Biol Cybern (2010) 103:255–271
DOI 10.1007/s00422-010-0396-4

ORIGINAL PAPER

Biological Cybernetics

Behavioral analysis of differential hebbian learning in closed-loop systems

Tomas Kulvicius · Christoph Kolodziejski · Minija Tamosiunaite · Bernd Porr · Florentin Wörgötter

Received: 23 March 2010 / Accepted: 31 May 2010 / Published online: 17 June 2010

Abstract Understanding closed loop behavioral systems is a non-trivial problem, especially when they change during learning. Descriptions of closed loop systems in terms of information theory date back to the 1950s, however, there have been only a few attempts which take into account learning, mostly measuring information of inputs. In this study we analyze a specific type of closed loop system by looking at the input as well as the output space. For this, we investigate simulated agents that perform differential Hebbian learning (STDP). In the first part we show that analytical solutions can be found for the temporal development of such systems for relatively simple cases. In the second part of this study we try to answer the following question: How can we predict which system from a given class would be the best for a particular scenario? This question is addressed using energy, input/output ratio and entropy measures and investigating their development during learning. This way we can show that within well-specified scenarios there are indeed agents which are optimal with respect to their structure and adaptive properties.

Keywords Adaptive systems · Sensorimotor loop · Learning and plasticity · Entropy · Input/output ratio · Energy · Optimal agents

1 Introduction

Behaving systems form a closed loop with their environment where sensor inputs influence motor output, which, in turn, will create different sensations. Simple systems of this kind are reflex-based agents which react in a stereotyped way to sensory stimulation, either by a retraction or an attraction reflex (Braitenberg Vehicles, Braitenberg, 1986). If the environment is not too complex, one can describe (linear) systems of this kind also in the closed loop case by methods from systems theory. For this, the transfer functions of agent and environment need to be known and the characteristics of the control-loop also needs to be taken into account.

The situation becomes much more complicated as soon as one allows the controller to adapt, for example by learning. Now the transfer function of the agent changes over time and thereby its interaction with the world, which not only influences its behavior but also the learning, resulting in an ongoing change of the behavior. It is exceedingly difficult to describe such non-stationary situations.

Two very general questions arise here. (1) To what degree is it possible to describe the temporal development of such adaptive systems using only knowledge about their initial configuration, their learning mechanism and knowledge about the structure of the world? and (2) Given a certain

Tomas Kulvicius and Christoph Kolodziejski have contributed equally to this work.

T. Kulvicius (✉) · C. Kolodziejski · F. Wörgötter
Bernstein Center for Computational Neuroscience, Department for Computational Neuroscience, III Physikalisches Institut - Biophysik, Georg-August-Universität Göttingen, Friedrich-Hund Platz 1, 37077 Göttingen, Germany
e-mail: tomas@physik3.gwdg.de

C. Kolodziejski
e-mail: kolo@physik3.gwdg.de

M. Tamosiunaite
Department of Informatics, Vytautas Magnus University, Vileikos g. 8, Kaunas, Lithuania
e-mail: m.tamosiunaite@if.vdu.lt

B. Porr
Department of Electronics & Electrical Engineering, University of Glasgow, Glasgow, GT12 8LT, Scotland
e-mail: B.Porr@elec.gla.ac.uk

complexity of the world can we predict which system from a given class would be the best (in some well defined sense)?

Clearly these questions are too general to be answered without constraining "system" and "world" much more. But even when doing so, the problem remains intricate due to the non-stationary closed loop configuration.

In this study, we will focus on systems that perform differential hebbian learning (Hebb 1949; Sutton and Barto 1981; Kosco 1986; Klopf 1988), related to spike-timing dependent plasticity (Markram et al. 1997; Saudargiene et al. 2004, 2005), for the learning of temporal sequences of paired sensor events. Temporal sequences of sensor events are common for animals and humans and exist as soon as the same event is registered first by a "far-sensor" (e.g., eye, ear, nose) and later by a "near-sensor" (e.g., touch-sensor, taste-bud). Our systems are initially built as reflex loops and the learning goal is to avoid this reflex. A simple example, also used here, is a robot that learns to avoid obstacles. Such a machine can start—like a Braitenberg Vehicle—with a touch-triggered (signal x_0, Fig. 1a) retraction reflex and learn to use a far-sensor (i.e., infrared signal x_1) to turn earlier and stop running into obstructions thereby avoiding the reflex (Porr and Wörgötter 2003a). Fig. 1a shows the general control diagram for such systems, discussed in several previous articles (Porr and Wörgötter 2003a,b, 2006; Kulvicius et al. 2007). Inputs arise from the system's own behavior and can be understood as disturbance-events D that enter the inner loop via the transfer function P_0 of the world and become a sensor event at sensor x_0. This sensor is the near-sensor and triggers the reflex motor action z. The far-sensor x_1 has received the same disturbance already earlier (via transfer function P_1) leading to a stimulation sequence: first x_1 then x_0 (Fig. 1b). This is depicted by the delay variable τ between inner and outer loop. During learning the influence of x_1 onto output z will grow via weight ω_1 leading to an earlier motor action and, thus, to the avoidance of the reflex.

Previous studies (Porr and Wörgötter 2003a,b, 2006; Kulvicius et al. 2007) have derived stability conditions for environments where the transfer functions were constant. However, while the agent interacts with its environment these transfer functions change. For example, the question arises whether the delay τ between the predicting event x_1 and the reflex trigger x_0 would change while learning to avoid obstacles. Thinking of an obstacle-avoiding robot, intuitively one would expect τ to get longer as the growing influence of x_1 should lead to a later and later triggering (by x_0) of the reflex until it is finally fully eliminated. This is shown in Fig. 1b trajectory (1) versus trajectory (2), where the robot beetle depicted uses two sets of antennas (short and long) for near (x_0) and far (x_1) sensing, respectively. The intuition of a growing τ is alluring, but just shows how even in the simplest cases our understanding of such adaptive systems can go astray. Because, as shown below and contrary

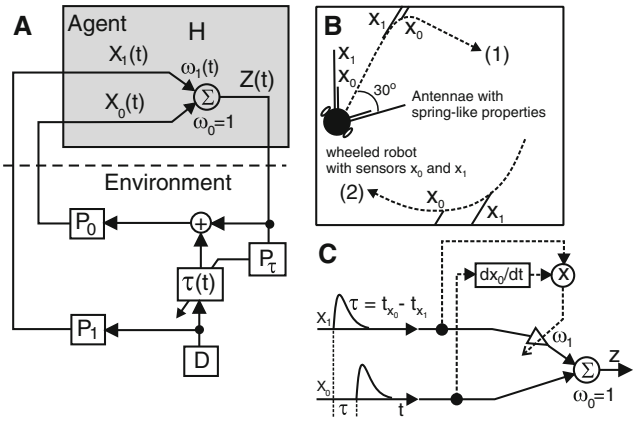

Fig. 1 **a** Schematic diagram of the closed-loop learning system with inputs x_0 and x_1, connection weights ω_0 and ω_1 and neuronal output z. P_0 and P_1 denote the reflexive and the predictive pathway, respectively. D defines the disturbance, where τ is the time difference between inputs x_0 and x_1 as shown in **c**. **b** Robot setup with short antennas (reflexive inputs, x_0) and long antennas (predictive inputs, x_1). The diagram shows a situation with an increase of the time difference between far- and near-sensor events during learning process ($\tau_2 > \tau_1$), depicted by the respective distance between the little (*solid*) triggering lines $x_{0,1}$ from trajectory (*dashed*) to wall. **c** Schematic diagram of the input correlation learning rule and the signal structure (ICO, Porr and Wörgötter 2006; Kulvicius et al. 2007)

to our naive intuition, such systems are better described by a τ which first grows and then shrinks again back to essentially its starting value. A deeper look into the development of these systems allows understanding why this happens and we can even derive an analytical approximation for the weight development in these cases.

In the second part of the study we define energy, input/output ratio and entropy measures for these systems and measure them in environments of different complexity. Using these measures we will first show that learning equalizes the energy uptake across agents *and* worlds. Strong differences which initially exist are being leveled out during learning. However, when judging learning speed and complexity of the resulting behavior one finds a trade-off and some agents will be better than others in the different worlds tested.

The analysis of closed-loop systems is a well established field in the engineering sciences, which also investigates "adaptive controllers". Very little, however, is known about adaptive controllers which interact with their environment by shaping non-stationary dynamics through their own learning (see Sect. 4). It had been shown that Shanon's Information Theory (Shannon 1948) can be applied for perception-action loops (Ashby 1956; Tishby et al. 1999; Touchette and Lloyd 2000). Few other studies exist that try to analyze closed loop systems from an agent-perspective asking about the information processing properties of such system in the context of what would be beneficial for the agent itself (Klyubin et al. 2007, 2008; Lungarella et al. 2005; Lungarella and Sporns

2006; Prokopenko et al. 2006). Even fewer attempts exist that consider learning (Lungarella and Sporns 2006; Porr et al. 2006). This study is, to our knowledge, one of the few which tries to address these issues in closed-loop learning systems. While our scenarios have strong constrains, the newly introduced information measures can be applied to a wide range of adaptive predictive controllers independent on the system setup.

The article is organized in the following way. First off all we will describe the environment and the adaptive controller of our system and define several system measures. Then we will present results from single experiments to demonstrate the basic behavior of the system and provide an analytical solution of its temporal development. Afterwards we will show results for the different system measures showing a statistical analysis for different agents and different environments. Finally we will discuss the question of "optimal robots" and will conclude our study with Sect. 4 relating our work to other approaches.

2 Methods

Note, all spatial measures in the following are given in arbitrary "size units" (short "units"), time is measured in "steps".

2.1 Agent

The structure of the simulated agent used for these simulations is shown in Fig. 1b. It is a Braitenberg Vehicle of diameter 40 units with two lateral wheels. It operates in a square arena of 400 × 400 units or a circular arena with diameter of 400 units, which can be empty ("simple world") or contain different numbers of obstacles ("complex worlds"). By default, the agent drives straight forward (dashed arrow) with speed $v = 1$ units per time step. It has two sensor-pairs, near-sensors and far-sensors, at the front; each sensor resembling a beetle's antenna, albeit here with ideal spring-like properties. Short near-sensors elicit the reflex signal x_0 and long far-sensors the predictive signal x_1. Triggering of a sensor will happen as soon as the agent gets close enough to an obstacle. Then the sensor signal x will be elicited according to:

$$x(t) = \beta x(t-1) + (1-\beta)\frac{\lambda_t}{\Lambda}, \tag{1}$$

where λ_t is the part of the antenna bent by an obstacle at time point t and Λ is the length of the antenna. The constant $\beta = 0.6$ defines the decay rate of the first order low pass implemented by the feedback $x(t-1)$. We use a fixed reflex antenna length of $\Lambda_0 = 10$ units and different antenna lengths for the predictive sensor of $\Lambda_1 = 40, 50, 60, 70, 80, 100, 120, 150, 200$. In the following we will use the antenna ratio $\frac{\Lambda_1}{\Lambda_0}$ to specify different robots.

To compute the agents's output z we use a linear summation neuron with an added-on winner-takes-all mechanism in order to prevent the robot from getting stuck in the corner or from oscillatory movements in case sensors on the left and right side are triggered at the same time. If sensors are triggered at the same time but one sensor is triggered more than the other ($x^R_{0,1} \neq x^L_{0,1}$) then z will follow strongest of the left and right sensor signals:

$$b_{0,1} = \text{sign}\left(x^R_{0,1} - x^L_{0,1}\right),$$
$$z = \omega_0 b_0 \max\left\{x^R_0, x^L_0\right\} + \omega_1 b_1 \max\left\{x^R_1, x^L_1\right\}, \tag{2}$$

where $x^{L,R}_{0,1}$ are sensory inputs from the left and right side obtained by Eq. 1 above.

In case $x^R_{0,1} = x^L_{0,1}$ we use a bias for the right side and calculate output by $z = \omega_0 x^R_0 + \omega_1 x^R_1$.

Note that winner-takes-all mechanism does not keep inputs from canceling out but creates lateral inhibition which means that, for instance, if the input on the right side is triggered stronger than that on the left side then the input from the left side will be ignored (inhibited) and the robot will turn to the left side until at some point sensors on the right and the left side will be triggered equally. If sensors are triggered equally then we use a bias for the right side (input from the left side is ignored) and as a consequence the robot will continue turning to the left until the obstacle is completely avoided. Thus, the winner-takes-all mechanism (lateral inhibition) together with built in bias helps preventing the robot from getting stuck in corners or from oscillatory movements.

Signal z is then directly used to change the robot's driving angle α. For the remainder of this study it is important to remember that z directly corresponds to the *change of the turning angle* $\mathrm{d}\alpha/\mathrm{d}t$:

$$\frac{\mathrm{d}\alpha}{\mathrm{d}t} = g_\alpha z(t), \tag{3}$$

where g_α is the steering gain. Positive output ($z > 0$) leads to a positive change of the turning angle $\mathrm{d}\alpha/\mathrm{d}t$, thus the robot would steer to the left, whereas negative output ($z < 0$) leads to a negative change of the turning angle $\mathrm{d}\alpha/\mathrm{d}t$ and corresponds to a rightward steering. From this the change of the robot position can be calculated for each time step and as a result of this setup the agent will avoid obstacles when moving through its arena.

We keep the weight w_0 fixed ($w_0 = 1$) and let only w_1 develop where initially we set $w_1 = 0$.

2.2 Learning rule

For learning we use the ICO (input correlation) rule (Porr and Wörgötter 2006), because of its intrinsic stability, given by (Fig. 1c):

$$\frac{d\omega_1}{dt} = \mu x_1 \frac{dx_0}{dt} \tag{4}$$

Note, that the typical low pass filtering of the input signals for ICO learning (Porr and Wörgötter 2006) is performed by the environment itself and by Eq. 1.

2.3 Closed-loop system

The general structure of the closed loop has been presented in Fig. 1a and was discussed in the introduction so that only a few explanations need to be added here.

In general we denote the transfer function of the agent by H and those of the environment by P. In Fig. 1a we have added the time variable t to all those components of which the temporal development is of interest in the context of this study: x_0, x_1, z, τ and ω_1. The other synaptic weight ω_0 is kept constant at 1.0.

2.4 Experimental procedure

We tested nine simulated robots with different antenna ratio $\Lambda_1/\Lambda_0 = [4, 5, 6, 7, 8, 10, 12, 15, 20]$ in four environments of different complexity. We used a circular environment with a diameter of 400 units where complexity was defined by the number of obstacles (3, 7, 14, or 21). We used square shaped obstacles of size 20 × 20 units that were placed at random positions in a circular manner at the perimeter of three imaginary circles with radii of 50, 120, and 190 points. This way we avoid deadlock situations and assure a free path along the whole circular arena. Several examples of the simplest and the most complex environments are shown in Fig. 2.

Two different types of experiments are being made. (1) Normal learning experiments where the robots actually learn while driving and (2) steady state experiments (called weight freezing), where we keep ω_1 for some time at a preset value for measuring the currently queried parameter(s) in a steady state situation. Then ω_1 will be increased and parameter(s) will be measured again and so on until we are reaching the final weight ω_1^f at which the reflex is not triggered anymore.

We used the following procedure for this. We set the weight ω_1 to specific values $(0, \Delta\omega_1, 2\Delta\omega_1, \ldots \omega_1^f$, where $\Delta\omega_1 = 10^{-3})$ and, for each ω_1, let the robot run for $N = 20000$ time steps without learning.

Such a procedure is motivated by the fact that the actual runtime is irrelevant (as explained above). Thus, by setting weights we can probe the robot's behavior for a longer period in a steady state situation in order to get more data for the analysis.

2.5 System measures

In the following we will present different measures used to evaluate temporal development and success of learning, and to find the optimal robot for a specific environment.

2.5.1 Temporal development

To analyze the temporal development we measure how the temporal difference τ between inputs x_1 and x_0 changes on average during learning. As events in these systems are very noisy we need to adopt a method by which the time-difference between two subsequent x_1, x_0 events is reliably measured. For this we use the weight freezing procedure and keep $\omega_1 = \text{const}$ for N time steps. We use a threshold with value $\theta = 0.02$ only for the x_0 signal to determine the time t_k when the signal x_0 reaches the threshold $(x_0 > \theta)$. Finally, we place a window $c_w = 300$ steps around these t_k values $(c_w \leq t_k < N - c_w, N = 20{,}000)$ and calculate the cross-correlation between x_1 and x_0 only inside this time window, with a window size related to STDP windows as reported in the literature (Markram et al. 1997; Bi and Poo 1998). This is given by:

$$C^k(t) = \sum_{T=-c_w}^{T=+c_w} x_1(t_k) \cdot x_0(t_k + T), \tag{5}$$

We determine the peak location of the cross-correlation as:

$$\tau_k = argmax\{C^k(t)\}. \tag{6}$$

Finally, we calculate the mean value of the obtained different time differences τ_k for the whole frozen time section (N steps) according to:

$$\overline{\tau} = \frac{1}{M} \sum_{k=1}^{M} \tau_k, \tag{7}$$

where M is the number of found threshold crossings. After increasing ω_1, this procedure is repeated until ω_1^f.

2.5.2 Energy

We measure how much energy the robot uses for a given task during the learning process. In physics the total kinetic energy of an extended object is defined as the sum of the translational kinetic energy of the center of mass and the rotational kinetic energy about the center of mass:

$$E_k = \frac{1}{2} m v^2 + \frac{1}{2} I \omega^2, \tag{8}$$

where m is the mass (translational inertia), I is the moment of inertia (rotational inertia), v and ω are the velocity and angular velocity, respectively. As we use a constant basic speed

v and all our robots have the same size we can simplify the previous equation and define the mean output energy as:

$$\overline{E_z} = \frac{g_\alpha^2}{2N} \sum_{t=0}^{N-1} z^2(t). \tag{9}$$

We note that the change of the turning angle $\frac{\mathrm{d}\alpha}{\mathrm{d}t} = g_\alpha z(t)$ is directly to be understood as the angular velocity w.

2.5.3 Input/output ratio

We define the input/output ratio H_z which measures the relation between reflexive and predictive contribution to the final output, and shows how this relation changes during the learning process. At the beginning of learning only the reflexive output will be elicited which would lead to zero value. With learning ratio should grow and reach a maximum when reflexive and predictive parts contribute to the output evenly. After that ratio should go down back to zero since the reflex is being avoided and at the end of learning only predictive reactions will be elicited.

We define the absolute value of the neuronal output for the x_0 pathway:

$$\begin{aligned} z_0(t) &= x_0(t) \cdot w_0, \\ |z_0| &= \sum_{t=0}^{N-1} |z_0(t)|, \end{aligned} \tag{10}$$

and for the x_1 pathway:

$$\begin{aligned} z_1(t) &= x_1(t) \cdot w_1(t), \\ |z_1| &= \sum_{t=0}^{N-1} |z_1(t)|, \end{aligned} \tag{11}$$

where N is the length of the sequence (here $N = 20000$ time steps). The total absolute value of neuronal output can be defined as:

$$\begin{aligned} z(t) &= z_0(t) + z_1(t). \\ |z| &= \sum_{t=0}^{N-1} |z(t)|, \end{aligned} \tag{12}$$

Finally, the input/output ratio can be calculated by the following equation:

$$H_z = -\left(\frac{|z_0|}{|z|} log_2 \frac{|z_0|}{|z|} + \frac{|z_1|}{|z|} log_2 \frac{|z_1|}{|z|} \right). \tag{13}$$

Note that this measure would be similar to an entropy measure if one would use the probabilities that an output z is generated by the reflex x_0 or predictor x_1 instead of the integrals $|z_{0/1}|$.

2.5.4 Path entropy

The following measure quantifies the complexity of the agent's trajectory during the learning process. The function z determines the state of the orientation of both wheels (particles) relative to each other as the relative speed of one particle against the other determines the turn angle and hence the orientation of the robot. If the robot only makes sharp turns then we would find for z ideally only two values: zero for "no turn" and one other (high) value for "sharp turn". In defining the path entropy H_p in an information theoretical way by *number of states taken divided by number of all possible states* this would yield a very low entropy as only two states out of many possible turns are taken. On the other hand the path entropy would reach its maximum value if all possible steering reactions will be elicited with equal probability.

Thus, in order to calculate the path entropy we need to get probabilities $p(z_i)$ of the output function z for each value z_i. To do that, first we calculate a cumulative distribution function of z by:

$$F_c(z) = \sum_{z_i \le z} f(z_i), \tag{14}$$

where $z = 0, \Delta z, \ldots 1$ (we used $\Delta z = 2 \times 10^{-3}$). Here $f(z_i) = 1$ if $z_i \le z$, and $f(z_i) = 0$ otherwise. From the cumulative distribution function we calculate a probability distribution function to be able to calculate the probability of the different values of z given by $p(z)$:

$$p(z) = \frac{\Delta F_c(z)}{\Delta z}. \tag{15}$$

Then we can define H_p in the usual way as:

$$H_p = -\sum_z p(z) log_2 p(z). \tag{16}$$

2.5.5 Speed of learning

To evaluate the speed of learning in our study we assess weight development and not time, noting that elapsed time is irrelevant. For instance, if the robot drives around a long time without touching obstacles (no learning events) this would not influence the weight. Learning is driven by events (x_1 and x_0 pairs) which is directly reflected by the weight growth and this we relate to the speed of learning. Hence we can determine the speed of learning of a specific agent by measuring at which weight the agent reaches the maximum input/output ratio value, where reflex and predictor contribute equally to the output. Thus, we define the learning speed S as being inversely proportional to this weight:

$$S = \left(\mathrm{argmax}\{H_z(\omega_1)\}\right)^{-1}, \tag{17}$$

with $\omega_1 = 0, \Delta\omega_1, \ldots, \omega_1^f$, where ω_1^f denotes the final weight at which the reflex x_0 is not triggered anymore.

Note in a given environment one finds that learning events can occur more or less often depending on the sensitivity of the reflex. In this case—to compare architectures at the reflex level—one would indeed want to measure time as such. We are, however, in the current study not concerned with this.

2.5.6 Optimality

In order to find an optimal robot for a specific environment we used an averaged optimality measure O which is a product of the speed of learning S and the final path entropy $H_p(\omega_1^f)$:

$$O = S \cdot H_p(\omega_1^f). \tag{18}$$

Note that we normalized values of S and $H_p(\omega_1^f)$ between zero and one before calculating the product in Eq. 18. This measure is based on the heuristic assumption that an "optimal robot" should be one which requires least learning time and still "makes the most of it" in the sense of producing the most complex paths. Therefore, we used the quotient of these two measures to define "optimal". In general optimality measures used to quantify behavior in engineered or self-organizing systems as well as in animals-observation are always in the eyes of beholder where combination of different features can be taken into account to describe what is deemed to be "optimal behavior" and this depends also on the specific task.

3 Results

3.1 Basic behavior of the system

The basic behavior of the obstacle avoidance agent is presented in Fig. 2 where we show simulation results for a circular environment with 3 and 21 obstacles. In panels a and b, we show weight development and corresponding driving trajectories (see insets) for the case where the robot was actually learning (no weight freezing here). The resulting weight curves for both cases are similar and we observe relatively rapid weight growth at the beginning of the learning and then slow saturation till the reflex is avoided and weights finally stabilize. Corresponding trajectories are color-coded where the blue color corresponds to reflex-driven behavior and the red color corresponds to predictor-driven behavior. Values for the color-coding were calculated by a contrast measure given in Appendix A.1. From the driving trajectories we can see that at the beginning the robots make sharp turns because of the initially built in strong reflex reaction whereby, as a consequence, the robot explores more or less the whole environment. With learning the predictor takes over which at the end leads to wall following behavior since learned steering actions are much weaker but are elicited earlier compared to the initially strong and late reflex reactions. Note that for the robots to learn wall-following behavior is not a *desired* navigational strategy. The learning goal of the robots is to learn avoiding obstacles without triggering the reflex (x_0, reflex avoidance learning). Since the robots do not have any additional "motivation" or "drive" function implemented, their behavior will be equilibrated (and, thus, not change anymore) as soon as they navigate in the environment without triggering the reflex. To avoid reflexes they learn reacting to early stimuli (x_1) but with much weaker steering reactions compared to the initial reflex (x_0) which as a consequence turns into wall following behavior. By reacting earlier the robots can use much less energy compared to late reflex reactions. The strategy of "reflex-avoidance" learning is known from neurons in the Cerebellum (Wolpert et al. 1998; Hofstötter et al. 2002).

Simulation results for the case where we used the weight freezing procedure are shown in Fig. 2c. This way we can, for different weights, show longer trajectories to better assess the robots' behavior. Here we plot selected trajectories for two different environments (3 and 21 obstacles) and for two different robots (antenna ratio 6 and 15). Trajectories for each case are presented in rows where the first trajectory corresponds to the reflexive driving behavior ($\omega_1 = 0$), the second and the third trajectory correspond to a mixture of reflexive and predictive behavior, and the last trajectory corresponds to the predictive driving trajectory when the reflex is finally fully avoided ($\omega_1 = \omega_1^f$). Here we obtained similar driving behavior as in the examples presented above. In general we observed that late, strong, and abrupt reflex reactions turn into early, weak and smooth predictive reactions whereby, as a consequence, a bouncing driving behavior turns into a wall following behavior.

3.2 Characterizing the temporal development

Figure 3 shows the results from one obstacle avoidance experiment in our standard empty square arena. Panels a and b show the development of the reflex (x_0) and predictor (x_1) signals over time (top panels), where the bottom panels show magnifications for the beginning and the end of the learning. As expected, x_0 shrinks substantially during learning, because the reflex signal is better and better avoided. It would finally fully vanish as theory predicts, leading to the stabilization of weights (Porr and Wörgötter 2006), only here—to be able to show how small x_0 signals look like—we have stopped the learning process before this final equilibrium had been reached (see Fig. 2a for a completed process).

The predictor signal in panel b also gets smaller which is due to the fact that at the beginning of learning the predictive antennas are bent all the way until the reflex antennas finally also hit the wall whereas after learning the reflex is avoided

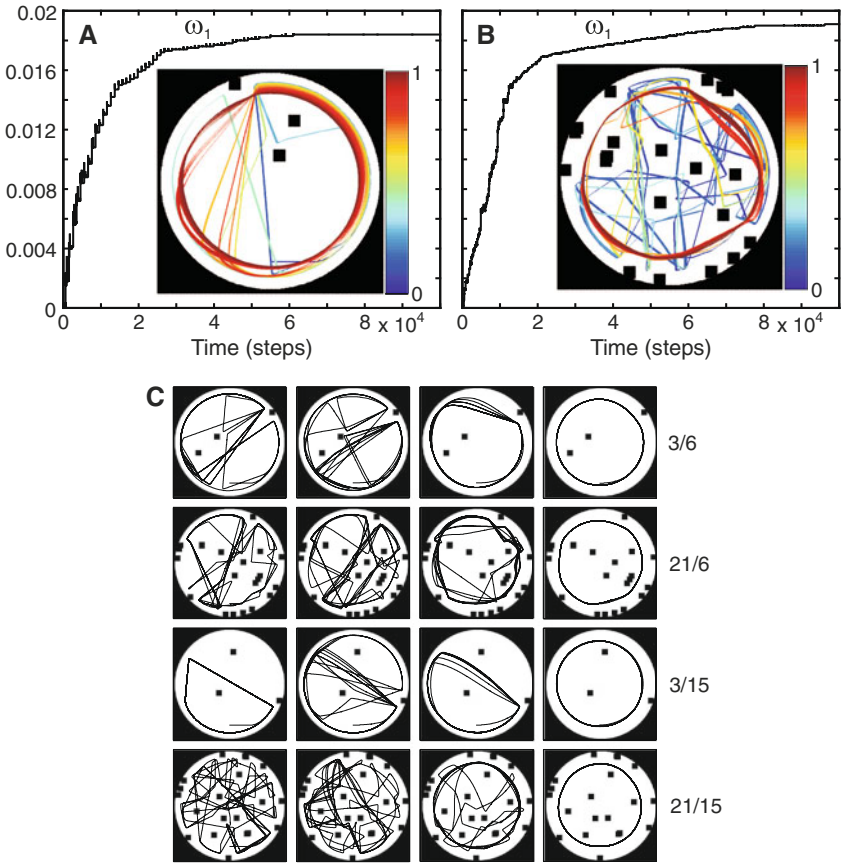

Fig. 2 Driving trajectories from single simulations in a circular environment with obstacles. **a, b** Weight development and corresponding driving trajectories obtained from individual simulations in an environment with 3 (**a**) and 21 obstacles (**b**). Trajectories are color-coded where a zero value corresponds to reflex-driven behavior and one corresponds to predictor-driven behavior. The following parameters were used: antenna ratio $\Lambda_1/\Lambda_0 = 6$, steering gain $g_\alpha = 50$, learning rate was $\mu = 5 \times 10^{-3}$ for the case **a**, and $\mu = 10^{-3}$ for the case **b**. **c** Driving trajectories obtained from individual simulations when using the weight freezing procedure in an environment with 3 (first and third row) and 21 obstacles (second and fourth row). For the two cases shown in the first two rows we used a robot with antennae ratio $\Lambda_1/\Lambda_0 = 6$ whereas for the third and fourth case antenna ratio $\Lambda_1/\Lambda_0 = 15$ was used. The same steering gain $g_\alpha = 50$ was used for all four cases. In the first column we show driving trajectories which correspond to the reflexive driving behavior ($\omega_1 = 0$), the second and the third column correspond to a mixture of reflexive and predictive behavior (approximately $1/3$ and $2/3$ of the learning process, respectively), and in the last column we show trajectories which correspond to the predictive driving behavior when the reflex is finally fully avoided ($\omega_1 = \omega_1^f$)

and the predictive antennas are not so strongly bent anymore. Panel c finally shows the development of the output signal z, which shrinks in amplitude but gets wider over time.

Of special interest is the development of τ during the learning process. Therefore, we carried out a simulation where we analyzed how the time difference τ depends on the synaptic weight ω_1 and the angle at which the robot hits the obstacle. For that we simulated our agent in a square and a circular environment without obstacles where we let the robot drive into a wall with different preset starting angles as shown in Fig. 4 (see insets). We varied the starting angle from 30 to 90° in the square arena and from 40 to 90° in the circular arena. Smaller angles were not possible here. In addition we also varied the weight ω_1 by setting it to a specific value $(0, \Delta\omega_1, \ldots,$ where $\Delta\omega_1 = 10^{-3})$. Results for both environments are shown in Fig. 4 where we plot the time difference τ between inputs x_1 and x_0 against the synaptic weight ω_1. Here each curve shows time differences for one specific preset angle at which the agent drives towards the wall. The obtained results are very similar for both cases where we can see that the time difference increases for all given angles with increasing weights. We can also see that the increase for large angles is less pronounced than that for small angles. In general we observe that curves for small angles are shorter than those for larger angles, which is due to the fact that a less strong weight may suffice to avoid a wall when approaching under a small angle, but will not under a large angle. In a real learning situation this would lead to the fact that at the beginning all angles lead to learning, whereas at the end only large ones will. If we assume that there is

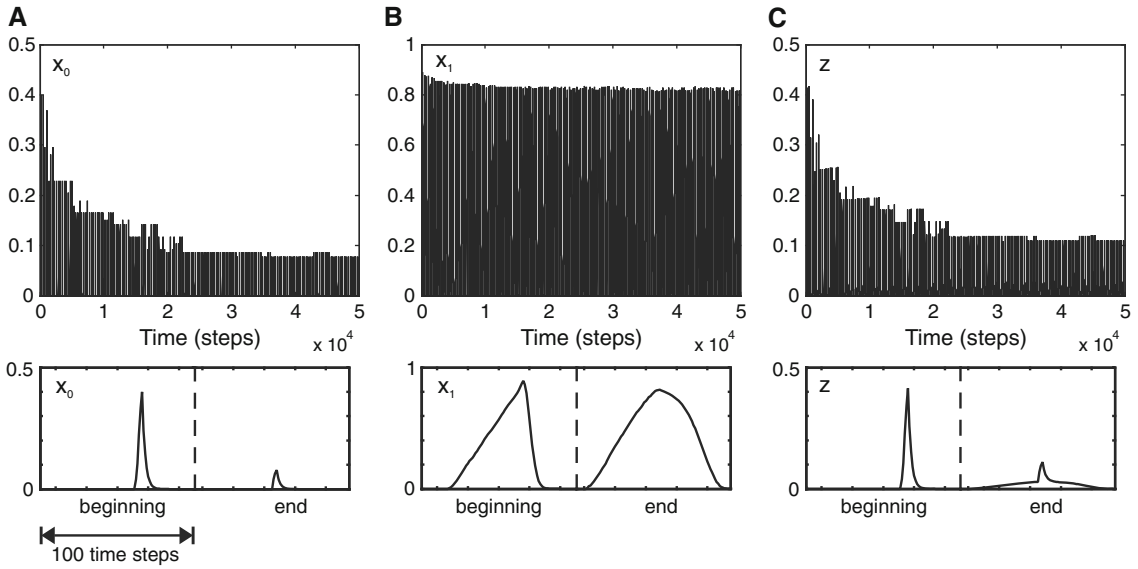

Fig. 3 Results from a simulation in a square arena without obstacles. **a, b** Inputs x_0 and x_1, respectively. **c** Neuronal output z. Bottom panels show signal shapes at the beginning and at the end of the learning. The following parameters were used: antenna ratio $\Lambda_1/\Lambda_0 = 5$, steering gain $g_\alpha = 50$, learning rate $\mu = 0.06$

no prior bias for any approach-angle (hence all angles will occur with equal probability without any learning), then this predicts that as soon as learning takes place an agent will *on average* experience τ values which follow (roughly) the average curve (grey) inside the "brushes" shown in Fig. 4.

To test this prediction, we analyzed the development of τ statistically by simulating nine different robots in four different environments. For the statistical evaluation we carried out 100 experiments for each specific case (in total 36 cases). All experiments were carried out by using the weight freezing procedure. Statistics are presented in Fig. 4c–f where we plot averaged results for all 100 experiments for each case. As discussed above we can see an increase of $\overline{\tau}$ at the beginning and then a decay later on. We can also observe that in general we get larger $\overline{\tau}$ values if we increase the antenna ratio which is obvious because longer antennas produce larger time differences between x_1 and x_0 events. In addition we observe that the time differences at the beginning of the development are smaller for simpler environments and are larger for more complex environments. The reason for this is that in a simple environment we get only those experiences where the robot drives into an obstacle placed close to the wall with a sharp angle or into the opposite wall when it is repelled from the obstacle (for trajectories see Fig. 2c cases 3/6 and 3/15), which leads to small, uniform values of $\overline{\tau}$ in panels c and d. In more complex environments the variety of experiences is much larger due to the more complex paths taken by the robot (see Fig. 2c cases 21/6 and 21/15) and this leads to the larger and more dispersed $\overline{\tau}$ values in panels e and f.

3.3 Analytical closed-loop calculation of the temporal development

3.3.1 Definitions

The analysis of the different signals and their changes makes it now possible to provide an analytical approximation for the temporal weight development. To do so we need to simplify the observed signal structure that we received throughout our experiments in a heuristic way. For the analytics, the reflexive signal x_0 consists of a linear rising and falling phase with identical slopes (see Fig. 5a). In contrast, for convenience, the shape of the predictive input x_1 is described by a concave quadratic function (see Fig. 5b). The definition of both, with their maximum being at $t = 0$, is as follows:

$$x_0(t) = \frac{A_0}{\sigma_0}(t+\sigma_0)\Theta(t+\sigma_0)\Theta(-t) + \frac{A_0}{\sigma_0}(\sigma_0 - t)\Theta(t)\Theta(\sigma_0 - t) \tag{19}$$

$$x_1(t) = A_1\left(1 - \frac{t^2}{\sigma_1^2}\right)\Theta(\sigma_1 + t)\Theta(\sigma_1 - t) \tag{20}$$

with Θ being the Heaviside step function. The parameters $A_{0/1}$ stand for the amplitude and $\sigma_{0/1}$ for the width of the simplified signal shapes $x_{0/1}$.

We will see that this simple definition will lead to a very good approximation of the system's behavior.

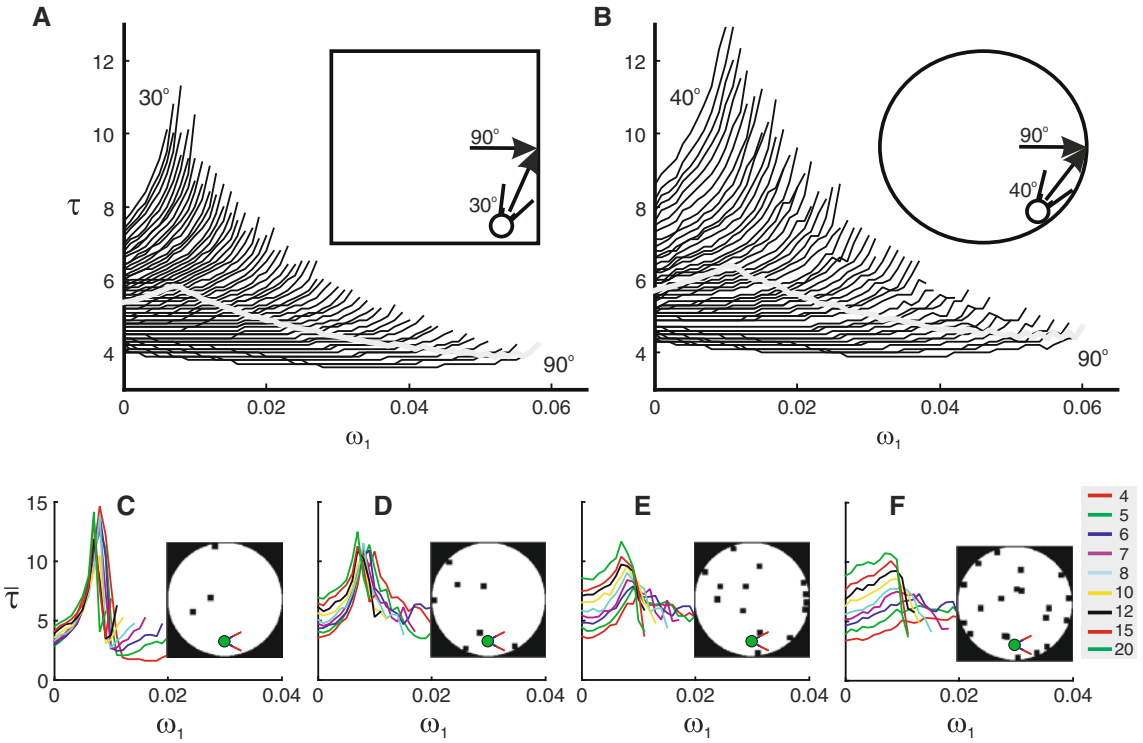

Fig. 4 Time difference τ between far- and near-sensory inputs x_1 and x_0 for a simualted wall avoidance task in a square (**a**) and a circular arena (**b**). τ is plotted against the weight ω_1 where each curve represents a certain angle with which the robot sets off to drive towards the wall of the specific arena as shown in the insets. Grey curve represents the average. The following parameters were used for all cases: antenna ratio $\Lambda_1/\Lambda_0 = 5$, steering gain $g_\alpha = 50$, weight change $\Delta\omega_1 = 10^{-3}$. **c–f** Statistics for time difference τ between inputs x_0 and x_1 obtained from a simulated obstacle avoidance task in a circular environment of different complexity with 3, 7, 14, and 21 obstacles (see insets for examples). Colored curves in each panel show the averaged results plotted against the weight ω_1 obtained from 100 experiments where different color represents results for the different robots defined by the antenna ratio Λ_1/Λ_0. The following parameters were used for all cases: steering gain $g_\alpha = 50$, weight change $\Delta\omega_1 = 10^{-3}$

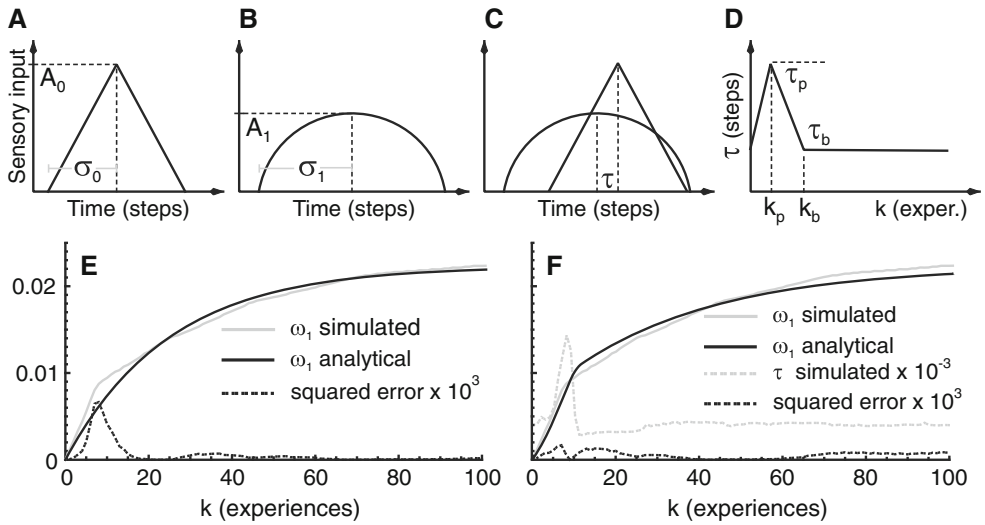

Fig. 5 **a–d** Structural simplifications of the input signals used for analytical calculations. **a** Reflex signal. **b** Predictive signal. **c** The relation between both signals including the temporal difference τ. **d** The development of τ-values over experience k. **e, f** Comparison of simulated and analytical results of weight development when using constant τ (**e**) and variable τ (**f**). In **f** we also show the development of τ (scaled by a factor of 10^{-3}) obtained from the simulation whereas the shape of τ used for analytical calculation is shown in (**d**) and corresponding parameters are given in Table 1. Additionally, we show the squared error scaled by a factor of 10^3

Table 1 Parameters extracted from an experiment

Parameters	a_0	A_1	σ_0	σ_1	ω_f	τ_b	μ_1	τ_p	k_b	k_p	μ_2
Values	0.6	0.85	43.75	5.75	0.0223	4	0.073	12	13	9	0.0523

The first part states the parameters and their values needed for both, constant-τ and variable-τ, approximations, whereas the second and the third part give particular parameters used for the respective, constant-τ and variable-τ, cases. We additionally indicate the learning rate by μ_1 and μ_2, relating them to the equation used to fit the data

3.3.2 Weight change per learning experience

As the weight change per time step is defined by the ICO-learning rule, the weight change per experience k is the integral over a single $x_1 - x_0$ experience using Eq. 4:

$$\omega'(k) := \frac{\mathrm{d}\omega(k)}{\mathrm{d}k} = \int_{-\sigma_1}^{\sigma_1} \mu x_1(t)\dot{x}_0(t-\tau)\,\mathrm{d}t \tag{21}$$

where k is defined as experience and $\dot{x} = \frac{\mathrm{d}x}{\mathrm{d}t}$. Next we include the heuristic equations for $x_0(t)$ and $x_1(t)$ describing the observed temporal development of the signal shapes (i.e., Eqs. 19 and 20) which allows us to integrate Eq. 21:

$$\begin{aligned}
\omega'(k) &= \int_{\tau-\sigma_0}^{\tau} \mu A_1 \left(1 - \frac{(t-\sigma_1)^2}{\sigma_1^2}\right)\frac{A_0}{\sigma_0}\,\mathrm{d}t \\
&\quad - \int_{\tau}^{\tau+\sigma_0} \mu A_1 \left(1 - \frac{(t-\sigma_1)^2}{\sigma_1^2}\right)\frac{A_0}{\sigma_0}\,\mathrm{d}t \\
&= \mu\frac{A_1 A_0}{\sigma_0}\left[t - \frac{1}{3}\frac{(t-\sigma_1)^3}{\sigma_1^2}\right]_{\tau-\sigma_0}^{\tau} \\
&\quad - \mu\frac{A_1 A_0}{\sigma_0}\left[t - \frac{1}{3}\frac{(t-\sigma_1)^3}{\sigma_1^2}\right]_{\tau}^{\tau+\sigma_0} \\
&= \mu\frac{2A_0 A_1 \sigma_0}{\sigma_1^2}\tau.
\end{aligned} \tag{22}$$

In order to avoid unnecessary complex case distinctions we used following constraints on $\tau : |\tau| < \sigma_1 - \sigma_0$ given from the hindsight of the actual τ development we will encounter.

When looking at the data one finds that it is reasonable to keep most variables, especially A_1, σ_0 and σ_1 and some others (see Table 1), constant. Clearly the amplitude of the reflex A_0 should shrink as this leads to weight stabilization. The parametrization of A_0, thus, writes as $A_0 = a_0(1 - \frac{\omega}{\omega_f})$, were we use the final weight value ω_f as a control parameter for the shrinking of reflex amplitude A_0.

After including the parametrization of A_0 into Eq. 22 we get:

$$\omega'(k) = \mu\frac{2a_0 A_1 \sigma_0}{\sigma_1^2}\tau\left(1 - \frac{\omega(k)}{\omega_f}\right) \tag{23}$$

Now the question arises whether a constant or a changing τ would be required for a good system description.

Analytical calculation of the weight development with constant τ: For a constant $\tau = \tau_b$ the solution of the first-order differential equation Eq. 23 using the initial condition $\omega(0) = 0$ is

$$\begin{aligned}
\omega(k) &= \omega_f\left(1 - \exp\left[-\mu\frac{2a_0 A_1 \sigma_0 \tau_b}{\omega_f \sigma_1^2}k\right]\right) \\
&= \omega_f\left(1 - \exp\left[-\mu\lambda k\right]\right) \quad \text{with}
\end{aligned} \tag{24}$$

$$\lambda = \frac{2a_0 A_1 \sigma_0 \tau_b}{\omega_f \sigma_1^2} \tag{25}$$

Analytical calculation of the temporal development including the temporal dependence of τ on k: Different from above, here we start with Eq. 22 and add the parameterizations of A_0 and τ to this equation using:

$$\tau(k) = \begin{cases} \tau_b + (\tau_p - \tau_b)\frac{k}{k_p} & \text{if } 0 \le k \le k_p \\ \tau_p - \frac{\tau_p - \tau_b}{k_p - k_b}k_p + \frac{\tau_p - \tau_b}{k_p - k_b}k & \text{if } k_p < k \le k_b \\ \tau_b\frac{k}{k_f} & \text{if } k > k_b \end{cases} \tag{26}$$

describing a linear increase in the beginning of learning which results in a τ-value of τ_p at experience k_p. It is followed by a linear decrease to the original τ-value of τ_b at experience k_b where it is kept fixed to the end (see Fig. 5d). This gives us three second-order differential equations, which we solve independently. Equations are structurally similar to Eq. 23 and their solutions are shown in Appendix A.2.

Results: We can now extract the necessary parameters from the robot experiments and test to what degree the different situations (constant vs. variable τ) describe the system correctly. Parameters are given in Table 1.

In Fig. 5e and f, we show the real weight change of the conducted experiment and the analytical solution for constant and variable τ. From the experimental data it can be seen that the weight ω_1 grows at two different rates. First, faster till experience $k = 10$ and then slower afterwards, which has been explained in Sects. 3.1 and 3.2, and was due to an initial increase in τ and then a decrease in τ to initial values. Consequently, the constant-τ solution (E) only captures the overall weight development, however, cannot reproduce the

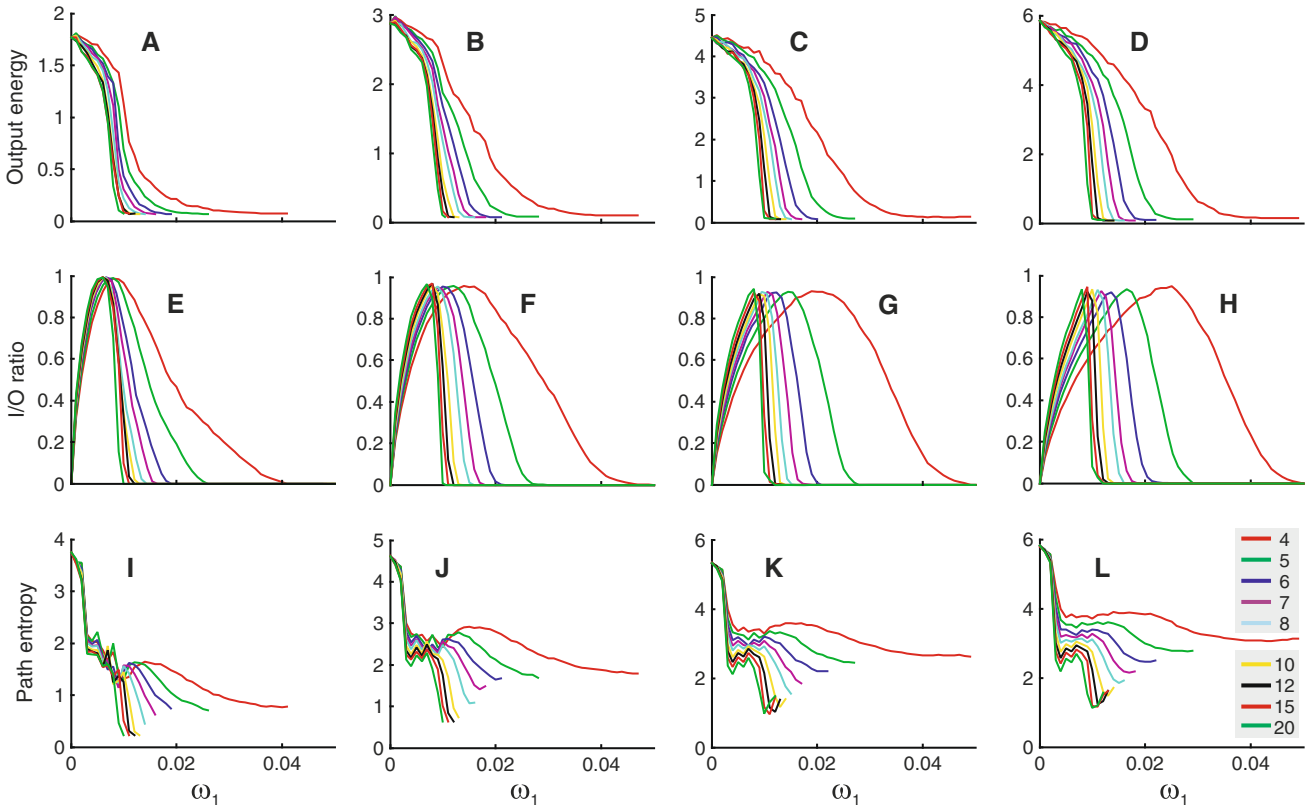

Fig. 6 Statistics for different measures obtained from a simulated obstacle avoidance task in a circular environment of different complexity with 3, 7, 14, and 21 obstacles (from left to right). **a–d** output energy E_z, **e–h** input/output ratio H_z, **i–l** path entropy H_p. Colored curves in each panel show averaged results plotted against weight ω_1 for a specific measure obtained from 100 experiments where different colors represent results for a specific robot defined by the antenna ratio Λ_1/Λ_0 (see **l**). The following parameters were used for all cases: steering gain $g_\alpha = 50$, weight change $\Delta\omega_1 = 10^{-3}$

change in weight growth around experience $k = 10$. The fit for variable τ is substantially better (F) and the different weight growths are much better reproduced. The remaining error arises from the required simplifications used to arrive at the analytical solution.

3.4 System measures

We analyzed the development of the system measures during learning by testing nine different robots in four different environments. For statistical evaluation we carried out 100 experiments for each specific case (in total 36 cases). All experiments were performed by using the weight freezing procedure. Statistics are presented in Fig. 6 where we plot averaged results for all 100 experiments for each measure.

Results for the energy development are shown in panels a–d. We can see that energy is gradually decreasing as sharp reflexive steering reactions turn into smooth predictive reactions and less energy is needed for shallow turns compared to sharp turns. We also observe that the energy consumption at the end of the development is similar across all robots and all environments.

Development of the input/output ratio is presented in panels e–h. As expected, we observe that at the beginning of the development the ratio equals zero since only the reflex contributes to the neuronal output and then increases as the synaptic weight of the predictive input grows. The ratio reaches a maximum when reflex and predictor contribute equally to the output. Thereafter the ratio decreases back to zero since with development we get less and less reflexive reactions and at the end of the development only predictive reactions are elicited. Different robots reach their maximum ratio at different weights. Similarly, a different steepness of the decay after the maximum is found. Results suggest that in given environments, robots with longer antennas are quicker learners compared to robots with shorter antennas. We can conclude that the input/output ratio measure can be used to evaluate the success and speed of learning of a specific agent in a given environment.

Results for the path entropy are presented in panels i–l where in most cases we see a rapid decay at the beginning

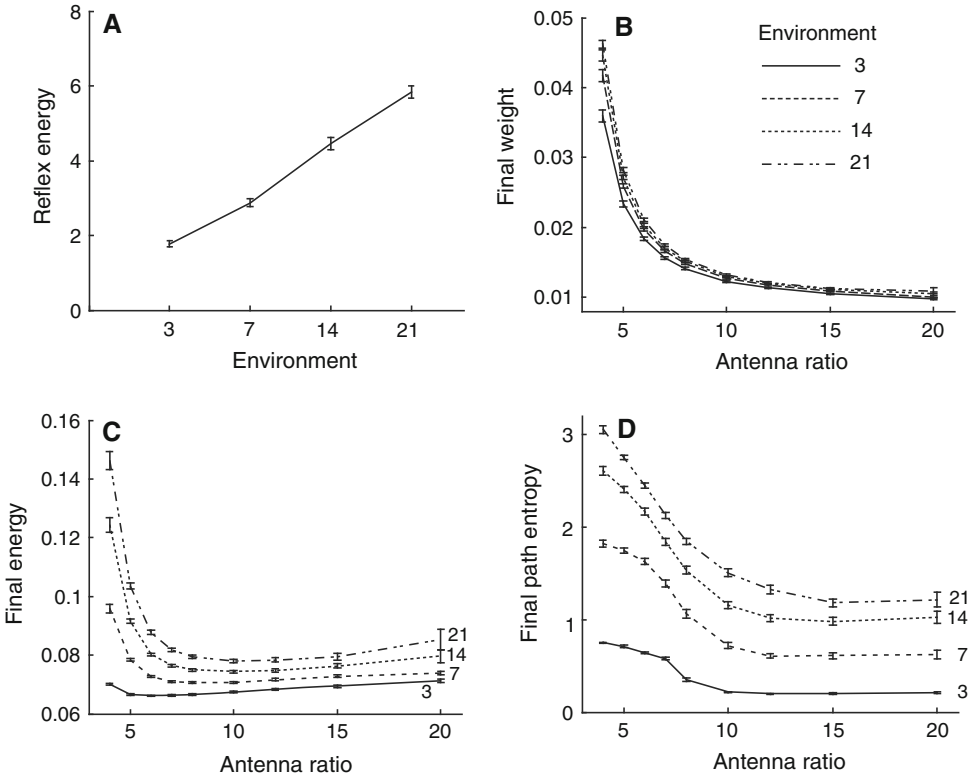

Fig. 7 Results from a simulated obstacle avoidance task. **a** Average reflex energy plotted against environment complexity defined by the number of obstacles. **b** final weight, **c** final energy, and **d** final path entropy plotted against the antenna ratio Λ_1/Λ_0. The error-bars represent confidence intervals (95%) of the mean

of the development followed by a small increase and a slow decay at the end. This tells us that the reflexive behavior at the very beginning of the development produces relatively complex paths whereas, when the predictor takes over, the driving trajectories become simpler, which leads to a decrease in path entropy. Usually there exists a transition phase during the development where the robot changes its driving trajectory in order to avoid obstacles producing more complex paths for some time and this is seen as a small increase in the path entropy curve. After that the path entropy slowly decreases since predictive reactions produce rather stereotypical and simple circular paths (see also Fig. 2c). We can also observe that robots with shorter antennas produce more complex paths compared to robots with longer antennas.

Summarized results for all robots and all environments are presented in Fig. 7. In panel a, we compare energy consumption of the reflexive behavior ($\omega_1 = 0$) where we see that energy consumption increases significantly with increase of environmental complexity. This suggests that by measuring reflex energy we can judge complexity of the environment, and that learning is not necessary for such an evaluation. In panel b, we compare the final weights (ω_1^f). Results demonstrate that there is no statistically significant difference between different environments except for the robots with short antennas (antenna ratio 4–8). Results for the final energy ($E_z(\omega_1^f)$) are compared in panel c. As expected, robots consume less energy in simple environments and more energy in more complex environments, although those differences are mush less pronounced compared to the pure reflex energy (panel a). We can also observe that robots with very long antennas are energetically slightly worse on average than robots with shorter antennas. In panel d, we compare the final path entropies ($H_p(\omega_1^f)$). Here we obtained similar results to those of the final energy where we see that robots produce more complex paths as the environmental complexity increases. Results also demonstrate that in general robots with shorter antennas produce more complex paths compared to robots with long antennas.

3.5 On optimal robots

In the following we are concerned whether there exists an optimal robot for a given environment. We compare the robots' performance with respect to different measures (Fig. 8). In panel a, we compare side by side energy consumption after learning, i.e., when the reflex x_0 is not triggered anymore. Here we can see that the minimal energy consumption shifts from robots with shorter antennas to robots with longer antennas as the environment's complexity increases, but differences (except for the shortest antennas) are small. As we can see in panel b the most complex paths are produced by shortest antennas ($\Lambda_1/\Lambda_0 = 4$) in all four environments. Concerning the speed of learning (for the speed measure see Eq. 17) we observe that robots with long antennas learn much quicker than robots with shorter antennas (panel c). Also the drop in performance, when getting into more complex

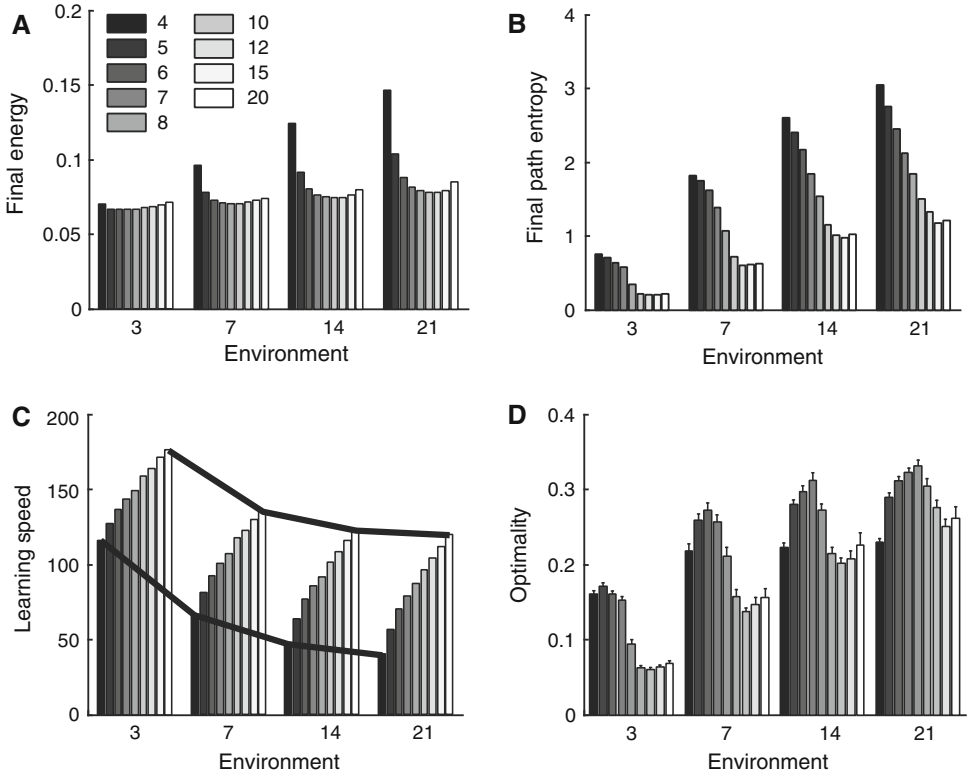

Fig. 8 Comparison of different robots in specific environments of different complexity obtained from a simulated obstacle avoidance task. Antenna ratios are given by the gray shading (see **a**). **a** energy, **b** path entropy, **c** learning speed, and **d** optimality. Average results are shown in each panel obtained from 100 experiments. In **d** error bars represent confidence intervals (95%) of mean

environments, is less for long antennas as compared to short ones (see lines in panel c). We remind the reader that speed of learning is given by the equilibrium point (peak of the input/output ratio, see Fig 6) where the reflex signal x_0 and the predictor signal x_1 contribute on average equally to the output.

In general one should think that "a good robot" would be one that produces, after learning, complex paths and learns those quickly. As we can see, however, there is a trade-off between these two constraints, the speed of learning and the path complexity for a given environment. As a consequence, by using the normalized product of these two quantities (see Eq. 18), panel d shows that there is an optimal robot existing for any given environment. For instance, the optimal robot for the simplest environment is the robot with antenna ratio five whereas in the most complex environment the robot with antenna ratio eight is the best. Based on the obtained results we can conclude that different robots adapt differently to a specific environment due to their different physical properties. Note, we did not consider using also the final energy for defining "optimality" because it does not alter the general picture. For most robots the final energy does not vary much (panel a) and, where its high (short antennas, complex worlds) and, thus, not optimal, including it in the measure would only emphasize the effect that robots with short antennas are best in simple worlds and vice versa (as stated above).

4 Discussion

In this study we have started to address the difficult question how to quantify continuous learning processes in behaving systems that change by differential hebbian plasticity. The central problem lies here in the closed loop situation, which leads—even in very simple linear cases—to an intricate interplay between behavior and plasticity.

In the first part of this study we have concentrated on the inputs and we could show how τ develops over time for different robots and in different worlds. The peaked characteristics of the development of τ during learning (Fig. 4) is a nice example of the mutual interaction between behavior and plasticity. Touching a wall with a shallow angle just does not occur anymore after some learning and the system finds itself in the domain of large approach angles where τ shrinks again, contrary to our naive first intuition, which had argued for a continuous growth of τ. This also leads to a biphasic weight development and it was possible to use the measured τ-characteristics, together with some assumptions on the amplitude change of x_0 and x_1, to quite accurately calculate such a weight development in an analytical way.

In the second part of this article, we have started to quantify the behavior of our little beetles by considering their output z. We have defined measures for energy, input/output ratio and entropy focusing on the question whether there is an optimal robot for a given environment.

4.1 The system identification problem in adaptive closed-loop systems

Several methods are known from the literature to address the model identification issue in a broader context. For example one can use a [Non-linear] Auto-Regressive Moving Average approach with or without exogeneous inputs ([N]ARMA[X], Box et al. (1994)) to arrive at a general model of behaving robot systems (Iglesias et al. 2008; Kyriacou et al. 2008), but these models contain many parameters for fitting and parameters do not have any direct physical meaning. Our attempts stop short of a complete model identification approach, which does not seem to be required for our system. Instead, here we could use a rather limited model with quite a reductionist set of equations (see Sect. 3.3), which was to some degree unexpected given the complexity of the closed loop behavior of our robots (Fig. 2).

We observed that signal shapes and timings change in a difficult way influencing the learning. As a consequence, it is not easy to find an appropriate description and the right measures for capturing such non-stationary situations. Fig. 1a shows the structure of our closed loops system and this diagram has been used in earlier studies for convergence analyzes (Porr and Wörgötter 2003a,b, 2006; Kulvicius et al. 2007). From this diagram it becomes clear that τ, z as well as $x_{0,1}$ are the relevant variables in our system. While learning is defined by the relation between inputs $x_{0,1}$ and, hence, τ; behavior is defined by output z.

Interestingly one finds in the first place that learning acts "equalizing". Robots with different initial (reflex) energy (Fig. 7a) become very similar after learning (Fig 7b, note the different scales in panel a and b). This finding can be understood from some older studies on closed-loop differential hebbian (ISO, ICO) learning. Fig. 1a shows that these systems will learn avoiding the reflex and that learning will stop once this goal has "just" been reached leading to an asymptotic equilibrium situation (Porr and Wörgötter 2003b). Furthermore, the systems investigated here are linear, hence all of them will in the end essentially require the same total effort for performing the avoidance reaction. These two facts explain why their energy is very similar in the end. The fact that robots are different, however, does surface when looking at the paths they choose after learning. Robots with long predictive antennas can never make sharp turns anymore and their paths are dominated by performing the same shallow turns again and again leading to little path variability and hence to a small final path entropy (Fig. 7d). On the other hand, these same long-antenna robots learn their task much faster than their short-antenna fellows: for the former, the equilibrium point between reflex and predictor (peak in the input/output ratio) is reached faster than for the latter (Fig. 6e–h).

This leads to a trade-off and by using the normalized product of learning speed times path entropy we found that for different environments different robots are optimal (Fig. 8d). Clearly, this type of optimality is to some degree just in the eyes of the beholder and one might choose to weigh the two aspects (learning speed and path complexity) differently by which other robots would be valued more than those currently called "optimal". Nonetheless, also with a different weighing one will observe that some robots would be better than others in the different worlds.

4.2 Information flow in adaptive closed-loop systems

In general the second part of the study relates to work focusing on information flow in closed-loop systems. There have been a few contributions to this topic. Tishby et al. (1999) introduced an Information-Bottleneck (IB) framework that finds concise representations for a system's input that are as relevant as possible for its output, i.e., concise description that preserves the relevant essence of the data. The relevant information in one signal with respect to the other is defined as the mutual information that the signal provides about the other. Although, the Information-Bottleneck framework was successfully applied in various applications, like data clustering (Slonim and Tishby 2000; Slonim et al. 2001), feature selection (Slonim and Tishby 2001), POMDPs (Poupart and Boutilier 2002), it conceptually differs from our study, since we are interested in the dynamics of sensory-motor systems during the learning process.

In the study of Klyubin et al. (2004, 2005, 2007, 2008) the authors used a Bayesian network to model perception-action loops. In their approach a perception-action loop is interpreted in terms of a communication channel-like model. They show that maximization of information flow can evolve into a meaningful sensorimotor structure (Klyubin et al. 2004, 2007). In Klyubin et al. (2005, 2008) the authors present a universal agent-centric measure, called "empowerment", which is defined as the information-theoretic capacity of an agent's actuation channel (the maximum mutual information for the channel over all possible distributions of the transmitted signal). The empowerment is zero when the agent has no control over its sensory input, and it is higher when the agent can control what it is sensing. In these studies it could be demonstrated that maximization of empowerment can be used for control tasks (such as pole balancing) as well as for an evolution of the sensorimotor system or even to construct contexts which can assign semantic "meaning" to the robot's actions (Klyubin et al. 2005, 2008). Similar to the work of Klyubin et al. (2004, 2005, 2007, 2008) in the study of Prokopenko et al. (2006) the authors used two measures called generalized *correlation entropy* and generalized *excess entropy* to alter the locomotion of a simulated modular robotic system (snake-like robot) by an evolution

process. The mentioned studies differ from our approach, since in these works information measures had been used to drive a sensorimotor adaptation on a relatively large time scales (simulating evolution by using genetic algorithms) whereas in our approach we use information measures to investigate the behavior of closed-loop system during on-line learning on relatively short time scales.

Lungarella et al. (2005) have shown that coordinated and coupled sensorimotor activity decreases the entropy and increases the mutual information within specific regions of the sensory space. In contrast to our study they analyzed information flow only on the sensory inputs whereas we consider inputs as well as outputs (input/output ratio, path entropy, energy). Also, different from our attempt, these authors analyzed the system in a reflex-based closed-loop scenario where no learning had been applied. Ay et al. (2008) and Der et al. (2008) used a predictive information measure (PI, mutual information between past and future sensor values) to evaluate behavioral complexity of agents and to use PI as an objective function for the agents' adaptation, however, similar to Lungarella et al. (2005), only looking at the input space.

An earlier study of Lungarella and Sporns (2006) has demonstrated that learning can affect information flow (transfer entropy) of the sensorimotor network of a behaving agent. In this study transfer entropy was used to analyze the causal structure of the loop, i.e., causal effects of sensory inputs on motor states and vice versa, whereas in our study we use system measures to analyze the system dynamics during learning with respect to the speed of learning and behavioral performance of an agent. Also differently from our approach Lungarella and Sporns (2006) used incremental reward based learning, which belongs to a different class of learning algorithms.

Our approach more closely relates to the study of Porr et al. (2006). They define the information value (called predictive information) only by the weights of the ISO learning rule (Porr and Wörgötter 2003b), where, different from our approach (see Eq. 13), sensory inputs and outputs are not included in this measure. In Porr et al. (2006) weights reflect the predictive power of their corresponding inputs: the larger the weights the higher the predictive information. Essentially this measure shows which inputs are more predictive in relation to the signal at x_0, whereas in our approach the measures of input/output ratio, path entropy and energy reflect the general behavior of the system, for example the contribution of reflex and predictor to the system's output.

Our measures, similar to those in Porr et al. (2006), are developed within the framework of predictive correlation based learning (specifically using the ICO-rule here). Nevertheless, these measures can be also used for other learning rules as long as the reflex and the predictive inputs can be identified. The previously discussed empowerment measure (Klyubin et al. 2005, 2008) is independent of the specific learning rule and can treat the system as a black box. As mentioned before empowerment is defined as channel capacity, which is the maximum mutual information over all possible distributions of the transmitted signal. This quantity is difficult to calculate and may require using a "detachable" world model that allows exact repetitions of certain behaviors in a particular situation (Klyubin et al. 2008). This means that it is not straightforward to use empowerment for analyzing on-line behavioral systems.

Here we used input/output ratio in order to see how the influence of predictor and reflex on the system output changes over time. This measure could be also used to investigate the dynamics of systems with many different inputs (also without defining predictive and reflexive inputs and independent on system setup) in order to analyze the contribution of different inputs to the performance during learning. For this, one would need to calculate input/output ratio of each sensory input independently. Note that the output-signal based measures used by us, for example our path entropy measure, can also be applied independently of the learning rule and the actual behavioral pattern. They could, thus, be used also in other systems, quantifying their (possibly entirely different) behavior and its variability. The proposed system measures could be also used for an analysis of the system dynamics with multiple subtasks, i.e., obstacle avoidance (negative tropism) and food retrieval (positive tropism), or multiple agent systems, for investigation of cooperative behavior.

In summary, in the current study we have analyzed closed loop behavioral systems which change by differential Hebbian learning. We were surprised to find that even these very simple systems are already too complex to fully deduct the system's behavior from the initial setup of system and world. Only together with some information on the general structure of the development of their descriptive parameters, analytical solutions can be still found for their temporal development. By using energy, input/output ratio and entropy measures and investigating their development during learning we have shown that within well-specified scenarios there are indeed agents which are optimal with respect to their structure and adaptive properties. As a consequence, this study may help leading to better understanding of the complex dynamics of learning&behaving systems. The fact that with learning optimal agents will exist (probably under any measure of optimality!) may make it necessary to reconsider evolutionary approaches as cited above (Klyubin et al. 2007, 2008; Prokopenko et al. 2006) in light of a different fitness function, which also takes the learning into account (Baldwin Effect, Baldwin 1896; Hinton and Nowlan 1987).

Acknowledgements This research was supported by the European funded PACO-PLUS project as well as by BMBF (Federal Ministry of Education and Research), BCCN (Bernstein Center for Computational Neuroscience)—Göttingen project W3 and BFNT project 3a

Appendix

A.1 Contrast measure

We obtained values for the color-coding in Fig. 2 as follows:

$$\begin{aligned} Z_0(k) &= \sum_{t=k}^{k+c_w-1} |\omega_0(t) \cdot x_0(t)|, \\ Z_1(k) &= \sum_{t=k}^{k+c_w-1} |\omega_1(t) \cdot x_1(t)|, \\ R(k) &= \frac{Z_1(k) - Z_0(k)}{Z_1(k) + Z_0(k)}, \end{aligned} \tag{27}$$

where $k = 0 \ldots N - w_r$, $N = 10^5$ is the length of the input sequence, and $c_w = 5 \times 10^3$ is the size of the sliding time window. Note that we normalized values of R between zero and one.

A.2 Analytical calculation of the temporal development including the temporal dependence of τ on k

From Eqs. 22 and 26 we derive three second-order differential equations, which we solve independently:

if $0 \le k \le k_p$

$$\omega'(k) = \mu \frac{2a_0 A_1 \sigma_0}{\sigma_1^2} \left(\tau_b + (\tau_p - \tau_b)\frac{k}{k_p}\right) \times \left(1 - \frac{\omega(k)}{\omega_f}\right), \tag{28}$$

if $k_p < k \le k_b$

$$\omega'(k) = \mu \frac{2a_0 A_1 \sigma_0}{\sigma_1^2} \left(\tau_p - \frac{\tau_p - \tau_b}{k_p - k_b} k_p + \frac{\tau_p - \tau_b}{k_p - k_b} k\right) \times \left(1 - \frac{\omega(k)}{\omega_f}\right), \tag{29}$$

if $k > k_b$

$$\omega'(k) = \mu \frac{2a_0 A_1 \sigma_0}{\sigma_1^2} \left(\tau_b \frac{k}{k_b}\right) \left(1 - \frac{\omega(k)}{\omega_f}\right). \tag{30}$$

The solution of these differential equations are as follows:

if $k \le k_p$

$$\omega(k) = \omega_f \left(1 - \exp\left[-\mu\tilde{\lambda}\frac{2k_p\tau_b + k(\tau_p - \tau_b)}{2k_p} k\right]\right), \tag{31}$$

if $k_p < k \le k_b$

$$\omega(k) = \omega_f - \omega_f \exp\left[-\mu\tilde{\lambda}\frac{(k^2 - 2k_p k + k_p k_b)\tau_b}{2(k_b - k_p)}\right] \times \exp\left[\mu\tilde{\lambda}\frac{(k^2 - 2kk_b + k_p k_b)\tau_p}{2(k_b - k_p)}\right], \tag{32}$$

if $k > k_b$

$$\omega(k) = \omega_f \left(1 - \exp\left[-\mu\tilde{\lambda}\frac{2k\tau_b + k_b(\tau_b - \tau_b)}{2}\right]\right), \tag{33}$$

where $\tilde{\lambda} = \lambda/\tau$ (see Eq. 25).

References

Ashby WR (1956) An introduction to cybernetics. Chapmann and Hall Ltd., London

Ay N, Bertschinger N, Der R, Güttler F, Olbrich E (2008) Predictive information and explorative behavior of autonomous robots. Eur Phys J B 63:329–339

Baldwin JM (1896) A new factor in evolution. Am Nat 30:441–451

Bi GQ, Poo MM (1998) Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. J. Neurosci. 18:10464–10472

Box G, Jenkins GM, Reinsel GC (1994) Time series analysis: forecasting and control. Prentice-Hall, Englewood Cliffs, NJ

Braitenberg V (1986) Vehicles: experiments in synthetic psychology. The MIT Press, Cambridge, MA

Der R, Güttler F, Ay N (2008) Predictive information and emergent cooperativity in a chain of mobile robots. In: Bullock S, Noble J, Watson R, Bedau MA, (eds) Artificial life XI: proceedings of the eleventh international conference on the simulation and synthesis of living systems. MIT Press, Cambridge, MA, pp 166–172

Hebb DO (1949) The organization of behavior. Wiley, New York

Hinton GE, Nowlan SJ (1987) How learning guides evolution. Complex Syst 1:495–502

Hofstötter C, Mintz M, Verschure PF (2002) The cerebellum in action: a simulation and robotics study. Eur J Neurosci 16:1361–1376

Iglesias R, Nehmzow U, Billings SA (2008) Model identification and model analysis in robot training. Robot Auton Syst 56:1061–1067

Klopf AH (1988) A neuronal model of classical conditioning. Psychobiology 16(2):85–123

Klyubin AS, Polani D, Nehaniv CL (2004) Organization of the information flow in the perception-action loop of evolved agents. In: 2004 NASA/DoD conference on evolvable hardware. IEEE Computer Society, pp 177–180

Klyubin AS, Polani D, Nehaniv CL (2005) Empowerment: a universal agent-centric measure of control. In: IEEE congress on evolutionary computation (CEC 2005), pp 128–135

Klyubin AS, Polani D, Nehaniv CL (2007) Representations of space and time in the maximization of information flow in the perception-action loop. Neural Comput 19:2387–2432

Klyubin AS, Polani D, Nehaniv CL (2008) Keep your options open: an information-based driving principle for sensorimotor systems. PLoS ONE 3:e4018

Kosco B (1986) Differential Hebbian learning. In: Denker JS (ed) Neural networks for computing: AIP conference proceedings, vol 151. American Institute of Physics, New York

Kulvicius T, Porr B, Wörgötter F (2007) Chained learning architectures in a simple closed-loop behavioural context. Biol Cybern 97:363–378

Kyriacou T, Nehmzow U, Iglesias R, Billings SA (2008) Accurate robot simulation through system identification. Robot Auton Syst 56:1082–1093

Lungarella M, Pegors T, Bulwinkle D, Sporns O (2005) Methods for quantifying the informational structure of sensory and motor data. Neuroinformatics 3:243–262

Lungarella M, Sporns O (2006) Mapping information flow in sensorimotor networks. PLoS Comput Biol 2:e144

Markram H, Lübke J, Frotscher M, Sakmann B (1997) Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. Science 275:213–215

Porr B, Wörgötter F (2003a) Isotropic sequence order learning. Neural Comput 15:831–864

Porr B, Wörgötter F (2003b) Isotropic-sequence-order learning in a closed-loop behavioural system. Philos Transact A Math Phys Eng Sci 361:2225–2244

Porr B, Wörgötter F (2006) Strongly improved stability and faster convergence of temporal sequence learning by using input correlations only. Neural Comput 18:1380–1412

Porr B, Egerton A, Wörgötter F (2006) Towards closed loop information: Predictive information. Constr Found 1(2):83–90

Poupart P, Boutilier C (2002) Value-directed compression of POMDPs. In: Becker STS, Obermayer K (eds) Advances in neural information processing systems, vol 15. pp 1547–1554

Prokopenko M, Gerasimov V, Tanev I (2006) Evolving spatiotemporal coordination in a modular robotic system. In: SAB 2006. pp 558–569

Saudargiene A, Porr B, Wörgötter F (2004) How the shape of pre- and postsynaptic signals can influence STDP: a biophysical model. Neural Comput 16:595–625

Saudargiene A, Porr B, Wörgötter F (2005) Synaptic modifications depend on synapse location and activity: a biophysical model of STDP. BioSystems 79:3–10

Shannon CE (1948) A mathematical theory of communication. Bell Syst Tech J 27:379–423

Slonim N, Tishby N (2000) Document clustering using word clusters via the information bottleneck method. In: Proceedings of the 23rd annual international acm-sigir conference on research and development in information retrieval

Slonim N, Tishby N (2001) The power of word clustering for text classification. In: Proceedings of the 23rd European colloquium on information retrieval research

Slonim N, Somerville R, Tishby N, Lahav O (2001) Objective classification of galaxy spectra using the information bottleneck method. Mon Notes R Astron Soc 323:270–284

Sutton RS, Barto AG (1981) Toward a modern theory of adaptive networks: expectation and prediction. Psychol Rev 88:135–170

Tishby N, Pereira FC, Bialek W (1999) The information bottleneck method. In: Proceedings of the 37-th annual allerton conference on communication, control and computing. pp 368–377

Touchette H, Lloyd S (2000) Information-theoretic approach to the study of control systems. Physica A 331:140–172

Wolpert DM, Miall RC, Kawato M (1998) Internal models in the cerebellum. Trends Cogn Sci 2:338–347