Taming the Beast: Guided Self-organization of Behavior in Autonomous Robots

Georg Martius[1,2,3] and J. Michael Herrmann[1,2,4]

[1] Bernstein Center for Computational Neuroscience Göttingen
[2] Institute for Nonlinear Dynamics, University of Göttingen
[3] Max Planck Institute for Dynamics and Self-Organization
[1−3]Bunsenstraße 10, 37073 Göttingen, Germany
[4] University of Edinburgh, School of Informatics, IPAB
10 Crichton Street, Edinburgh EH8 9AB, U.K.
georg@nld.ds.mpg.de, michael.herrmann@ed.ac.uk

Abstract. Self-organizing processes are crucial for the development of living beings. Practical applications in robots may benefit from the self-organization of behavior, e.g. for the increased fault tolerance and enhanced flexibility provided that external goals can also be achieved. We present several methods for the guidance of self-organizing control by externally prescribed criteria. We show that the degree of self-organized explorativity of the robot can be regulated and that problem-specific error functions, hints, or abstract symbolic descriptions of a goal can be reconciled with the continuous robot dynamics.

1 Introduction

Intrinsically motivated but non-trivial behavior is an important prerequisite for autonomous robot development. Self-organization of robot control is a promising approach, where the resulting behavior is characterized by on-going exploration or by a refinement of those behavioral traits that can be called natural for a specific robot in a particular environment [1,2]. Animals, including humans, acquire their behavioral repertoire in a similar way, behavioral elements are developed autonomously and are further refined during the whole life span. Nevertheless, modulatory effects on the self-organizing behavior can be imposed as well by the environment. Animals can learn by imitation or by downright teaching from superior fellows. Furthermore, behavior is subject to the dictate of drives that are partly intrinsic and partly external to the agent. Finally, humans derive goals for their own behavior from rational reasoning.

Incentives for behavioral adaptation is an interesting subject for study in behavioral science where the interference of such higher forms of learning with the underlying self-organization does not seem to be a problem. In robotics, however, the situation is different. Although promising examples exist [1,3,4], self-organization of behavior is 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.

S. Doncieux et al. (Eds.): SAB 2010, LNAI 6226, pp. 50–61, 2010.
© Springer-Verlag Berlin Heidelberg 2010

Usually, goal-oriented behavior is achieved by directly optimizing 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 allow for different 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 [5,6]. In this 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. Self-organizing systems tend to have a high tolerance against failures and degrade gracefully, which is an advantage that should not be given up when developing systems aiming to achieve tasks in practical applications. Although being interested in the wider context, we will be dealing in this particular study with a specific approach to self-organizing control, namely homeokinetic learning [7].

What can we expect from a *guided homeokinetic controller*? It has been shown earlier that a variety of behaviors can emerged from the principle of homeokinesis [1,2]. This process of self-organization selects certain elements from the space of action sequences such that a set of behaviors is realized. The emerging behaviors show a coherent sensorimotor dynamics of the particular robot in its environment. The goal is now to shape the self-organization process to produce desired or preferred behaviors within a short time. Part of the idea is to channel the exploration of the homeokinetic controller around certain behaviors, such that control modes can be found which match the given robotic task.

In the present paper, we will discuss three mechanisms of guidance. The first one uses online reward signals to shape the emerging behaviors and is briefly discussed in Section 3. A second mechanism for guiding consists in the incorporation of supervised learning e. g. by specific nominal motor commands that we call *teaching signals* (Section 4). Using distal learning [8] we study the utilization of teaching signals in terms of sensor values in Section 5. In Section 6 we propose a third mechanism that allows for the specification of mutual motor teaching. The latter two are presented here for the first time.

2 Self-organized Closed Loop Control

Self-organizing control for autonomous robots can be achieved by establishing an intrinsic drive towards behavioral activity as described by the homeokinetic principle [7], for details cf. [1,2].

The dynamical evolution of the sensor values $x \in \mathbb{R}^n$ of the robot is described by

$$x_{t+1} = \psi(x_t) = M(x_t, y_t, \mathcal{A}) + \xi_{t+1}. \tag{1}$$

where M is the internal predictive model that maps the sensations x and the actions $y \in \mathbb{R}^m$ to the predicted sensory inputs, $\mathcal{A}$ is a set of parameters and ξ

is the mismatch between the predicted and the actually observed sensor values. In this study, the internal model M is implemented as a linear neural network:

$$M(x_t, y_t, \mathcal{A}) = Ay_t + Sx_t + b, \tag{2}$$

where $\mathcal{A} = (A, S, b)$. The actions y are generated by a controller function

$$y_t = K(x_t, C, h) = g(Cx_t + h) \tag{3}$$

where $g(\cdot)$ is a componentwise sigmoidal function, we use $g_i(z) = \tanh(z_i)$, C is a weight matrix and h is a bias vector.

The parameters $\mathcal{A}$ of the model are adapted online to minimize the prediction error $\|\xi\|^2$ (Eq. 1) via gradient descent. However, the minimization is ambiguous with respect to A and S because y is a function of x, see (3). In contrast to our earlier approach [5], we introduce a bias into the model learning in order to capture the essential part of the mapping by the matrix A. This is achieved by the adaptation of A based on a prediction error that is obtained for a discounted S term, i. e.

$$\Delta A = \epsilon_A (\xi_{t+1} + \delta S x_t) y_t^\top, \tag{4}$$
$$\Delta S = \epsilon_A \xi_{t+1} x_t^\top, \tag{5}$$

where a small value of $\delta = 0.001$ fully serves the purpose and $\epsilon_A = 0.1$ is a learning rate.

If the parameters of the controller (C, h) are also adapted by the minimization of the prediction error $\|\xi\|^2$ then stable but typically trivial behaviors are achieved. The robot may get trapped in any state with $\xi = 0$ which happens prevalently when it is doing nothing. There are, however, specific cases where such a principle can be successfully applied: If the drive for activity is provided from outside or brought about by e. g. evolution [9], or if a homeostatic rule is applied to, for instance, the neural activity [10,11]. The homeokinetic paradigm [7,1] instead suggests to use the so-called *postdiction error*. This error is the mismatch

$$v_t = x_t - \hat{x}_t \tag{6}$$

between true sensor values x_t and reconstructed sensor values $\hat{x}_t$ that are defined using Eq. 1 as

$$\hat{x}_t = \psi^{-1}(x_{t+1}) \tag{7}$$

assuming that ψ is invertible. If $\hat{x}_t$ (rather than x_t) had been actually observed then by definition the best possible prediction based on the present model M (1) would have been made. The error functional minimizing the postdiction error v_t is called *time-loop error* (TLE) and can be approximated by

$$E_{TLE} = \|v_t\|^2 = \xi_{t+1}^\top \left(L_t L_t^\top\right)^{-1} \xi_{t+1}, \tag{8}$$

where $L_{t,ij} = \frac{\partial \psi(x_t)_i}{\partial x_{t,j}}$ is the Jacobian matrix of ψ at time t. Thus another important feature of this error quantity becomes evident: The minimization of v entails

the minimization of the inverse Jacobian. This in turn means that small eigenvalues of L are increased. Thus the controller performs stabilization in inverted time, i. e. destabilization in forward time. This eliminates the trivial fixed points (in sensor space) and enables spontaneous symmetry breaking phenomena. The reader might wonder why the system does not start to behave chaotically or reach uncontrollable oscillations. The reason is that the destabilization is limited by the nonlinearities $g(\cdot)$ and that the TLE is invariant to oscillation frequencies as discussed in [12]. Intuitively, the homeokinesis can be understood as the drive to sustain a non-trivial behavior that can be predicted by the internal model. Since the internal model is very simple smooth behaviors are preferred. Fig. 1 illustrates how the homeokinetic controller is connected to a robot.

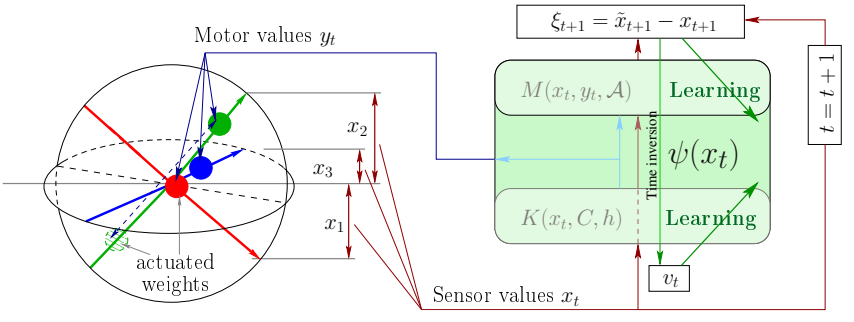

Fig. 1. The Homeokinetic controller connected to the SPHERICAL robot in the sensorimotor loop. The SPHERICAL robot is driven by weights that are moved along the axes by actuator and is equipped with axis-orientation sensors (x_i). The homeokinetic controller consists of the controller function K and the predictor M, both together form ψ (Eq. 1). The TLE is obtained by propagating ξ_{t+1} through ψ in inverted time.

The TLE (8) can be minimized by gradient descent which gives rise to a parameter dynamics that evolves simultaneously with the state dynamics:

$$x_{t+1} = \psi(x_t) + \xi_{t+1}, \tag{9}$$

$$C_{t+1} = C_t - \epsilon_C \frac{\partial}{\partial C} E_{TLE} \quad \text{and} \quad h_{t+1} = h_t - \epsilon_h \frac{\partial}{\partial h} E_{TLE}, \tag{10}$$

where $\epsilon_C = \epsilon_h = 0.1$ is chosen for the learning rate. We use a fast synaptic dynamic for the learning of the controller and the model such that the system adapts quickly. Assuming sensory noise, the TLE is never zero nor has a vanishing gradient such that the rule (10) produces an 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 time span and transitions are hard to predict.

Let us consider as a first example a robot with two wheels that is equipped with wheel velocity sensors. In the beginning the robot rests, but after a short time it autonomously starts to drive forward and backward and to turn. If a

wall is encountered such that the wheels stop the robot will immediately stop the motors and eventually drive in the free direction. A more complex example for the self-organization of *natural* behaviors is provided by the SPHERICAL robot (Fig. 1) which starts to roll around different internal axes as we will see below. Furthermore, high-dimensional systems such as snake- or chain-like robots, quadrupeds, and wheeled robots have been successfully controlled [13]. 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. 10. In this paper we will propose a mechanism to guide the self-organizing behaviors towards desired behaviors.

3 Guided Self-organizing Control

How can we guide the learning dynamics such that a given goal is realized by the self-organizing process? One option is to modify the lifetime of the transient behaviors depending on a given reward signal. For this purpose we can explicitly modify the frequencies of occurrence of different behaviors and obtain more of a desired and less of an undesired behavior. The prediction error ξ occurs as a factor in the learning rule (8), i. e. the lifetime of well predictable behavior is extended such that the original TLE already contains a reward for predictability in this formalism. When applying this method to the SPHERICAL robot (Fig. 1) we can, for example, achieve fast locomotion by rewarding high velocity and obtain curved driving and spinning modes when rewarding rotational velocity around the upwards axis, see [5] for more details.

A second and more stringent form of guidance will be studied 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 hard if the explorative dynamics is to be preserved.

Guided self-organization (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 a number of ways to guide the homeokinetic controller which we will discuss in the following.

4 Guidance by Teaching

First we will describe how the problem-specific error functions (PSEF) can be integrated and then we will consider a few examples. 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 allow the same procedure. Unfortunately, the simple sum of both gradients is likely to steer the system out of its working regime and we cannot easily identify a fixed weighting between the two gradients that would satisfy an adequate pursuit of the goal and maintaining explorativity. One reason is that the nonlinearities (cf. Eq. 3) in the TLE cause 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 of the sensorimotor loop such that both gradients become compatible. It turns out that this transformation can be obtained using the natural gradient with the Jacobian matrix of the sensorimotor loop as a metric. The update for the controller parameters C is now given by

$$\frac{1}{\epsilon_C} \Delta C_t = -\frac{\partial E_{TLE}}{\partial C} - \gamma \frac{\partial E_G}{\partial C} \left(L_t L_t^\top \right)^{-1}, \tag{11}$$

where E_G is the PSEF and $\gamma > 0$ is the guidance factor deciding the strength of the guidance. For $\gamma = 0$ there is no guidance and we obtain the unmodified dynamics, cf. (10).

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 signals y_t^G and the actual motor values, thus

$$E_G = \|\eta_t^G\|^2 = \|y_t^G - y_t\|^2. \tag{12}$$

Since y_t is functionally dependent on the controller parameters (3), the gradient descent can be performed, i.e. the derivative reads $\frac{\partial E_G}{\partial C_{ij}} = -\eta_{t,i}^G \, g_i' \, x_{t,j}$, where $g_i' = \tanh' \left(\sum_{j=1}^n C_{ij} x_{t,j} + h_i \right)$. A similarly motivated approach is homeotaxis [14], where an action error is added to the TLE as well, however the error was minimized in one step, and not along its gradient.

An evaluation of the guidance mechanism has been performed using the TwoWheeled robot, which was simulated in our realistic robot simulator LpzRobots [15]. 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 signals are followed most of the time but

with occasional exploratory interruptions, especially when the teaching signals 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 signals dictate. The exploration around the teaching signals might be useful in general to find modes which are better predictable or more active.

5 Sensor Teaching and Distal Learning

Let us now transfer the motor teaching paradigm to sensor teaching signals. This is a useful way of teaching because desired proprioceptive sensor values can be more easily obtained than motor values, for instance by passively moving the robot or parts of the robot. This kind of teaching is also commonly used when humans learn a new skill, e. g. think of a tennis trainer that teaches a new stroke by moving the arm and the racket of the learner. Thus, a series of nominal sensations can be acquired that can serve as teaching signals. Setups where the desired outputs are provided in a different domain than the actual controller outputs are called *distal learning* [8]. Usually a forward model is learned that maps actions to sensations (or more generally to the space of the desired output signals). Then the mismatch between a desired and actual sensation can be back-propagated to obtain the required change of action. The back-propagation can also be done using an inversion of the forward model which we have already at hand, see Eqs. 1 and 7. The idea is actually very simple, namely calculating motor teaching signals from sensor teaching signal using the inverted model by solving $x_t^D = M(x_{t-1}, y_{t-1}^G, \mathcal{A})$ w.r.t. y^G, cf. Eq. 2, which can in turn be inserted into Eq. 12. Afterwards we apply the motor teaching mechanism (Section 4).

The potential of this method will become more obvious in the following more complex example. We use a simulated robot named the SPHERICAL which is of relatively simple shape, but involves a complicated control problem, see Fig. 1. We will consider the goal of restricting the movements of the robot to rotations around one of its axes. The robot is actuated by three internal weights that are movable along orthogonal axes. Thus a single change in the positions of the weights results in a change of the center of mass of the robots and thus in a certain rolling movement. Control has to take into account strong inertia effects and a non-trivial map between motor actions and body movements. Let us first consider the behavior without guidance ($\gamma = 0$). From a resting initial situation, the rule (10) induces an increasing sensitivity by noise amplification until a coherent physical movements develop. Shortly afterwards a regular rolling behavior is executed which breaks down infrequently to give way for different movement patterns. In particular the rolling modes around one of the internal axes are seen to occur preferably, see Fig. 2(a,b). This modes are characterized by small sensor values for the rotation axis whereas the remaining two sensor values oscillate.

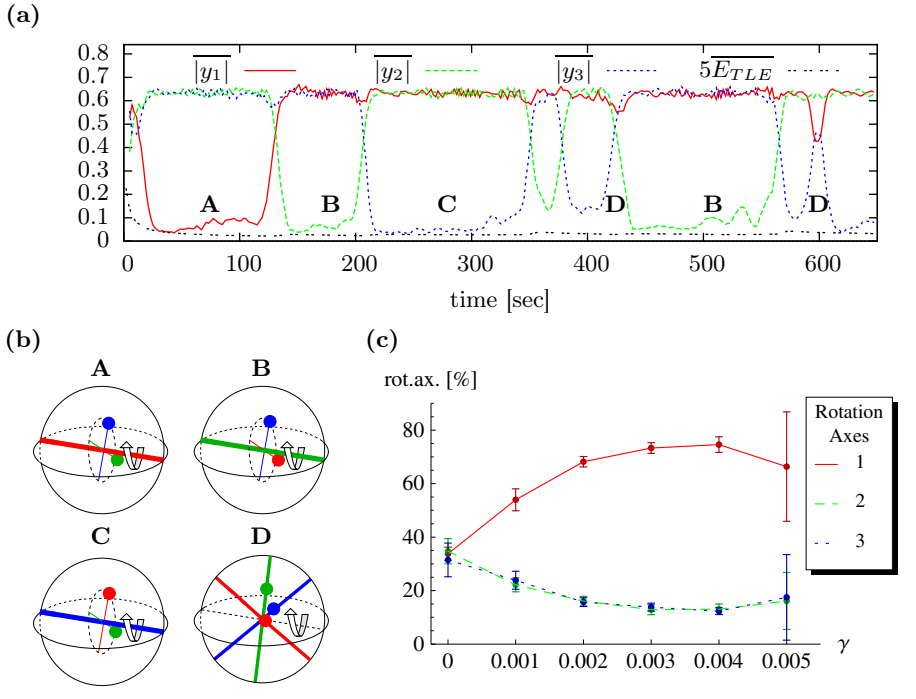

Fig. 2. The SPHERICAL robot without guidance explores its behavioral options. With guidance it prefers a specific axis of rotation. **(a)** Amplitudes of the motor value oscillations ($y_{1\ldots3}$) and the TLE (E_{TLE}) averaged over 10 sec (scaled for visibility) without guidance ($\gamma = 0$). Corresponding behaviors are indicated with letters **A-D**. **(b)** Sketch of four typical behaviors (**A-D**), namely the rolling mode around the three internal axis (**A-C**) and around any other axis (**D**); **(c)** Behavior for the distal learning task. The percentage of rotation around each of the internal axes is shown for different values of the guidance factor γ (no teaching for $\gamma = 0$). The rotation around the red (first) axis is clearly preferred for non-zero γ (mean and standard deviation are plotted for 10 runs each of a duration of 60 min).

In order to guide the robot into the rotation around the first axis we use a distal teaching signal where the first component is zero and the remaining two components contain the current sensor values such that they do not generate any learning signal (i.e. the mismatch is zero). The teaching signal vector is formally $x_t^G = (0 \;\; x_{t,2} \;\; x_{t,3})^\top$, where $x_{t,1\ldots3}$ are the sensor values at time t. As a descriptive measure of the behavior, we used the index of the internal axis around which the highest rotational velocity was measured at each moment of time. Figure 2(c) displays for different values of the guidance factor (γ) and for each of the axes the percentage of time it was the major axis of rotation. Without guidance there is no preferred axis of rotation as expected. With distal learning the robot shows a significant preference for a rotation around the first axis up to 75 %. For overly strong teaching, a large variance in the performance occurs. This is caused by a too strong influence of the teaching

signal on the learning dynamics. Remember that the rolling modes can emerge due to the fine regulation of the sensorimotor loop to the working regime of the homeokinetic controller, which cannot be maintained for large values of γ. We may ask why is it not possible to force the controller to stay in the rotational mode around the first axis? When the robot is in this rotational mode the teaching signal is negligible. However, the controller's drive to be sensitive will increase the influence of the first sensor such that the mode becomes unstable again.

To summarize, the SPHERICAL robot with the homeokinetic controller can be guided to move mostly by rotation around one particular axis, by specifying the constancy of a single sensor as a teaching signal.

6 Guidance by Cross-Motor Teaching

Finally we will propose a guidance mechanism with internal teaching signals. As an example we want to influence the controller to prefer a mirror-symmetry in the motor patterns. This can be achieved by using the motor value of one motor as the teaching signal for another motor and vice versa. For two motors, this can be expressed as: $y^G_{t,1} = y_{t,2}$ and $y^G_{t,2} = y_{t,1}$, where y^G_t is again the nominal motor value vector, see Eqs. 11 and 12. This self-supervised teaching induces soft constraints which reduce the effective dimension of the sensorimotor dynamics and thus guide the self-organization along a sub-space of the original control problem.

Let us consider the TWOWHEELED robot again and suppose the robot should move mostly straight, not get stuck at obstacles or in corners and cover substantial parts of its environment. We will see that all this can be achieved by a simple guidance of the homeokinetic controller where both motors are mutually teaching each other.

For experimental evaluation we placed the robot in an environment cluttered with obstacles and performed many trials for different values of the guidance factor. 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 turning. For the same reason the angular velocity should be lowered. In Fig. 3 the behavioral quantification and a sample trajectory are plotted. Additionally the relative area coverage is shown, which reflects how 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. Note that 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. 3(b), such as sensitivity (exploration) and predictability (exploitation) remain.

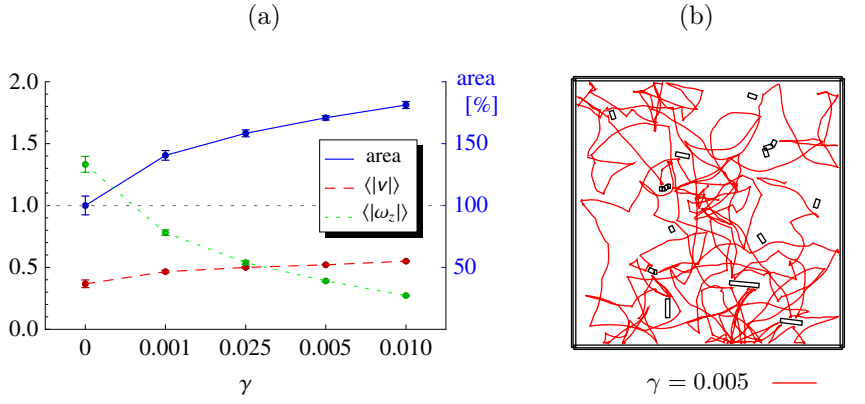

Fig. 3. Behavior of the TwoWheeled robot when guided to move preferably straight. **(a)** Mean and standard deviation (of 5 runs each 20 min) of the area coverage (**area**), the average velocity $\langle|\boldsymbol{v}|\rangle$, and the average angular velocity $\langle|\omega_z|\rangle$ for different values of the guidance factor γ. Area coverage (box counting method) is given in percent of the case without guidance ($\gamma = 0$) (**right axis**). The robot is driving straighter and its trajectory covers more area for larger γ; **(b)** An example trajectory of the robot with $\gamma = 0.005$.

7 Discussion

We have presented here two new methods for guiding self-organizing behavior that are based on teaching signals. Desired motor patterns were specified by means of an error function that was integrated into the learning dynamics. The strength of guidance can be conveniently adjusted. Because teaching information is often given in the sensor space whereas learning is performed in the motor representation, a transformation is necessary which is obtained from the adaptive internal world model. The feasibility of both approaches was demonstrated by robotic experiments.

We introduced cross-motor teachings in order to be able to specify relations between different motor channels. If it is known or desired that certain degrees of freedom of a robot should move in a coherent way, e. g. symmetrical or anti-symmetrical, then these relation can be injected as soft constraints that reduce the effective dimensionality of the system. As an example, the TwoWheeled robot showed that by enforcing the symmetry between the left and right wheel the behavior changes qualitatively to straight motion.

The exploratory character of the controller is nevertheless retained and helps to find a behavioral mode even if the specification of the motor couplings is partially contradictory. The resulting behaviors are not enforced by the algorithm. For example the TwoWheeled robot can choose freely between driving forward or backward whereas in direct teaching the direction of driving is obviously dictated by an external teacher. Furthermore, it is evident that the robot remains sensitive to small perturbations and continues to explore its environment.

Guided self-organization using cross-motor teachings shares some properties with other approaches to autonomous robot control such as evolutionary algorithms [16] and reinforcement learning (RL) [17]. Evolutionary algorithms can optimize the parameters of the controller and are able to produce the same behaviors as we found in this study, cf. [18,19,20]. A critical experiment would investigate high-dimensional systems that cannot be decomposed into identical components.

A further difference is that self-organizing control is merely modulated by guidance, whereas evolutionary algorithms tend to converge to a static control structure. RL uses discrete actions or a parametric representation of the action space. In either case, high-dimensional systems will cause slow convergence. Preliminary experiments with a chain-like robot (cf. [13]) show a clear advantage of cross-motor teaching in comparison to generic RL although similar relations among the actions in RL compensate part of this drawback. Natural actor-critics [21] may bring a further improvement of the RL control, but natural gradients can also be incorporated here. A decisive advantage of cross-motor teaching may be that goal-directed behaviors emerge within the self-organization of the dynamics from a symbolic description of the problem and do not need continuous training data such as in imitation learning [22].

It is, however, clearly an interesting option to adapt cross-motor teaching to an imitation learning scenario. Although delayed rewards are still non-trivial for continuous domains, RL can cope with them in principle, while the guidance with rewards [5] requires instantaneous rewards.

Acknowledgment. The project was supported by the BMBF grant #01GQ0432.

References

1. Der, R., Hesse, F., Martius, G.: Rocking stamper and jumping snake from a dynamical system approach to artificial life. Adapt. Beh. 14, 105–115 (2006)
2. Hesse, F., Martius, G., Der, R., Herrmann, J.M.: A sensor-based learning algorithm for the self-organization of robot behavior. Algorithms 2(1), 398–409 (2009)
3. Stefano, N.: 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 (2006)
4. Tani, J.: Learning to generate articulated behavior through the bottom-up and the top-down interaction processes. Neural Networks 16(1), 11–23 (2003)
5. Martius, G., Herrmann, J.M., Der, R.: Guided self-organisation for autonomous robot development. In: Almeida e Costa, F., Rocha, L.M., Costa, E., Harvey, I., Coutinho, A. (eds.) ECAL 2007. LNCS (LNAI), vol. 4648, pp. 766–775. Springer, Heidelberg (2007)
6. Prokopenko, M.: Guided self-organization. HFSP Journal 3(5), 287–289 (2009)
7. Der, R.: Self-organized acquisition of situated behavior. Theory in Biosciences 120, 179–187 (2001)
8. Jordan, M.I., Rumelhart, D.E.: Forward models: Supervised learning with a distal teacher. Cognitive Science 16(3), 307–354 (1992)
9. Nolfi, S., Floreano, D.: Learning and evolution. Auton. Robots 7(1), 89–113 (1999)

10. Di Paolo, E.: Organismically-inspired robotics: Homeostatic adaptation and natural teleology beyond the closed sensorimotor loop. In: Murase, K., Asakura, T. (eds.) Dyn. Systems Approach to Embodiment and Sociality, pp. 19–42 (2003)
11. Williams, H.: Homeostatic plasticity in recurrent neural networks. In: Schaal, S., Ispeert, A. (eds.) From Animals to Animats: Proc. 8th Intl. Conf. on Simulation of Adaptive Behavior, vol. 8. MIT Press, Cambridge (2004)
12. Der, R., Martius, G.: From motor babbling to purposive actions: Emerging self-exploration in a dynamical systems approach to early robot development. In: Nolfi, S., Baldassarre, G., Calabretta, R., Hallam, J.C.T., Marocco, D., Meyer, J.-A., Miglino, O., Parisi, D. (eds.) SAB 2006. LNCS (LNAI), vol. 4095, pp. 406–421. Springer, Heidelberg (2006)
13. Der, R., Martius, G., Hesse, F., Güttler, F.: Videos of self-organized behavior in autonomous robots (2009), `http://robot.informatik.uni-leipzig.de/videos`
14. Prokopenko, M., Zeman, A., Li, R.: Homeotaxis: Coordination with persistent timeloops. In: Asada, M., Hallam, J.C.T., Meyer, J.-A., Tani, J. (eds.) SAB 2008. LNCS (LNAI), vol. 5040, pp. 403–414. Springer, Heidelberg (2008)
15. Martius, G., Hesse, F., Güttler, F., Der, R.: LpzRobots: A free and powerful robot simulator (2009), `http://robot.informatik.uni-leipzig.de/software`
16. Nolfi, S., Floreano, D.: Evolutionary Robotics. In: The Biology, Intelligence, and Technology of Self-organizing Machines. MIT Press, Cambridge (2001); 1st Print (2000), 2nd Print (2001)
17. Sutton, R.S., Barto, A.G.: Reinforcement learning: Past, present and future. In: SEAL, pp. 195–197 (1998)
18. de Margerie, E., Mouret, J.B., Doncieux, S., Meyer, J.A.: Artificial evolution of the morphology and kinematics in a flapping-wing mini UAV. Bioinspiration and Biomimetics 2, 65–82 (2007)
19. Ijspeert, A.J., Hallam, J., Willshaw, D.: Evolving Swimming Controllers for a Simulated Lamprey with Inspiration from Neurobiology. Adaptive Behavior 7(2), 151–172 (1999)
20. Mazzapioda, M.G., Nolfi, S.: Synchronization and gait adaptation in evolving hexapod robots. In: Nolfi, S., Baldassarre, G., Calabretta, R., Hallam, J.C.T., Marocco, D., Meyer, J.-A., Miglino, O., Parisi, D. (eds.) SAB 2006. LNCS (LNAI), vol. 4095, pp. 113–125. Springer, Heidelberg (2006)
21. Peters, J., Vijayakumar, S., Schaal, S.: Natural Actor-Critic. In: Gama, J., Camacho, R., Brazdil, P.B., Jorge, A.M., Torgo, L. (eds.) ECML 2005. LNCS (LNAI), vol. 3720, pp. 280–291. Springer, Heidelberg (2005)
22. Peters, J., Schaal, S.: Natural Actor-Critic. Neurocomputing 71(7-9), 1180–1190 (2008)