots, 2007.

[53] K. Matsuoka, Sustained oscillations generated by mutually inhibiting neurons with adaptation, Biological Cybernetics 52 (6) (1985) 367–376.

[54] T. M. Mitchell, Machine Learning, McGraw Hill, 1997.

[55] F. Mondada, E. Franzi, P. Ienne, Mobile robot miniaturisation: A tool for investigation in control algorithms, in: Proceedings of the 3rd International Symposium on Experimental Robotics, Springer Verlag, 1993.

[56] L. Mu, R. E. Ritzmann, Kinematics and motor activity during tethered walking and turning in the cockroach, Blaberus discoidalis, Journal of comparative physiology. A 191 (2005) 1037–1054.

[57] G. N. Orlovsky, T. G. Deliagina, S. Grillner, Neural Control of Locomotion: From Mollusk to Man, Oxford University Press, 1999.

[58] G. B. Parker, Z. Lee, Evolving neural networks for hexapod leg controllers, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, vol. 2, 2003.

[59] F. Pasemann, Discrete dynamics of two neuron networks, Open Systems and Information Dynamics 2 (1993) 49–66.

[60] F. Pasemann, Dynamics of a single model neuron, International Journal of Bifurcation and Chaos 2 (1993) 271–278.

[61] F. Pasemann, M. Hild, K. Zahedi, So(2)-networks as neural oscillators, in: J. Mira, J. Alvarez (eds.), Computational Methods in Neural Modeling: Proceedings of the 7th International Work-Conference on Artificial and Natural Networks, vol. 2686, Springer Berlin, 2003.

[62] F. Pasemann, M. Huelse, K. Zahedi, Evolved neurodynamics for robot control, in: V. M (ed.), European Symposium on Artificial Neural Networks, vol. 2, 2003.

[63] J. M. Porta, E. Celaya, Body and leg coordination for omnidirectional walking in rough terrain, in: Proceedings of the 3rd International Conference on Climbing and Walking Robots, Professional Engineering Publishing, 2000.

[64] M. H. Raibert, Legged Robots that Balance, Cambridge, MIT Press, 1986.

[65] A. L. Ridgel, R. E. Ritzmann, Effects of neck and circumoesophaegeal connective lesions on posture and locomotion in the cockroach, Journal of comparative physiology. A, Neuroethology, sensory, neural, and behavioral physiology 191 (6) (2005) 559–573.

[66] R. E. Ritzmann, A. Bueschges, Insect walking: From reduced preparations to natural terrain, in: G. North, R. J. Greenspan (eds.), Invertebrate Neurobiology, Cold Spring Harbor Laboratory Press, 2007.

[67] R. E. Ritzmann, A. J. Pollack, Wind activated thoracic interneurons of the cockroach: II. Patterns of connection from ventral giant interneurons, Journal of Neurobiology 19 (1988) 589–611.

[68] R. Rojas, Neural NetworksA Systematic Introduction, Springer Verlag, 1996.

[69] D. E. Rumelhart, J. L. McClelland, Parallel Distributed Processing: Explorations in the Microstructure of cognition, Volume 1: Foundations, MIT Press, 1986.

[70] B. L. Rutter, W. A. Lewinger, M. Blumel, A. Buschges, R. D. Quinn, Simple muscle models regularize motion in a robotic leg with neurally-based step generation, in: Proceedings of the IEEE International Conference on Robotics and Automation, 2007.

[71] J. Schmidt, H. Fischer, A. Bueschges, Pattern generation for walking and searching movements of a stick insect leg II Control of motoneuronal activity, Journal of Neurophysiology 85 (2001) 354–361.

[72] R. Smith, Open dynamics engine v0.5 user guide, http://www.ode.org/ode-0.5-userguide.html (2004).

[73] H. Takemura, M. Deguchi, J. Ueda, Y. Matsumoto, T. Ogasawara, Slip-adaptive walk of quadruped robot, Robotics and Autonomous Systems 53 (2) (2005) 124–141.

[74] A. Twickel, Obstacle perception by scorpions and robots, Master's thesis, University of Bonn, Germany (2004).

[75] T. Yamaguchi, K. Watanabe, K. Izumi, Neural network approach to acquiring free-gait motion of quadruped robots in obstacle avoidance, Artificial Life and Robotics 9 (4) (2005) 188–193.

[76] S. N. Zill, Invertebrate neurobiology: Sensory processing in reverse for backward walking, Current Biology 17 (12) (2007) R462–R464.