Strongly improved stability and faster convergence of temporal sequence learning by utilising input correlations only

Bernd Porr[1], and Florentin Wörgötter[2,3]

[1]Department of Electronics & Electrical Engineering, University of Glasgow, Glasgow, GT12 8LT, Scotland, B.Porr@elec.gla.ac.uk,
[2]Department of Psychology, University of Stirling, Stirling FK9 4LA, Scotland
[3]Bernstein Center of Computational Neuroscience, University Göttingen, Germany, worgott@chaos.gwdg.de

Abstract

Currently all important, low-level, unsupervised network learning algorithms follow the paradigm of Hebb, where input- and output activity are correlated to change the connection strength of a synapse. However, as a consequence, classical Hebbian learning always carries a potentially destabilising autocorrelation term which is due to the fact that every input is in a weighted form reflected in the neuron's output. This self-correlation can lead to positive feedback, where increasing weights will increase the output and vice versa, which may result in divergence. This can be avoided by different strategies like weight normalisation or weight saturation which, however, can cause different problems. Consequently, in most cases, high learning rates cannot be used for Hebbian learning leading to relatively slow convergence. Here we introduce a novel correlation based learning rule which is related to our ISO-learning rule (Porr and Wörgötter, 2003a), but replaces the derivative of the *output* in the learning rule with the derivative of the reflex *input*. Hence the new rule utilises input correlations only, effectively implementing strict heterosynaptic learning. This looks like a minor modification, but leads to dramatically improved properties. Elimination of the output from the learning rule removes the unwanted, destabilising autocorrelation term allowing us to use high learning rates. As a consequence we can mathematically show that the theoretical optimum of *one-shot learning* can be reached under ideal conditions with the new rule. This result is then tested against four different experimental setups and we will show that in all of them very few (and sometimes only one) learning experiences are needed to achieve the learning goal. As a consequence the new learning rule is up to 100 times faster and in general more stable than ISO-learning.

1 Introduction

Probably all existing correlation based learning algorithms rely currently on Donald Hebb's famous paradigm (Hebb, 1949), that connections between network units should be strengthened if the two connected units are simultaneously active (Oja, 1982; Kohonen, 1988; Linsker, 1988). The Hebb-rule can be formalised as

$$\Delta\rho_j = \mu u_j f(v) \tag{1}$$

where ρ_j is the connection strength and the output is calculated from the weighted sum $v = \sum_j \rho_j u_j$. The factor μ is called the learning rate. The linear operator f is just the identity operator $f = v$ for classical Hebbian learning (Hebb, 1949) and it is the derivative $f = v'$ for differential Hebbian learning (Kosco, 1986).

In spite of their success, Hebbian type learning algorithms can be unstable because of the existing *autocorrelation* term in the learning rule. This can be seen if we replace v in Eq. 1 by the weighted sum. Apart from the cross correlation terms we get $\Delta\rho_j \propto \mu\rho_j u_j f(u_j)$. Hebbian learning is only stable if this autocorrelation term is zero, or can be compensated for by means of additional measures taken (Oja, 1982; Bienenstock et al., 1982; Miller, 1996b; Porr and Wörgötter, 2003a). In the general case, however, this term leads to an exponentially growing instability and to network divergence.

Hebb rules have been employed in a wide variety of unsupervised learning tasks and during the last years we had focused on the specific problem of temporal sequence learning (Porr and Wörgötter, 2001; Porr and Wörgötter, 2003a). In this case two (or more) signals exist which are correlated to each other, but with certain delays between them. In real life this can happen, for example, when heat radiation precedes a pain signal when touching a hot surface or when the smell of a prey arrives before the predator is close enough to see it hiding in the shrubs. Such situations occur often during the lifetime of a creature and in these cases it is advantageous to learn reacting to the earlier stimulus, not having to wait for the later signal. Temporal sequence learning enables the animal to react to the earlier stimulus. Thus, the animal learns an *anticipatory* action to avoid the late unwanted stimulus. From a more theoretical perspective such situations are related to classical and/or instrumental conditioning and in early studies correlation-based, stimulus-substitution models have been used to address the problem of how to learn such sequences (Sutton and Barto, 1981). Soon these methods were, however, superseded by reinforcement learning algorithms (Sutton, 1988; Watkins, 1989; Watkins and Dayan, 1992) partly because those algorithms had favourable mathematical properties (Dayan and Sejnowski, 1994) and partly because convergent learning could be achieved in behaving systems (Kaelbling et al., 1996). Relations to biophysics, however, seem to exist more to the dopaminergic reward-based learning system (Schultz et al., 1997) than to (differential) Hebbian learning through long term potentiation (LTP) at glutamatergic synapses (Malenka and Nicoll, 1999); for a review see (Wörgötter and Porr, 2005). Therefore, in a series of recent papers we have tried to show that it is possible to solve reinforcement learning tasks by correlation based (Hebbian) rules realising that such tasks can often be embedded into the framework of sequence learning which allows for a Hebbian formalism (Porr and Wörgötter, 2003a,b). However, we had to discover that the Hebbian learning rule, which we had designed to address problems of temporal sequence learning produces exactly the same autocorrelation instability which often prevented convergence.

To solve this problem, in this study we present a novel, heterosynaptic learning rule which allows implementation of fast and stable learning. This learning rule has been derived from ISO learning (Porr and Wörgötter, 2003a), which belongs to the class of differential Hebbian learning rules (Kosco, 1986). ISO learning, however, suffers from the problem discussed above. It, too, contains the destabilising autocorrelation term and only for the limiting case of $\mu \rightarrow 0$ we have been able to prove that this term vanishes (Porr and Wörgötter, 2003a), but only when using a set of orthogonal input filters.

However, a very simple alteration of ISO learning eliminates its autocorrelation term completely: *If we correlate only inputs with each other this term does not exist any more*. More specifically we define an error signal at one of the inputs and correlate this error signal with the other inputs. Consequently, our rule can be used in applications where such an error signal can be identified which is the case, in particular, in closed loop feedback control.

We will in this study first derive the convergence properties of input correlation (*"ICO"*) learning, showing that one-shot learning is the theoretical limit for the learning rate. As an additional advantage it will become clear that input filtering does not rely on orthogonal filters at the different inputs. Any input characteristic will suffice as long as the whole system contains an (additional) low-pass filter component. This however, can also come from the transfer function of the environment in which the learning system is embedded. The advantage of now being able to choose almost arbitrary input filters will now for the first time also allow approximating far more complex (e.g., non-linear) output characteristics than was possible with ISO-learning.

In the second part of this study we will compare ICO-learning with its equivalent differential Hebbian learning rule, namely the ISO-learning rule. This comparison, performed on a simulated and real benchmark test, will demonstrate that input correlation learning is indeed much faster and more stable than the older ISO-learning. Finally, we will present a set of experiments from different application domains which show that one-shot learning can be approached when using the ICO-rule. These applications have been specifically chosen to raise confidence that ICO-learning can be applied in a variety of different situations.

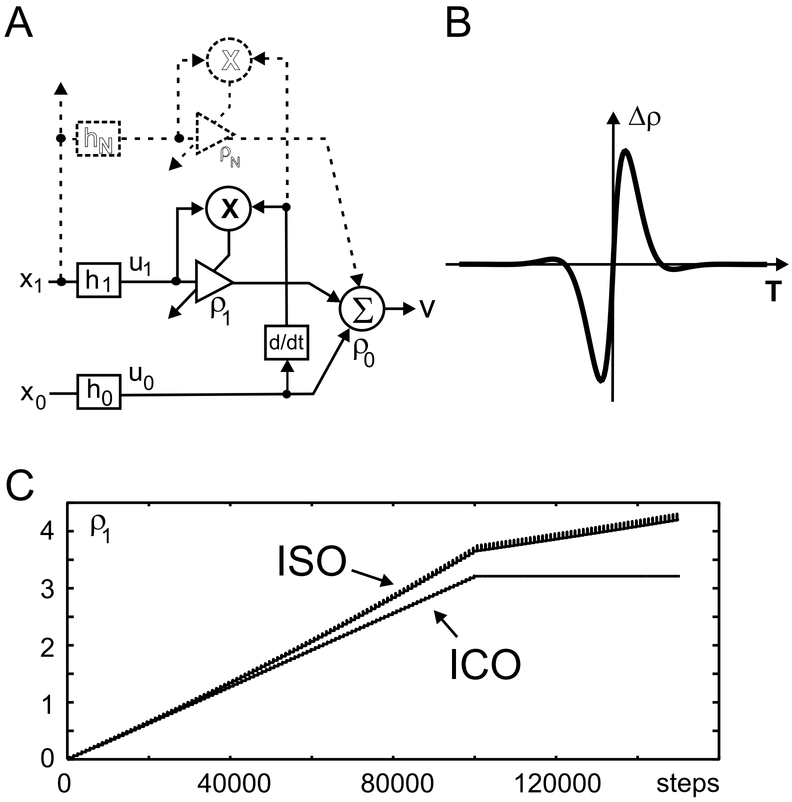

Figure 1: For caption see next page.

Caption to figure 1: Circuit and weight change: A) General form of the neural circuit in an open-loop condition. Inputs x_k are filtered by resonators with impulse response h_k and summed at v with weights ρ_k. The symbol d/dt denotes the derivative. The amplifier symbol denotes a changeable synaptic weight, $\otimes$ is a correlator and $\sum$ is a summation node. The filters $h_1 \ldots h_N$ form a filter bank to cover a wider range of temporal differences between the inputs. B) Weight change curve. Shown is the weight change for two identical resonators H_0, H_1 with $Q = 0.51, f = 0.01$. The two inputs x_0 and x_1 receive delta pulses $x_1(n) = \delta(n)$ and $x_0(n) = \delta(n - T)$. The temporal difference between the inputs is T. The resulting weight change after infinite time is $\Delta\rho$. C) Behaviour of the weight ρ_1 for ICO learning as compared to ISO learning. Pairs of delta pulses are applied as in B. The time between the delta pulses was set to $T = 25$. The pulse-sequence was repeated every 2000 time steps until step 100,000. After step 100,000 only input x_1 receives delta pulses. The learning rate was $\mu = 0.001$.

2 Input Correlation learning

2.1 The neural circuit

Fig. 1A shows the basic components of the neural circuit. In contrast to Porr and Wörgötter 2003a we will for the mathematical formalism employ here the z-transform instead of the Laplace transform. This is due to the fact that the z-space provides a simple way to express the correlation and thus allows a straightforward proof of convergence and stability (see also appendix A).

The learner consists of two inputs x_0 and x_1 which are filtered with functions h.

$$\begin{aligned} u_0 &= x_0 * h_0 \\ u_j &= x_1 * h_j \end{aligned} \tag{2}$$

where the signal x_1 is filtered by a filter-bank of N filters which are indexed by j.

The filter functions $h_1 \ldots h_N$ represent a filter bank with different characteristics so that it is possible to generate complex shaped responses (Grossberg, 1995). The filtered inputs u_k converge onto a single learning unit with weights ρ_k and its output is given by:

$$v = \sum_{k=0}^{N} \rho_k u_k \tag{3}$$

The output will determine the *behaviour* of the system, but not its learning.

To make ICO learning comparable with ISO-learning, for h we will use mostly resonators as in our previous work. We will, however, later also employ other filter-functions if applicable. In discrete time the resonator responses are given by:

$$h(n) = \frac{1}{b} e^{an} \sin(bn) \leftrightarrow H(z) = \frac{1}{(z - e^{p})(z - e^{p^*})} \tag{4}$$

where p^* is the complex conjugate of p. Note, z-transformed functions are denoted by capital letters or as $\rho(z)$ in case of Greek letters. The index for the time steps is n. The real and imaginary parts of p are defined as $a = \mathrm{Re}(p) = -\pi f/Q$ and $b = \mathrm{Im}(p) = \sqrt{(2\pi f)^2 - a^2}$ respectively which is the definition for continuous time. The transformation into discrete time is performed by the exponential e^p in Eq. 4 which is called the impulse invariance method. The parameter $0 \leq f < 0.5$ is the frequency of the resonator normalised to a sampling rate of one. The so called quality $Q > 0.5$ of the resonator defines the decay rate. We will mostly employ a very low quality ($Q = 0.6$) which results in a rapid decay.

2.2 The learning rule

The learning rule for the weight change ρ_j is:

$$\frac{d\rho_j}{dt} = \mu u_j \frac{du_0}{dt} \qquad j > 0 \tag{5}$$

where only input signals are correlated with each other. Comparing Eq. 1 with the new learning rule we see that the output v has been replaced by the input u_0. The derivative indicates that the learning rule implements differential learning (Kosco, 1986). Thus, we have differential *heterosynaptic* learning.

Weight changes can be calculated by correlating the resonator responses of H_0 and H_1 in the z-domain. In the open loop case, this is straightforward and only differs formally from the Laplace domain used in Porr and Wörgötter (2003a) yielding the same weight change curves. Fig. 1B shows the weight change curve for $N = 1$, $H_0 = H_1$ (for parameters see legend). Weights increase for $T > 0$ and decrease for $T < 0$, which means that a sequence of events $x_1 \rightarrow x_0$ leads to a weight increase at ρ_1, whereas the reverse sequence $x_0 \rightarrow x_1$ leads to a decrease. Thus, learning is predictive in relation to the input x_0. Weights stabilise if the input x_0 is set to a constant value (or if x_1 is set to zero).

Fig. 1C shows the behaviour of ICO-learning as compared to ISO-learning in the open loop case for the relatively high learning rate of $\mu = 0.001$. Clearly one sees that ISO-learning contains an exponential instability, which leads to an upward bend in the straight line and prevents weight stabilisation even when setting $x_0 = 0$ at time step 100,000. This is different for ICO-learning which does not contain this instability.

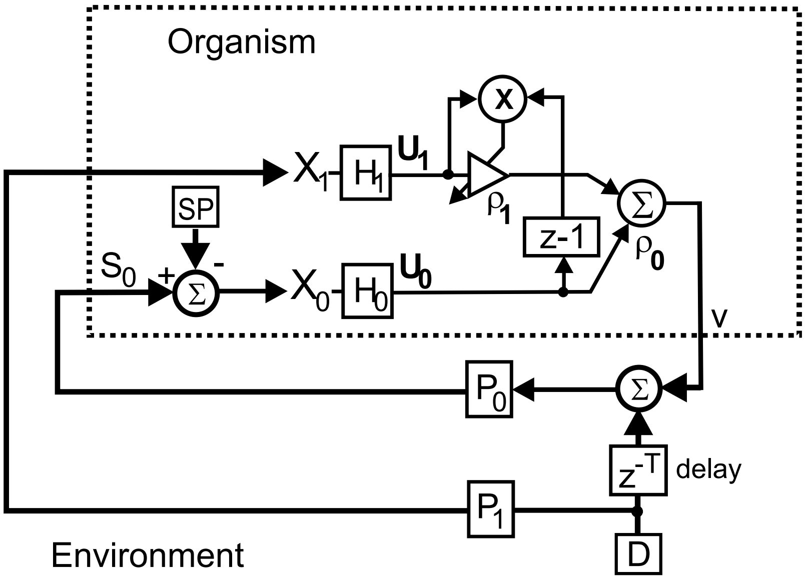

Figure 2: ICO learning embedded into its environment. Σ is a linear summation unit. Except for the constant set-point SP, the inside of the organism resembles ICO learning as shown in Fig. 1 A, but here shown without the filter bank and transformed into the z-domain. D is a disturbance which is delayed by T time-steps. The term $z-1$ denotes a derivative in the z-domain. Transfer functions P_0 and P_1 represent the environment and establish the feedback from the motor output v to the sensor inputs x_0 and x_1. S_0 represents the input before subtracting the setpoint.

3 ICO learning embedded in the environment

3.1 The closed loop circuit – General setup and learning goal

ICO learning is designed for a closed loop system where the output of the learner v feeds back to its inputs x_j after being modified by the environment. The resulting structure (Fig. 2), similar to that described in Porr et al. (2003), is that of an subsumption architecture where we start with an inner feedback loop which is superseded by an outer loop (Brooks, 1991). For a more detailed discussion of such nested structure we refer to Porr et al. (2003).

Feedback loop: Initially only a stable inner reflex or *feedback loop* exists which is established by the transfer function of the organism H_0, the transfer function of the environment P_0, the weight $\rho_0 \neq 0$ and the (here constant) set-point SP. Such a reflex could, for example, be the retraction reaction of an animal, when touching a hot surface. In such an avoidance scenario X_0 would represent the input to a pain receptor, with a desired state of $SP = 0$. Hence a correctly designed reflex will indeed re-establish this desired state, but only in a reactive way, hence, only *after* the disturbance D has upset the state at X_0 for a short while. The delay parameter z^{-T} is here introduced to define the timing relation between inner, late and outer, early (predictive) loop.

Thus, the transfer function H_0 establishes a fixed reaction of the organism by transferring sensor inputs into motor actions. The transfer function P_0 establishes the environmental feedback from the motor output to the sensor input of the organism.

The *goal* of the feedback loop is to keep the set-point SP at S_0 as precise as possible. In this context X_0 can be understood as an *error signal which has to be minimised*. Without loosing generality we will set the set-point SP for all theoretical derivations from now on to zero ($SP = 0$) which means that $S_0 = X_0$ and we interpret the sensor input as the error signal.

Learning Goal: We are going to explain now how learning is achieved. Initially the outer loop, formed by H_1, P_1, is inactive because $\rho_1 = 0$. It receives the disturbance D at sensor input X_1 earlier than the inner loop. In our example, one could think of a heat radiation signal which is felt already *before* touching the hot surface. However, a naive system will not react in the right way, withdrawing the limb before touching, as can be seen in very young children, who will hurt themselves in such a situation.

Hence, the learning goal for this system is to grow ρ_1 such that an earlier appropriate reaction will be elicited after learning. As a consequence, after learning X_0 will, in an ideal case, never leave the set-point again and, in a way, one could think of this as the reflex being shifted earlier in time. In the general case there will be a filter-bank where every filter has its own corresponding weight $\rho_j, j > 0$.

In the following subsections we will establish the formalism for treating such closed loop systems and provide a convergence proof. The main result of this section is that we will show that ICO-learning approaches on-shot-learning in a stable convergence domain provided the inner loop represents a stable feedback controller or, in other words, provided the reflex creates an appropriate and stable reaction. Readers not interested in the mathematical derivations, which rely on the application of some methods from control theory, might consider skipping this section.

3.2 Stability Proof

3.2.1 Responses to a disturbance

The stability of a feedback system can be evaluated by looking at its impulse response to a disturbance. The actual reaction of the feedback system to a disturbance D can be calculated easily in the z-domain. In the simplest case the disturbance is a delta pulse which is just $D = 1$ in the z-domain. In more complex scenarios (like in the experiments) the disturbance is a random event for which we assume that it is bounded and stable. Thus, we apply a disturbance D and observe the changes, for example, at the sensor input X_0:

$$X_0 = Dz^{-T}P_0 + X_0 H_0 \rho_0 P_0 \tag{6}$$

We can now solve for X_0 and get:

$$X_0 = Dz^{-T}\frac{P_0}{1-\rho_0 P_0 H_0} \tag{7}$$

This equation provides the response of the feedback loop to a disturbance D. We demand here that the feedback is designed in a way that X_0 is stable and always decays to zero after a disturbance has occurred. For a general stability analysis of feedback loops we refer the reader to D'Azzo (1988).

In addition we introduce

$$F = X_0 H_0 = z^{-T}\frac{P_0 H_0}{1-\rho_0 P_0 H_0} \tag{8}$$

which is the response of the feedback loop at U_0 to a delta pulse ($D = 1$). We will need this term later for the stability analysis.

A pure feedback loop cannot maintain the set-point all the time because the reaction to a disturbance D by the feedback loop is always too late. Thus, from the point of view of the feedback it is desirable to *predict* the disturbance D to preempt the unwanted triggering of the feedback loop (Palm, 2000). Fig. 2 accommodates this in the most general way by a formal "delay" parameter z^{-T}, which assures that the input x_1 receives the disturbance D earlier than input x_0.

This establishes a second *predictive pathway*, which is inactive at the start of learning ($\rho_1 = 0$). The learning goal is to find a value for ρ_1 so that the learner can use the earlier signal at x_1 to generate an anticipatory reaction which prevents x_0 from deviating from the set-point SP. Generally the predictive pathway is set up as a filter bank where the input x_1 feeds into different filters which generate the predictive response.

The response of the system to a disturbance D *with* the predictive pathway can be obtained in the same way as demonstrated for the feedback loop:

$$X_0 = \frac{P_0[DP_1\sum_{k=1}^{N}\rho_k H_k + Dz^{-T}]}{1-P_0\rho_0 H_0} \tag{9}$$

The goal is now to find a distribution of weights ρ_k so that the condition $X_0 = 0$ is satisfied all the time. In other words: find weights which assure that the input X_0 never deviates from the set-point.

3.2.2 Analysis of Stability

Learning rule in the z-domain: Stability is achieved if the weights ρ_j converge to a finite value. We will prove stability in the z-domain which has two advantages: the derivative can be expressed in a very simple form and the closed loop can be eliminated. The result also provides absolute values of the weights after a disturbance has occurred. Eq. 5 can be rewritten in the z-domain:

$$(z-1)\rho_j(z) = \mu[(z-1)U_0(z)]U_j(z^{-1}) \tag{10}$$

where $(z-1)$ is the derivative. Since the z-transform is not such a commonly used formalism, we refer the reader to appendix A for a detailed description of some of the used methods to arrive at Eq. 10. Note that the weight $\rho_j(z)$ is the z-transformed version of $\rho_j(t)$. The change of the weight $\rho_j(z)$ on the left side is expressed in the same way as the derivative on the right side. This formulation also takes into account that any change of the weight $\rho_j(z)$ might have an immediate impact on the values of U_0 and U_j. Thus, we do not assume here that learning operates at low learning rates μ. At this point we allow for any learning rate.

Calculating the weight: To calculate the weight $\rho_j(z)$ we need the filtered reflex input $U_0 = X_0 H_0$ which can be directly obtained from Eq. 9.

The resulting weight $\rho_j(z)$ can now be evaluated using:

$$\rho_j(z) = \mu F \left[D P_1 \sum_{k=1}^{N} \rho_k H_k + D z^{-T} \right] D^- P_1^- H_j^- \tag{11}$$

where we will abbreviate from now on the time reversed functions $H(z^{-1})$ by H^-.

Solving for $\rho_j(z)$ gives:

$$\rho_j(z) = \frac{\mu F D D^- P_1 P_1^- \sum_{k \neq j, k=1}^{N} \rho_k(z) H_k H_j^- + z^{-T} \mu F D D^- P_1^- H_j^-}{1 - \mu F D D^- P_1 P_1^- H_j H_j^-} \tag{12}$$

which is the value of the weight $\rho_j(z)$ after a disturbance D.

To get a better understanding of the equation above we restrict ourselves now to just one filter in the predictive pathway and set $N = 1$. In that case the sum in the numerator vanishes to give:

$$\rho_1(z) = \frac{z^{-T} \mu F D D^- P_1^- H_1^-}{1 - \mu F D D^- P_1 P_1^- H_1 H_1^-} := \frac{M}{K} \tag{13}$$

Thus, we have a result that can be analysed for the stability of weight $\rho_1(z)$.

Stability criterion: A system is bounded-input bounded-output stable if its impulse response and its corresponding transfer function Y satisfies the following condition:

$$|Y(e^{i\omega})| < \sum_{n=-\infty}^{n=+\infty} |y(n)| < \infty \tag{14}$$

for any ω (Diniz, 2002, p.63). In the following discussion we assume that all functions can be expressed as fractions of polynomials. This is possible as long as the system behaves approximately linearly. Thus, the functions have zeroes and poles in the z-domain. To keep the transfer function $|H(e^{i\omega})|$ of Eq. 14 bounded one has to demand that the unit circle does not contain any poles. Otherwise we would get unlimited exponential growth over time.

Hence, stability analysis requires two components: We need to show that the numerator M in Eq. 13 remains bounded and that the denominator K contains no additional poles.

Numerator M is bounded: We discuss first the numerator M of Eq. 13. It can be interpreted as a correlation between two signals: The first signal FD is the response of the feedback loop F to the disturbance D. The second signal $DP_1H_1 = U_1$ is the response of the predictive pathway P_1H_1 to the disturbance D. Now, the question is under which conditions the correlation between these two signals is stable. The one signal is the impulse response of the stable feedback loop F. Stable feedback loops behave like low-pass filters. Thus, they generate a damped exponential which decays to a constant value. The other signal is the response of the predictive pathway. This signal will also be dominated by a low pass characteristic because the filter H_1 is, by definition, a resonator with a strong low pass characteristics. Furthermore we note that environmental transfer functions (here P_1) generically establish a low pass filter as discussed in Porr et al. (2003). Both signals, the response of the feedback loop F and the response of the predictive pathway converge to zero for infinite time. Hence, it can be assumed that, with great generality the correlation of these two low pass signals also converges. Thus, the numerator poses no threat to stability.

Denominator K has no additional poles: In the next step the denominator K has to be assessed. As we have to test if the denominator creates additional poles. The denominator consists of amplitude terms $DD^-P_1P_1^-H_1H_1^-$ because in general $|Y(\omega)|^2 = Y(z)Y(z^{-1})|_{z=e^{i\omega}}$. These terms are real valued as is the learning rate rendering the denominator K of Eq. 13 as:

$$K = 1 - \mu F|DP_1H_1|^2 \neq 0|_{z=e^{i\omega}} \tag{15}$$

which, for stability, is supposed to be unequal zero to prevent additional poles.

Thus, a simple stability criterion can be stated by:

$$max(\mu|F||DP_1H_1|^2)|_{z=e^{i\omega}} < 1 \tag{16}$$

for all ω. If this criterion is maintained we do not get additional poles. If F, D and P_1 are known, H_1 and μ can be designed in such a way that our stability criterion is met.

Hence, we need to discuss only the one remaining complex function F, which is the impulse response of the feedback loop at U_0. This loop is by construction stable. Also, we remember that DP_1H_1 is the impulse response of the predictive pathway. Weight change results directly from the product of these two functions weighted by the learning rate (see Eq. 5). The question is: Up to which learning rate does this product obey Eq. 16. We note that, if the disturbance D, the gain of the feedback loop F or the gain of the predictive pathway $P_1 H_1$ increases in amplitude, learning becomes faster and this is permitted as long as the effective learning rate:

$$\mu |F| |DP_1 H_1|^2|_{z=e^{i\omega}} = \tilde{\mu} < 1 \tag{17}$$

is below one. In other words the system must not produce an overshoot during its first learning experience.

On the other hand, this also means that one is allowed to increase μ up to that critical value, which leads to the fact that *we can reach one shot learning with ICO learning.*. This is one of the central results of this study.

Behaviour of the final value of ρ_1: Eq. 13 provides us also with the final value of the weight $\rho_1(z)$. To gain a better understanding of the result we multiply Eq. 13 by $H_1 P_1$:

$$\rho_1(z) H_1 P_1 = z^{-T} \frac{G}{1-G} \tag{18}$$

where the constant $G = \mu F D D^- P_1 P_1^- H_1 H_1^-$ is the same for the numerator and the denominator. The expression on the right hand side of Eq. 18 is a formal description of a feedback controlled amplifier. This amplifier can be inverting or non-inverting depending on the sign of the function G. The sign is only determined by the impulse response of the feedback loop F because all the other terms in G are positive.

As a second relevant observation we note that this means that the term on the left hand side of Eq. 18 will have the *same sign* as the feedback reaction F and, because of the delay term z^{-T} it will act *at the moment* when the feedback would be triggered.

More than one filter ($N > 1$): After having understood the case with just one filter $N = 1$ we can now generalise to the case $N > 1$. Thus, we are getting back to Eq. 12. Comparing Eq. 12 with Eq. 13 shows that the stability criteria from the special case also apply to the general case: The denominators are the same in both cases so that the criterion Eq. 16 still holds. The only difference is the sum over correlations between different resonators (H_k correlated with H_j). The crucial question here is whether or not the correlation of these resonator responses is stable. The answer is affirmative because the correlation of one resonator H_k with another one H_j is just the weight change for the case $T = 0$ of the learning rule (see Fig. 1B). This weight change is stable for the same reason as given above: the correlation of two low pass filtered delta pulses is bounded. Thus, ICO learning is also stable for a filter bank which is embedded in a closed loop. The absolute values of the weights in a filter bank are not easy to understand because of the correlations between the filter functions H_j and H_k. These correlations do not play a role after successful learning because then x_0 is constant and therefore any weight change is suppressed anyway.

4 Applications

This section has two purposes: it will compare the performance of ICO learning with differential Hebbian (ISO-) learning and will show that ICO learning can be applied successfully to different application domains. In sections 4.1 and 4.2 we use a biologically inspired task which will be first performed as a simulation and then as a real robot experiment where a robot was supposed to retrieve “food disks”. This task is similar to the one described in Verschure et al. (2003) and to the second experiment in Porr and Wörgötter (2003b). In the simulation we will compare ISO learning with ICO learning and show that the latter is able to perform one shot learning under ideal noise free conditions. The actual robot experiment will show that ICO learning also operates successfully in a physically embodied system where ISO learning fails. Other complex control examples will be presented in the last two experiments using different setups.

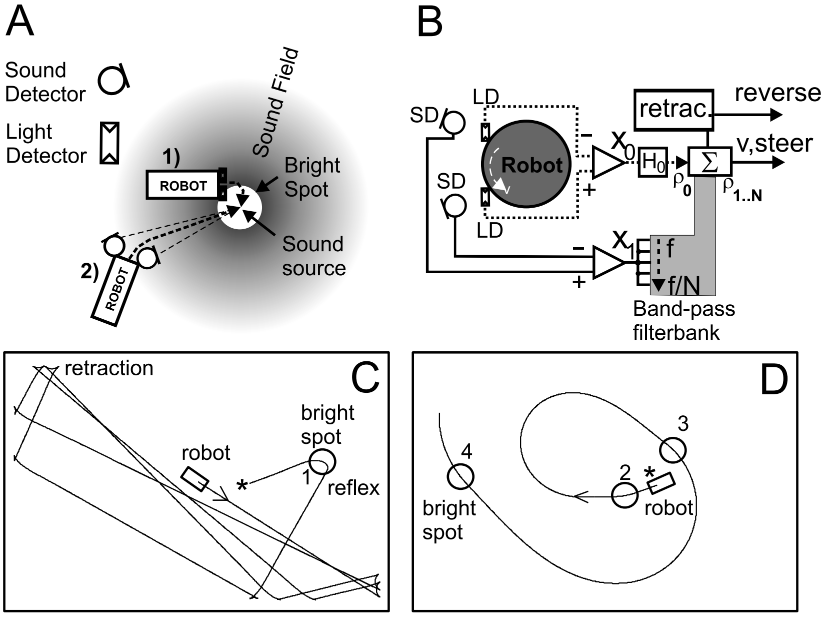

Figure 3: For caption see next page.

Caption to Figure 3: The robot simulation. A) The robot has a reflex mechanism (1) which elicits a sharp turn as soon as it touches the disk laterally and thereby pulls the robot into the centre of the disk. The disk also emits "sound". The robot has the task to use this sound field to find the disk from a distance (2). B) The robot has two touch detectors (LD) which establish with the filter H_0 and the fixed weight ρ_0 the reflex reaction by $x_0 = LD_l - LD_r$. The difference of the signals from two sound detectors (SD) feed into a filter bank. The weights $\rho_1 \dots \rho_N$ are variable and are changed either by ISO or ICO learning. Apart from the reflex reaction at the disk the robot has a simple retraction mechanism when it collides with a wall ("retraction", not used for learning). The output v is the steering angle of the robot. C) Basic behaviour until the first learning experience. The trace at "" continues in D) where the robot has learned to target the disks from a distance. The example here uses ICO learning with $\mu = 5 \cdot 10^{-5}$. Other parameters: filters are set to $f_0 = 0.01$ for the reflex, $f_j = 0.1/j, j = 1 \dots 5$ for the filter bank where $Q = 0.51$ for all filters. Reflex weight was $\rho_0 = 0.005$.*

4.1 The simulated robot

This section presents a benchmark application which compares Hebbian (ISO) learning with our new input correlation (ICO) learning. Fig. 3A presents the task where a simulated robot has to learn to retrieve "food disks" in an arena. The food disks are also emitting simulated sound signals. Two sets of sensor signals are used. One sensor-type (x_0) reacts to (simulated) touch and the other sensor-type (x_1) to the sound. The actual choice of these modalities, however, is not important for the experiment, but this creates a natural situation where sound precedes touch. Hence, learning must use the sound sensors which feed into x_1 to generate an anticipatory reaction towards the "food disk" (Verschure et al., 2003). The circuit diagram is shown in Fig. 3B. The reflex reaction is established by the *difference* of two touch detectors (LD), which cause a steering reaction towards the white disk. Hence x_0 is a transient signal that occurs only during touching of a disk. As a consequence, x_0 is equal to zero if both LDs are not stimulated, which is the trivial case of not touching a disk at all, or when they are stimulated *at the same time* which happens during a straight encounter with a disk. The latter situation occurs after successful learning which, as explained below, leads to the head-on touching of the disks. The reflex has a constant weight ρ_0 which always guarantees a stable reaction. The predictive signal x_1 is generated by using two signals coming from the sound detectors (SD). The signal is simply assumed to give the Euclidean distance ($r_{r,l \to s}$) of the left "l" or right "r" microphone from a sound source "s". The difference of the signals from the left and the right microphone $r_{r \to s} - r_{l \to s}$ is a measure of the azimuth of the sound source to the robot. Successful learning leads to a turning reaction which balances both sound signals and results ideally in a straight trajectory towards the target disk ending in a head-on contact. After having encountered a disk, the disk is removed and randomly placed somewhere else.

An example of successful learning is presented in Fig. 3C and D. The robot first bumps into walls. Eventually, it drives through the disk which provides the robot with the first learning experience. In this example just one experience has been sufficient for successful learning: The trace in Fig. 3D continues the trace from C. Such one-shot learning can be achieved with ICO learning but not with ISO learning. This will be tested now more systematically by comparing the performance for ISO and ICO learning in a few hundred simulations.

We quantify successful and unsuccessful learning for increasing learning rates μ. Learning was considered successful when we received a sequence of four contacts with the disk at a sub-threshold value of $|x_0| < 0.2$. We recorded the actual number of contacts until this criterion was reached. Hence four contacts represent our statistical threshold for deciding between chance and actually successful learning. The choice of a threshold of 0.2 has two reasons: First, when x_0 is below the threshold the robot visibly heads for the centre of the "food disk". Second, the signal x_0 has only discrete values because of a discrete arena of 600×400 where the robot has a size of 20×10. Even if the robot heads perfectly towards the food disk there will be very often a temporal difference between the left and the right sensor because of the discrete representation of both the robot and the round shaped food disk (diameter 20) leading to a small remaining value of x_0 (aliasing effect).

The log-log plots of the number of contacts in Fig. 4A,B show that both rules follow a power law. The similarity of the curves for small learning rates reflect the mathematical equivalence of both rules for $\mu \rightarrow 0$.

The dependence of failures on the learning rate is quite different for ISO- as compared to ICO learning. For differential Hebbian (ISO) learning (Fig. 4B), errors increase roughly exponentially up to a learning rate of $\mu = 10^{-4}$. This behaviour reflects errors caused by the autocorrelation terms. Above $\mu = 10^{-4}$ failures reach a plateau with some statistical variability. For ICO-learning (Fig. 4A) failures remain essentially zero up to $\mu = 0.0002$; the learned behaviour diverges only above that value. In contrast to the ISO-rule, this effect is here due to "over-learning" where the learning gain of the predictive pathway is higher than the gain of the feedback loop. Thus, the predictive pathway becomes unstable already during the first learning experience. This means that the effective learning rate (Eq. 17) has exceeded one. The actual learning rate μ is lower because it is multiplied with the gains of the feedback reaction F and the predictive pathway DH_1P_1 which depend on the actual experimental setup.

For two different learning rates ($\mu = 5 \cdot 10^{-6}, 5 \cdot 10^{-5}$) the weights $\rho_j, j > 0$ and the reflex input x_0 are plotted in Fig. 5. The data have been taken from four simulations of Fig. 4. Thus, success has been measured in the same way as before, requiring $|x_0|$ to be below 0.2 for four consecutive learning experiences. At the low learning rate (A,B,C,D) weights converge to very similar values

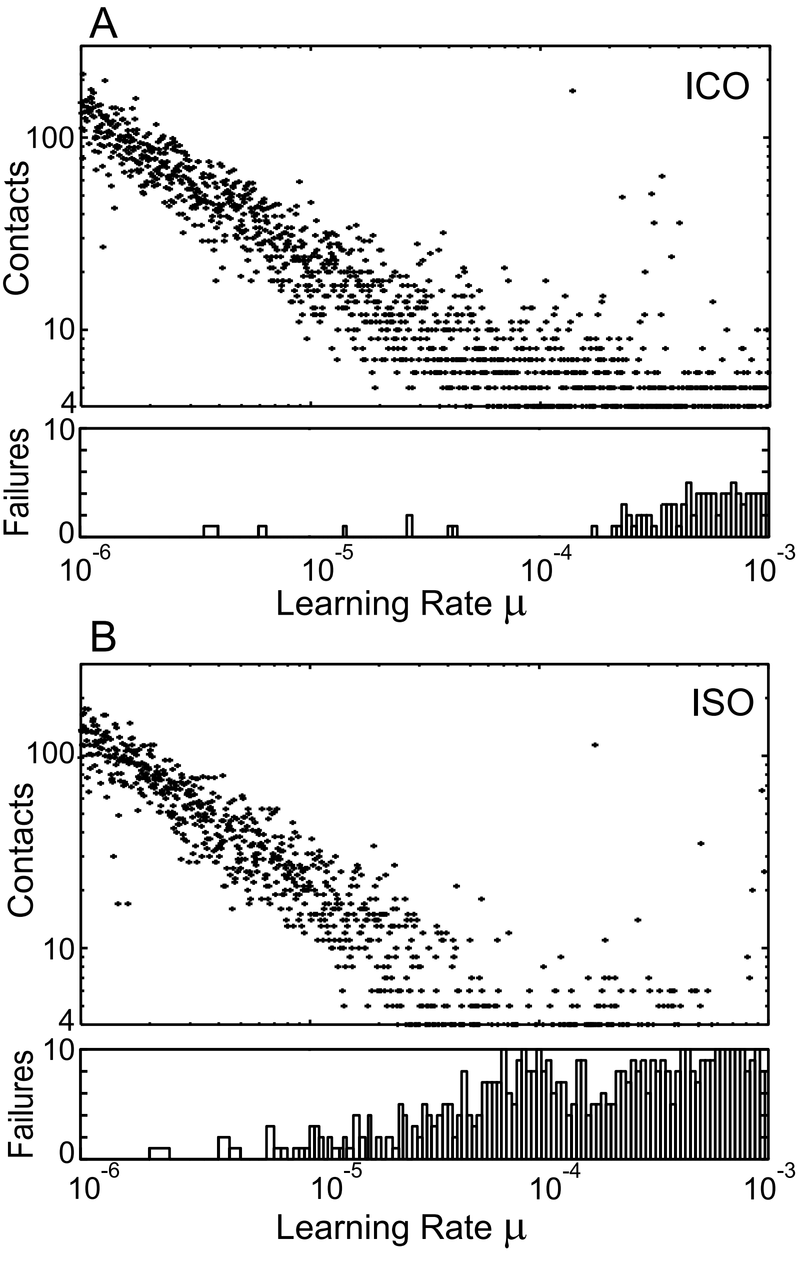

Figure 4: Results from the simulated robot experiment. A) Results from ICO learning and B from the ISO learning. Log-log plots show how many contacts with the target were required for successful learning at a given learning rate μ. Histograms show how many times learning was not successful. The bin size was set to 10 experiments which gives an equal spacing on the log x-axis. Failures are shown on a linear axis.

for ISO- as well as ICO-learning. This is not surprising as for low learning rates the autocorrelation term in ISO learning is small. However, even for such low learning rates the weights drift for the ISO learning case. This can

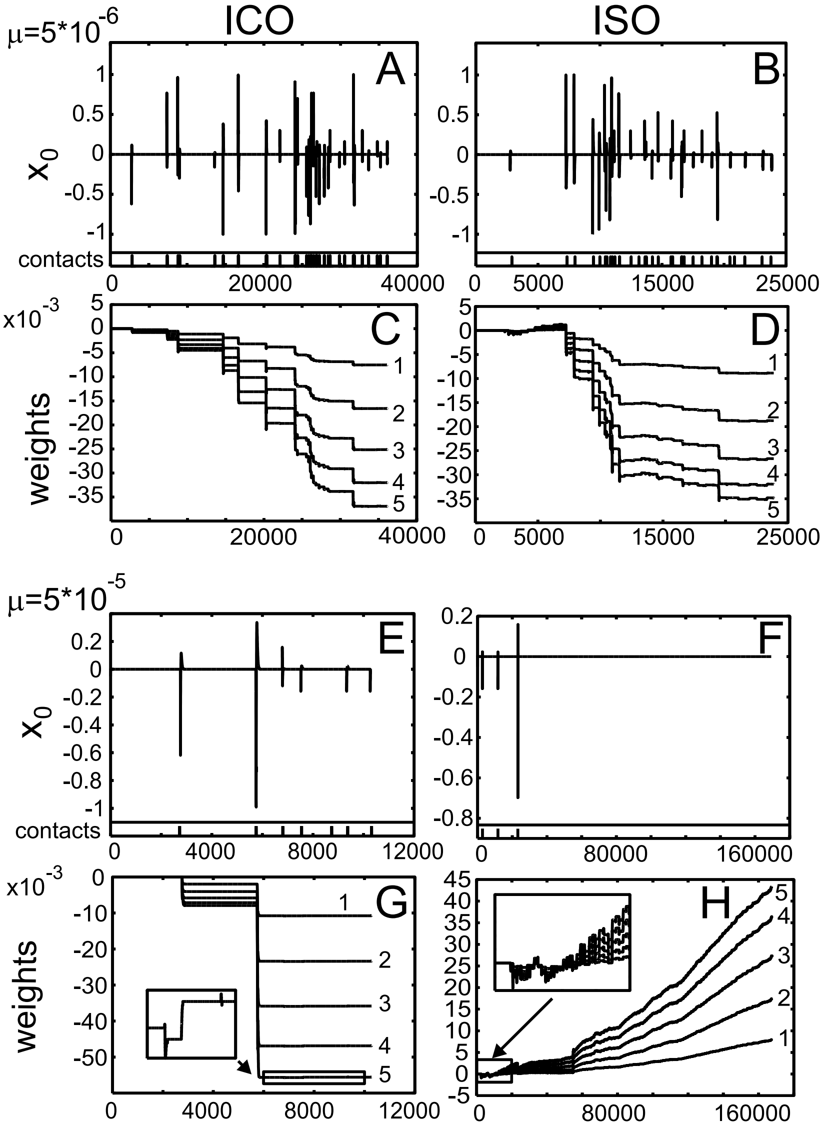

Figure 5: Comparing ICO and ISO learning in individual simulated robot experiments. A,B,E,F) Plots of the reflex input x_0 of the contacts with the food source and (C,D,G,H) of the weights for two learning rates: A-D) $\mu = 5 \cdot 10^{-6}; E - H) 5 \cdot 10^{-5}$ for the two different learning rules ISO- and ICO-learning. The inset in (G) shows steps from $6,000 \ldots 10,000$ plotted with a y-range of $-55.72 \cdot 10^{-3} \ldots - 55.54 \cdot 10^{-3}$. The inset in (H) shows steps from $0 \ldots 20,000$ plotted with a y-range of $-0.001 \ldots 0.0025$.

be seen in particular between steps $3000 - 7,000$ in (D): Although there are no contacts and, thus, x_0 is zero weights drift upwards because of non-zero inputs to the filter bank through x_1. ICO learning (C) does not show any weight drift because of three reasons: First, a constant input at x_0 keeps the weights constant. Second, the predictive input x_1 is zero at the moment x_0

is triggered. This is the case after successful learning as seen, for example, in Fig. 5C between steps $32,000\ldots36,000$. Third, the derivative (u_0') of the *filtered* input x_0 is symmetric so that the weight change is effectively zero. All these factors contribute to stability. Even in the case that x_0 always receives small transients learning is stable. Transients can occur due to aliasing in the simulation or in the real robot due to mechanical imperfections. Such transients trigger unwanted weight change. However, they do not destabilise learning if x_0 is understood as an error signal which always counteracts unwanted weight change. For example, a transient at the reflex input x_0 causes the robot to learn a too strong steering reaction to the left. The next time the robot enters the food disk the too strong left turn causes an error signal at x_0 which reduces the steering reaction again. Thus, one finds that in these cases weights will occasionally grow or shrink due to transients in x_0. However, the weights will be brought back to their optimal values if x_0 carries a proper error signal.

In the experiments with high learning rates (Fig. 5E,F,G,H) learning is very fast resulting in stable weights for ICO after just two learning experiences, which appear in panel (E) as large peaks. After the second peak weights undergo only minimal change. In fact, the "almost head-on" contacts (small peaks in x_0) between steps $6,000\ldots10,000$ of Fig. 5G cause the weights to become more positive again. This is demonstrated the inset of Fig. 5G which indicates that learning has initially caused a slight overshoot of the weights.

A different behaviour is observed for ISO learning (Fig. 5H): After the second contact with the "food disk" the system starts to diverge. The autocorrelation term dominates learning, leading to exponential growth of the weights. After step $22,000$ the reflex input x_0 is zero which means that only the autocorrelation terms change the weights. Behaviourally we observe that the robot first learns the right behaviour, namely driving towards the food disk. This behaviour corresponds to negative weights as seen in Fig. 5C,D,G. After step $10,000$, however, the weights drift to positive values which is behaviourally an *avoidance behaviour*. This behaviour becomes stronger and stronger so that the robot will never touch the food disk again. This unwanted ongoing learning is due to the movements of the robot which cause a continuously changing sound signal x_1 resulting in a non-vanishing auto-correlation term. Thus, while ICO learning (Eq. 5) is stable for both low and high learning rates its differential Hebbian counterpart ISO learning is only stable at low learning rates.

The benchmark tests above have provided an ideal condition for learning where just one "food disk" was in the arena. This gave a perfect correlation between proximal and distal sensor. Having three "food disks" in the arena at the same time renders learning more difficult (Fig. 6). Now, we have no longer a simple relationship between the reflex input x_0 and the predictor x_1. The sound fields from the different "food disks" superimpose onto each other so that the distal information is distorted. However, ICO learning also manages

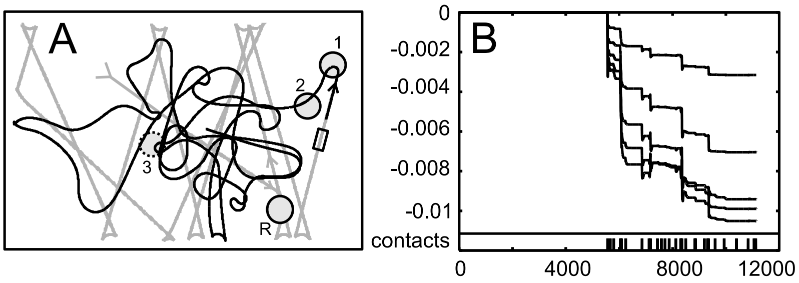

Figure 6: ICO-learning simulation with three simultaneously present food disks. The parameters are the same as for the last simulations. A) Trace of the robot simulation for the whole simulation. The trace before learning kept in gray to differentiate it from the learning behaviour. Initially, learning is switched off for the first 1000 steps to demonstrate purely reflexive behaviour when encountering the disk at "R". The following first three learning experiences are marked as "1–3". B) Weight development during learning. The learning rate was set again to $\mu = 5 \cdot 10^{-5}$.

this scenario without any problems. Fig. 6A depicts the trace of a run starting just before the first learning experience. Panel B shows the corresponding weight development which is stable as well. Again, ISO learning is not able to perform this task at this high learning rate (data not shown).

In summary the simulations demonstrate that ICO learning is much more stable than the Hebbian ISO learning rule. ICO learning is able to operate with high learning rates approaching one shot learning under ideal noise-free conditions.

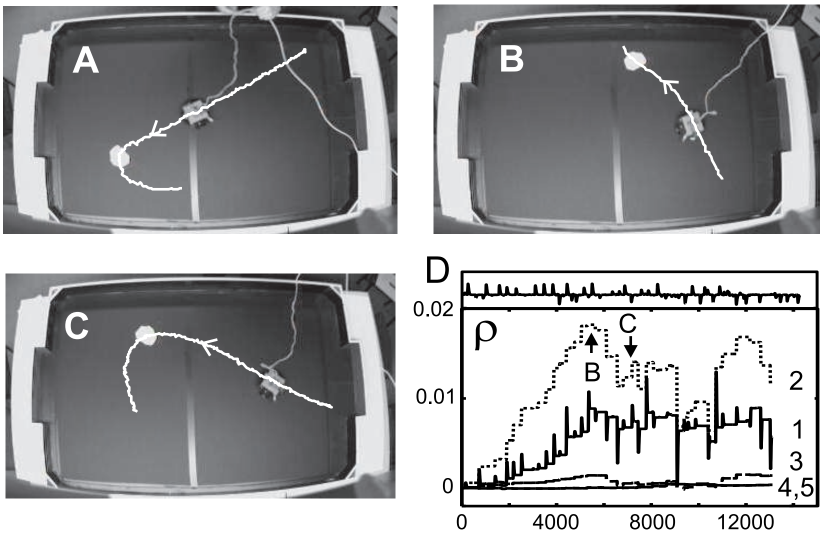

Figure 7: Experiment with a real robot. A-C show traces during the run which lasted 8:46 min. A is taken from the start of the run at 0:06 showing the first reflex reaction, B and C show learned targeting behaviours after 3:45 (14 contacts) and 4:48 (18 contacts) respectively. The development of the weights and the trace x_0 *is shown in D. The values of the weights for Panels B and C are indicated by arrows. Parameters: Learning rate was set to* $\mu = 0.00002$*, the reflex weight to* $\rho_0 = 40$*, and the video input image* $v(\Xi, \Upsilon)$ *was* $\Xi = [1 \dots 96] \times \Upsilon[1, 64]$ *pixels. The scan-line for the reflex was at* $\Upsilon = 50$*, scan-line for the predictor was at* $\Upsilon = 2$*. The reflex* x_0 *and the predictive signal* x_1 *were generated by creating a weighted sum of thresholded gray levels:* $x_{0,1}(\Upsilon) = \sum_{\Xi=1}^{96} (\Xi - (96/2))^2 \Theta(v(\Xi, \Upsilon) - 128)$ *where* Θ *is the Heaviside function. The predictive input is split up into a filter bank of* 5 *filters. The predictive filters have* $100, 50, 33, 25, 20$ *taps where all coefficients are set to one. The reflex pathway is set up with a resonator set to* $f_0 = 0.01$ *and* $Q = 0.51$*. The camera was a standard pinhole camera with a rather narrow viewing angle of* $\pm 35°$.

4.2 The real robot

In this section we will show that the same "food-disk" targeting task can also be solved by a real robot. This is not self-evident because of the complications that arise from the embodiment of the robot and its situatedness in a real environment. See Ziemke 2001 for a discussion of the embodiment principle. In addition, we will show that it is possible to use other filters than resonators in the predictive pathway.

As before, the task of the robot is to target a white disk from a distance. Similar to the simulation the robot has a reflex reaction which pulls the robot into the white disk just at the moment the robot drives over the disk. This reflex reaction is achieved by analysing the bottom scan-line of a camera mounted on the robot. The predictive pathway is created in a similar way: A scanline from the top of the image, which views the arena at a greater distance from the robot (hence "in its future") is fed into a filter-bank of five filters. In contrast to the simulation these filters are set up as *FIR filters* with different numbers of taps where all coefficients are set to one. Thus, the only thing such a filter does is to smear the input signal out over time while the response duration is limited by the number of filter taps. We had two reasons for choosing such filters. First, in contrast to ISO learning, we do not need orthogonality between the reflex pathway and the predictive pathway. Thus, it is possible to employ different filter functions in the different pathways. This made it possible to solve a problem that exists with this robot setup: Because we used a camera with a rather narrow angle we had to put the "food disk" rather centrally in front of the robot. The FIR filters generate step responses which result in a clearly observable behavioural change after learning as soon as the food disk enters the visual field of the robot. Resonator responses are "too smooth" and reflex and learned reaction look too similar.

The reflex behaviour before learning is shown in Fig. 7A where the robot drives exactly straight ahead until it encounters the white disk. Only when it sees the disk directly in front of it a sharp and abrupt turning reaction is generated. Learning rate was set to the highest possible value such that at higher learning rates the system started to diverge. Learning needs longer than in the simulation: About ten contacts with the white disk are needed until a learned behaviour can be seen. Examples for successful learning are shown in panels B,C. Now the robot's turning reaction sets in from a distance of about 50 cm from the target disk. Thus, the robot has learned anticipatory behaviour.

The real robot is subject to complications which do not exist in the simulation. The inertia of the robot, imperfections of the motors and noise from the camera render learning more difficult than in the simulation. As a consequence of this we obtain a non-zero reflex input x_0 all the time as shown in the top trace of Fig. 7D. This is also reflected in the weight development: The weights change during the whole experiment. However, they do not diverge. Rather, they oscillate around their best value. The experiment can be run for a few hours without divergence.

Another reason for weight change is the limited space in the arena. This effect can be drastic if the robot, for example, is caught in a corner of the arena. Imagine the robot first encounters a “food disc” and then directly bumps into a wall. The bump causes then a retraction reaction which changes the input x_0 and therefore the reflex reaction. Consequently learning is affected by such movements. Another aspect is the human operator who throws the food disks in front of the robot. If the food disk is thrown in too late in front of the robot the timing between x_1 and x_0 is different which also leads to wrong correlations.

All additional error sources like noisy data impose an upper limit for the learning rate. This limit, however, is not the theoretical one (Eq. 16) but a practical limit to protect the robot from learning the wrong behaviour during its first learning experience (Grossberg, 1987).

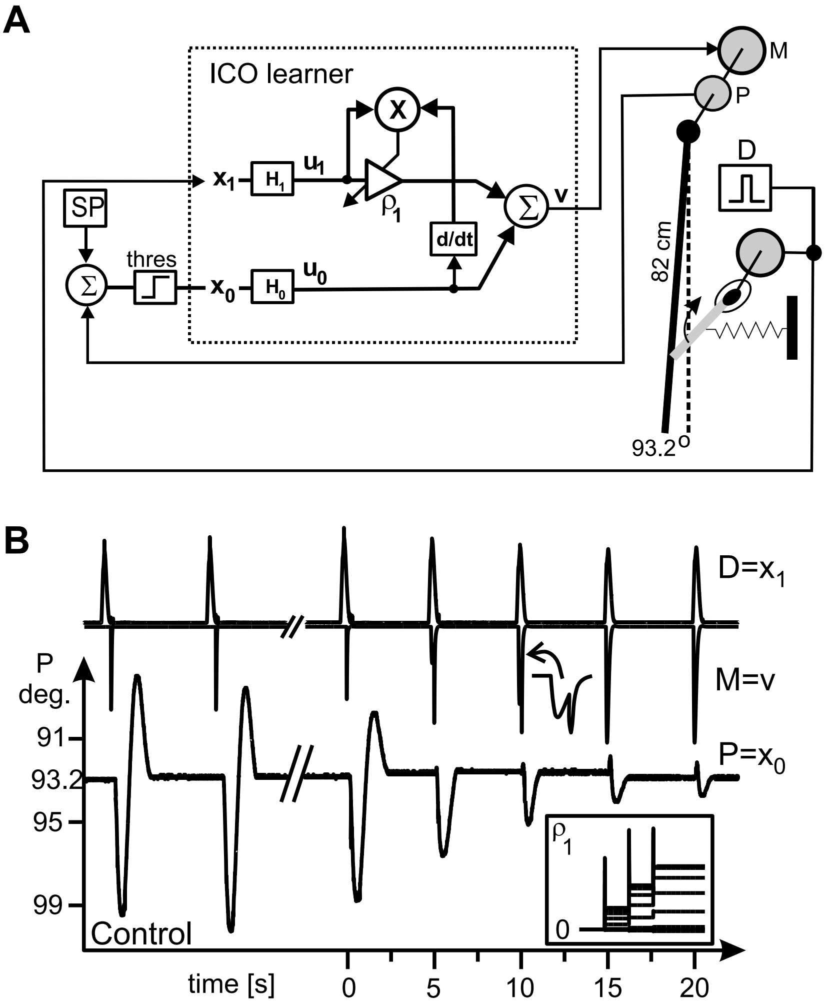

Figure 8: For caption see next page.

Caption to Fig. 8: A) Setup of the mechanical system. The position of the main arm is maintained by a PI controller controlling motor force M with $\rho_0 = 6$; its position is measured by a potentiometer P, $SP = 100°$, effective equilibrium point (EEP) reached $= 93.2°$. Note, the effective equilibrium point will only be identical to the set-point for an ideal controller at infinite gain. A disturbance D is introduced by an orthogonally mounted smaller arm. System parameters were: sampling interval $5\ ms$, $\mu = 2 \times 10^{-5}$, $f_0 = 10\ Hz$, $Q_0 = 0.6$, $Q_1^j = 0.6$, $f_1^j = \frac{20\ Hz}{j}$, $j = 1, \ldots, 10$. B) Signal traces D, M, and P from one experiment. The inset ρ_1 shows the development of the connection weights. Disturbances are compensated after about four trials and weights stabilise.

4.3 Control Applications

In the next two sections we will demonstrate that ICO-learning can be used also in more conventional control situations. To this end, we note first that a reflex is conceptionally very similar to a conventional closed loop controller, where a setup is maintained by a feedback reaction from the controller as soon as a disturbance is being measured. In the next section we will show anticipatory control of a mechanical arm as well as feed-forward compensation of a heat pulse in a temperature controlled container, such as those commonly used for chemical reactions. Mainly we will try to demonstrate that also in these situation ICO learning converges very fast, which may make it applicable in more industrial scenarios, too.

4.3.1 The mechanical arm

To show that ICO learning is also able to operate with a classical PI controller we have set up another mechanical system. In addition we show in this example how weights can be kept stable if the input x_0 is too noisy. Recall that weight stabilisation occurs as soon as $x_0 = 0$ (Eq. 5). To assure this, we employ here a threshold around the SP creating an interval within which x_0 was set to zero.

For our mechanical arm (Fig. 8A,B) a conventional PI controller defines the reflexive feedback loop controlling arm position $P = x_0$. The PI controller replaces the resonator H_0 in this case. To stop the weights from fluctuating we employ a threshold at x_0 of $\Theta = \pm 1°$ around the set-point. Disturbances ($D = x_1$) arise from the pushing force of a second small arm mounted orthogonally to the main arm. A fast reacting touch sensor at contact point measures D. Force D is transient (top trace in Fig. 8B) and the small arm is pulled back by a spring. A moderately high learning rate was chosen to demonstrate how the system develops in time. The second trace $M = v$ shows the motor signal of the main arm. Close inspection reveals that during learning this signal first becomes bi-phasic (small inset curve), where the earlier component corresponds to the learned part and the later component to the PI controller's reaction. At the end of learning only the first component remains (note the forward shift of M with respect to D). Trace $P = x_0$ shows the position signal of the main arm. In the control situation learning was off and a bi-phasic reaction is visible with about 10° position deviation (peak to peak). During learning this deviation is almost fully compensated after four trials. Inset curves ρ_1 at the bottom show that the connection weights have stabilised after the fourth trial. The fifth trial is shown to demonstrate the remaining variability of the system's reaction.

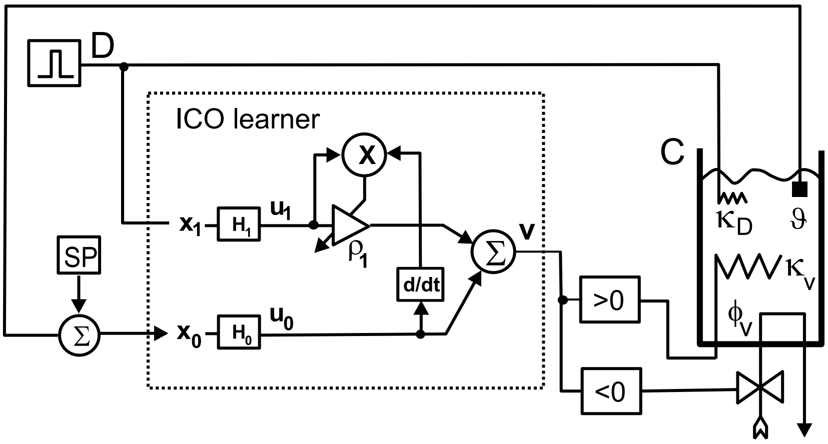

Figure 9: Learning to keep the temperature ϑ in a container C constant against external disturbances. Container volume was 500 *ml, Main heat source was provided by a* 500 *W coil heater (κ_v), main cooling source by pulse-width modulated, valve-controlled water flow through a copper coil (ϕ_v) with max.* 750 *ml/min at* 17° *C. The disturbance heat source (κ_D) received pulses of* 1000 *W from D. Data acquisition and control was performed with a USB-DUX board. Sampling rate was* 1 *Hz. The resonator in the feedback loop was set to $f_0 = 0.2$ Hz, $Q_0 = 0.51$ and its corresponding weight to $\rho_0 = 50$. H_1 is a filter bank of resonators with parameters given below.*

4.3.2 Temperature control

Fig. 9 shows anticipatory temperature control against heat spikes, which could, in a real plant, be potentially very damaging. A feedback loop with a resonator H_0 guarantees a constant temperature SP in a container. The actual temperature is controlled by an electric heater (κ_v) and by a cooling system (ϕ_v). The system can be considered as non-linear because cooling and heating are achieved by different techniques. The demanding task of learning here is to predict temperature changes which are caused by another heater κ_D which is switched on occasionally. In a real application this heater would be rather a second thermometer or other sensor that is able to predict the deviation from the set-point SP.

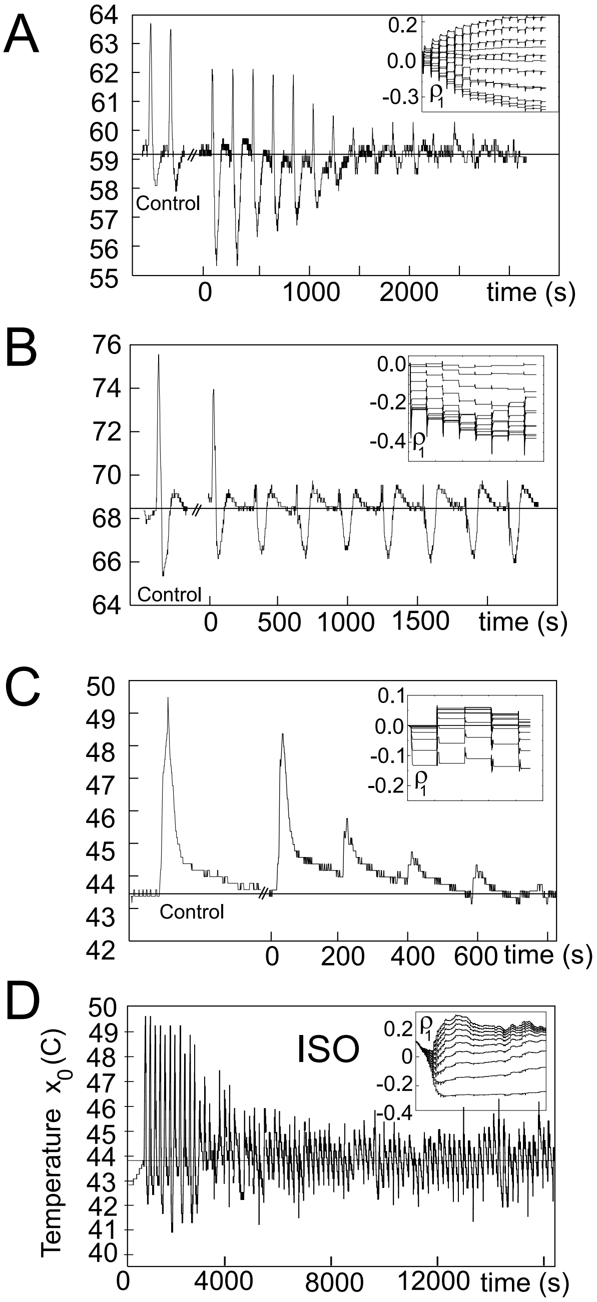

Figure 10: For caption see next page.

Caption to Figure 10: Temperature control experiments. Parameters of the filter bank H_1 *are* $Q_1^j = 0.51$ $f_1^j = \frac{0.1\ Hz}{j}$*, with for A,B:* $j = 1, \ldots, 12$ *and for C,D:* $j = 1, \ldots, 10$*. Experiments with different parameters: A)* $SP = 60° C$*,* $EEP = 59.2° C$*,* $\rho_0 = 250$*, disturbance pulse duration* 10 *s,* $\mu = 4 \times 10^{-8}$*. B)* $SP = 70° C$*,* $EEP = 68.4° C$*,* $\rho_0 = 250$*, disturbance pulse duration* 20 *s,* $\mu = 4 \times 10^{-7}$*. C)* $SP = 44.0° C$*,* $EEP = 43.5° C$*,* $\rho_0 = 150$*, disturbance pulse duration* 12 *s,* $\mu = 7.5 \times 10^{-7}$*. D) Same as in C, except for higher feedback gain of* $\rho_0 = 250$ *(*$EEP = 43.9° C$*) and lower learning rate of* $\mu = 2 \times 10^{-11}$*, but using ISO-learning as denoted in the figure. Note that the levels of the input signals at* ρ_0 *and* $\rho_j, j > 0$ *are different. This leads to different absolute values for* ρ_0 *and* $\rho_j, j > 0$*.*

Several temperature experiments have been performed at different setpoints. In part A a high gain and small μ was used and learning compensates over- and undershoot in about 15 trials. Part B shows that with a high gain and a high learning rate the heat spike is compensated in a single trial, which could represent a vital achievement in a real plant. In this case, compensation of the undershoot, however, takes much longer (not shown). In part C a low gain was used and the system reacts rather slowly. Learning compensates the overshoot after four trials and the effective equilibrium point is now again reached, which was not the case before learning. In all situations (A-C), weights essentially stabilise and drift only slightly around their equilibrium, because no threshold was used at x_0. These small oscillations are similar to the behaviour of the weights in the real robot experiment, which were also oscillating around their equilibrium. Furthermore, we note that learning already sets in strongly in the first trial immediately influencing the output.

In part D we show how the system reacts when using the Hebbian learning rule (ISO-learning). We observe bad convergence even for rather small learning rates of $\mu = 2 \times 10^{-11}$, which is more than 1000 times smaller than those used for panels A-C. These findings mirror the results of the simulations performed above. Some compensation occurs, but weights drift much more. To achieve this specific result, a higher gain had to be used than in the equivalent experiment shown in C. With a lower gain convergence was never reached probably due to the noise in the signals. It should also be noted that this experiment was the best out of 20 using ISO-learning.

5 Discussion

In the current paper we have presented a modification of our old ISO-learning rule, which has led to a dramatic improvement of convergence speed and stability. Mathematically we were able to show that under ideal noise free conditions ICO learning approaches one-shot learning.

The relations of these types of differential Hebbian learning rules (Kosco, 1986; Klopf, 1986; Sutton and Barto, 1987; Roberts, 1999) to temporal sequence learning and to reward-based learning, most notably TD- and Q-learning (Sutton and Barto, 1998); and its embedding in the existing literature has been discussed by us to a great extent in a set of older papers (see in particular Wörgötter and Porr 2005 for a summary). Here we would like to restrict the discussion only to the relevant novel features of ICO-learning.

Now, we have to discuss the different application domains of ICO versus ISO learning. ICO and ISO learning are identical, when using an orthogonal filter set for the condition of $\mu \rightarrow 0$. In this situation, the autocorrelation term of ISO learning vanishes and convergence is guaranteed for ISO learning as well (Porr et al., 2003). The advantage of ISO learning as compared to ICO learning is its "isotropy": all inputs can self-organise into reflex inputs or predictive inputs, depending on their temporal sequence (see Porr and Wörgötter 2003a for a discussion on this property). For ICO learning one needs to build the predefined subsumption architecture (Fig. 2) into the system from the beginning. This means that we have to set up a feedback system which has a desired state and an error signal ($x_0 \rightarrow 0$) which drives learning. In the context of technical control applications this is usually given so that ICO learning is the preferred choice against ISO learning (D'Azzo, 1988). In biology, however, self-organisation is the key aspect. ISO learning has the ability to self organise which pathways become reflex pathways and which pathways become predictive pathways. Reflex pathways can be superseded by other pathways which in turn can become reflex pathways. This also means that ISO learning is able to use any input as an error signal whereas ICO learning can only use x_0 as an error signal. By hierarchically superseding reflex loops ISO learning is able to self-organise subsumption architectures (Brooks, 1989; Porr et al., 2003), which is not possible with ICO learning.

The filter-bank here is used to generate an appropriate behavioural response. In contrast to our older ISO learning it is possible to use other filter functions like step functions for the filter bank which has been demonstrated in the real robot experiment. The only restriction imposed on the filter-bank is that it should establish a low pass characteristic. This characteristic is has to be established by the closed loop not by the open loop. This means that the actual filter in the filter bank need not to posses a low pass characteristic but the closed loop established by the environment. Filter banks have been employed in other learning algorithms, for example in TD-learning (Sutton and Barto, 1998). In contrast to our learning scheme the filters there are used only for the critic and not for the actor. In other words: they are used to smear out the conditioned stimulus so that it can be correlated with the unconditioned stimulus.

In terms of synaptic plasticity ISO- and ICO learning differ substantially: While ISO learning can be interpreted as a homo-synaptic learning rule (Porr et al., 2004), ICO-learning is strictly heterosynaptic. The neuronal literature on heterosynaptic plasticity normally emphasises that it is essentially a modulatory process which modifies (conventional) homo- synaptic learning (Bliss and Lomo, 1973; Markram et al., 1997), but cannot lead to plasticity on its own (Ikeda et al., 2003; Bailey et al., 2000; Jay, 2003). However, evidence was also found for a more direct influence of heterosynaptic plasticity in Aplysia siphon sensory cells (Clark and Kandel, 1984), in the Amygdala (Humeau et al., 2003) and in the limbic system (Beninger and Gerdjikov, 2004; Kelley, 2004).

As a consequence, heterosynaptic learning rules have so far mostly been used to emulate modulatory processes, for example, by the implementation of three-factor learning rules, trying to capture dopaminergic influence in the Striatum and the Cortex (Schultz and Suri, 2001). To our knowledge ICO is the only learning rule that operates strictly heterosynaptically, which, for network learning and plasticity, might open new avenues as compared to the well established Hebb rules (Oja, 1982; Kohonen, 1988; Linsker, 1988; MacKay, 1990; Rosenblatt, 1958; von der Malsburg, 1973; Amari, 1977; Miller, 1996a).

For example, the tremendous stability of ICO, which is guaranteed for $x_0 = 0$ or can be enforced by using a threshold ($x_0 < \Theta$), will allow designing stable nested or chained architectures of several ICO-learning units where the "primary" units in such an architecture are controlled by the feedback neuronal activity of the "secondary" ones. Hence, the secondary neurons in such a setup would provide the x_0 signal by ways of an internal feedback loop, which takes the role and replaces the behavioural feedback employed here. Not only does this shed an interesting light on neuronal feedback loops like the cortico-thalamic loops (Alexander et al., 1986; Morris et al., 2005) but it might also offer interesting possibilities for novel network architectures, where stability can be built into the system by ways of such loops.

Like ISO learning ICO learning develops a forward model (Palm, 2000, p.592) of the reflex reaction established by H_0, ρ_0 and P_0. The forward model is represented by the resonators and weights $H_j, \rho_j, j > 0$ (Porr et al., 2003). The main advantage of ICO learning against ISO learning is that it is not limited to resonators (H_j) as filters. We have shown here that instead of resonators simple FIR filters can be used for the filter bank. The required low pass characteristic came from the environment. The FIR filter was, however, just an example. Future research has to systematically explore which linear and non-linear filters are suitable for ICO learning.

Finding a target with a simulated or real world device has been employed in earlier works. The oldest model with hand tuned fixed weights has been employed by Walter (1953) where his tortoise had to find its home cage. To find the optimal weights Paine and Tani (2004) have recently employed a genetic algorithm which is able to solve a T-maze task. Their simulated robots need 63 generations. When it comes to learning basically two paradigms are employed: reinforcement learning or Hebbian learning. In reinforcement learning Q-learning seems to be the learning rule of choice. Q-learning generates optimal policies to retrieve a reward where a policy associates a sensor input with an action. The Q-value evaluates the policy if it leads to a reward or not. The higher the Q-value the more probable is the future or immediate reward.

Q-learning has been successfully to applied by Bakker et al. (2002) to a T-maze task: The robot has to learn that a road sign at the entrance of the T-maze gives the clue if the reward is in the left or in the right arm. To solve this task the simulated Kephera robot needed 2,500 episodes. Thrun (1995) also employs Q-learning to find a target. In contrast to Bakker et al., however, the robot navigates freely in an environment. This task probably comes closest to our task. Successful targeting behaviour is established after approximately 20 episodes. Our robot needed approximately 15 contacts with the white disk to find it reliably. However, after 20 episodes the success rate in the experiment by Thrun is still very poor. Further 80 episodes are needed to bring the success rate up to 90%. Our robot has already a comparable success rate of 90% after these 15 contacts, given that the camera can see the disk. The different convergence speeds suggest that Thrun has employed a lower learning rate.

The other learning rule which has been employed to solve targeting tasks is Hebbian learning. In Verschure and Voegtlin (1998) and Verschure et al. (2003) the robot has the task to find targets. Similar to our robot, their robot is equipped with proximal and distal sensors. The proximal sensors trigger reflex reactions. The task is to use the distal sensors to find the targets from the distance. In contrast to our heterosynaptic learning, Verschure and Voegtlin employ Hebbian learning and not heterosynaptic learning. In order to limit unbounded weight growth they modified the Hebbian learning rule. In Verschure et al. (2003) this has been done directly by adding a decay term proportional to the weight. In Verschure and Voegtlin (1998) infinite weight growth is counteracted by inhibiting the signals from the distal sensors or in other words the conditioned stimuli. Unfortunately a direct comparison of the performances with our experiment is not possible because it is not clear from Verschure and Voegtlin (1998); Verschure et al. (2003) how many contacts with the target were needed to learn the behaviour.

Touzet and Santos (2001) have systematically compared different reinforcement learning algorithms applied to obstacle avoidance. Such systematic approaches are difficult to achieve because of different hardware platforms, different environments and different ways of documenting the robot runs. Thus, a systematic evaluation of the different learning rules will be subject of further investigation.

Acknowledgements

We thank David Murray Smith for fruitful comments on the manuscript. We are also grateful to Thomas Kulvicus and Tao Geng who constructed the mechanical arm.

A Using the z-transform for the convergence proof

In this section we are describing in detail how we transformed the learning rule Eq. 5 into the z-domain. The z-transform of a sampled (or time discrete) signal $x(n)$ is defined as:

$$X(z) = \sum_{n=-\infty}^{\infty} x(n) z^{-n} \tag{19}$$

The capital letter $X(z)$ denotes the z-transform of the original signal $x(n)$. The z-transform is the discrete version of the Laplace transform which in turn is a generalised version of the Fourier transform. The original signal and its z-transform are equivalent if convergence can be guaranteed (Proakis and Manolakis, 1996).

The z-transform has a couple of useful properties which simplify the convergence proof shown in section 3.2.2.

- **Convolution:** The z-transform can be applied not only to signals but also to filters. Filtering in the time domain means convolution of the signal $x(n)$ with the impulse response $h(n)$ of the filter. In the z-domain it is just a multiplication with the transformed impulse response:

  $$x(n) * h(n) \Leftrightarrow X(z)H(z) \tag{20}$$

  For example, Eq. 2 turns into $U_j = X_j H_j$ in the z-domain where the capital letters indicate the z-transformed functions. Once transformed into the z-domain equations can be solved by simple algebraic manipulations. For example Eq. 6 can be solved for X_0 by subtracting $X_0 H_0 \rho_0 P_0$ from both sides and then dividing the equation by $1 - \rho_0 P_0 H_0$.

- **Correlation:** The correlation of two signals can be derived from the convolution (Eq. 20) by recalling that a correlation is just a convolution where one signal is reversed in time. Time reversal $x(-n)$ in the z-domain $X(z^{-1})$ leads directly to a formula for correlation:

  $$x(n) * h(-n) \Leftrightarrow X(z)H(z^{-1}) \tag{21}$$

- **Derivative:** The derivative in the z-space can be expressed as an operator (Bronstein and Semendjajew, 1989):

  $$\frac{d}{dn} \Leftrightarrow (z-1) \tag{22}$$

With that background it is now possible to z-transform the learning rule Eq. 5:

$$\rho_j' = \mu u_j u_0' \qquad \Leftrightarrow \qquad (z-1)\rho_j = \mu U_j(z^{-1})(z-1)U_0(z) \tag{23}$$

which is equation Eq. 10. Note that the derivative on the right hand side is not time reversed because it belongs to U_0.

References

Alexander, G., DeLong, M., and Strick, P. (1986). Parallel organization of functionally segregated circuits linking basal ganglia and cortex. *Annu Rev Neurosci*, 9:357–381.

Amari, S. I. (1977). Neural theory of association and concept-formation. *Biol Cybern*, 26(3):175–185.

Bailey, C. H., Giustetto, M., Huang, Y. Y., Hawkins, R. D., and Kandel, E. R. (2000). Is heterosynaptic modulation essential for stabilizing Hebbian plasticity and memory? *Nat Rev Neurosci*, 1(1):11–20.

Bakker, B., Linåker, F., and Schmidhuber, J. (2002). Reinforcement learning in partially observable mobile robot domains using unsupervised event extraction. In *Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Lausanne.

Beninger, R. and Gerdjikov, T. (2004). The role of signaling molecules in reward-related incentive learning. *Neurotoxicity Research*, 6(1):91–104.

Bienenstock, E., Cooper, L., and Munro, P. (1982). Theory for the development of neuron selectivity, orientation specifity and binocular interpretation in visual cortex. *J. Neurosci*, 2:32–48.

Bliss, T. and Lomo, T. (1973). Long-lasting potentiation of synaptic transmission in the dentrate area of the anaesthetized rabbit following stimulation of the perforant path. *J Physiol*, 232(2):331–356.

Bronstein, I. and Semendjajew, K. (1989). *Taschenbuch der Mathematik*. Harri Deutsch, Thun and Frankfurt/Main, 24 edition.

Brooks, R. A. (1989). How to build complete creatures rather than isolated cognitive simulators. In VanLehn, K., editor, *Architectures for Intelligence*, pages 225–239. Erlbaum, Hillsdale, NJ.

Brooks, R. A. (1991). Intelligence without representation. *Artificial Intelligence*, 47:139–159.

Clark, G. A. and Kandel, E. R. (1984). Branch-specific heterosynaptic facilitation in aplysia siphon sensory cells. *PNAS*, 81(8):2577–2581.

Dayan, P. and Sejnowski, T. (1994). Td(λ) converges with probability 1. *Mach. Learn.*, 14(3):295–301.

D'Azzo, J. J. (1988). *Linear Control System analysis and design*. McGraw-Hill, New York.

Diniz, P. S. R. (2002). *Digital Signal Processing.* Cambridge university press, Cambridge.

Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive Science*, 11:23–63.

Grossberg, S. (1995). A spectral network model of pitch perception. *J Acoust Soc Am*, 98(2):862–879.

Hebb, D. O. (1949). *The organization of behavior: A neurophychological study.* Wiley-Interscience, New York.

Humeau, Y., Shaban, H., Bissière, S., and Lüthi, A. (2003). Presynaptic induction of heterosynaptic associative plasticity in the mammalian brain. *Nature*, 426(6968):841–845.

Ikeda, H., Akiyama, G., Fujii, Y., Minowa, R., Koshikawa, N., and Cools, A. (2003). Role of ampa and nmda receptors in the nucleus accumbens shell in turning behaviour of rats: interaction with dopamine and receptors. *Neuropharmacology*, 44:81–87.

Jay, T. (2003). Dopamine: a potential substrate for synaptic plasticity and memory mechanisms. *Prog Neurobiol*, 69(6):375–390.

Kaelbling, L. P., Littman, M. L., and Moore, A. W. (1996). Reinforcement learning: A Survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Kelley, A. E. (2004). Ventral striatal control of appetitive motivation: role in ingestive behaviour and reward-related learning. *Neuroscience and Biobehavioural Reviews*, 27:765–776.

Klopf, A. H. (1986). A drive-reinforcement model of single neuron function. In Denker, J. S., editor, *Neural Networks for Computing: Snowbird, Utah*, volume 151 of *AIP conference proceedings*, New York. American Institute of Physics.

Kohonen, T. (1988). *Self-Organization and Associative Memory.* Springer, Berlin, Heidelberg, New York, 2 edition.

Kosco, B. (1986). Differential hebbian learning. In Denker, J. S., editor, *Neural Networks for computing: Snowbird, Utah*, volume 151 of *AIP conference proceedings*, pages 277–282, New York. American Institute of Physics.

Linsker, R. (1988). Self-organisation in a perceptual network. *Computer*, 21(3):105–117.

MacKay, D. J. (1990). Analysis of linsker's application of hebbian rules to linear networks. *Network*, 1:257–298.

Malenka, R. C. and Nicoll, R. A. (1999). Long-term potentiation — a decade of progress? *Science*, 285:1870–1874.

Markram, H., Lübke, J., Frotscher, M., and Sakman, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps. *Science*, 275:213–215.

Miller, K. D. (1996a). Receptive fields and maps in the visual cortex: Models of ocular dominance and orientation columns. In Donnay, E., van Hemmen, J., and Schulten, K., editors, *Models of Neural Networks III*, pages 55–78. Springer-Verlag.

Miller, K. D. (1996b). Synaptic economics: Competition and cooperation in correlation-based synaptic plasticity. *Neuron*, 17:371–374.

Morris, B., Cochran, S., and Pratt, J. (2005). PCP: from pharmacology to modelling schizophrenia. *Curr Opin Pharmacol*, 5(1):101–106.

Oja, E. (1982). A simplified neuron model as a principal component analyzer. *J Math Biol*, 15(3):267–273.

Paine, R. W. and Tani, J. (2004). Motor primitive and sequence self-organisation in a hierachical recurrent neural network. *Neural Networks*, 17:1291–1309.

Palm, W. J. (2000). *Modeling, Analysis and Control of Dynamic Systems.* Wiley, New York.

Porr, B., Saudargiene, A., and Wörgötter, F. (2004). Analytical solution of spike-timing dependent plasticity based on synaptic biophysics. In Thrun, S., Saul, L., and Schölkopf, B., editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA.

Porr, B., von Ferber, C., and Wörgötter, F. (2003). Iso-learning approximates a solution to the inverse-controller problem in an unsupervised behavioural paradigm. *Neural Comp.*, 15:865–884.

Porr, B. and Wörgötter, F. (2001). Temporal hebbian learning in rate-coded neural networks: A theoretical approach towards classical conditioning. In Dorffner, G., Bischof, H., and Hornik, K., editors, *Artificial Neural Networks — ICANN 2001*, volume 2130, pages 1115–1120, Berlin. Springer.

Porr, B. and Wörgötter, F. (2003a). Isotropic Sequence Order learning. *Neural Comp.*, 15:831–864.

Porr, B. and Wörgötter, F. (2003b). Isotropic sequence order learning in a closed loop behavioural system. *Roy. Soc. Phil. Trans. Math., Phys. & Eng. Sciences*, 361(1811):2225–2244.

Proakis, J. G. and Manolakis, D. G. (1996). *Digital Signal Processing.* Prentice-Hall, New Jersey.

Roberts, P. D. (1999). Temporally asymmetric learning rules: I. Differential Hebbian Learning. *Journal of Computational Neuroscience*, 7(3):235–246.

Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.*, 65(6):386–408.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, 275:1593–1599.

Schultz, W. and Suri, R. E. (2001). Temporal difference model reproduces anticipatory neural activity. *Neural Comp.*, 13(4):841–862.

Sutton, R. (1988). Learning to predict by method of temporal differences. *Machine Learning*, 3(1):9–44.

Sutton, R. and Barto, A. (1981). Towards a modern theory of adaptive networks: Expectation and prediction. *Psychological Review*, 88:135–170.

Sutton, R. S. and Barto, A. (1987). A temporal-difference model of classical conditioning. In *Proceedings of the Ninth Annual Conference of the Cognitive Science Society*, pages 355–378, Seattle, Washington.

Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: An Introduction.* Bradford Books, MIT Press, Cambridge, MA, 2002 edition.

Thrun, S. (1995). An approach to learning mobile robot navigation. *Robotics and Autonomous systems*, 15:301–319.

Touzet, C. and Santos, J. F. (2001). Q-learning and robotics. In *IJCNN'99, European Simulation Symposium*, Marseille.

Verschure, P. and Voegtlin, T. (1998). A bottom-up approach towards the acquisition, retention, and expression of sequential representations: Distributed adaptive control III. *Neural Networks*, 11:1531–1549.

Verschure, P. F. M. J., Voegtlin, T., and Douglas, R. J. (2003). Environmentally mediated synergy between perception and behaviour in mobile robots. *Nature*, 425:620–624.

von der Malsburg, C. (1973). Self-organization of orientation sensitive cells in the striate cortex. *Kybernetik*, 14(2):85–100.

Walter, W. G. (1953). *The Living Brain.* G. Duckworth, London.

Watkins, C. and Dayan, P. (1992). Q-learning. *Machine Learning*, 8:279–292.

Watkins, C. J. (1989). *Learning from delayed rewards.* PhD thesis, University of Cambridge, England.

Wörgötter, F. and Porr, B. (2005). Temporal sequence learning, prediction and control - a review of different models and their relation to biological mechanisms. *Neural Comp*, 17:245–319.

Ziemke, T. (2001). Are robots embodied? In *First international workshop on epigenetic robotics Modeling Cognitive Development in Robotic Systems*, volume 85, Lund.