Tipping the Scales: Guidance and Intrinsically Motivated Behavior

Georg Martius[1] and J. Michael Herrmann[2]

[1] Max Planck Institute for Mathematics in the Sciences, Inselstr. 22, 04103 Leipzig, Germany
[2] University of Edinburgh, School of Informatics, IPAB, 10 Crichton St, Edinburgh EH8 9AB, U.K.
martius@mis.mpg.de, michael.herrmann@ed.ac.uk

Abstract

We propose a novel approach to learning in autonomous robots that relies on the dynamical maintenance of an actively sensitized sensorimotor loop. Very weak learning cues are sufficient to orient a robot towards the desired behavior which is then selected from the intrinsic exploratory movements rather than imposed by a control command. The learning paradigm is a form of guided self-organization and is complementary to both active and intrinsically motivated learning. We present a systematic analysis of the learning algorithm in a robot control task and demonstrate its remarkable scalability with respect to the degrees of freedom of the system.

Introduction

Learning in autonomous agents implies an active involvement of the agent in the acquisition of new behavior. Lopez and Oudeyer (2010) ask for a unified formalism for active and intrinsically motivated exploration and observe a convergence of approaches from machine learning and developmental psychology towards a new perspective for developmental robotics. While a number of examples exist that impressively demonstrate the virtues of this view, it appears that a different sets assumptions are required that may eventually turn out to limit the possibility of on-going learning, scaling and transfer across domains. Since a more extended discussion is beyond the present scope we should mention here merely that the present approach aims at a relaxation of some of these assumptions. We will use only a local world model

While some variants of intrinsically motivated learning try to extract controllable options (Singh et al., 2004; Martius et al., 2008; Jung et al., 2011) we will use here a related approach (Martius and Herrmann, 2010) in order to improve the sensitivity with respect to given learning signals (cues). We implement in this way a form of self-organized curiosity (Schmidhuber, 1991; Herrmann, 2001) for the cues which substantially improves goal-related learning in an autonomous robot. We will show examples where the learning time within this approach scales very nicely with the complexity of the problem.

We start from an approach to self-organization of robot control (Der, 2001; Martius et al., 2011) which aims at robotic behaviors that are characterized by on-going exploration and that can be called natural for a specific robot in a particular environment (Der et al., 2006; Hesse et al., 2009). Animals, including humans, can be assumed to acquire their behavioral repertoire in a similar way: Behavioral elements are developed autonomously and are composed and refined later in order to realize more complex goals. The resulting behavior is, nevertheless, subject to an on-going developmental modulation throughout the whole life span.

In robotics, many promising examples for autonomous behavioral adaptation and generation have been studied for instance by Herrmann (2001); Tani (2003); Der et al. (2006); Nolfi (2006); Oudeyer et al. (2005). Self-organization of behavior is, nevertheless, still a field of active exploration. Further questions such as the interaction of learning by self-organization and learning by supervision or by external reinforcement are just starting to gain scientific interest.

Usually, goal-oriented behavior is achieved by direct optimization of the parameters of a control program such that the goal is approached more closely. The learning system must receive information about whether or not the behavior actually approaches the goal. This information may be available via a reward signal in reinforcement learning or by a fitness function in evolutionary algorithms. We will consider similar types of goal-related information when aiming at a combination of self-organizing control with external drives. For this combination the term *guided self-organization* (GSO) was proposed by Martius et al. (2007); Prokopenko (2009). In a general perspective, GSO is the combination of goal-oriented learning and developmental self-organization. Each of the two learning paradigms bring about their particular benefits and GSO aims at combining them in an optimal manner. For instance, self-organizing systems tend to have a high tolerance against failures and degrade gracefully, which is also desirable in task-oriented applications. when developing systems aiming to achieve tasks in practical applications. We will deal in with a specific approach to self-organizing control, namely homeokinetic learning.

Homeokinetic learning generates self-organized behavior which can serve as intrinsic motivation of the robot to become engaged with its environment. The robot learns to maintain an active low-level sensorimotor loop without abstract or specific information. Here we will study the possibility of including high-level information into this dynamical systems approach such that the robot can learn to reach a goal or to optimize its behavior according to external standards.

What can we expect from a *guided homeokinetic controller*? It has been shown earlier by Der et al. (2006) and Hesse et al. (2009) that a variety of behaviors can emerge from the principle of homeokinesis. The emerging behaviors show a coherent sensorimotor dynamics of the particular robot in its environment. With additional guidance the exploration of the homeokinetic controller can be channeled around desired or preferred behaviors such that control modes can be quickly found which match the given robotic task.

The behavior is essentially driven by intrinsic self-organization, while the goal is easily taken up by the system due to the optimal sensitivity of the homeokinetic control. In a sense, we are not considering here an approach to robot learning but rather an on-going dynamic realization of the (external or internal) hints as part of an exploratory regime.

In the present paper, we will advance our study of guided self-organization of behavior, presented in Martius and Herrmann (2010), by an application to a high-dimensional system. In order to keep the paper self-consistent, we introduce the homeokinetic control principle in the next section and present then the guidance by supervised *teaching cues*. The latter are the basis for the guidance by cross-motor teaching that can be implemented by the specification of abstract motor relations. We will extend this framework and apply it to the locomotion of bracelet-like robots with up to 40 DoF.

Self-Organized Closed-Loop Control

Self-organizing control for autonomous robots can be achieved by an intrinsic drive towards active and predictable behavior as described by the homeokinetic principle (Der, 2001). We assume that the dynamics of the sensor values $x \in \mathbb{R}^n$ of the robot can be written as

$$x_{t+1} = \psi(x_t) + \xi_{t+1} \tag{1}$$

where ψ is the internal model maintained and adapted by the robot to predict future sensor values and ξ is the prediction error. The motor values (actions) $y \in \mathbb{R}^m$ are generated by a controller implemented simply as a parametric map or one-layer neural network:

$$y_t = K\left(x_t, \mathcal{C}_t\right) = g\left(C_t x_t + h_t\right) \tag{2}$$

where $g(\cdot)$ is a sigmoid function with $g_i(z) = \tanh(z_i)$. The controller parameters $\mathcal{C}$ consist of a weight matrix C and a bias vector h. We compose the map ψ from a forward model $M(x, y, \mathcal{A})$ and the controller $K(x, \mathcal{C})$ (Eq. 2) as

$$\psi(x_t) = M(x_t, y_t, \mathcal{A}_t) = M(x_t, K(x_t, \mathcal{C}_t), \mathcal{A}). \tag{3}$$

The function M is initially unknown, but the robot adapts it continuously in order to minimize the prediction error ξ_t by

$$\mathcal{A}_{t+1} = \mathcal{A}_t - \epsilon_a \frac{\partial}{\partial \mathcal{A}_t} \|\xi_t\|^2. \tag{4}$$

If the parameters $\mathcal{C}$ were also adapted in this way then stable but typically trivial behaviors would be produced unless specific information is given to the robot.

The homeokinetic principle which we are going to use here normally does not need any specific information in order to produce a variety of elementary behaviors in a robot. We will show that this principle for the self-organization of behavior offers also a new perspective for learning in robots. That is, if additional information is available then a homeokinetically controlled robot can use this information more efficiently. This follows from the strongly enhanced sensitivity of the learning system and establishes a novel approach to learning in robots.

The homeokinetic principle suggests to use the so-called *time-loop error* (TLE) which is based on the reconstructed sensor values $\hat{x}_t$. Using Eq. 1 and assuming for now that ψ is invertible we define

$$\hat{x}_t = \psi^{-1}\left(\psi(x_t) + \xi_{t+1}\right) = \psi^{-1}\left(x_{t+1}\right) \tag{5}$$

which are sensor values that would have made the prediction perfect. Intuitively $\hat{x}_t$ is obtained by going forward in time from x_t to x_{t+1} and then backward in time to $\hat{x}_t$. This sequence is called the time loop and thus the TLE is

$$E_{TLE} = \|v_t\|^2 \quad \text{with} \quad v_t = x_t - \hat{x}_t \tag{6}$$

which minimizes the mismatch between true sensor values x_t and their reconstruction $\hat{x}_t$.

In linear approximation we obtain $v_t \approx L_t^{-1} \xi_{t+1}$, where the matrix $L_t = \frac{\partial \psi(x_t)}{\partial x_t}$ is the Jacobian of ψ at time t. Note that v_t can only be calculated after x_{t+1} is available. We account for non-invertible L by using a regularized inverse. The TLE

$$E_{TLE} = \|v_t\|^2 \approx \xi_{t+1}^\top \left(L_t L_t^\top\right)^{-1} \xi_{t+1}, \tag{7}$$

minimizes the norm of v (Eq. 6) and accounts for the error ξ (Eq. 1) only as much as it is transformed by the inverse dynamics of the system. This reveals another important feature of this error quantity, namely to minimize the norm of the inverse Jacobian. This results in an increase of predominantly the small eigenvalues of L. Therefore, the controller performs a destabilization in time. This eliminates the trivial fixed points (in sensor space) and enables spontaneous symmetry breaking which shows in the robot e. g. as a transition

from rest to a directed movement. Nevertheless, the system does not start to behave chaotically or enters uncontrollable oscillations because the destabilization is limited by the nonlinearity $g(\cdot)$ (Eq. 2). Intuitively, homeokinesis can be understood as the drive towards non-trivial behaviors that are still predictable by the internal model. Since the internal model is simple, smooth behaviors are preferred. Fig. 1 illustrates how the homeokinetic controller is connected to a robot.

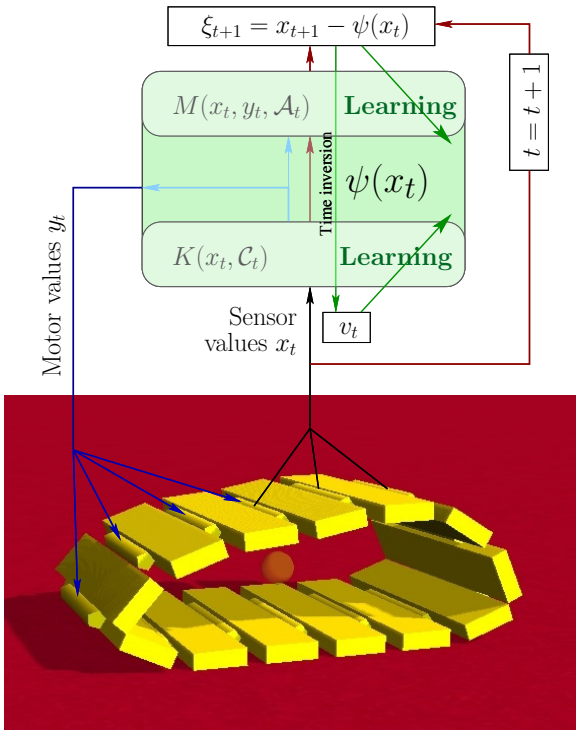

Figure 1: The homeokinetic controller connected to the ARMBAND robot. The ARMBAND consists here of $m = 13$ flat segments that are connected by actuated joints. It receives sensory inputs x_i from the joint position sensors. The control architecture consists of the controller K and the predictor M which are combined to form ψ, see Eqs. 1 - 2. The transparent ball indicates the center of mass of the robot. It is used for evaluation of performance but not for control.

The TLE (Eq. 7) is minimized by gradient descent which gives rise to a parameter dynamics that evolves simultaneously with the state dynamics, see e. g. (Hesse et al., 2009).

$$\begin{aligned} C_{t+1} &= C_t - \epsilon_c \frac{\partial}{\partial C} E_{TLE} \\ h_{t+1} &= h_t - \epsilon_c \frac{\partial}{\partial h} E_{TLE} \end{aligned} \tag{8}$$

The learning rates $\epsilon_c \approx \epsilon_A$ for the controller and the model are chosen such that the system adapts on the behavioral time scale. Because of unavoidable sensory noise, the TLE is never zero, neither does it have a vanishing gradient. The rule (Eq. 8) produces therefore a continuously itinerant trajectory in the parameter space, i. e. the robot traverses a sequence of behaviors that are determined by the interaction with the environment. These behaviors are, however, waxing and waning and their transitions are hard to predict.

As an example, consider a robot with two wheels that is equipped with wheel velocity sensors. In the beginning the robot rests, but after a short time the homeokinetic learning rule initiates autonomous forward, backward or turning movements. If a wall is encountered that causes the wheels to stop, the robot will immediately reduce the motor speed and change the internal parameter to regain sensitivity. Eventually it will drive in a free direction. A more complex example for the self-organization of *natural* behaviors was provided by a spherical robot (Martius and Herrmann, 2010) that is actuated by movable internal masses. After a short time the robot starts to roll around one of its internal axes, but switches to a different axis every so often. Furthermore, high-dimensional systems such as serpentoid or catenoid robots, quadrupeds, hexapods and wheeled robots have been successfully controlled (see Martius et al. (2011)).

It is of particular interest that the control algorithm induces a preference for movements with a high degree of coordination among the various degrees of freedom. All the robotic implementations demonstrate the emergence of play-like behavior, which are characterized by coordinated whole body movements seemingly without a specific goal. The coordination among the various degrees of freedom arises from their physical coupling that is extracted and enhanced by the controller, because each motor neuron is adapted to be sensitive to coherent changes in all degrees of freedom due to Eq. 8.

Guided Self-Organizing Control

How can we guide the joint dynamics of state (1) and parameters (8) in order to realize a given goal by the self-organizing process? One option is to modify the lifetime of the transient behaviors depending on a given reward signal, see Martius et al. (2007). A second and more stringent form of guidance was proposed by Martius and Herrmann (2010) and will be augmented and applied to a high-dimensional system in the present paper. We will formulate the problem in terms of problem-specific error functions (PSEF) that indicate an external goal by minimal values. A trivial example of such an error function is the difference between externally defined and actually executed motor actions. This is a standard control problem which, however, becomes difficult if the explorative dynamics is to be preserved.

GSO focuses on this interplay between the explorative dynamics implied by homeokinetic learning and the additional drives. The challenge in the combination of a self-organizing system with external goals becomes clear when recalling the characteristics of a self-organizing system. One important feature is the spontaneous breaking of symmetries

of the system. This is a prerequisite for spontaneous pattern formation and is usually achieved by self-amplification, i.e. small noisy perturbations cause the system to choose one of several symmetric options while the intrinsic dynamics then causes the system to settle into this asymmetric state. A nonlinear stabilization of the self-amplification forms another ingredient of self-organization. These two conditions which we will call our working regime, are to be met for a successful guidance of a self-organizing system. There are several ways to guide the homeokinetic controller which we will discuss in the following.

Guidance by Problem-Specific Teaching

First we will describe how problem-specific error functions (PSEF) can be integrated. Recall that the adaptation of the controller parameters is done by performing a gradient descent on the time-loop error. The PSEF must depend functionally on the controller parameters in order to enable the same procedure. Unfortunately, the simple sum of both gradients (of the time-loop error and of the PSEF) is likely to steer the system out of its working regime. Furthermore, we cannot easily identify a fixed weighting between the two gradients that would satisfy an adequate pursuit of the goal while maintaining explorativity. One reason is that the nonlinearity (Eq. 2) in the TLE causes the gradient to vary over orders of magnitude. A solution to this problem can be obtained by scaling the gradient of the PSEF according to the Jacobian matrix (see 7) of the sensorimotor loop such that both gradients become compatible. This transformation is essentially a natural gradient with the Jacobian matrix of the sensorimotor loop as a metrics. The update for the controller parameters C is now given by

$$\frac{1}{\epsilon_C}\Delta C_t = -\frac{\partial E_{TLE}}{\partial C} - \gamma \frac{\partial E_G}{\partial C}\left(L_t L_t^\top\right)^{-1}, \qquad (9)$$

where E_G is the PSEF and $\gamma \geq 0$ is the guidance factor deciding the strength of the guidance. For $\gamma = 0$ there is no guidance and we re-obtain the unmodified dynamics (Eq. 8).

For clarity we will start with a very simple goal, namely we want a robot to follow predefined motor actions called *teaching signals* in addition to the homeokinetic behavior. We can define the PSEF as the mismatch η_t^G between motor teaching cues y_t^G and the actual motor values, thus

$$E_G = \|\eta_t^G\|^2 = \|y_t^G - y_t\|^2. \qquad (10)$$

Since y_t is functionally dependent on the controller parameters (Eq. 2), the gradient descent can be performed, i.e. the derivative reads $\frac{\partial E_G}{\partial C_{ij}} = -\eta_i^G\, g_i'\, x_j$, where $g_i' = \tanh'\left(\sum_{j=1}^n C_{ij}x_j + h_i\right)$ (all quantities at time t). A similarly motivated approach is in linear systems is homeotaxis (Prokopenko et al., 2008).

An evaluation of the guidance mechanism has been performed using the TWOWHEELED robot, which was simulated in the realistic robot simulator LPZROBOTS (Martius et al., 2011). The motor values determine the nominal wheel velocities and the sensor values report the actual wheel velocities of both wheels. We provided to both motors the same oscillating teaching signal. The resulting behavior is a mixture between the taught behavior and self-organized dynamics depending the value of γ. For $\gamma = 0.01$ the teaching cues are followed most of the time but with occasional exploratory interruptions, especially when the teaching cues have a small absolute value. In this case the system is closer to the bifurcation point where the two stable fixed points for forward and backward motion meet. These interruptions cause the robot, for example, to move in curved fashion instead of strictly driving in a straight line as the teaching cue suggest. The exploration around the teaching signals might be useful in general to find modes which are better predictable or more active.

Interestingly, we can similarly define a mechanism that uses teaching cues in terms of sensor values (Martius and Herrmann, 2010).

Guidance by Cross-Motor Teaching

Guidance mechanism can also use internal teaching signals. As an illustrative example, consider the mirror-symmetry that is preferred in many control systems. We will first follow this idea and describe a simple implementation following this example before we generalize this scheme later in order to apply it to high-dimensional systems. In either case, motor values of some motors will be used as teaching signals for other motors.

Pairwise symmetries. For two motors, guidance can be introduced by

$$y_{t,1}^G = y_{t,2} \qquad \text{and} \qquad y_{t,2}^G = y_{t,1}, \qquad (11)$$

where y_t^G is the vector of nominal motor values, see (9, 10). For experimental evaluation we placed the TWOWHEELED robot in an environment cluttered with obstacles and performed many trials for different values of the guidance factor. The robot was rewarded for straight movement and was therefore expected not to get stuck at obstacles or in corners and cover substantial parts of its environment. In order to quantify the influence of the guidance we recorded the trajectory, the linear velocity, and the angular velocity of the robot. We expect an increase in linear velocity because the robot is to move straight instead of circling. For the same reason the angular velocity should be lowered. In Fig. 2 the behavioral quantification and a several sample trajectories are plotted. Additionally the relative area coverage is shown, which indicates that much more area of the environment was covered by the robot with guidance compared to freely moving robot. As expected, the robot shows a distinct decrease in mean turning velocity and a higher area coverage with increasing values of the guidance factor until the guidance becomes dominant and the performance drops. In the

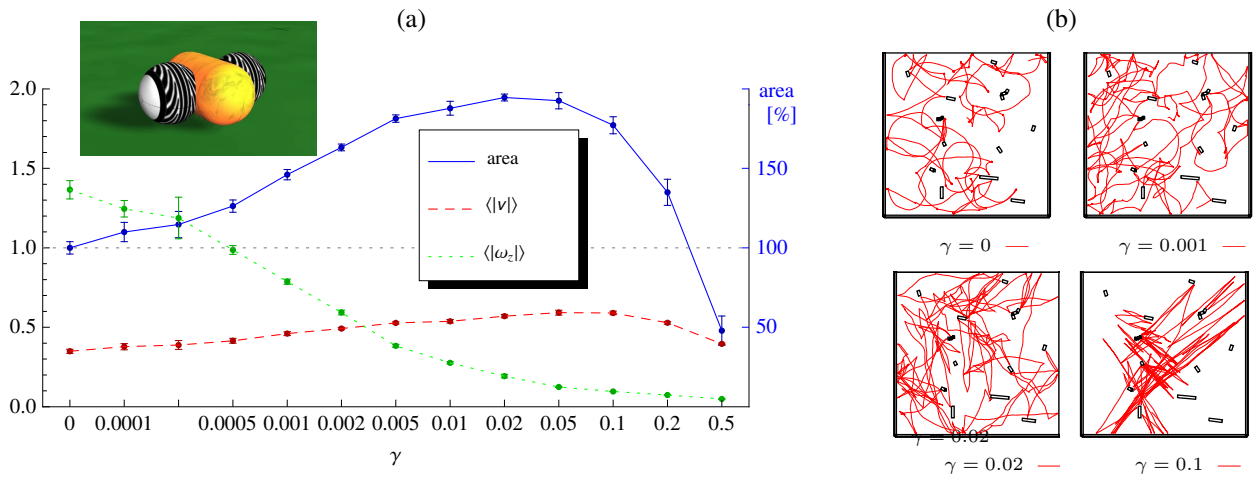

Figure 2: Behavior of the TWOWHEELED robot when guided to move preferably straight. **(a)** Mean and standard deviation (of five runs each 20 min) of the area coverage (**area**), the average velocity $\langle|\boldsymbol{v}|\rangle$, and the average angular velocity $\langle|\omega_z|\rangle$ for different values of the guidance factor γ. Area coverage (box counting method with 300×300 boxes) is given in percent relative to case $\gamma = 0$ (**right axis**). The robot is driving straighter and its trajectory covers more area for larger γ. The inset shows a screenshot of the simulation. **(b)** Example trajectories for different guidance factors. Note that for $\gamma = 0.1$ still many boxes are visited but less well spread. Parameters: $\epsilon_c = \epsilon_a = 0.01$, update rate 100 Hz.

normal regime the robot is still performing turns and drives both backwards and forwards and that it does not get stuck at the walls, as seen in the trajectory in Fig. 2(b), is because the sensitivity (exploration) and predictability (exploitation) of the controller remain. If the guidance is too strong the favorable properties of the self-organizing behavior are lost such that the robot stalls or performs repetitively the same motion. Note that already very small values of γ yield a high effect of the guidance.

Permutation relations. In a more general cross-motor teaching setup, each motor has one incoming and one outgoing connection, such that there is still only one teaching signal per motor. The connections can be described by a permutation π_m of m motors that assigns each motor a source of teaching input. The teaching signal is then given by

$$\left(y_t^S\right)_i = \left(y_t\right)_{\pi_m(i)} \qquad \text{for } i = 1, \ldots, m. \tag{12}$$

With a cyclic schema of connections a group of motors can be synchronized. In the following experiment we use a rotation-symmetric motor connection setting to show that a high-dimensional chain-like robot can quickly develop a locomotion behavior.

The ARMBAND robot consists of a sequence of flat segments placed in a ring-like configuration, where subsequent segments are connected by motor-operated hinge joints. As a result we obtain a robot with the appearance of a bracelet or chain, see Fig. 1. Each joint provides a sensor value of the current position. The motor values define target joint positions, which typically cannot be reached due to substantial physical constraints and underactuation. In this way the controller obtains informative feedback from the robotic body. Since the robot is symmetric there is by construction no preferred direction of motion, meaning that the homeokinetically controller robot will move forward or backward with equal probability. The robot cannot turn or move sideways, but it can produce a variety of postures and locomotion patterns.

With the method of cross-motor teaching we can select different symmetries, such that the robot is more likely to perform a directed motion. For that we define the permutation used in Eq. 12 as

$$\pi_m(i) = (i + k + \lfloor m/2 \rfloor) \mod m, \tag{13}$$

where $k \in \{-1, 0, 1\}$. Coarsely speaking, this connects motors on the opposite side of the robot with a shift to one or the other side in a way that depends on k. The choice of k reflects the desired direction of motion and depends on whether the number of joints m is even or odd. If m is even then $k = -1$ and $k = 1$ are used for both directions (forward or backward) and $k = 0$ represents a point symmetric connection setup. In the latter case the robot will not prefer a direction of motion and the behavior is similar to the case without guidance. For odd values of m, which is used here, $k = 0$ and $k = 1$ need to be used, resp., for backward and forward motion. In the following experiments the robot has $m = 13$ motors. The motor connections for $k = 1$ are shown in Fig. 3. Each motor connection is displayed by an arrow pointing to the receiving motor. Note that the connections are directed and a motor is not teaching the same motor from which it is receiving teaching cues. For $k = 0$ (and n

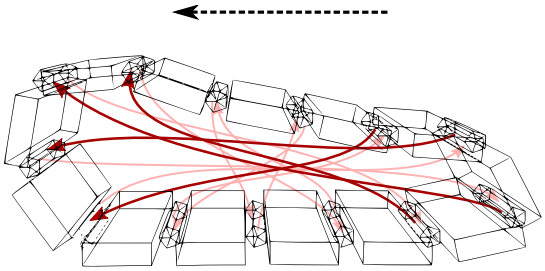

Figure 3: ARMBAND robot with cross-motor connections. Links are connected by hinge joints that are actuated by servo motors. The curved arrows indicate unidirectional cross-motor connections. For these connections the robot preferably moves leftwards. All links are identical, but four links are drawn boldly for better visibility.

odd) all arrows are inverted, meaning that for each connection the sending and receiving motors would swap roles.

Results

To evaluate the performance we conducted for different values of the guidance factor γ_S five trials each 30 min long. In a first setting the cross-motor connections were fixed ($k = 1$) for the entire duration of the experiment. We observed the formation of a locomotion behavior after a very short time. Note that this behavior requires all joints of the robot to be highly coordinated. As a quantitative measure of the performance we calculate the horizontal velocity $\boldsymbol{v}$ using the center of mass of the robot. Thus, the velocity is a scalar and we define forward motion if $\boldsymbol{v} > 0$ and backward motion if $\boldsymbol{v} < 0$. In this experiment we expect the robot to move only forward, because a fixed cross-motor connection setup was used. The average velocity of the robot increased distinctively with raising guidance factors, see Fig. 4(a). For excessively large values of the guidance factor γ_S the velocity goes down again. This occurs for two reasons: First, the cross-motor teaching has a too strong influence on the working regime of the homeokinetic controller and second the actual motor pattern of the locomotion behavior does not perfectly obey the relations between the motor values, not all motor values are exactly equal. Again, already a small value of γ is sufficient to achieve the goal. It appears the self-organizing system needs only very little influence to be guided into the desired regions of the behavior space.

Without guidance the robot moves equally to both directions but with comparably low velocity. This can be seen at the mean of the absolute velocity in Fig. 4(a). If the value of the guidance factor is chosen conveniently, the robot moves in one direction with varying speed see Fig. 4(b) for 3 velocity traces. The velocity traces are seen to have a peak followed by a dip before a more steady regime is attained. It appears that the controller learning surpasses a more optimal configuration with respect to the velocity, but there the tradeoff between self-organizing and guidance is not met. Later strong fluctuations may occur that reflect the explorative nature of the homeokinetic part. The locomotive behavior can also be seen in Video 1, see Ref. (Supplement, 2011), for a low value of guidance factor ($\gamma_s = 0.001$) and in Video 2 for a medium value of guidance factor ($\gamma_s = 0.003$).

In a second setup, we changed the cross-motor connections every 5 min, i. e. k was changed from 0 to 1 and back. A value of $k = 0$ should lead to a negative velocity and a $k = 1$ to a positive velocity. To study the dependence on the guidance factor and to measure the performance we use the average absolute velocity ($\langle|\boldsymbol{v}|\rangle$) and the correlation of the velocity with the configuration of the connections ($\rho(\boldsymbol{v}, k)$), see Fig. 5(a). Without guidance ($\gamma_S = 0$) there is, as expected, no correlation with the supposed direction of locomotion. For a range of values of the guidance factor we find a high total locomotion speed with a strong correlation to the supposed direction of motion. Note that the size of the correlation depends on the length of the intervals of one connection setting. For long intervals the correlation will approach one. In Fig. 5(b) the velocity of the robot is plotted for different runs with the same value of the guidance factor that was used in the previous experiment ($\gamma_S = 0.003$). We observe that the robot changes the direction of motion shortly after the configuration of connections was changed, see also Video 3 at Supplement (2011).

The locomotion of the robot is essentially influenced by the number of cross-motor connections. For that we use again the fixed connectivity. In a series of simulations a number $0 \leq l \leq m$ equally spaced cross-motor connections (Fig. 3) are used. With increasing l the robot start to locomote earlier. Full performance is reached already if 8 out of the 13 connections are used, see Fig. 6(a).

In order to study the scaling properties of the learning algorithm we varied the number of segments m of the robot and thus the dimensionality of the control problem. The results are astonishing, see Fig. 6(b): The behavior is learned with the same speed also for large number (40) of segments. There is no scaling problem here for the following reason. In the closed loop with an approximate feedback strength (self-regulated by the homeokinetic controller) the robot needs only very little influence to roll. The length of the robot can even help because other behavioral modes (e. g. wobbling) are damped increasingly due to gravitational forces. For the same reason, small robots are slower than medium ones. Large robots are again slower because the available forces at the joints become too weak. The experiment illustrates that specific behaviors can be achieved in a high-dimensional robot by using cross-motor teachings. Cross-motor connections can break the symmetry between the two directions of motion such that a locomotory behavior is produces quickly. When the connections are switched later during runtime, the behavior of the robot changes reliably.

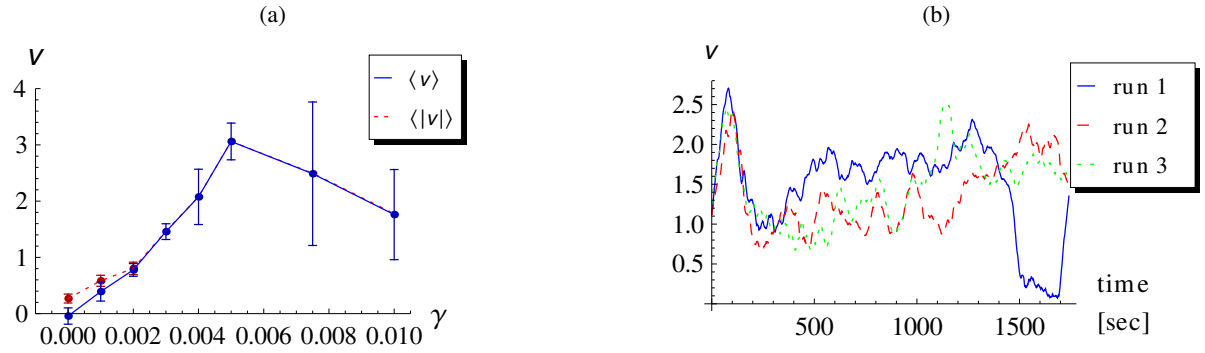

Figure 4: Performance of the ARMBAND robot with constant cross-motor teaching. **(a)** Mean and standard deviation of the average velocity $\langle \boldsymbol{v} \rangle$ and the average absolute velocity $\langle |\boldsymbol{v}| \rangle$ of five runs for different guidance factors γ_S. **(b)** Velocity of the robot $\bar{v}$ (average over 1-minute sliding window) for three runs at $\gamma_S = 0.003$, $k = 1$, $\epsilon_c = \epsilon_a = 0.1$, 100 Hz update rate.

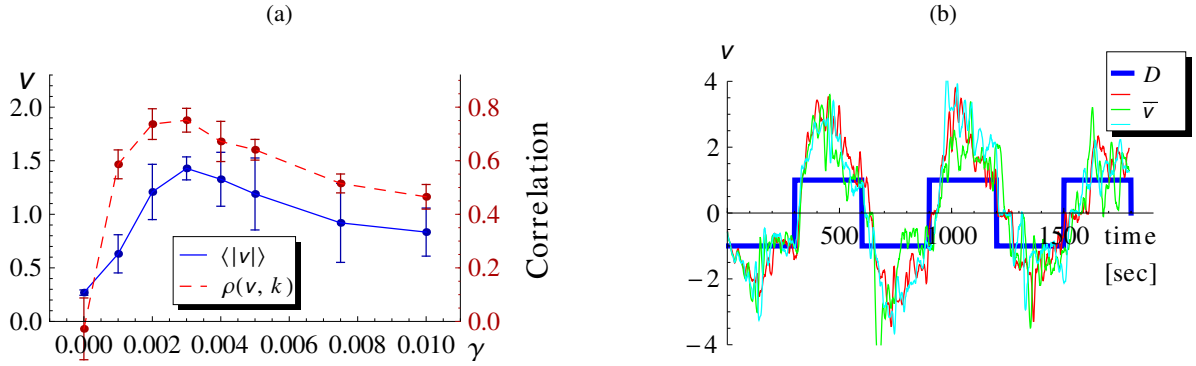

Figure 5: Performance with switching cross-motor teaching. **(a)** Mean and standard deviation of the average absolute velocity $\langle |\boldsymbol{v}| \rangle$ and the correlation $\rho(\boldsymbol{v}, k)$ of the velocity with the configuration of the connections of five runs for different guidance factors γ_S. **(b)** Velocity (average over 10-seconds sliding window) for three runs of the robot with a supposed direction of motion D. Parameters as in Fig. 4.

The guidance mechanism can also be transferred to sensor space using the direct sensor teaching, which was discussed above and was proposed by Martius and Herrmann (2010). One obtains a cross-sensor teaching analogously to the definitions given above. This can become useful, for example, if a certain behavior is demonstrated by a human operator by passively moving the robot. In the case of the ARMBAND robot, one can easily imagine that the robot is pushed along the ground such that a locomotion pattern is formed. Based on the sensor readings, the correlations between the sensor channels can be determined and serve as a basis for the construction of a specific cross-sensor teaching configuration.

Discussion

We have presented here two mechanisms to guide the homeokinetic self-organization of behavior. The first one uses desired motor patterns that were introduced into the learning dynamics by means of an additional error function. The strength of guidance can be conveniently adjusted. We have considered also cross-motor teaching as a new way of using the directed teaching to select desired behaviors. The approach introduced here is realized by a permutation of the motors signal for teaching. We applied this algorithm to a bracelet-like robot (ARMBAND) with many degrees of freedom and demonstrated the accelerated development of locomotion behavior from scratch. Even the relearning to the opposite direction of motion is possible very quickly. Since the learning is very fast and the performance changes gradually with changing γ, the guidance factor could be adapted automatically. Most striking is the scaling of the algorithm to higher dimensions. In the present case the performance did not decrease when the robot was enlarged to have 40 DoF. This is a result of the exploitation of the embodiment by the self-organization process.

The exploratory character of the controller is retained under guidance and helps to find a behavioral mode even if the specification of the motor teaching signals are partially contradictory. For example, the TWOWHEELED robot can choose freely between driving forward or backward, because the behavior-space is only partially constrained. Furthermore, it is evident that the robot remains sensitive to small perturbations and continues to explore its environment. The

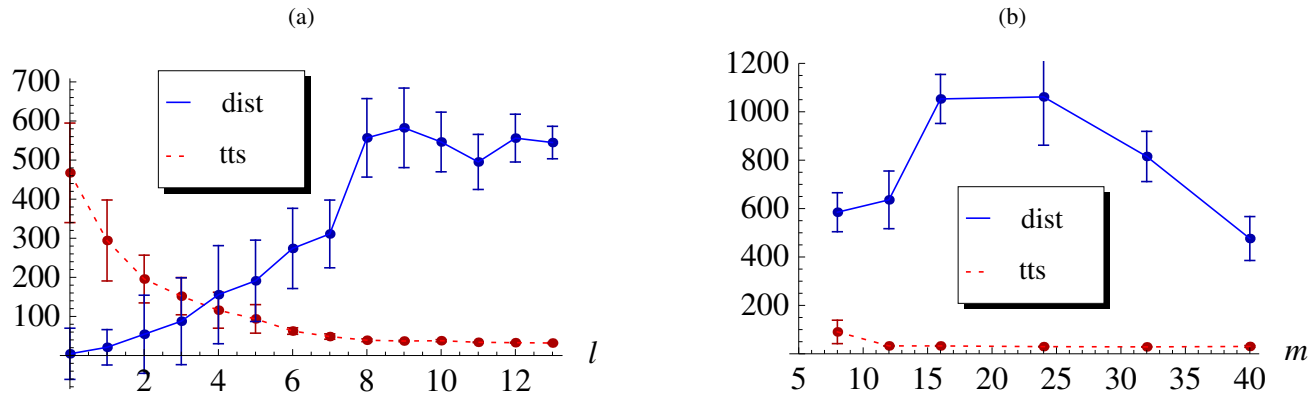

Figure 6: Scaling of learning time and performance for different robot complexity. The plots show mean and standard deviation of the distance traveled by the robot ('dist' in units of 1 segment size) and of the time-to-start ('tts' in seconds) of 20 runs à 10 min ($\gamma = 0.003$). **(a)** Performance as a function of the number of cross-motor connections l (equally spaced around a robot with $m = 13$ joints). **(b)** Performance for different numbers of segments m (DoF) with full cross-motor connectivity ($l = m$).

constraints are not strictly enforced by the algorithm but the self-organization can find a mode that fits better to the particular embodiment. The presented experiments with the ARMBAND demonstrate this effect. The guidance signal alone would synchronize all motors to the same value (same phase in the oscillations) which does not lead to a locomotion behavior whereas the combined learning dynamics leads to a smooth and adaptive locomotion, see Video 3 (Supplement, 2011).

Acknowledgment: The project was supported within the National Bernstein Network by the BMBF grant #01GQ0432 at BCCN Göttingen. We are grateful to Ralf Der and Theo Geisel for encouragement and discussions.

References

Der, R. (2001). Self-organized acquisition of situated behaviors. *Theory in Biosciences*, 120:179–187.

Der, R., Hesse, F., and Martius, G. (2006). Rocking stamper and jumping snake from a dynamical system approach to artificial life. *Adaptive Behavior*, 14(2):105–115.

Herrmann, J. M. (2001). Dynamical systems for predictive control of autonomous robots. *Theory in Biosciences*, 120:241–252.

Hesse, F., Martius, G., Der, R., and Herrmann, J. M. (2009). A sensor-based learning algorithm for the self-organization of robot behavior. *Algorithms*, 2(1):398–409.

Jung, T., Polani, D., and Stone, P. (2011). Empowerment for continuous agent-environment systems. *Adapt. Beh.*, 19:16–39.

Lopez, M. and Oudeyer, P.-Y. (2010). Active learning and intrinsically motivated exploration in robots: Advances and challenges. *IEEE Transactions on Autonomous Mental Development*, 2(2):65–69.

Martius, G., Fiedler, K., and Herrmann, J. M. (2008). Structure from behavior in autonomous agents. In *Proc. IEEE IROS 2008*, pages 858 – 862.

Martius, G. and Herrmann, J. (2010). Taming the beast: Guided self-organization of behavior in autonomous robots. In Doncieux, S. et al., editors, *From Animals to Animats 11*, volume 6226 of *LNCS*, pages 50–61. Springer.

Martius, G., Herrmann, J. M., and Der, R. (2007). Guided self-organisation for autonomous robot development. In Almeida e Costa, F. et al., editors, *Proc. Advances in Artificial Life, (ECAL 2007)*, volume 4648 of *LNCS*, pages 766–775. Springer.

Martius, G., Hesse, F., Güttler, F., and Der, R. (2011). LPZROBOTS: A free and powerful robot simulator. `http://robot.informatik.uni-leipzig.de`.

Nolfi, S. (2006). Behaviour as a complex adaptive system: On the role of self-organization in the development of individual and collective behaviour. *ComPlexUs*, 2(3-4):195–203.

Oudeyer, P.-Y., Kaplan, F., Hafner, V. V., and Whyte, A. (2005). The playground experiment: Task-independent development of a curious robot. In *AAAI Spring Symp. on Developmental Robotics, 2005*, pages 42–47, Stanford, California.

Prokopenko, M. (2009). Guided self-organization. *HFSP Journal*, 3(5):287–289.

Prokopenko, M., Zeman, A., and Li, R. (2008). Homeotaxis: Coordination with persistent time-loops. In Asada, M. et al., editors, *SAB*, volume 5040 of *LNCS*, pages 403–414. Springer.

Schmidhuber, J. (1991). A possibility for implementing curiosity and boredom in model-building neural controllers. In Meyer, J. A. and Wilson, S. W., editors, *Proc. Simulation of Adaptive Behavior: From Animals to Animats*, pages 222–227. MIT.

Singh, S., Barto, A., and Chentanez, N. (2004). Intrinsically motivated reinforcement learning. In *Proc. 18th Annual Conf. Neural Information Proc. Systems*, Vancouver, BC, Canada.

Supplement (2011). Videos for this article. `http://robot.informatik.uni-leipzig.de/research/supplementary/ECAL2011`.

Tani, J. (2003). Learning to generate articulated behavior through the bottom-up and the top-down interaction processes. *Neural Networks*, 16(1):11–23.