2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

Adaptive Neural Oscillators with Synaptic Plasticity for Locomotion Control of a Snake-Like Robot with Screw-Drive Mechanism

Timo Nachstedt, Florentin Wörgötter, Poramate Manoonpong,
Ryo Ariizumi, Yuichi Ambe, and Fumitoshi Matsuno

Abstract— **Central pattern generators (CPGs) play a crucial role for animal locomotion control. They can be entrained by sensory feedback to induce proper rhythmic patterns and even store the entrained patterns through connection weights. Inspired by this biological finding, we use four adaptive neural oscillators with synaptic plasticity as CPGs for locomotion control of our real snake-like robot with screw-drive mechanism. Each oscillator consists of only three neurons and uses adaptive mechanisms based on frequency adaptation and Hebbian-type learning rules. It autonomously generates proper periodic patterns for the robot locomotion and can be entrained by sensory feedback to memorize the patterns. The adaptive CPG system in conjunction with a simple control strategy enables the robot to perform self-tuning behavior which is robust against short-time perturbations. The generated behavior is also energy efficient. In addition, the robot can also cope with corners as well as move through a complex environment with obstacles.**

I. Introduction

Central Pattern Generators (CPGs) are neural circuits that are able to produce periodic outputs without requiring any periodic input [1], [2]. It has been widely shown that CPGs play a crucial role in the creation and coordination of animal locomotion. While no input is required to produce basic periodic output, sensory feedback is critical for shaping and tuning the output pattern to produce appropriate periodic motion of limbs or limbless bodies [2].

Several works have employed this concept to develop different types of CPG models ranging from detailed biophysical to pure mathematical models [2], [3], [4], [5], [6] for controlling robot locomotion. Most of them generally perform as purely reactive oscillators which get entrained by an external input but only temporary [4], [6]. If the input is removed the oscillators immediately return to their inherent dynamics. Other works show adaptive oscillators [7], [8] having the ability to memorize the effect of the input or perturbation. For example, Righetti et al. [9] introduced a frequency adaptation rule for a general time-continuous oscillatory system. Nakamura et al. [10] applied actor-critic reinforcement learning to obtain the suitable CPG parameters for generating stable biped locomotion to deal with environmental changes. Buchli and Ijspeert [11] presented a schema to autonomously adapt the frequencies of CPGs used to control a quadruped robot. Inoue and Ma [12] used a genetic algorithm to obtain an adaptive CPG for controlling a snake-like robot and to deal with different friction environments.

While all those adaptive CPG mechanisms are impressive in their own right, most online optimizations suffer from long adaptation times. The fastest presented configuration for an adaptive Hopf oscillator in [9] still needs more than 2000 oscillation periods to adapt to an external frequency. The optimization in [8] takes more than 1500 periods, but also modifies eleven CPGs with five parameters per CPG. However, in [7] a system of two coupled oscillators learns the individual frequencies and the mutual phase relation within approximately 30 oscillation periods.

Recently, we have developed an adaptive neural oscillator with synaptic plasticity [13]. Following a minimal approach, it consists of only three simple artificial neurons and is able to quickly adapt to and remember an external periodic input within few periods of the external signal. Here, the adaptive neural oscillator is for the first time used to control our snake-like robot using screw-drive units connected by active joints [14]. Compared to other snake-like robots that are driven by undulation movements [12], [15] or crawler mechanisms [16], this robot can use its screw drive units to generate propulsion for locomotion on any side of the body which is in contact with the environment. Therefore, it can cope with narrower spaces. In addition, the screw drive units also offer more degrees of freedom in the movement. For each screw-drive element, one adaptive neural oscillator is used as a CPG-like structure with a feedback mechanism to adaptively control the rotation speed. The feedback mechanism is based on the angle sensors of the active joints connecting the different screw elements. A simple control strategy is employed to control those active joints. It allows only one joint to be freely movable at a time while the other joints are fixed at zero position. This way, not only feedback values but also flexible and stable locomotion is obtained.

Employing the adaptive CPGs together with the control strategy, the screw speeds of the robot are autonomously and quickly tuned to proper values. The reached configurations are robust against short-time perturbations and allow energy efficient straight-forward locomotion. Furthermore, it can help the robot to successfully cope with corners and even to move through a complex environment with obstacles.

This research was supported by Emmy Noether grant MA4464/3-1 of the Deutsche Forschungsgemeinschaft and Bernstein Center for Computational Neuroscience II Göttingen (BCCN grant 01GQ1005A, project D1) as well as by the HeKKSaGOn network.

Timo Nachstedt, Poramte Manoonpong and Florentin Wörgötter are with the Bernstein Center for Computational Neuroscience, Third Institute of Physics, at the University of Göttingen in 37077 Göttingen, Germany. {nachstedt,poramate,worgott}@physik3.gwdg.de.

Ryo Ariizumi, Yuichi Ambe and Fumitoshi Matsuno are with the Department of Mechanical Engineering and Science, Graduate School of Engineering at the Kyoto University in Kyoto, 606-8501, Japan. {ariizumi.ryo.53n,ambe.yuichi.23n}@st.kyoto-u.ac.jp, matsuno@me.kyoto-u.ac.jp.

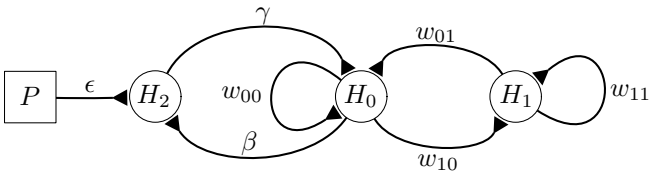


Fig. 1. The adaptive neural oscillator with synaptic plasticity.

II. Adaptive Neural Oscillator With Synaptic Plasticity

We use standard additive time-discrete neurons H_i, $i \in \{0, \ldots, N-1\}$, where N is the number of neurons. The neural activity $a_i(t+1)$ at time step $t+1$ depends on the neural outputs $o_j(t)$ at the previous time step t and the corresponding synaptic weights $w_{ij}(t)$. The neural output is given by a sigmoid transfer function σ of the activity. In this contribution σ is always chosen to be the hyperbolic tangent:

$$a_i(t) := \sum_{j=0}^{N-1} w_{ij}(t) o_j(t-1) \ , \tag{1}$$

$$o_i(t) := \tanh(a_i(t)), \qquad i = 0, \ldots, N-1 \ . \tag{2}$$

It is known that a fully connected two neuron network of this type produces quasi-periodic output if the synaptic weights are chosen according to an SO(2)-matrix [17] :

$$\begin{pmatrix} w_{00} & w_{01} \\ w_{10} & w_{11} \end{pmatrix} = \alpha \cdot \begin{pmatrix} \cos(\varphi) & \sin(\varphi) \\ -\sin(\varphi) & \cos(\varphi) \end{pmatrix} \tag{3}$$

with $-\pi < \varphi < \pi$ and $\alpha > 1$. For $\alpha = 1 + \epsilon$ with $\epsilon \ll 1$ the output is nearly sine-shaped with an angular frequency $\omega \approx \varphi$.

The adaptive neural oscillator with synaptic plasticity is shown in Fig. 1. It consists of an SO(2)-oscillator formed by the neurons H_0 and H_1, an additional neuron H_2 and an external perturbation P. Three additional synapses with synaptic weights $\beta := w_{20}$, $\gamma := w_{02}$ and $\epsilon := w_{2P}$ are introduced. They are governed by Hebbian-type learning rules based on correlation and relaxation terms driving the weights towards predefined relaxation values β_0, γ_0 and ϵ_0. The parameters $A, B > 0$ determine the influence of the individual terms [13]:

$$\beta(t+1) = \beta(t) - A\ o_0(t)\ o_2(t) - B\ (\beta(t) - \beta_0) \ , \tag{4}$$

$$\gamma(t+1) = \gamma(t) - A\ o_2(t)\ o_0(t) - B\ (\gamma(t) - \gamma_0) \ , \tag{5}$$

$$\epsilon(t+1) = \epsilon(t) + \ A\ P(t)\ o_2(t) - B\ (\epsilon(t) - \epsilon_0) \ . \tag{6}$$

The parameter φ in equation (3) is modified based on the following frequency adaptation rule with the learning rate μ:

$$\varphi(t+1) = \varphi(t) + \mu\ \gamma(t)\ o_2(t)\ w_{01}(t)\ o_1(t) \ . \tag{7}$$

With an appropriate choice of parameters the adaptive oscillator governed by above equations is able to adapt to signals with an external frequency f_{ext} within a wide frequency range. After adaptation the external perturbation P can be removed from the system while it maintains to oscillate at the learned frequency. The reconfiguration of the oscillator can be interpreted as an interplay of short-term synaptic plasticity at the synapses β,γ and ϵ and long-term synaptic plasticity at the synapses governed by the SO(2)-weight matrix [13].

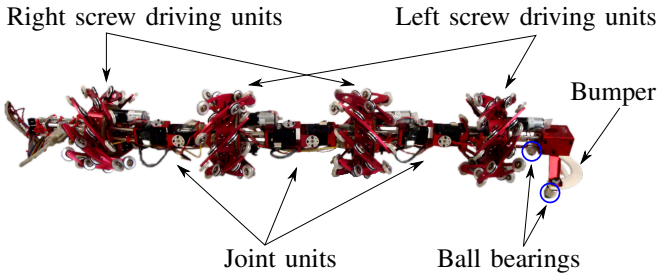


Fig. 2. Prototype of the snake-like robot with screw drive mechanism.

III. Snake-Like Robot with Screw Drive Mechanism

A prototype of the snake-like robot with screw drive mechanism [14] is shown in Fig. 2. It consists of two types of screw drive units called left and right screw units. Each unit is composed of an outer part that actually rotates and an inner part equipped with a DC motor used to drive the outer part and its encoder measuring the rotation. There are eight blades connected to the outer part where each blade has four passive wheels. The angle α between the rotation axis of the screw unit and the one of the passive wheels is positive for the right screw drive units and negative for the left ones. The individual screw drive units are connected by active joint units allowing two degrees of freedom, pitch (up-down) and yaw (left-right) movements. Every joint actuator can operate in an angle range from $-\pi/2$ to $\pi/2$. Right and left screw units are connected alternately with the foremost unit being oriented left.

To prevent the robot from just rotating the inner screw unit parts the foremost screw unit is connected to a head element equipped with two ball bearings with ground contact. Furthermore a rigid bumper is attached to the head element to increase the capability to cope with obstacles.

By rotating a screw unit around its rotation axis a force in the direction of the axis of the passive wheel on the ground is generated. If left and right units are rotated in alternating directions, a forward or backward propulsion of the whole robot is generated, respectively. Different movements like rotations or lateral locomotion can be achieved by different combinations of the rotation directions of the screw units.

IV. Control Strategy

The proposed control strategy consists of two main parts. On the one hand for every screw unit there is a closed-loop control loop formed by the unit's servo motor, its angle sensor and one adaptive neural oscillator with synaptic plasticity as a CPG to control the rotational velocity. On the other hand there is a feedback mechanism modulating the rotational velocities of the screw units based on angle sensor signals from the yaw joints. The idea is to use the joint angle

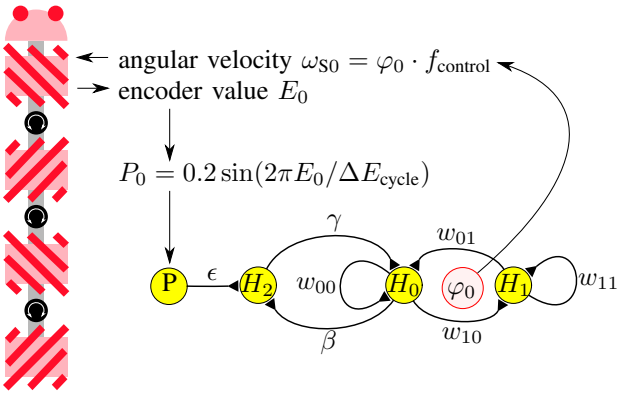


Fig. 3. Basic control schema for one screw unit: Based on the encoder value E read out from the screw and the known encoder value difference ΔE_{cycle} representing one full rotation cycle, a sine-shaped perturbation P is calculated and fed into the adaptive neural oscillator with synaptic plasticity. The current internal value φ of the oscillator is used to control the angular velocity $\omega = \varphi \cdot f_{\text{control}}$ of the unit where f_{control} is the update frequency of the neural network. This control loop exists for every individual screw unit.

feedback mechanism to provide a reflex-like control signal while the CPG control loop should adapt according to those reflexes and thereby avoid them. While using CPGs to control undulation movements of snake-like robots is a common approach (e. g. [18]) to our best knowledge until now there has not been any attempt to control rotational actuators of snake-like-robots similar to the presented one with the help of CPGs. In addition, using joint angle feedback and reflex avoidance in general are well-known concepts but have not yet been applied in the presented way for snake-like robots.

A. Adaptive Neural Oscillators as CPGs for Screw Units

The screw units of the robot are velocity controlled. For $\alpha = 1.01$ the internal variable φ_i of the ith neural oscillator approximates the angular velocity ω_{Oi} of its periodic output. This signal can thus be directly used to control the angular velocities ω_{Si}, $i = 0, \ldots, 3$, of the screw units. We implement one neural oscillator per screw unit. Regarding the update frequency $f_{\text{control}} = 10s^{-1}$ of the neural network we therefore have

$$\omega_{Si} := \pm f_{\text{control}} \cdot \omega_{Oi} = \pm f_{\text{control}} \cdot \varphi_i \tag{8}$$

where the signs are chosen alternately to obtain a forward movement.

Reading out the encoder value E_i provided at every screw unit and using the number ΔE_{cycle} of encoder steps that corresponds to one full rotation of the screw, the current rotation angle of every unit can be determined. Furthermore, it can be transformed into a sine shaped signal P_i that is fed back into the neural oscillator:

$$P_i = 0.2 \sin(2\pi E/\Delta E_{\text{cycle}}) \ . \tag{9}$$

Thereby, the factor 0.2 is introduced to adapt the amplitude of the perturbation to the amplitude of the neural output signals of the oscillator. The complete control loop for the first screw is illustrated in Fig. 3.

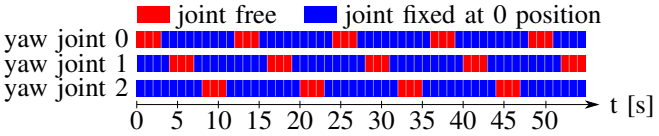


Fig. 4. Yaw joint servo motor free-run-mode protocol: To simplify and stabilize the system at any time there is at maximum one servo in free-run mode. The corresponding interval for every motor has a length of three seconds. In between the individual intervals there is always a time span of one second in which all joints are fixed at zero position.

B. Joint Angle Feedback Mechanism

As long as the motors at the screw units are powerful enough the control loop shown in Fig. 3 will always be self-consistent. To observe self-tuning behavior of the screw speeds an additional feedback mechanism is necessary. One promising possibility to realize such a mechanism is to use the angle sensors of the yaw joint actuators connecting the different screw units. By setting the corresponding servo motors into a free-run mode the measured joint angles are directly influenced by the current screw speed configuration.

However, it turns out by setting all joints into free-run mode simultaneously the robot gets very unstable and very difficult to control. Furthermore, it is very hard to determine clear rules how to react and adapt to different feedback signals. Therefore, we choose an engineering approach to simplify the system. At any given time only one yaw joint servo motor is set into free-run mode while the other two yaw joints are fixed with maximum power at central zero position. In between changing the currently freed joint a short interval is introduced in which all joints are fixed at zero position. This is to prevent the abrupt zero setting movement of one joint to influence the joint angle of the next free joint. The detailed joint-freeing-strategy is shown in Fig. 4.

According to this strategy, we have to only deal with one non-zero joint angle at any given time. Thus, a reasonable feedback mechanism can be determined heuristically. The basic idea is to modify the screw unit speeds which is set by the CPG output such that a tendency towards a rotation movement with a direction contrary to the observed joint angle is created. All feedback connections are scaled by a global feedback coupling strength η. Figure 5 shows the detailed feedback strength for every screw unit and every yaw joint angle sensor. They are empirically determined and tuned. As long as the feedback connections are appropriate to bring the robot back into a straight line shape, the exact coupling values are rather irrelevant.

C. Complete Control Schema

Figure 6 gives an overview of the complete control schema. The rotational velocities of the screw units are given by the sum of the output of the CPG driven control loop and the modulations of the joint angle feedback.

In general, higher or lower rotational velocities are reflected by the screw units' encoder signals, thereby creating perturbation signals P_i with different frequencies then the CPGs. Following the dynamics of the adaptive neural oscillator with synaptic plasticity the difference between the

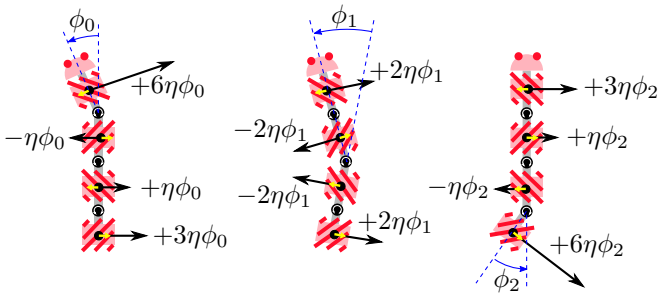


Fig. 5. Feedback mechanism: The angular velocities of the screw units are modulated based on the signals of the yaw joint angle sensors. The figure illustrates the individual feedback connections. The small yellow arrows show the default rotation directions of the screw units for forward locomotion. The large black arrows indicate in which direction the rotational velocity is modified for a positive joint angle signal ϕ_0, ϕ_1 or ϕ_2, respectively. The corresponding labels give the precise modification strength where η is a global feedback coupling strength. The signs and ratios of the individual feedback connections are determined heuristically by thinking in terms of a rotational movement.

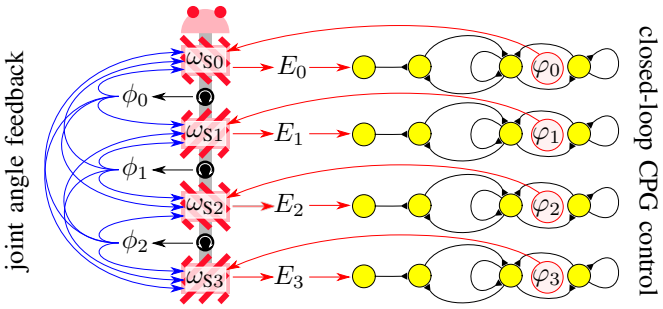


Fig. 6. Overview of the complete control schema: The rotational velocity ω_{Si} of screw unit $i \in \{0, 1, 2, 3\}$ is given by a sum of two velocity components. It is calculated based on the internal variable φ_i of the corresponding CPG whose input is the encoder signal E_i and the modulations depending on the yaw joint angles ϕ_0, ϕ_1 and ϕ_2.

internal and the external frequency leads to a growing of the ϵ-synapses which increases the influence of the external signal on the adaptive oscillator and accelerates the adaptation process.

The adaptation of the CPG frequencies leads to a more stable locomotion of the robot expressed by the decreasing bending amplitudes of the joints in free-run mode. As bending amplitudes decrease, the modulations of the screw units' velocities also decrease and the differences between the CPG frequencies and the external signals vanish. As a result, the synaptic weights β and γ get smaller and reduce the influence of the external signals on the adaptive oscillators. In this state, the synaptic weights β, γ and ϵ start to decay synchronously towards β_0, γ_0 and ϵ_0, respectively, while there are only minor adaptations of the internal φ-parameters.

V. Experiments and Results

A. Straight-Line Recovery

The first experiment is to test whether the joint angle feedback mechanism is adequate to bring the snake-like robot into a straight-line shape (all joints at zero position) when perturbations are applied.

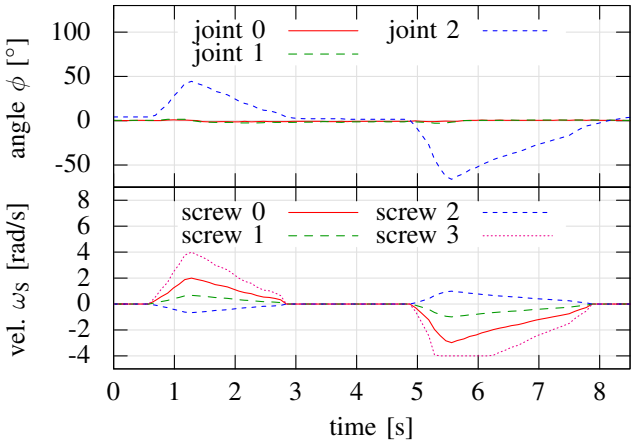


Fig. 7. Testing the joint angle feedback mechanism: Yaw joints 0 and 1 were fixed at the zero position, joint 2 was in free-run mode. The CPG output for every screw unit was set to 0 and the CPG adaption mechanism was disabled. The robot was manually perturbed along the yaw axis of joint 2. Due to the joint-angle feedback mechanism, the perturbation indirectly changed rotational velocities of the screw units; thereby the robot returned back into the straight-line configuration.

The basic speeds of all screw units were set to zero ($\phi_i = 0$) and the learning mechanism of the adaptive neural oscillators was disabled (i.e., $\mu = 0$). One yaw joint servo motor was set to free-run mode while all other servo motors were fixed at zero position. The global feedback coupling constant was $\eta = 0.06$ rad s^{-1}/°. In this configuration, we manually perturbed the system along the free joint axis and observed whether the robot was able to reproduce a straight-line shape.

Figure 7 shows the time series of one of those experiments where the servo motor of joint 2 was set into free-run mode. Obviously, the feedback mechanism was capable of dealing with perturbations at this joint in both rotational directions and reproduced the straight-line configuration within approximately three seconds in both cases. Similar results were also found when joint 1 or joint 2 was set to the free-run mode. For every joint the experiments were repeated at least three times with qualitatively consistent results. These results show effective functionality of the joint angle feedback mechanism. We recommend readers to also see part one of the supplementary video of this experiment at `http://manoonpong.com/ICRA2013/ANO/suppl.mp4`.

B. Screw Speed Adaptation

For studying the adaptive behavior of the system, we choose an arbitrary set of initial CPG frequencies resulting in different initial screw unit velocities. The global feedback coupling constant was set to $\eta = 0.06$ rad s^{-1}/°. The parameters for the adaptive neural oscillators were $A = 1.0$, $B = 0.01$, $\beta_0 = 0.00$, $\gamma_0 = 1.00$, $\epsilon_0 = 0.01$ and $\mu = 1.0$. During the experiments, the robot moved on a flat terrain without obstacles. Note that, due to limitations in cable length and available space, the robot was moved to a different position or orientation at some points of time.

Figure 8 shows a chosen example of our experiments.

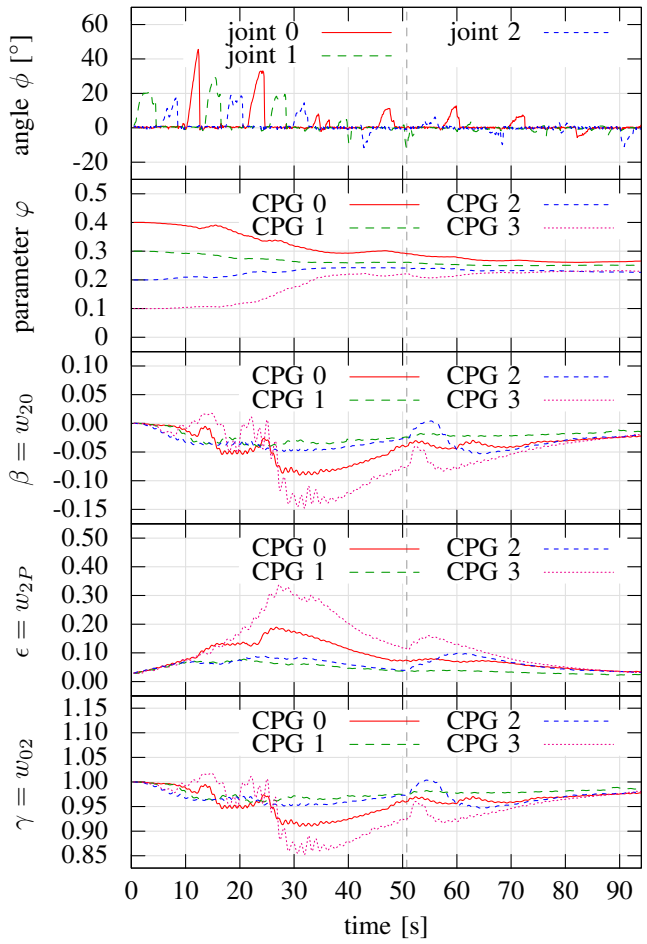


Fig. 8. Self-Tuning of screw unit velocities: All CPGs were initialized with different frequencies such that the foremost screw unit had the fastest initial rotational velocity and velocities descended towards the backward direction of the robot. The first plot shows the measured yaw angles of joints 0,1, and 2. Because of the introduced free-run-mode protocol at every time there was at maximum one angle deviated from zero. The second plot shows the internal parameter φ of each CPG that determines the basic screw unit speeds according to $\omega_{i,\text{screw}} := f_{\text{control}} \cdot \varphi_i$ with $f_{\text{control}} = 10\ \text{s}^{-1}$. The last three plots show the changes of the synaptic weights β, ϵ and γ during the adaptation process. During the experiment, the robot was moved and placed into a different orientation at $t = 50, 77$ s due to space and cable limitations. In the plot, this is indicated by the dashed vertical line. Finally, the control parameters converged leading to a stable locomotion.

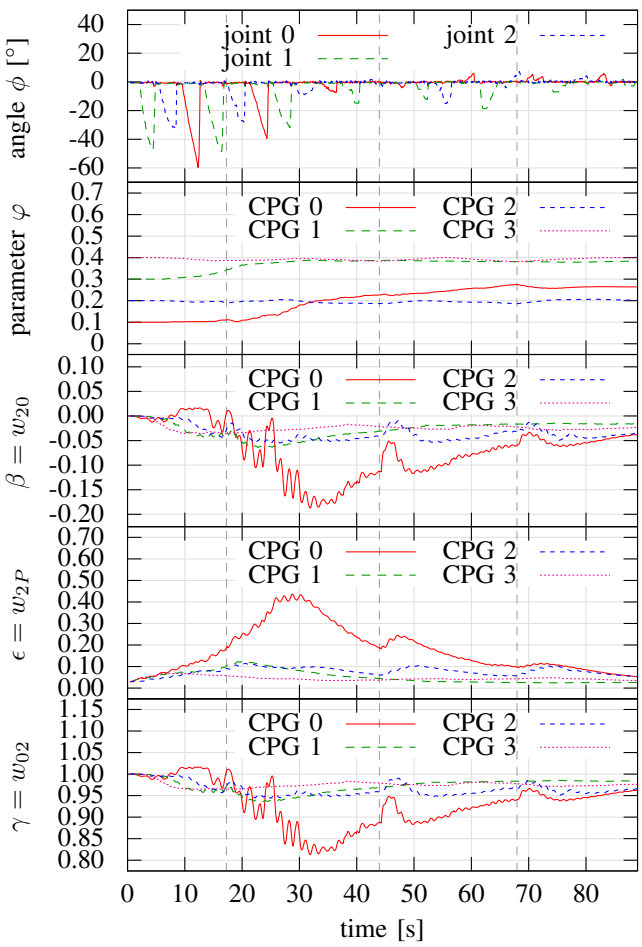


Fig. 9. Self-Tuning of screw unit velocities with a different final state: The CPGs were initialized with $\varphi_0 = 0.1$, $\varphi_1 = 0.3$, $\varphi_2 = 0.2$ and $\varphi_3 = 0.2$. Compared to Fig. 8 a different final configuration of the screw speeds was reached corresponding to a movement with a strong sideward component. Again, the dashed vertical lines indicate the times at which we manually changed the robot orientation.

According to the initial ϕ_i-values of the CPGs, the initial rotational velocities of the screw units were $\omega_{S0} = -4.0$ rad/s, $\omega_{S1} = +3.0$ rad/s, $\omega_{S2} = -2.0$ rad/s and $\omega_{S3} = +1.0$ rad/s. As can be seen in the top row of Fig. 8, this configuration leaded to high bending of the joint that was being in free-run mode. According to the joint angle feedback mechanism, this leaded to large modulations of the screw unit velocities as determined by the CPG control loops. Note that the internal parameters of the CPGs were not directly affected by those modulations.

Due to space and cable limitations during the experiments the robot was moved and placed to a different location or orientation. This introduced a small perturbation to the system. Hence, P_i was no longer in phase with the neural output o_1 of the corresponding neural oscillator. This was detected by the dynamics of the adaptive oscillators with synaptic plasticity and leaded to rapid changes of the synaptic weights β, γ and ϵ (see small bumps in Fig. 8). As a consequence, the influence of the external signal on the oscillator was again increased, thereby the adaptation mechanism tried to synchronize P_i and o_1 again.

While some bumps of the synaptic weights after changing the robot orientation are clearly visible in Fig. 8, nothing of this character can be observed for the parameter φ. This illustrates the ability of the neural oscillator with synaptic plasticity to damp short-time disturbances.

In Fig. 8, all screw speeds converged towards a common value leading to approximate forward locomotion of the robot. However, the control schema does not demand this behavior. Instead of global goal-directed adaptation the described mechanism is only a local adaptation process where the internal goal is to avoid yaw joint bending. This can also be achieved by different final locomotion behaviors. For example, Fig. 9 shows an adaptation process resulting in a movement direction with a high sideward component.

TABLE I

ENERGY CONSUMPTION IN STATIC AND ADAPTIVE CASES AND THEIR COMPARISON

initial configuration				adapt	I_J [mA]	I_S [mA]	P [W]	R [%]
φ_0	φ_1	φ_2	φ_3					
0.2	0.2	0.2	0.2	no	331(3)	608(9)	12.3(2)	2.8(1.6)
				yes	301(3)	617(12)	11.9(2)	
0.1	0.2	0.3	0.4	no	538(2)	777(7)	17.4(1)	30.0(7)
				yes	298(3)	642(7)	12.2(1)	
0.4	0.3	0.2	0.1	no	400(3)	744(4)	14.9(2)	22.4(9)
				yes	291(2)	602(6)	11.6(1)	
0.3	0.4	0.2	0.2	no	386(2)	660(7)	13.7(1)	15.9(7)
				yes	290(2)	599(5)	11.5(1)	
0.1	0.3	0.2	0.4	no	437(3)	660(9)	14.5(2)	17.3(9)
				yes	295(4)	629(6)	12.0(1)	

Abbreviations are: Average joint servo currents I_J, average screw unit motor currents I_S and average power consumption P for different initial conditions $(\varphi_0, \varphi_1, \varphi_2, \varphi_3)$. In every case the values for a controller without any adaptation and with all joints fixed at zero position are compared to the proposed controller with the help of the power consumption reduction R. The values in brackets indicate the uncertainties of the values.

Obviously, the actual reached final configuration depends on the initial screw speed configuration.

C. Energy consumption

We assume that the final configurations of the adaptation process lead to energy-efficient locomotion of the robot. To examine this assumption we investigate the total power consumption of the joint servo motors and the one of the screw unit motors by measuring the corresponding average currents I_J and I_S. The operating voltage for the joint servo motors is $U_J = 15$ V, the screw unit motors are driven with a voltage $U_S = 12$ V. The average complete power consumption of the robot is therefore given by $P = I_J U_J + I_S U_S$.

For a couple of different initial configurations $(\varphi_0, \varphi_1, \varphi_2, \varphi_3)$ the power consumption P_{static} for a static controller without adaptation and without the free-run mode protocol for the joint servos is compared to the power consumption P_{adapt} measured in the converged final state of the controller described above. From these two conditions (static and adaptive cases), the power consumption reduction $R = 1 - P_{\text{adapt}}/P_{\text{static}}$ is determined.

Table I gives the obtained results. Every row represents a single experiment. In all test cases a significant reduction of the power consumption can be observed. In most cases the current consumption reduces for the joint servo motors as well as for the screw unit motors. This is due to a higher degree of coordination of the different screw unit speeds that avoids opposing forces and torques and maintains a straight-line configuration of the robot without requiring high joint torques.

D. Obstacle Avoidance

The proposed control mechanism consisting of the adaptive CPGs with the joint angle feedback mechanism (cf. Fig. 6) and the free-run-mode protocol (cf. Fig. 4) does not only enable the self-tuning behavior as well as energy efficient locomotion shown above but also helps the robot to autonomously overcome certain obstacles.

As a basic example for this we investigate the robot's capability to cope with corners as shown in Fig 10. Corners with angles between $10°$ and $90°$ clockwise were prepared and then we checked whether the robot was able to pass them in clockwise and counterclockwise directions. The initial CPG frequency configuration was $\varphi_0 = \varphi_1 = \varphi_2 = \varphi_3 = 0.3$ corresponding to a rotational velocity of $\omega_{Si} = 3$ rad/s for all screw units. Each corner scenario was tested with a simple non-adaptive and all-joints-fixed controller and with the proposed control mechanism (i.e., advanced controller). Scenarios that mark a qualitative change in terms of whether the robot is able to cope with the presented corners when modifying the angle monotonously are repeated at least three times.

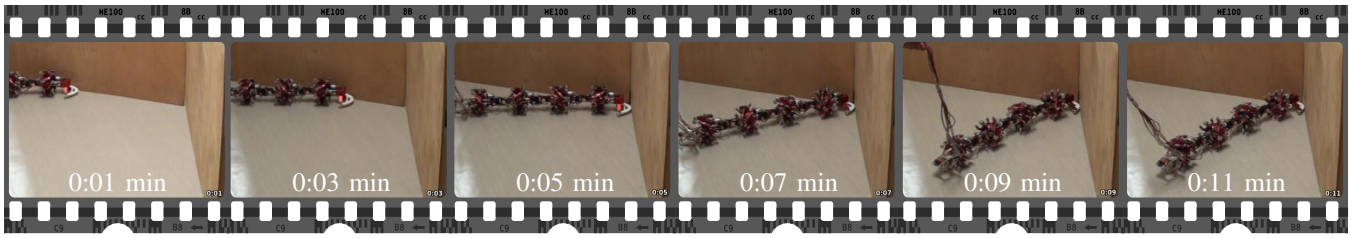
Results show that with the simple non-adaptive controller, i. e. with all joints fixed at zero position and all screws driven at identical speeds, the robot can cope with clockwise corners with a maximum angle of $50°$. For angles larger than or equal to $60°$ the robot got stuck similar to the one shown in Fig. 10a. In contrast, using the advanced controller the robot was able to pass clockwise corners with the maximum angle of $90°$ (see Fig. 10b). We recommend readers to also see part two of the supplementary video of this experiment at `http://manoonpong.com/ICRA2013/ANO/suppl.mp4`.

For counterclockwise corners both the simple and the adaptive controller are able to cope with all prepared angles. This is due to the orientation of the foremost screw unit which creates a force in the front left direction. The force component to the left helps to overcome the corner obstacle.

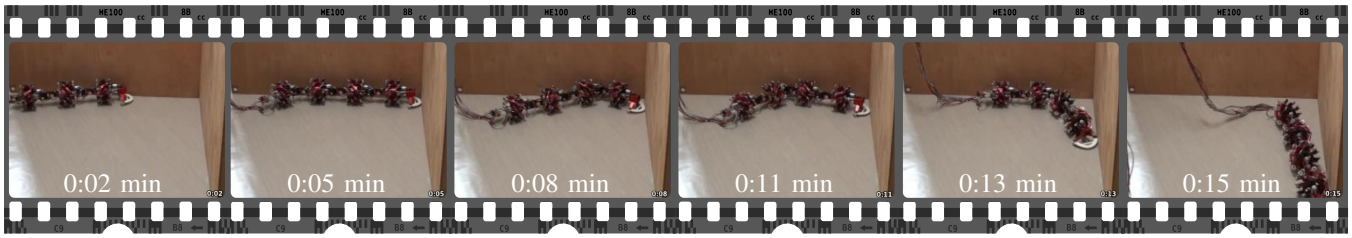
In addition to the tests with the corner situations, we also test the robot in a complex environment with obstacles. Results show that using the advanced controller the robot can successfully navigate through the environment (see Fig. 11). We recommend readers to also see part three of the supplementary video of this experiment at `http://manoonpong.com/ICRA2013/ANO/suppl.mp4`.

VI. CONCLUSION

This paper presented adaptive locomotion of the snake-like robot with screw-drive mechanism. The locomotion includes self-tuning behavior which is robust against short-time perturbations and energy-efficient. In addition, the robot can adapt its locomotion to cope with corners as well as move through a complex environment with obstacles. All these behaviors are controlled by the adaptive neural oscillators with synaptic plasticity in conjunction with a simple control strategy. The adaptive oscillators, on the one hand, provide quick adaptability to the robot due to their synaptic plasticity mechanisms. The control strategy with a free-run mode, on the other hand, gives flexibility to the robot and also ensures its stable locomotion. More demanding tasks will be related to implement goal-directed behavior control allowing the robot to learn to move towards a given goal.



(a) 70° clockwise corner with a simple non-adaptive controller.



(b) 90° clockwise corner with the adaptive controller with the free-run mode protocol.

Fig. 10. Comparison of the capability to cope with clockwise corners between a primitive and the proposed controller.

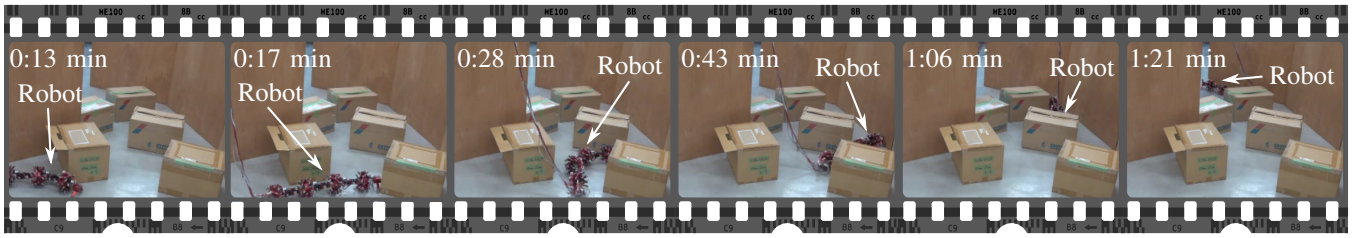


Fig. 11. Locomotion through a complex environment with obstacles.

REFERENCES

[1] E. Marder and D. Bucher, "Central pattern generators and the control of rhythmic movements." *Curr. Biol.*, vol. 11, no. 23, pp. R986–R996, Nov. 2001.

[2] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: a review." *Neural Networks*, vol. 21, no. 4, pp. 642–653, May 2008.

[3] J. Hellgren and S. Grillner, "Computer simulation of the segmental neural network generating locomotion in lamprey by using populations of network interneurons," *Biol. Cybern.*, vol. 68, no. 1, pp. 1–13, 1992.

[4] K. Matsuoka, "Sustained oscillations generated by mutually inhibiting neurons with adaptation," *Biol. Cybern.*, vol. 52, no. 6, pp. 367–376, 1985.

[5] J. Buchli, L. Righetti, and A. J. Ijspeert, "Engineering entrainment and adaptation in limit cycle systems : From biological inspiration to applications in robotics." *Biol. Cybern.*, vol. 95, no. 6, pp. 645–664, Dec. 2006.

[6] J. Morimoto, S. Hyon, G. Cheng, D. Bentivegna, and C. Atkeson, "Modulation of simple sinusoidal patterns by a coupled oscillator model for biped walking," in *Proc. IEEE International Conference on Robotics and Automation (ICRA 2006)*. IEEE, 2006, pp. 1579–1584.

[7] J. Nishii, "A learning model for oscillatory networks," *Neural Networks*, vol. 11, no. 2, pp. 249–257, 1998.

[8] D. Marbach and A. J. Ijspeert, "Online Optimization of Modular Robot Locomotion," in *Proc. IEEE International Conference on Mechatronics and Automation 2005*, no. July, 2005, pp. 248–253.

[9] L. Righetti, J. Buchli, and A. J. Ijspeert, "Dynamic Hebbian learning in adaptive frequency oscillators," *Physica D*, vol. 216, no. 2, pp. 269–281, Apr. 2006.

[10] Y. Nakamura, T. Mori, M.-a. Sato, and S. Ishii, "Reinforcement learning for a biped robot based on a CPG-actor-critic method," *Neural Networks*, vol. 20, no. 6, pp. 723–735, 2007.

[11] J. Buchli and A. J. Ijspeert, "Self-organized adaptive legged locomotion in a compliant quadruped robot," *Auton. Robots*, vol. 25, no. 4, pp. 331–347, 2008.

[12] K. Inoue, T. Sumi, and S. Ma, "CPG-based control of a simulated snake-like robot adaptable to changing ground friction," in *Proc. IEEE International Conference on Intelligent Robots and Systems (IROS 2007)*. Ieee, Oct. 2007, pp. 1957–1962.

[13] T. Nachstedt, F. Wörgötter, and P. Manoonpong, "Adaptive neural oscillator with synaptic plasticity enabling fast resonance tuning," in *Artificial Neural Networks and Machine Learning - ICANN 2012*, ser. Lecture Notes in Computer Science, A. E. Villa, W. Duch, P. Érdi, F. Masulli, and G. Palm, Eds. Springer Berlin Heidelberg, 2012, vol. 7552, pp. 451–458.

[14] H. Fukushima, S. Satomura, T. Kawai, M. Tanaka, T. Kamegawa, and F. Matsuno, "Modeling and Control of a Snake-Like Robot Using the Screw-Drive Mechanism," *IEEE Trans. Robot.*, vol. 28, no. 3, pp. 541–555, 2012.

[15] Z. Lu, S. Ma, B. Li, and Y. Wang, "Serpentine Locomotion of a Snake-like Robot Controlled by Cyclic Inhibitory CPG Model," in *Proc. IEEE International Conference on Intelligent Robots and Systems (IROS 2005)*, 2005, pp. 96–101.

[16] J. Borenstein, M. Hansen, and A. Borrell, "The OmniTread OT-4 serpentine robotdesign and performance," *J. Field Robot.*, vol. 24, no. 7, pp. 601–621, 2007.

[17] F. Pasemann, M. Hild, and K. Zahedi, "SO(2)-Networks as Neural Oscillators," in *Computational Methods in Neural Modeling*, J. Mira, Ed. Springer Berlin / Heidelberg, 2003, no. 2, pp. 1042–1042.

[18] J.-K. Ryu, N. Y. Chong, B. J. You, and H. I. Christensen, "Locomotion of snake-like robots using adaptive neural oscillators," *Intel. Serv. Robot.*, vol. 3, pp. 1–10, 2010.