Isotropic Sequence Order Learning

Bernd Porr and Florentin Wörgötter

Department of Psychology, University of Stirling, Stirling FK9 4LA,
Scotland

In this article we present an isotropic, unsupervised algorithm for temporal sequence learning. No special reward signal is used such that all inputs are completely isotropic. All input signals are bandpass filtered before converging onto a linear output neuron. All synaptic weights change according to the correlation of bandpass-filtered inputs with the derivative of the output. We investigate the algorithm in an open- and a closed-loop condition, the latter being defined by embedding the learning system into a behavioural feedback loop. In the open-loop condition we find that the linear structure of the algorithm allows analytically calculating the shape of the weight change which is strictly hetero-synaptic and follows the shape of the weight change curves found in spike-time dependent plasticity. Furthermore, we show that synaptic weights stabilise automatically when no more temporal differences exist between the inputs without additional normalising measures. In the second part of this study, the algorithm is is placed into an environment which leads to closed sensor-motor loop. To this end a robot is programmed with a pre-wired retraction reflex reaction in response to collisions. Through ISO-learning the robot achieves collisions avoidance by learning the correlation between his early range-finder signals and the later occuring collision signal. Synaptic weights stabilise at the end of learning as theoretically predicted. Finally we discuss the relation of ISO-learning with other drive reinforcement models and with the commonly used "temporal difference" (TD-) learning algorithm. This study is followed up by a mathematical analysis of the closed-loop situation in the accompanying article.

1 Introduction

A central goal of every autonomous agent is to maintain homoeostasis (Ashby, 1956), without which it will eventually disintegrate (viz. ”die”). A generic way to achieve this is by reacting to a disturbance of the homoeostasis with a closed loop negative feedback mechanism (a reflex), which will compensate for the disturbance by means of a (motor) reaction. Thus, the simplest form of sensible autonomous behaviour can be obtained by designing an agent whose (re-)actions are reflex based (Brooks, 1989). This type of behaviour is even found in rather primitive animals like amoebas, which retract their philopodia when encountering a potentially damaging chemical gradient.

Such sensor-motor reflex-loops represent typical feedback reaction systems, because a reflex will always be elicited only *after* a sensor event has already been encountered; as the word “feedback” implies. The reaction delay which is unavoidably associated with every reflex-loop can in the worst case even lead to fatal situations. Thus, in any kind of improved behaviour the acting agent will try to avoid reflexes, for example, by predicting one sensor event from another earlier occurring event (at a different sensor). This takes place when predicting pain from the heat which radiates from a hot surface in order to prevent a retraction reflex by means of an anticipatory avoidance reaction. In this example heat radiation and pain are causally related. Many other similar causal relations exist during the life of an animal; for example, between smell and taste when foraging or between vision and touch when exploring. In all of these cases a temporal sequence of sensor events occurs, which needs to be learned in order to avoid reflex reactions to the later event. Thus, temporal sequence learning is a dominant aspect of animal behaviour. It requires a late event which serves as a reference to which the earlier event temporally relates. The goal is to learn this specific temporal relation.

In artificial systems temporal sequence learning can be achieved, for example by classical Hebbian learning (Hebb, 1967) in combination with delays Levy and Minai (1993), by differential Hebbian learning (Kosco, 1986; Klopf, 1986) or by the very influencial TD-reinforcement learning algorithm (TD=temporal difference) (Sutton, 1988; Montague et al., 1993; Dayan and Sejnowski, 1994; Abbott and Blum, 1996; Dayan et al., 2000; Rao and Sejnowski, 2001; Haruno et al., 2001; Schultz and Suri, 2001). In TD-learning the “later event” is represented by a designated reference signal (mostly a reward- or punishment-signal) to which the prediction of the learner is explicitly compared. The reference signal, thus, represents an explicitly defined, so-called evaluative feedback for the learning, which stops when prediction and reward match. This may pose a problem as pointed out by Klopf (1988), who had emphasised that evaluative feedback cannot exist in autonomously acting agents, which normally cannot rely on any external, evaluative, (teacher-like) signal. Klopf's differential Hebbian algorithm is indeed non-evaluative and belongs to the so called class of drive reinforcement models. This issue, however, is still rather controversial. Klopf's arguments are convincing, while, on the other hand, evidence exists that dopamine could indeed serve as such a possible reward-like reference signal in the brains of higher mammals (Schultz et al., 1997; Schultz and Suri, 2001) which can respond to complex learning situations such as instrumental (operand) conditioning. Less complex forms of learning such as basic classical conditioning, however, can even be observed in very simple creatures (for example, Aplysia), which do not have a reward system — or at least one has not yet been discovered (Kandel et al., 1983).

One aspect of the current study, therefore, is to design an algorithm in which sequence order learning takes place in a reward-free, unsupervised way by means of a temporal-Hebb learning rule which is isotropic with respect to the inputs (hence the name ”ISO-learning”, which stands for Isotropic Sequence Order learning)[1]. Thus, the algorithm is strictly based on the causal relation between its inputs, which is in reality often given by the ”properties of the world” as described by the examples above. The ”reference-signal” is just the latest occurring signal (which often has the highest initial synaptic weight); a situation, which can change during learning.

The paper is organised in the following way. Firstly we will introduce the algorithm in an open-loop paradigm. Its linear structure allows an analytical treatment of some of its main characterising features. More complex aspects will be addressed with simulations. In this part of the study it will also become clear that all input lines are mathematically equivalent in our algorithm. Furthermore, we will show that the algorithm performs strict hetero-synaptic learning. A detailed comparison of ISO-learning with other algorithms is given in Appendix B.

In the second part of this study we will embed our algorithm in a behavioural loop by means of a robot experiment. This creates a self-referential system and leads to stability. As a consequence of the fact that learning is

[1] The self-referential structure of this abbreviation is meant to hint at the self-referential behavioural loop introduced in the second part of this article.

hetero-synaptic, we find that synaptic weights will self-stabilise as soon as the reflex input becomes silent.

One central aspect of this and the following paper is to show that unsupervised open-loop ISO-learning inherently turns into a reference-based system as soon as it is embedded into a non-evaluative environment which leads to a closed sensor-motor loop. This could be expected from the results of Klopf (1988) but we will show analytically in the second paper (Porr et al., 2002) that such a closed loop system creates — by means of the learning process — a "forward model" of its environment. Temporal sequence learning using the ISO-learning algorithm can therefore be understood as finding a solution to the specific inverse controller problem which replaces a reflex by its forward model.

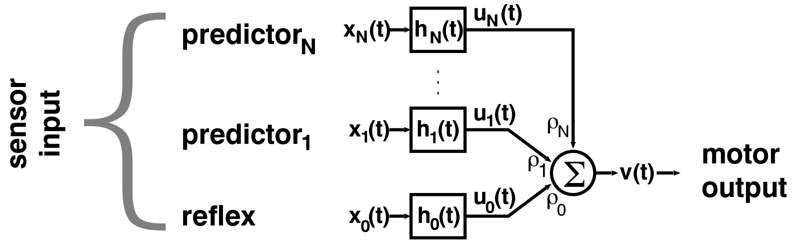

Figure 1: The basic circuit in the time domain.

2 Open Loop: ISO-Learning

First we will describe the algorithm itself and its characteristics without behavioural feedback.

We consider a system of $N+1$ linear filters h receiving inputs x and producing outputs u. The filters connect with corresponding weights ρ to one output unit v (Fig. 1).

In the Introduction we have emphasised that all input lines of our algorithm are mathematically equivalent. It should be remembered, however, that *functionally* many times there are distinctive differences between them. As a consequence, we will use x_0 to denote the one unit which will later represent the reflex pathway. This has no mathematical consequences and is done only for convenience. The output v is then given as:

$$v = \rho_0 u_0 + \sum_{k=1}^{N} \rho_k u_k \tag{1}$$

Learning (weight change) takes place according to a differential Hebb rule (Eq. 2):

$$\frac{d}{dt}\rho_j = \mu u_j v' \qquad \mu \ll 1 \tag{2}$$

where the weight change depends on the correlation between u_j and the derivative of v. An extensive discussion how this rule relates to TD learning and to other differential Hebbian learning rules – as introduced by Klopf (1986, 1988) and Kosco (1986) – is given in the Discussion and in Appendix B. Here we only note that other differential Hebbian learning rules use filtering and derivates in different pathways as compared to ISO-learning (see Fig. 12 for circuit diagrams).

All weights can change (also ρ_0). The constant μ is adjusted such that all weight changes occur on a much longer time scale (i.e., very slowly) as compared to the decay of the responses u. Thereby the system operates in the steady state condition.

In general, the system which we consider shall operate in continuous time (e.g. with neuronal rate codes) and it shall be able to handle continuous input functions $x(t)$ of arbitrary shape.

The transfer function h shall be that of a *bandpass* which transforms a δ-pulse input into a damped oscillation (Fig. 2A) and is specified in the Laplace-domain:

$$h(t) \leftrightarrow H(s) = \frac{1}{(s+p)(s+p^*)} \tag{3}$$

where p^* represents the complex conjugate of the pole $p = a + ib$. It is important to note that such a bandpass is only stable if its pole-pair is located on the left complex half-plane, otherwise an amplified oscillation is obtained.

Real and imaginary parts of the poles are given by

$$a \;:= \mathrm{Re}(p) \;= -\pi f/Q \tag{4}$$

$$b \;:= \mathrm{Im}(p) \;= \sqrt{(2\pi f)^2 - a^2} \tag{5}$$

where f is the frequency of the oscillation. The damping characteristic of the resonator is reflected by $Q > 0.5$. Small values of Q lead to a strong damping.

The use of resonators (band-pass filters) is motivated by biology because oscillatory neuronal responses (Traub, 1999) and band-pass filtered response characteristics (at virtually all sensory front-ends, cell-membranes (Shepherd, 1990) and ion-channels like NMDA) are very prevalent in neuronal systems. Several examples for the utilisation of such bandpass filtered responses provide Grossberg and Schmajuk (1989) with their spectral timing model which they have used in different applications (Grossberg, 1995; Grossberg and Merrill, 1996).

Thus, the main idea is to use a neuron which gets bandpass filtered sensor signals at its inputs and generates a motor output. Later, one of these bandpasses (h_0) has the special task to provide the input for a reflex like reaction. The other bandpass filtered sensor signals are candidates for generating an earlier motor reaction through learning.

2.1 Analytical findings - Open-loop condition

2.1.1 Timing dependence of weight change

Here we address the question how the timing between the input signals influences the weight change.

In order to perform analytical calculations we will introduce two restrictions, which we will now use very often throughout the theoretical parts of this article. They will be waived later:

i) We will consider only two resonators, thus, $N = 1$.

ii) Accordingly we have to deal with only two input functions x_0, x_1 and we define them as (delayed) δ-pulses:

$$\begin{aligned} x_0(t) &= \delta(t - T), \qquad T \geq 0 && (6) \\ x_1(t) &= \delta(t) && (7) \end{aligned}$$

The first restriction is necessary because the analytical treatment of the case $N > 1$ is very intricate and largely impossible.

Concerning the second restriction we note that the theory of signal decomposition allows composing any causal input function from δ-pulses. Thus, the second constraint this is not really a restriction.

The delay T assures a well-defined causal relation between both inputs, where x_0 (the latter of the two) is the timing reference (the reflex input). Especially the section on the robot implementation will show that the algorithm (with $N > 1$) is very robust with respect to variations in T.

In general we use as an initial condition: $\rho_0 = 1$ and $\rho_1 = 0$.

For the analytical treatment we will only consider the weight change at ρ_1. (In fact, a little later we will show that the algorithm normally operates always in a domain where ρ_0 changes very little.)

Because we assume steady-state, we can rewrite the product in the learning rule (Eq. 2) as a correlation integral between input and output:

$$\begin{aligned} \rho_1 &\rightarrow \rho_1 + \Delta\rho_1 && (8) \\ \Delta\rho_1(T) &= \mu \int_0^\infty u_1(T+\tau)v'(\tau)d\tau && (9) \end{aligned}$$

Similar to other approaches (Oja, 1982) we compute the weight change for the initial development of the weights as soon as learning starts, because this is indicative of the continuation of the learning. Therefore, we assume $\rho_1(t) = 0$ for $t = 0$

and Eq. 9 turns into:

$$\rho_1(T)_{t=0} = \mu \int_0^\infty u_1(T+\tau)u_0'(\tau)d\tau \tag{10}$$

In simple cases (e.g., for $h_0 = h_1$) this integral can be solved directly. A general solution, which can also be extended to cover more than two inputs, requires to apply the Laplace transform using the notational convention: $x(t) \leftrightarrow X(s)$, for a transformation pair of functions in the time and the Laplace domain.

The linearity of our system allows solving the integral in Eq. 10 analytically, which is possible with the help of Plancherel's theorem (see the Appendix A for this rater unknown theorem). Applying it together with the shift theorem $x(t - t_0) \rightarrow X(s)e^{-t_0 s}$ to Eq. 10 we get:

$$\begin{aligned} \Delta\rho_1 &= \mu \frac{1}{2\pi} \int_{-\infty}^{+\infty} H_1(-i\omega) \left[i\omega e^{-Ti\omega} H_0(i\omega) \right] d\omega && (11) \\ &= \mu \frac{1}{2\pi} \int_{-\infty}^{+\infty} H_1(i\omega) \left[-i\omega e^{Ti\omega} H_0(-i\omega) \right] d\omega && (12) \end{aligned}$$

Note that symmetry of Plancherel's theorem is broken due to the exponential term. Equation 11 represents a Fourier transform and Eq. 12

its inverse. Both integrals can be evaluated with the method of residuals. Eq. 12, however, offers the advantage that we can neglect the right complex half plane, because it leads to contributions for negative time (i.e. $t < 0$) only (McGillem and Cooper, 1984; Stewart, 1960). Thus, of the four residuals (poles) for H_1 and H_0 only those of H_1 need to be considered because those of H_0 have flipped their sign in Eq. 12 and appear now on the right complex half-plane. We get as the final result:

$$\rho_1(T)_{t=0} = \mu \frac{b_1 M \cos(b_1 T) + (a_1 P + 2a_0 |p_1|^2) \sin(b_1 T)}{b_1 (P + 2a_1 a_0 + 2b_1 b_0)(P + 2a_1 a_0 - 2b_1 b_0)} e^{-T a_1} \quad T \geq 0 \tag{13}$$

$$\rho_1(T)_{t=0} = \mu \frac{b_0 M \cos(b_0 T) + (a_0 P + 2a_1 |p_0|^2) \sin(b_0 T)}{b_0 (P + 2a_0 a_1 + 2b_0 b_1)(P + 2a_0 a_1 - 2b_0 b_1)} e^{-T a_1} \quad T < 0 \tag{14}$$

where $M = |p_1|^2 - |p_0|^2$ and $P = |p_1|^2 + |p_0|^2$. If we assume identical resonators $H_0 = H_1 = H$, we get

$$\Delta\rho_1(T)_{t=0} = \mu \frac{1}{4ab} \sin(bT) e^{-aT} \tag{15}$$

which is identical to the impulse response of the resonator itself apart from a different scaling factor.

The corresponding weight change curves are plotted in Fig. 2b,c. The curves show that synaptic weights are strengthened if the presynaptic signal arrives before the postsynaptic signal and vice versa. The biological relevance of the learning curves becomes especially clear in the case $H_0 = H_1$. This learning curve with identical resonators is similar to the curves obtained in neuro-physiological experiments exploring spike timing dependent synaptic plasticity (STDP or "temporal Hebb") (Markram et al., 1997; Bi and Poo, 1998; Zhang et al., 1998; Abbott and Nelson, 2000; Fu et al., 2002)[2]. Furthermore we find for this case (Fig. 2b) that the location of the maximum of the learning curve T_{opt} falls in the interval:

$$\frac{\lambda}{2\pi} < T_{opt} < \frac{\lambda}{4}, \qquad \frac{1}{2} < Q < \infty \tag{16}$$

where $\lambda = 1/f$ is the wave-length of the resonator.

[2] In order to reproduce STDP in a biophysical model the signals x and u require a different interpretation involving NMDA-conductances and back-propagating action potentials. This the topic of a follow-up study which is currently in preparation (Saudargiene, Porr and Wörgötter in prep.).

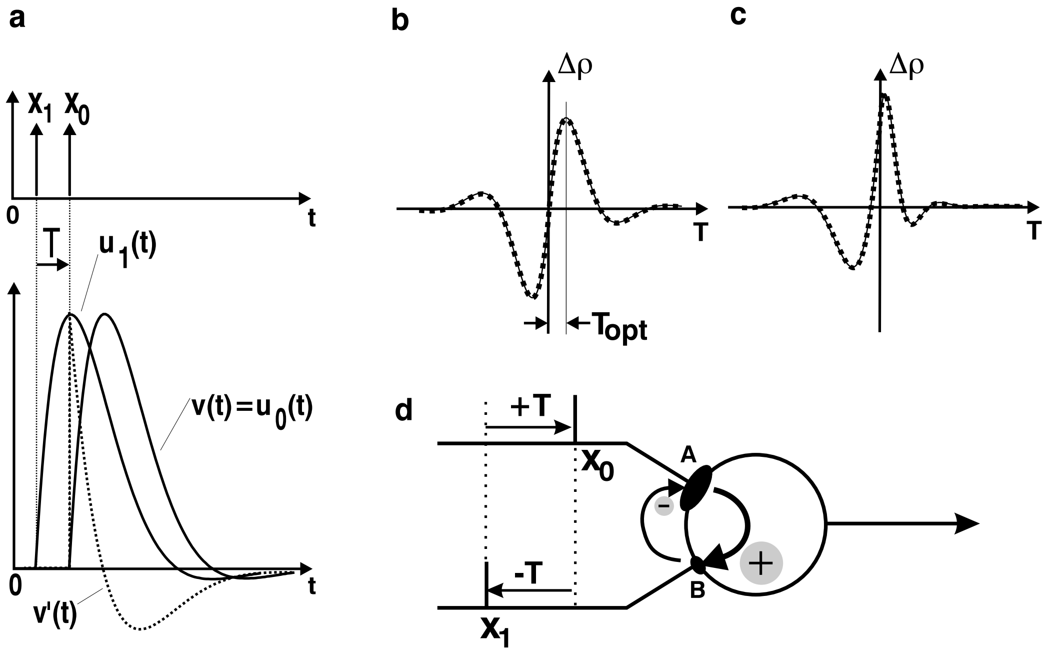

Figure 2: Input functions and the initial weight change for $t = 0$ according to Eqs. 13 and Eqs. 14. (a) shows the inputs x, the impulse responses u for a choice of two different resonators h and the derivative of the output v'. (b) shows the initial weight change $\rho_1(T)_{t=0}$ for $H_1 = H_0$, $Q = 1$, $f = 0.01$ (arbitrary units) and (c) for resonators with different frequencies $f_0 = 0.01$, $f_1 = 0.02$ but with the same $Q = 1$. The solid lines in (b) and (c) represent the analytical solutions derived from Eqs. 13/14 and the dotted lines simulation results resulting from the numerical integration of Eq. 9 with the same parameters for f and Q. For that purpose the two filters H_0 and H_1 get two different inputs $x_1(t) = \delta(t)$ and $x_0(t) = \delta(t - T)$. This pulse-sequence was repeated every 2000 time steps. After 400000 time steps the weight ρ was measured and plotted against the temporal difference T. The learning rate was set to $\mu = 0.001$. (d) Schematic explanation of the mutual weight change at a strong (A) and a weak synapse (B) with two subsequent delta pulses at the inputs x_1 and x_0 (for further explanations see text).

The isotropic setup of the algorithm in principle also leads to weight changes at ρ_0. It is, however, evident that the change in ρ_0 is (very) small when the contribution from the other inputs $\rho_k,\ k \geq 1$ is small. This is most easily seen when considering Fig. 2d which shows a situation which arises after some learning by using the standard initial conditions. The size of the synapses depicts the momentarily existing weight values. The input sequence is such that a weight increase arises at synapse B from the influence of input line A onto line B ($+T$ in learning curve), whereas weight decrease occurs at synapse A due to the inverse causal ($-T$) influence of input line B onto line A. The degree of change is depicted by the plus and minus signs, showing that the decrease of A is smaller than the increase of B. For two similar inputs a simple rule of thumb is that the weight-change $\Delta\rho$ roughly follows the weight value of *the other* input scaled by the learning rate μ, while the sign of the change depends on the temporal sequence of events:

$$\begin{aligned}\Delta\rho_{late\ input} &\approx \mu\ \rho_{early\ input} & (17)\\ \Delta\rho_{early\ input} &\approx -\mu\ \rho_{late\ input} & (18)\end{aligned}$$

As a result the strong input roughly maintains its strength while the contributions from the other inputs are small. This is the typical case when learning is guided by a strong reflex and the organism has the task to build up predictive pathways which should be weaker but more precise in order to prevent the disturbance.

We note that the above obtained analytical results can be extended to cover the most general system structure as represented in Fig. 1 with $N > 1$. Equation 1 turns into:

$$V(s) = \sum_{k=0}^{N} \rho_k U_k(s) \tag{19}$$

keeping it in the LAPLACE domain, because then we can directly obtain:

$$\Delta\rho_j(T) = \mu \frac{1}{2\pi} \int_{-\infty}^{+\infty} -i\omega V(-i\omega) U_j(i\omega) d\omega, \tag{20}$$

which is the general form of Eq. 9 in the LAPLACE domain. It should be noted that for all $\Delta\rho_j$ this integral can still be evaluated analytically in the same way as in the special case with two resonators discussed above. In the following equations we will use always use the index j for the input weights and k for the summation of the output-signal v.

2.1.2 Weight change when x_0 becomes zero

In this section we address the question of weight development when the reference input (reflex) becomes silent ($x_0 = 0$) at some point during learning. This is motivated by the cases discussed in the introduction, where the goal of learning is to avoid (late, painful, damaging) reflex reactions. Thus, setting

$x_0 = 0$ corresponds to the condition when the reflex has successfully been avoided. Note, that we are now left with just one input (x_1) asking if its synaptic weight will continue to change. This would correspond to a situation of *homo-synaptic learning* (e.g. homo-synaptic LTP, Guo-Quing and Poo (1998)). This section will show that our algorithm does not perform homo-synaptic learning. Instead the synaptic weight of x_1 stabilises as soon as $x_0 = 0$. Thus, ISO-learning is purely *hetero-synaptic learning*.

We use the same three restrictions (i-ii) as above and start with equation 20 inserting 19 into it. We set $x_0 = 0 \leftrightarrow X_0 = 0$ and the weight change becomes:

$$\Delta\rho_j = \mu \frac{1}{2\pi} \sum_{k=1}^{N} \rho_k \int_{-\infty}^{+\infty} -i\omega H_k(-i\omega) H_j(i\omega) d\omega \tag{21}$$

For $N = 1$ we get:

$$\begin{aligned} \Delta\rho_1 &= \mu \frac{1}{2\pi} \rho_1 \int_{-\infty}^{+\infty} -i\omega H_1(-i\omega) H_1(i\omega) d\omega & (22) \\ &= -\mu \frac{i}{2\pi} \rho_1 \int_{-\infty}^{+\infty} \omega |H_1(i\omega)|^2 d\omega & (23) \end{aligned}$$

$H_1(i\omega)H_1(-i\omega) = |H(i\omega)|^2$ is valid since the transfer functions can always expressed as products of complex conjugate pole-pairs. Multiplying $H_1(i\omega)$ with $H_1(-i\omega)$ leads to products of a complex number with its conjugate counterpart which renders its absolute value.

Since all transfer functions are symmetrical in relation to the real axis the frequency response $|H(i\omega)|^2$ is also symmetrical which leads to symmetrical responses in Eq. 23 at $|H_1(i\omega)|^2$. Due to ω in Eq. 23 the entire integral becomes anti-symmetrical and thus zero[3]. Thus, the weights stabilise if only x_1 is active.

This result can be summarised in a rather intuitive way: With $N = 1$ and $x_0 = 0$ there is an input signal only at x_1. The weight change in that case is a correlation of a damped sine wave with it's derivative which is a damped cosine wave. The correlation of a sine with a cosine is always zero.

[3] In a practical application (e.g. digital IIR filter) this is only true if the frequency responses of the input X_1 and the transfer function H_1 vanish for high frequencies to avoid that the integral becomes ill defined ($\infty - \infty$). In other words: the transfer functions must contain a low-pass term. This reflects the aspect that the time course of the input functions must be predictable (KALMAN filter property).

We have not attempted to calculate the behaviour of the weights for $N > 1$, which is very tedious if not impossible. Instead we will show simulation results for this later. However, the above argument can be extended by the Fourier theorem of wave decomposition to more inputs, because each sine wave from a resonator is multiplied by its cosine counterpart. Thus, we expect also for $N > 1$ a zero correlation and a stop of the weight development as soon as $x_0 = 0$.

2.2 Simulations - Open-loop condition

In this section we perform simulations with the neuronal circuit from Fig. 1. The simulations have the purpose to validate the theoretical results from the last section and to explore the more complex situations (especially $N > 1$) which are not analytically treatable.

The simulations were performed under LINUX on a Athlon processor using C++. The resonators were implemented as time-discrete IIR filters in the z-domain. We used the impulse invariant transformation from the s-plane to the z-plane and calculated the coefficients for the filters according to McGillem and Cooper (1984). We used normalised time steps resulting in normalised filter frequencies in the range $f = [0 \ldots 0.5]$. In all applications we used frequencies less or equal to $f_{\text{max}} 0.1$ in order to avoid sampling artifacts.

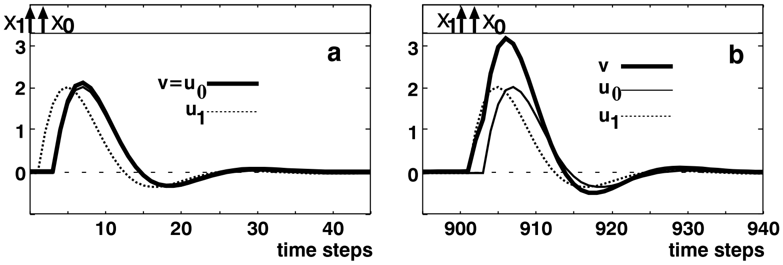

Figure 3: Simulation results with a circuit with two inputs, hence $N = 1$ (see Fig. 1). Input pulse sequences were repeated every 100 time-steps, the first starting at zero. Both resonators had values of $Q_{0,1} = 1$ and $f_{0,1} = 0.1$. The other parameters were $\mu = 0.01$ and $T = 2$. a) Result for $t = 0$, b) for $t = 900$.

2.2.1 One filter in the predictive pathway: N=1

As before, we begin with simplest case $N = 1$: one resonator in the reflex pathway x_0 and one resonator in the predictive pathway x_1 and use the same restrictions as above (i-ii).

Signal shape: Fig. 3a shows for $t = 0$ the δ-pulses at $x_{0,1}$ and the responses u_0 and u_1 from the resonators H_0 and H_1, respectively. Before learning the output v is identical to the signal u_0 because the weights were set to $\rho_0 = 1$ and $\rho_1 = 0$. The actual weight change of ρ_1 is caused by repeated pairing of the δ-pulses at x_0 and x_1. The result after 9 pairings is depicted in Fig. 3b. The comparison between Fig. 3a and Fig. 3b shows that the onset of the output v has shifted towards the earlier event x_1. Before learning it was identical to the resonator response u_0 in the reflex pathway. After learning the output is a superposition of both signals $u_{0,1}$ which leads to an onset which occurs together with the early onset of u_1. Thus, the circuit is able to "detect" the δ-pulse at x_1 as a predictor of the δ-pulse x_0.

Learning curve: Using the same setup we can vary the interval T and plot the change of ρ_1 in dependence of T for the initial learning step (i.e., for $t = 0$ after one correlation). This was simulated using identical resonators $H_0 = H_1$ but also with different resonators $H_0 \neq H_1$. The results are shown together with the analytical findings in Fig. 2b,c having used the same parameters in both the simulation and the analytical calculation. Thus, the analytically calculated weight change curves are reproduced by the simulation results.

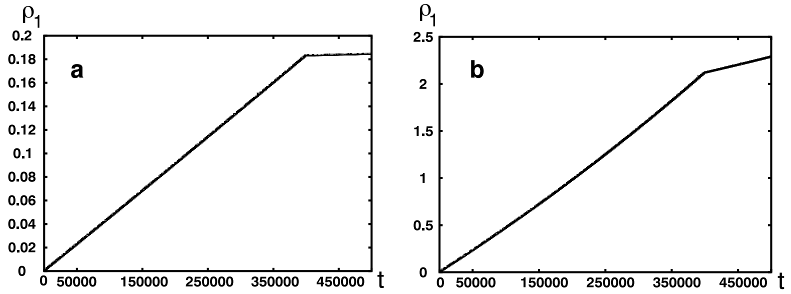

Figure 4: Simulated development of the weight ρ_1 for the case of two inputs ($N = 1$). Parameters were $f_{0,1} = 0.01$ and $Q_{0,1} = 1$. The inputs are triggered at a temporal difference of $T = 15$: $x_0 = \delta(t - T)$ and $x_1 = \delta(t)$. The pairing of the delta pulses is repeated every 2000 time steps. The learning rate is set to $\mu = 0.001$ in (a) and to $\mu = 0.01$ in (b).

Development of ρ_0: In all cases discussed so far both weights were allowed to change and substantial changes in ρ_1 were found for about 10-50 pairings, while we have claimed that ρ_0 remains stable. An easy intuition why this basically holds can be gained by using the "rule of thumb" defined above (Eq. 17,18). From this it is clear that the change of ρ_0 remains tiny for a prolonged time in our setup because ρ_1 equals zero at the beginning and μ is very small. In simulations we found that ρ_0 starts to change by more than 1% only after about 50000 learning steps when using a standard learning rate of $\mu = 0.001$ and $\rho_1 = 0,\ \rho_0 = 1$ as the usual initial conditions. Note that in the robot experiments, which will be shown below, the learning goal is reached after not more than 20 pairings. During this time the change in ρ_0 is minuscule.

Several other relevant cases could occur.

1. For example another interesting initial condition would be setting the weights to the same initial values (e.g. $\rho_0 = \rho_1 = 0.5$). This will still lead to a weight growth at ρ_1 (until about learning step 100000), but now ρ_0 will drop from the beginning. Functionally this could be interpreted as a situation where the reflex input becomes weaker, while the anticipatory pathway continues to take over. This could reflect a situation where the reflex has not been used for a long time, because then it is reasonable to allow the reflex to disappear leading to $\rho_0 = 0$. The only measure which has to taken is to stop the weight from changing its sign by keeping it finally at zero.

2. In conditions where the reflex saves the organism from life-threatening situations the weight ρ_0 can always be set to a fixed value.

3. In conditions where we have multiple synaptic weights of similar strength (i.e., $N > 1$), we can expect that the system's development will be dominated by stimulus-sequence induced symmetry breaking effects. This can lead to rather complex patterns which would require a more detailed analysis, which is beyond the scope of this article.

Weight stabilisation for $x_0 = 0$: The analytical results (Eq. 22) predict that ρ_1 should stabilise as soon as $x_0 = 0$. This, however, also requires that the learning rate μ is zero, which in reality cannot be ultimately achieved. The following simulation results show the effect of the learning rate on the development of the weights and compare the analytically obtained result with those obtained for more realistic situations. The simulation to test this was performed the following way: first we triggered the two resonators with paired δ-pulses. Then the input x_0 was switched off (i.e.: $x_0 = 0$) at $t = 400,000$ and only the input x_1 was still active.

Fig. 4 shows the weight development of ρ_1 over time for two different learning rates μ. With a low learning rate the weight ρ_1 approximately stabilises when the input x_0 is switched off (see Fig 4a) whereas with a higher learning rate the weight continues to grow. Weight stabilisation can be very desirable during learning but so is a high learning rate. These conflicting demands therefore lead to a trade-off, which needs to be taken care of in practical applications (like the robot application later in the text).

2.2.2 More than one filter in the predictive pathway

The setup with only one resonator ($N = 1$) in the predictive pathway has the disadvantage that there is only one specific temporal interval T_{opt} where learning (weight change) is at the maximal rate. The use of an array of resonators with different frequencies in the predictive pathway removes this disadvantage (see inset in Fig. 5). The system should now be able to learn more than only one time interval properly. We have set up such a system with an array of 10 resonators in the predictive pathway. We triggered this array with the same δ-pulse ($x_1 = \delta(t)$). The reflex pathway was triggered by a delayed δ-pulse ($x_0 = \delta(t - T)$; $T = 10$). The initial condition for learning was set to $\rho_0 = 1$; $\rho_k = 0$; $k \geq 1$ as before.

Signal shape: Fig. 5 shows the resonator responses u_k scaled with their momentarily existing weights ρ_k (top) at time $t = 390,000$ during learning. The scaled response of u_0 (a, dashed line) is still the biggest at this time. The diagram also shows the output signal v and its derivative during the learning process (also t=390,000, bottom). Additionally, the output signal is shown which is generated when silencing the input x_0 (c, dotted line, bottom, t=400,000).

The output v is a superposition of all resonator outputs. It can be seen that it has a first and a second maximum (marked with 1 and 2 in Fig. 5). The second maximum is due to the resonator response from the reflex pathway u_0 and vanishes when the input x_0 is switched off (see dotted curve in c).

The first maximum is generated by superposition of the responses $\rho_k u_k, k > 0$ (i.e. all except u_0). In general we have observed that this superposition

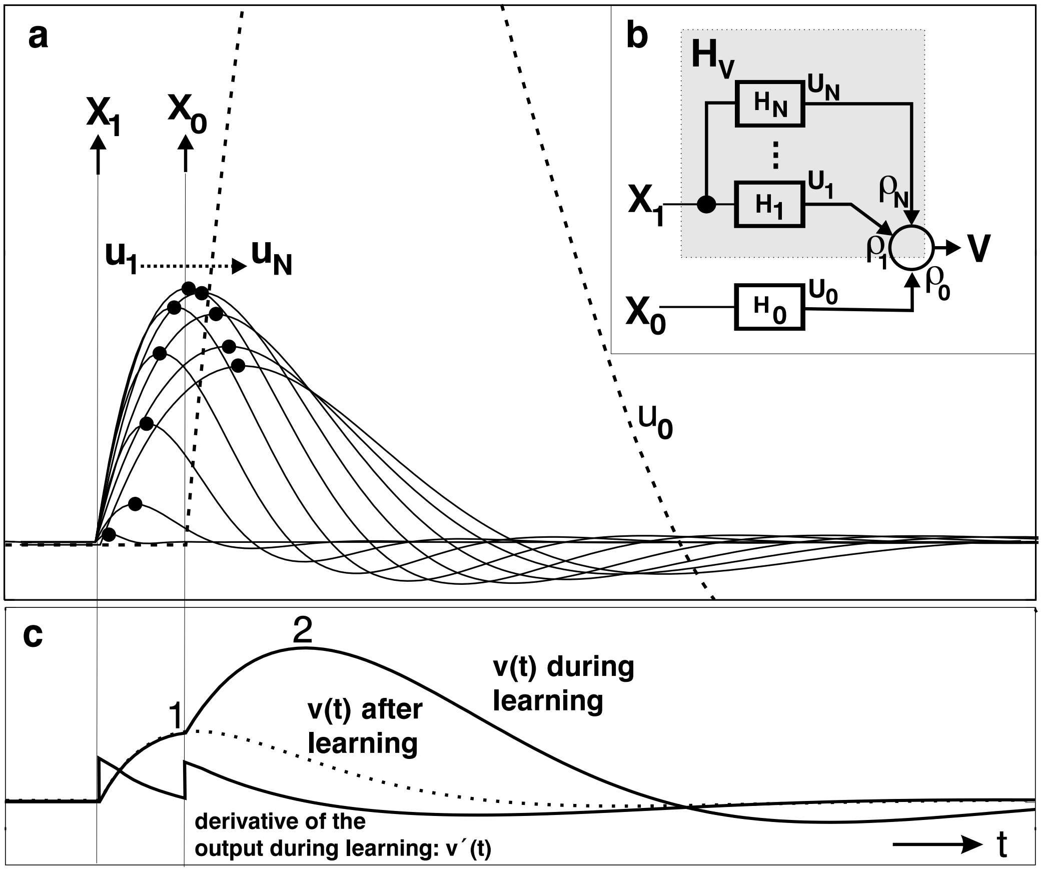

Figure 5: Multiple filters ($N = 10$) in the predictive pathway: Filter responses (a), the neuronal circuit (b) and its output during learning and after learning (c). The neuronal circuit (b) consists of a filter bank where the filter frequencies are set to $f_k = \frac{5f_0}{k}$; $k \geq 1$ and $f_0 = 0.01$. The learning rate was set to $\mu = 0.0005$ and $Q = 1$. The filter bank gets two different inputs $x_0(t) = \delta(t)$ (reflex-pathway) and $x_1(t) = \delta(t - T)$ (predictive pathway), $T = 10$. The delta pulses are repeated every 2000 time steps. After the 400,000 time steps x_0 is set to zero. The contribution of the signals $u_k\rho_k$ to the output v triggered by $x_1(t)$ is called H_V and is marked by the shaded box in (b). The weighted resonator responses $\rho_k u_k$ after learning are shown in (a). The output signals during learning (time step 390000) and after learning (after time step 400000) are shown in (c).

process will always try to generate the first maximum as close as possible to x_0. This can be understood by the ongoing amplification of an initially existing asymmetry in the system in the following way. At the first learning step the derivative of v is zero before x_0 and then follows the shape of the v'-curve as shown in the diagram. Thus, there is one resonator response whose shape matches the v'-curve best (best positive correlation). Obviously, it is that particular resonator which has its maximum at (or closest to) the maximum of the v'-curve (second cusp, first is still zero). For this resonator we get the highest correlation result (Eq. 9) and, thus, the strongest weight-growth at the beginning of learning. The other weights grow less strong and their growth rate is approximately (inversely) related to the distance of their resonator maximum from x_0. As a consequence we get a distribution of weight values which follows the shape outlined by the y-position of the resonator maxima as shown in the top panel by the dots on the curves. Superposition of these weighted responses, thus, leads to a maximum at of v at x_0. This line of argumentation continues to hold also for the following learning steps, because the theoretical results suggest that the contribution of the correlation of the first part of the v'-curve (first cusp) with the $u_k, k > 0$, which would correspond to homo-synaptic learning, is zero in all cases (see Eq. 21-23) thereby not affecting the weight change. Thus weight change continues to follow the distribution of the maxima in Fig. 5a. The resonator with the lowest frequency (f_l) determines the longest delay $T_{max} = \frac{1}{f_l}$ which can be learned. Equivalently the shortest delay is $T_{min} = \frac{1}{f_h}$ where f_h is the resonator with the highest frequency. Within the range $[T_{min}, T_{max}]$ any T causes an output with a maximum which always coincides with the location of x_0, provided there are enough resonators to allow for a sufficiently accurate superposition process.

Learning curve: As in the case of only two resonators; the dependence of the weight change on the temporal distance T can be explored. Now, however, we have to monitor N changeable weights. For this experiment, we have chosen the same standard setup using paired δ-pulses with a temporal delay of T, but now we use 15 resonators ($N = 15$) in the predictive pathway. Their frequencies are chosen such that 10 resonators have a frequency which is higher and 5 resonators one which is lower than f_0 (see Fig. 6). Every second weight change curve is shown in Fig. 6 for $t = 0$ where we varied T from -150 to 150. Every curve in this diagram represents one weight ρ_k of a specific resonator h_k in dependence of T. The curve plotted with the thick line belongs to the resonator h_k which has the same frequency like the resonator h_0, hence $f_k = f_0$. The other weight change curves belong to resonators in the predictive pathway which have different frequencies compared to f_0. It can be seen that every weight change curve has a specific T where weight

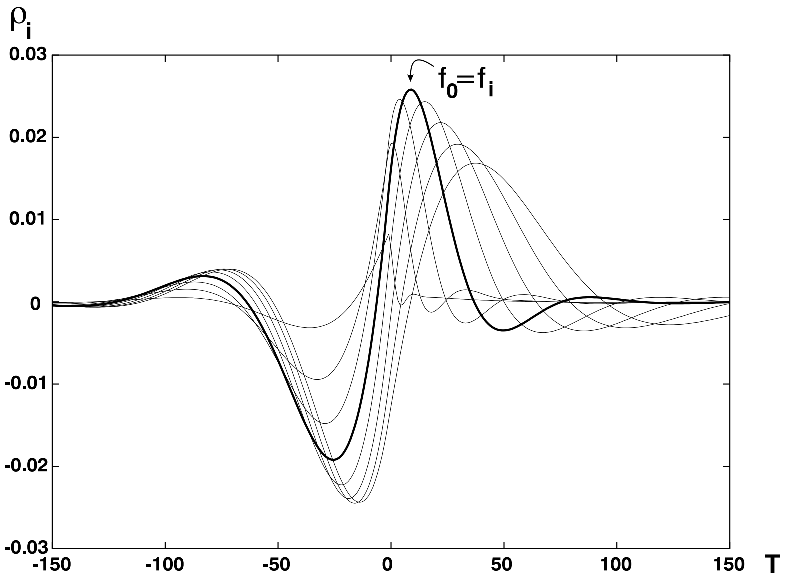

Figure 6: Weight changes ρ_j dependent of the temporal distance T with a filter bank of resonators ($N = 15$) set up as in Fig. 5b. The filter frequencies are set to $f_k = \frac{5f_0}{k}$; $k \geq 1$ with $f_0 = 0.01$ and $Q = 1$. The learning rate was set to $\mu = 0.0001$ and $Q = 1$. The case $f_0 = f_k$ is marked with a thick line and reproduces the curve in Fig.2b. The filter bank gets two different inputs $x_1(t) = \delta(t)$ (predictive pathway) and $x_0(t) = \delta(t - T)$ (reflex pathway). The delta pulses are repeated every 2000 time steps. After the 400,000th time step the weight ρ_j was measured and plotted against to the temporal difference T. Only every second curve is plotted.

change is maximal or (in support of the argument used to explain the first maximum in Fig.5) the other way round: for specific values of T and large N there exists always one particular resonator which shows maximum weight change.

Another interesting result is that the weight change curve with $f_k = f_0$ is identical to the weight change curve with only one resonator (see Fig. 6). The fact that both weight change curves are the same is due to the linearity of our model.

In summary, in an array of different resonators every resonator is only responsible for a specific and limited range of temporal intervals so that such an array is able to cover a wide range of different temporal intervals. The weight change curves for the different weights give precise information which resonator yields the maximum contribution to the output signal.

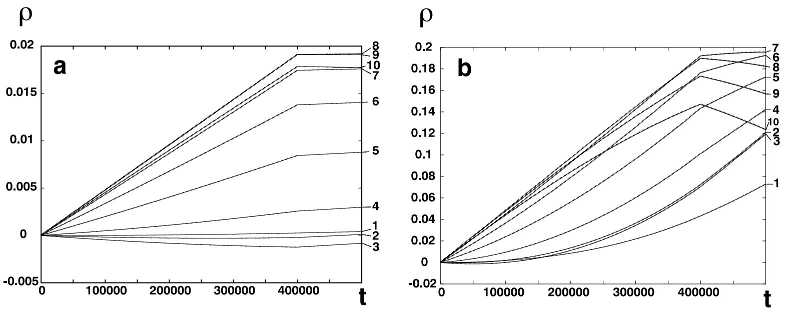

Figure 7: Weight change of multiple resonators $N = 10$ in dependence of the learning rate. The neuronal circuit (see Fig. 5b) consists of a filter bank where the filter frequencies are set to $f_k = \frac{0.1}{k}$; $k \geq 1$ where the index k is also used as a label for the different curves in this figure ($Q = 1$ in both cases). The filter bank gets two different inputs $x_1(t) = \delta(t)$ (predictive pathway) and $x_0(t) = \delta(t - T)$ (reflex pathway) with $T = 10$. The delta pulses are repeated every $2000th$ time step. After $400,000$ time steps x_0 was set to zero. The learning rate was set to $\mu = 0.0001$ in (a) and to $\mu = 0.001$ in (b).

Weight stabilisation for $x_0 = 0$: Next we ask whether the weights also stabilise in a *multi*-resonator setup if the reflex pathway x_0 becomes zero (compare to Fig. 7). We use the same setup as before ($N = 10$ and paired δ-pulses with $T = 10$). The test was performed in the same way as above by setting x_0 to zero at time $t = 400,000$. Fig. 7 shows that the weights stabilise in the limit of $\mu \rightarrow 0$. Thus, we find again that the crucial parameter for an approximate weight stabilisation is the learning rate μ, which is too high in b.

Because of the complexity of the mathematics, above we were not able to give robust analytical arguments for weight stabilisation in the multi-resonator case. We could only argue that the individual resonator responses (sine-waves) should be orthogonal to the derivative of the output (cosine wave) as soon as $x_0 = 0$, (see dashed curve in Fig. 6) leading to zero value of the correlation integral. The experimental findings in Fig. 7 support this notion. Thus, also in the multi-resonator case we obtain the desired property of weight stabilisation in the limit of $\mu \rightarrow 0$. Homo-synaptic learning does not take place even with more than two resonators in total.

Let us in the context of $N > 1$ also briefly consider more than one predictive pathway with several sensors that operate independently. In this case, the isotropy of the algorithm leads to the situation that learning will continue *between* those sensor-inputs even after the reference (reflex) input x_0

has become silent. Weight changes of the other (non-reflex) weights, however, will normally remain small because after learning the absolute values of them are small which leads only to minor cross influences as will be shown in the robot example below (Fig. 11). Thus, even in such a situation weight will be (approximately) stable.

Another stabilising factor arises if we place the learning algorithm in a closed behavioural loop (see also next section). In the closed loop paradigm analysed in the second study (Porr et al., 2002) we found that a perturbation of the weight ρ_1 – which disturbs the final condition $x_0 = 0$ – leads to a counterforce which reestablishes the original weight. Taken together these arguments show that weights might indeed undergo small drifts after removal of the reflex, but these drifts do not lead to a divergence. This is supported by the robot experiments, where we never observed weight divergence even after prolonged runs.

3 Closed Loop: The Robot experiment

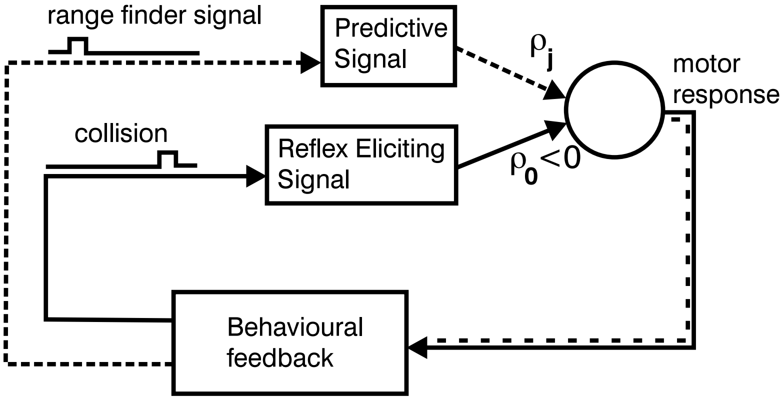

Figure 8: Simple sensor motor feedback with prediction which is make explicit with the example of collision avoidance. The solid lines depict a pre-wired reflex loop which exists before learning. This reflex loop performs a reflex reaction — in this case a retraction reaction (motor response) when the collision sensor (reflex eliciting signal) has been triggered. Learning has the task to learn that the earlier range finder signal (predictive signal, dashed pathway) can be used to generate an earlier motor reaction to prevent the collision (reflex).

The task in this robot experiment is collision avoidance. The built in reflex behaviour is a retraction reaction after the robot has hit an obstacle (Fig. 8, solid pathway). This represents a typical feedback mechanism with the desired state that the signal at the collision sensor should remain zero. In order to prevent that the robot leaves the desired state it can use other sensor modalities which can *predict* a looming collision. In our case this is achieved with range finders (Fig. 8, dashed pathway). The learning algorithm has the task to learn the existing temporal correlation between the range finder- and the collision sensor signals. After learning the robot can generate a motor reaction already in response to the range finder signals and thereby avoid the retraction reflex. Functionally the reflex will be eliminated and the "predictive pathway" takes over after learning.

Up to this point the algorithm had been treated in a pure open-loop condition, where learning was entirely unsupervised. The robot experiments shown below create a situation where the behavioural reaction influences the sensor inputs, thereby creating a closed-loop situation (Fig. 8). Unsupervised learning thereby turns into something which we would call "self-referenced" learning in order to distinguish it from "reinforcement" learning which requires an explicitly defined reference signal (punishment or reward), which is not present in closed loop ISO-learning. The theoretical treatment of this situation in the second article (Porr et al., 2002) will clarify that these two situations are fundamentally different.

The robot's circuit diagram is shown in Fig. 9; a detailed description, which includes a list of the robot's control parameters is given in Appendix C. The robot has three collision sensors and two range finders. All signals are filtered by band pass filters and converge onto two neurons which generate two different motor outputs: one controls the robot's speed and the other the robot's steering angle. The speed of the robot is set to a fixed value and its steering to zero so that the undisturbed robot drives straight forward. The built in retraction behaviour is generated by the dotted pathways where the collision sensors trigger highly damped sine waves in the corresponding resonators. This signal is sign-inverted and directly transmitted to the motors. Essentially, it consists of just one single half wave which, leads to the retraction reaction. The weights are initially set to minus one and effectively do not change during learning so that the retraction behaviour remains always the same. The dotted collision sensor pathways with their strong weights which determine the motor output are together with the arising behavioural feedback equivalent to the reflex loop discussed in Fig. 8.

The range finder signals (solid lines) react at a distance of about 15 cm

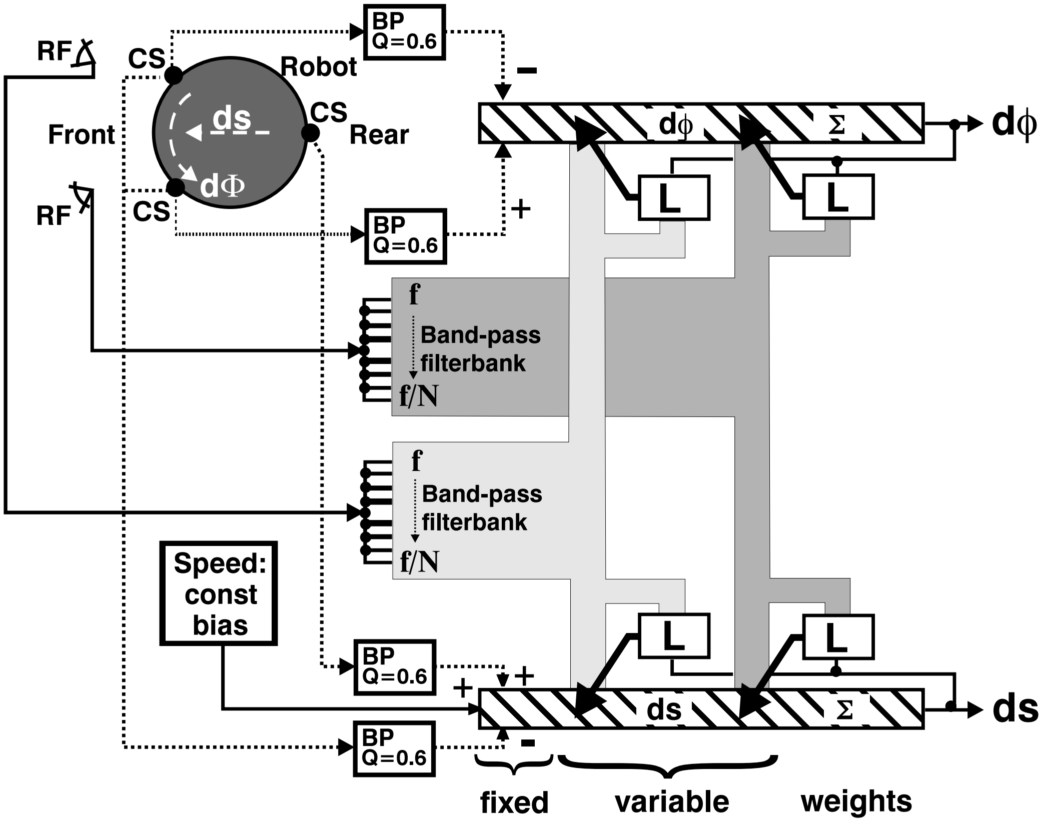

Figure 9: Robot Circuit: The robot consists of three collision sensors (CS), two range finders (RF) and two output neurons for speed (ds) and steering angle ($d\phi$). These output neurons represent simple simple summation circuits (indicated by $\sum$). The robot has a reflex behaviour which is established by the signals from the collision sensors (dotted lines) which are fed into 4 band pass filters H_0 with $f_0 = 1Hz$ and $Q_0 = 0.6$. The output of the band pass filters is summed at the neurons for speed (ds) and steering angle ($d\phi$). The corresponding weights are adjusted in such a way that the robot performs an appropriate retraction reaction if either of the collision sensors is triggered. The task of learning is to use the signals from the range finders (RF) to predict the trigger of the collision sensor (CS). The two signals from the left and the right range finder are fed into two filter-banks with $N = 10$ resonators with frequencies of $f_k = \frac{1Hz}{k}$; $k \geq 1$ and $Q = 1$ throughout. The 20 signals from the two filter bank converge on both the speed neuron and on the neuron responsible for the steering angle. Learning rate was $\mu = 0.00002$. L depicts the implementation of the learning rule (Eq. 2).

from an obstacle and are therefore able to predict a collision. However, the temporal delay between the range finder signal and the collision signal is variable and depends on the actual motion trajectory of the robot. In order to cope with a rather wide range of temporal delays we used the same approach as in section 2.2.2 and implemented two resonator filter-banks which get their signals from the two range finders. Filter banks consist of 10 resonators covering approximally a temporal interval between 50ms and 500ms. These resonator signals converge onto both the speed- and the steering neuron. Their weights are initially set to zero.

Depending on the initial conditions, different solutions were found by the robot to avoid obstacles. One solution, for example, is that the robot after learning simply stops in front of an obstacle or that it slightly oscillates back and forth. This type of behaviour may look trivial but is entirely compatible with the learning goal avoiding obstacles. More commonly we observed a different type of solution where the robot continuously drives around and uses mainly his steering to generate avoidance movements. Other solutions do not seem to be possible and have not been observed. Furthermore, we observed that the robot always found one of these solutions after sufficiently long learning.

Fig. 10 shows episodes of the robot behaviour and its signals for one selected example trajectory. The signals shown in Fig. 10c,d corresponds to a situation where the robot still collides with the walls. Corresponding collision points are marked in Fig. 10a by small letters c and d. As expected, learning leads to a change of the temporal relation between the range finder signal and the collision signal. This can be seen by the different lengths of T depicted in Fig. 10c,d and is due to the learned motor output which is increasingly dominated by the range finder signal. This supports the filter bank approach which we have used in the robot experiment. Finally, Fig. 10e depicts a situation where the robot has learned to avoid the obstacles ($CS = 0$).

Note that the low pass component of the band pass filters smoothes the rater noisy range finder signals which substantially adds to the robustness of the algorithm. Furthermore, pure noise signals are not correlated to other sensor signals and do not contribute to learning.

The change of the weights in the robot example shall now be compared with the results from the simulations. We find that the weights approximately stabilise also in our robot experiments (Fig. 11). Their actually values depend on the solution found. The situation in the robot experiment, however, is more complicated than in the simulations shown earlier, because the $ds-$ and $d\phi-$neurons get signals from more than two sensors at the same time. Thus, very often triplets of temporal correlations exist, like during a slanted

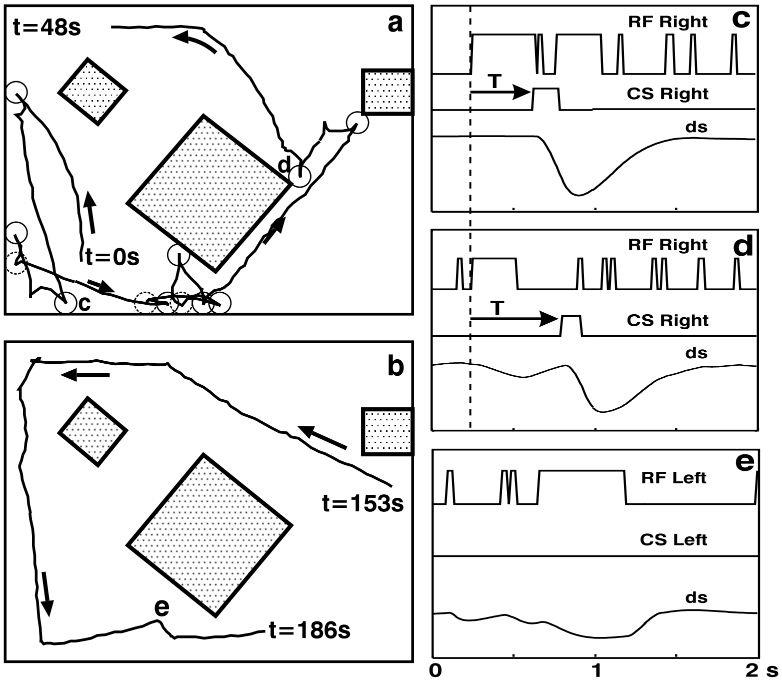

Figure 10: a) Manually reconstructed robot movement trace in an arena (240 $cm \times 200$ cm) with three obstacles (shaded) at the onset of learning. Motors were not entirely balanced leading to a curved start of the trajectory. Many collisions (circles, solid=forward-, dashed=backward collision) occur and trapping at obstacles happens. After a collision a fast reflex-like retraction&turning reaction is elicited. b) Robot movement trace after successful learning of the temporal correlation between signals at RF and CS. No more collisions occur, the trajectory is smooth. A complete movie of this trial can be viewed at `http://www.cn.stir.ac.uk/predictor/real` *— movie 1. c-e) Signals at RF (top), CS (middle), and motor control signal ds (bottom) for different learning stages. c) Signals occuring at the early collision marked 'c' in part a of this figure. A stereotyped motor reaction is elicited in response to the CS signal. d) Signals occurring at the late collision 'd'. Motor reactions occur in response to RF but are not sufficient to avoid the collision. When it occurs a strong motor reaction is again elicited. e) Signals occurring at the curve marked 'e' in (b). Smooth motor reactions occur in response to RF, CS remains silent because no collision occurs.*

wall approach we obtain first a signal from the right, then one from the left range finder and finally that from the right collision sensor. After successful learning the collision sensor remains silent but we are still left with sequences of ranger finder events. Thus, learning continues, though at a smaller rate even after the last collision has happened.

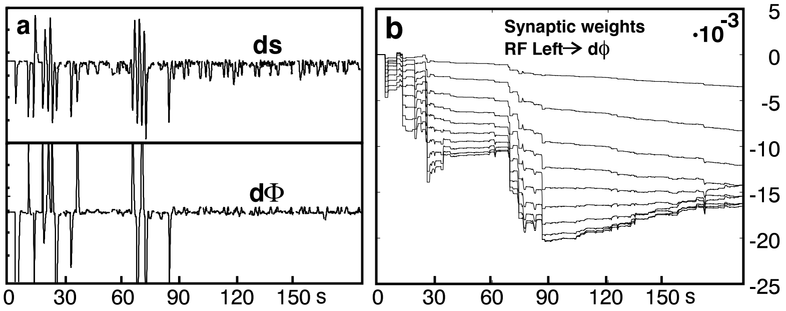

Figure 11: a) Complete motor signal-traces for ds and $d\phi$ and b) development of the synaptic weights for the same trial as in Fig. 10.

As a central observation this shows that our system continues to operate *without* a designated reference signal (because x_0 is zero now). Learning continues between the remaining inputs.

This can, for example, be seen in Fig. 11 when looking at the development of the weight from the left range finder to $d\phi$ which continues to change after the last collision has occurred (at $t = 85s$). Ultimately, the earlier of the two range finder signals would dominate, but this will lead to a stable situation only for very simple (e.g., circular) trajectories where an unchanging relation between both range finder signals is forced upon the robot.

An equivalent reward-retrieval situation has also been simulated. These results shall not be presented here in order to limit the size of this report, but can be viewed at `http://www.cn.stir.ac.uk/predictor/animat/`.

4 Discussion

In this study we have developed an isotropic algorithm for sequence order learning (ISO-learning) in which learning relies only on the temporal order of its inputs. This has the advantage that all input signals are treated equally and that learning takes place between all of them. Thus, it represents a form of unsupervised sequence order learning.

4.1 Basic properties

ISO-learning is only driven by the temporal relation between pre- and post-synaptic signals. As a consequence our learning rule is related to learning based on spike timing dependent plasticity (STDP) (Gerstner et al., 1997; Markram et al., 1997; Zhang et al., 1998; Bi and Poo, 1998; Roberts, 1999; Xie and Seung, 2000; Kistler and van Hemmen, 2000; Song et al., 2000; Song and Abbott, 2001; Fu et al., 2002), but our algorithm uses time-continuous functions and not spike-trains as input signals. The measured curves for STDP are based on the relation between individual (pre- and post-synaptic) spikes. Curves with different characteristic shapes have been observed (Abbott and Nelson, 2000), such as that similar to our Fig. 2, but also (e.g.) inverted versions of it have been measured. Thus, the question arises how these curves relate to the average firing rate of the neurons (Gerstner et al., 1997)? This question was specifically addressed in the studies of Roberts (1999) and Xie and Seung (2000) and they found that the temporal derivative of the postsynaptic impulse rate directly relates to the temporal hebbian STDP curve, or with sign inversion to the anti-hebbian curve. This shows that a computational link exists between rate-based ISO-learning and the spike based STDP results because we found that the hebbian STDP curve will be obtained analytically by integrating the ISO-learning rule over time (see Eqs. 10-14).

In the second part of this study we have introduced a closed loop situation by means of behavioural feedback. We implemented a primary reflex loop which is distinguished from all other inputs only by the fact that it initially carries the largest synaptic weight. In general, such closed loop reflex loop situations have the disadvantage that any *re*-action will only occur *after* an incoming sensor event. This inherent disadvantage of feedback loops leads to a general objective for improving animal behaviour which is to find a mechanism which prevents the reflex (Palm, 2000; Wolpert and Ghahramani, 2000). Sequence order learning can achieve this by creating earlier, anticipatory actions. In addition, we have shown that weights stabilise as soon as the reflex has been successfully avoided. Due to the isotropy of the inputs any other input line can take on the role of the reference signal during learning and the initial reflex can even be unlearned or reduced in strength - a situation which is observed in many physiological reflexes.

4.2 Practical aspects

A convenient aspect of ISO-learning which leads to a very limited computational effort is the use of IIR-filters in our approach. With such filters it is possible to generate a smooth and long-lasting response already when using only two resonators. Such a response can bridge already a very long temporal difference T and is therefore able to generate a basic predictive reaction. Additional filters contribute to a more and more *precise timing*. Thus the basic temporal correlation between X_1 and X_0 can established by one filter ($N = 1$) and then improved by adding more and more filters. In other approaches (Sutton and Barto, 1981, 1982; Klopf, 1988) often delay-lines are used for the predictive input X_1. The discrete structure of these algorithms requires many more delay elements as compared to the analogue operating ISO-learning because they need a delay element for every unit time-step. Thus, the computational effort is much higher if a broad temporal range has to be covered.

4.3 Evaluative versus non-evaluative models

A fundamental difference exists between reference-based (reward-based) algorithms (e.g. TD-learning) and the so-called drive reinforcement algorithms such as differential hebbian learning and ISO-learning.

Probably the most influential method for reference-based temporal sequence learning is the temporal difference (TD) learning algorithm Sutton (1988). TD-learning has the goal to generate an output v which predicts a reference (reward r) by the help of its (sensorial) input signals x. This goal is achieved by minimising a prediction error δ between reference and output. Thus, learning relies on the predefined reference which acts like a teacher signal in supervised learning.

The direct comparison between the two algorithms shows that (as mentioned before) the reference- (reward-) pathway and the error calculation of TD learning is replaced by the reflex-pathway in our algorithm. Mathematically the reflex pathway is not distinct from the other pathways in ISO-learning, functionally; however, it drives the output with an initially strong weight: as described in section 2.2.1, the strongest input dominates the learning behaviour of the *other* inputs/weights. Klopf (1988) called this "drive reinforcement learning". In Appendix B we will compare the different drive reinforcement models that exist in the literature with each other. Here we just note that our algorithm therefore belongs to the class of pure *unsupervised* learning algorithms opposed to TD learning which is *supervised* using the reference (reward) as teaching signal. The initially strong weight in the reflex pathway of ISO-learning can be interpreted as a boundary condition preventing the output from becoming arbitrary. Introducing boundary conditions is typical practice of unsupervised, especially, Hebbian learning (Miller, 1996).

The structural differences of our learning algorithm and TD learning suggest different neuronal substrates. The TD learning circuit consists of two different components: The predictive circuit and the error-signal circuit. Usually these two circuits are identified with different neuronal subsystems: the error circuit with the dopamine system and the predictive circuit with cortical or other dopamine modulated brain areas. Strong evidence exist supporting this by the work of Schultz et al. (1997) and it is also known that reward based learning plays a substantial role in animal behaviour such as during instrumental (operand) conditioning paradigms and during action planning (Dayan and Abbott, 2001).

Our algorithm, on the other hand, suggests only one neuronal circuit because all pathways are equivalent as supported by Hauber et al. (2001). It is conceivable that such a system coexists to the reward based learning system(s), because in an autonomous agent any reward based system needs to be bootstrapped by "first correlative experiences", such as those used by our system to drive learning.

In the one-circuit scenario, our learning rule, based on temporal relation between pre- and post-synaptic signals, would have to be represented by internal neuronal variables like NMDA-dynamics or Ca^{2+} concentration. The direct relation of our learning rule with the shape of the spike-time dependent plasticity (STDP) curves (Fig.2b) indicates that it should be relatively straight-forward to redesign our model into a biophysically more realistic one, which directly relies on such internal neuronal variables and which uses spike trains as inputs. This has recently been attempted by Rao and Sejnowski (2001) using the TD-learning algorithm but the relation between TD-learning and STDP is less direct and, accordingly, the transition between those two models is bit more intricate (Dayan, 2002).

When talking to specialists in the field of temporal sequence learning we were asked to also explain to what degree our learning rule is different to the one used in TD-learning. This aspect is quite technical and we refer the reader to Appendix B for an in-depth discussion.

4.4 Closed loop condition

Hebbian learning rules like the one used here belong to the class of unsupervised learning rules. Unsupervised learning seems to be the obvious choice for creating the first and earliest stages of autonomous behaviour, because it does not require external (teacher-like) knowledge. Instead it purely relies on self-organisation based on the correlation structure of the inputs. Such unguided self-organisation processes, however, can also lead to a situation where nonsensical correlations are learned leading in the end to an undesired network behaviour. The standard solution to avoid this problem is the introduction of boundary conditions which keep the self-organisation process within sensible margins. In practise this is either done heuristically by the network designer, or, as a better choice, boundary conditions are introduced such that they intrinsically (and in a natural way) represent the structure of the problem to which the self-organisation process is applied.

In the case of our unsupervised temporal sequence learning algorithm the same is achieved by embedding the learning circuit in an environment which leads to a closed loop situation. The causal relation which naturally exists between many different pairs of sensor events (pain follows heat, taste follow smell, etc.) as described in the introduction creates an implicit boundary condition for our algorithm by using the latest incoming event (the one which drives the reflex) as the temporal reference for learning. The environment has two properties in our model: it provides feedback and it contains disturbances, but very clearly it does not provide any reward or any other teaching signal. Klopf (1988) called this feedback loop ”non-evaluative” since there is nothing in the environment which evaluates the organism's performance. Instead, here ISO-learning becomes *self-referenced* (von Foerster, 1960; Maturana and Varela, 1980): the actions of the learner influence its own learning without any evaluation process.

Robotics is the discipline which can clarify the concepts of autonomous behaviour and interaction with a complex environment quite naturally (Brooks, 1997). In the field of temporal sequence learning Vershure has been working for over 10 years in using robot applications (Verschure and Pfeifer, 1992; Verschure and Voegtlin, 1998). In his words every organism undergoes tree steps of development: pre-wired reflex (fixed connections), adaptive control (classical Hebbian learning of sequences of sensor inputs) and reflective, contextual control (goal oriented learning). In Vershure's terminology adaptive control has no goals but builds up temporal associations with “proximal” and “distal” sensors. At the stage of the reflective control a goal is introduced in the form of a reward or punishment when, for example, that an object has successfully been found.

Our study shows that this distinction may be too rigid. The behavioural pattern, observed in our robot, seems to be punishment-guided, which would place it at the advanced level of reflective control. The unsupervised, non-evaluative, but self-referenced structure of the robot's interaction with the environment, however, places it at the more simple level of adaptive control. This shows that autonomous agents can develop rather complex behavioural patterns by means of simple nested feedback loop systems, without having to evaluate their own behaviour. Of course, we would not argue against the importance of higher learning schemes and it is also quite sensible to distinguish between increasingly higher levels starting with non-evaluative schemes, which are surpassed by evaluative (reward/punishment) schemes, which in turn are followed up by contextual learning and finally by the different stages of cognitive learning. But it seems advisable to treat these different stages in a less separatist way allowing for a broader transition range between them.

In the accompanying article (Porr et al., 2002) we will derive a theoretical treatment of the closed loop ISO-learning situation. We will show analytically that the predictive pathway learns to approximate the inverse controller of the reflex pathway thereby creating a forward-model of the control situation.

Acknowledgements

We are grateful to Christian von Ferber, Leslie Smith, Richard Reeve and in general to the members of the CCCN-seminar for their helpful comments during various stages of this work. This study was supported by grants from SHEFC RDG "INCITE" and by the European funding "ECOVISION".

A Plancherel's theorem

This theorem is rather unknown, therefore we state it here as:

$$\begin{aligned}\int_0^\infty f_1(t)f_2(t)dt &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} F_1(i\omega)F_2(-i\omega)d\omega \qquad (24)\\ &= \frac{1}{2\pi}\int_{-\infty}^{+\infty} F_1(-i\omega)F_2(i\omega)d\omega \qquad (25)\end{aligned}$$

where F is the Laplace transform of f (Stewart, 1960). If we set $f_1 = f_2 = f$ it becomes the more commonly used theorem of Parseval.

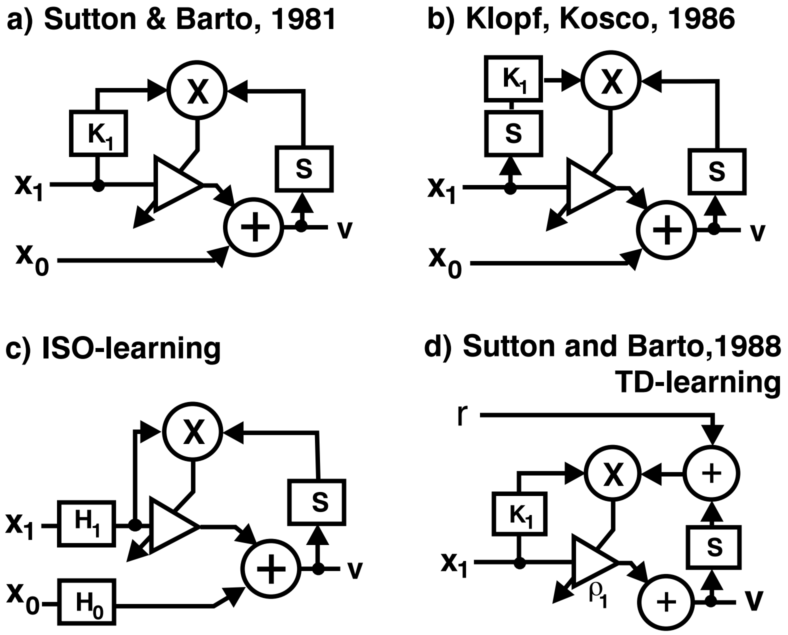

Figure 12: Comparison of three drive reinforcement algorithms (a-c) and TD-learning (d) in Laplace *notation. Transfer functions are denoted as H, K, the derivative operator as s. X_0 represents the unconditioned and X_1 the conditioned input. The amplifier symbol denotes the changing synaptic weight. Note that diagram (c) is drawn with a fixed weight at X_0 to make it more easily comparable to the other diagrams. All models use a derivative of the postsynaptic signal in order to control the weight change. Both Sutton and Barto-models (a,d) use low-pass filters K only in the conditioned pathway, Klopf's model (b) is identical to model (a) with the exception of an additional temporal derivative at this input. Only in ISO-learning all inputs are filtered, which together with the output-derivative generates orthogonal behaviour, leading to weight stabilisation (for further explanations see text).*

B Comparison between TD-, differential Hebbian- and ISO learning

One has to distinguish drive reinforcement models such as those by Sutton and Barto (1981), Klopf (1986, 1988) and Kosco (1986) (to which ISO-learning also belongs) from reference-based reinforcement models such as TD-learning (Sutton, 1988; Dayan and Sejnowski, 1994).

We will first discuss the differences between the different drive reinforcement models which are shown in Fig. 12a–c. The central difference between

ISO-learning and the other techniques is that in ISO-learning all inputs are filtered before they are summed at the output neuron. In the other algorithms the inputs are summed in an un-filtered way. A low-pass filter is only applied to the conditioned stimulus when it enters the learning pathway (i.e., before the correlator "×"). This leads to a fundamentally different behaviour of ISO-learning because as a result ISO-learning produces an orthogonal behaviour between input and output (after filtering the inputs resemble sine waves, thus the derivative of the output resembles a [sum of] cosine[s], see Fig. 2). This orthogonal behaviour, which crucially relies on the filtering of all pathways leads to the inherent and desired weight stabilisation-property of ISO-learning which does not arise in the other drive-reinforcement algorithms without additional measures taken. Furthermore we note that Klopf has introduced an additional derivative at the conditioned input, because he focuses on signal changes.

TD-Learning belongs to yet another category of sequence learning algorithms. The difference arises from the fact that TD is evaluative (reference-based, Fig. 12d), whereas the drive reinforcement models operate in a non-evaluative way. This elementary difference has already been discussed at great length in the main text. Here we focus on the aspect that TD-learning also uses some kind of derivative, which suggests a strong structural similarity between TD- and ISO-learning methods. In spite of this apparent similarity, however, our approach is more strongly related to Kalman filtering than to TD-learning. This is due to the combination of linear filtering and applying a derivative. The predictive property of our algorithm thereby arises from the fact that every low pass filtered function is smooth, which leads to the situation that its derivative linearly predicts its future development. This property of low pass filtered signals is well known in signal theory and is mainly used in the Kalman filter theory (Bozic, 1994).

TD-learning, instead, calculates a temporal difference error δ (thus, similar to the famous δ-rule by Widrow and Hoff (1960)) by means of subtracting subsequent output values from each other and relating this error value to the reward: $\delta(t) = r(t) + v(t+1) - v(t)$. The second group of terms seems to be related to the derivative used in our approach. This mathematical similarity, however, carries a distinctively different interpretation, which can be understood as follows: The goal of TD-learning is that the output $v(t)$ should at any point in time predict the total remaining reward

$$v(t) = \sum_{s \geq t}^{T} r(s) \tag{26}$$

at the end of learning. Take the example of a rat exploring a maze where at each intersection a decision about a turn has to be made creating a temporal sequence of events. Each turn leads to a different reward (e.g. food) to be picked up along the way. This clarifies the concept of "total *remaining* reward" until the end of the maze is reached at T. Furthermore it is known that the total remaining reward can be iteratively approximated using the next following prediction value $v(t+1)$ to yield something like the total remaining *expected* reward:

$$\sum_{s \geq t}^{T} r(s) \approx r(t) + v(t+1) := e(t, t+1) \tag{27}$$

During learning this total remaining expected reward e is compared with its actual prediction v in order to define the prediction error δ. Thus, $\delta(t) = e(t, t+1) - v(t)$, leading to the apparent similarity of the resulting temporal difference terms $v(t+1) - v(t)$ in TD-learning with the derivative used by us. From this interpretation, however, it is quite clear that the term $v(t+1)$ arises only in conjunction with $r(t)$. This kind of conjunction cannot be found in our algorithm because it is reward-free. Furthermore, the structure of TD-learning is slightly acausal looking forward in time using $v(t+1)$ to calculate $\delta(t)$. This, and the reward-based structure of TD-learning makes it also rather difficult to associate it with spike-time dependent plasticity as attempted by Rao and Sejnowski (2001), see Dayan (2002) for a discussion. Thus, the formal similarities between both rules do not seem to warrant treating them as equal. These interpretations continue to hold also for the time-continuous version of TD-learning designed by Doya (2000).

C The Robot

1) Hardware: A modified commercial robot ("rug warrior", 16 *cm* diameter) was used. Two active wheels are driven by DC motors, steering is achieved through different DC-levels. Average speed was adjusted to

0.45 m/s using a control parameter $c = 0.6$. In order to detect mechanical contact the robot has three microswitches CS_l, CS_r, CS_b in a triangular configuration (Fig.11a) Visual signals are generated by two multiplexed, infrared emitting, active range finders RF_l, RF_r with an angle of 70° between them. Infrared reflection is detected by an infrared sensor centred between the emitters which operates in synchrony with them. The detection range was adjusted to $0.5 \ - \ 15.0 \ cm$. Interfacing between robot and computer is done tethered via a conventional I/O card.

2) Sensor characteristics: Sensor signals are band-pass filtered as in many biological systems. This is achieved by feeding the raw signal into a band-pass with transfer function $h(t)$. The output functions of the band-pass filters are denoted as u_k and normalised to one. The band-pass characteristics of all collision sensors $h_r(t)$ is identical with $Q = 0.6$ and $f = 1 \ Hz$. The signal from each vision sensor is fed in parallel into a filter bank of ten band-pass filters. Its frequencies are set to $f_k = \frac{10}{i} \ Hz, \ k = 1, 2, \ldots 10$; Q is set to 1.0, throughout. The filter bank approach assures that large and varying temporal intervals between vision sensor- and collision sensor signals can be covered.

3) Neuronal Circuitry: The robot has two neurons; one which controls the speed ds, the other the steering angle $d\phi$. Normal operation is straight forward motion ($ds = const, \ d\phi = 0$). Both neurons receive inputs from all sensors in a direct feed-forward connectivity.

4) Unconditioned retraction reaction: The unconditioned retraction reaction uses only the collision sensor signals. These signals drive the output neurons in such a way that an avoidance movement with a motion vector pointing away from the site of stimulation is elicited.

5) Learning: All band-pass filter outputs u_k from the collision- and the vision sensors converge onto both neurons where they are summed according to their synaptic weights ρ_k. Only the synaptic weights from the vision-sensors are allowed to change starting from zero. The change of the weights is achieved by the learning rule Eq. 2. The constant μ is set to 0.00002.

6) Neuronal Output (resulting from 4 & 5): The output of the neurons is defined as: $ds = c - \rho_0^{ds}[h_r(t) * (CS_l + CS_r - CS_b)] + l_{ds}$ and $d\phi = \rho_0^{d\phi}[h_r(t) * (CS_l - CS_r)] + l_{d\phi}$. The asterisk denotes a convolution operation. The variables l_{ds} and l_{d_ϕ} represent the total sum of all learned contributions that converge onto the ds- and $d\phi$-neuron respectively. Learning follows Eq. 2. The synaptic weights in unconditioned reaction are kept constant at $\rho_0^{ds} = 0.15$ and $\rho_0^{d\phi} = -0.5$.

References

Abbott, L. and Blum, K. (1996). Functional significance of long-term potentiation for sequence learning and prediction. *Cereb. Cortex*, 6:406–416.

Abbott, L. and Nelson, S. B. (2000). Synaptic plasticity: taming the beast. *Nature Neuroscience supplement*, 3:1178–1179.

Ashby, W. R. (1956). *An introcduction to cybernetics*. Methnen+Co LTD, London.

Bi, G.-q. and Poo, M.-m. (1998). Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strengh, and postsynaptic cell type. *J. Neurosci.*, 18(24):10464–10472.

Bozic, S. M. (1994). *Digital and Kalman filtering: an introduction to discrete-time filtering and optimum linear estimation*. E. Arnold, London.

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. (1997). Intelligence without representation. In John, H., editor, *Mind Design II*, chapter 15, pages 395–420. MIT-press, Cambridge, Mass.

Dayan, P. (2002). Matters temporal. *TRENDS in Cognitive Sciences*, 6(3):105–106.

Dayan, P. and Abbott, L. F. (2001). *Theoretical Neuroscience*. MIT Press, Cambridge MA.

Dayan, P., Kakade, S., and Montague, P. R. (2000). Learning and selective attention. *Nature Neuroscience supplement*, 3:1218–1223.

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

Doya, K. (2000). Reinforcement learning in continous time and space. *Neural Networks*, 12(1):219–245.

Fu, Y.-X., Djupsund, K., Gao, H., Hayden, B., Shen, K., and Dan, Y. (2002). Temporal specifity in the cortical plasticity of visual space representation. *Science*, 296:1999–2003.

Gerstner, W., Kreiter, A. K., Markram, H., and Herz, A. V. (1997). Neural codes: Firing rates and beyond. *Proc Natl. Acad. Sci USA*, 94:12740–12741.

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

Grossberg, S. and Merrill, J. (1996). The hipocampus and cerebellum in adaptively timed learning, recognition and movement. *J. Cogn. Neurosci.*, 8:257–277.

Grossberg, S. and Schmajuk, N. (1989). Neural dynamics of adaptive timing and temporal discrimination during associative learning. *Neural Networks*, 2:79–102.

Guo-Quing, B. and Poo, M.-M. (1998). Synaptic modifications in cultured hippocampus neurons. *J Neurosci.*, 18(24):10464–10472.

Haruno, M., Wolpert, D. M., and Kawato, M. (2001). Mosaic model for sensorimotor learning and control. *Neural Comp.*, 13:2201–2220.

Hauber, W., Bohn, I., and Grietler, C. (2001). NMDA, but not dopamin D_2 receptors in the rat nucleus accumbens are involved in guidance of the instrumental behaviour by stimuli predicting reward magnitude. *J. Neurosci.*, 20(16):6282–6288.

Hebb, D. O. (1967). *The organization of behavior.* Science Ed., New York.

Kandel, E., Abrams, T., Bernier, L., Carew, T., Hawkins, R., and Schwartz, J. (1983). Classical conditioning and sensitization share aspects of the same molecular cascade in aplysia. *Cold Spring Harb Symp Quant Biol*, 48(2):821–830.

Kistler, W. M. and van Hemmen, J. L. (2000). Modeling synaptic plasticity in conjunction with the timing of pre- and postsynaptic action potentials. *Neural Comp.*, 12:385–405.

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

Klopf, A. H. (1988). A neuronal model of classical conditioning. *Psychobiol.*, 16(2):85–123.

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

Levy, W. B. and Minai, A. A. (1993). Sequence learning in a single trial. In *Proceedings of the 1993 INNS World Congress on Neural Networks II*, pages 505–508, New Jersey. Erlbaum.

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.

Maturana, H. and Varela, F. J. (1980). *Autopoiesis and cognition: the realization of the living.* Reidel, Dordrecht.

McGillem, C. D. and Cooper, G. R. (1984). *Continous and discrete signal and system analysis.* CBS publishing, New York.

Miller, K. D. (1996). 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.

Montague, P. R., Dayan, P., and Sejnowski, T. J. (1993). Foraging in an uncertain environment using predictive hebbian learning. *NIPS*, 6:598–605.

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

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

Porr, B., von Ferber, C., and Wörgötter, F. (2002). Iso-learning approximates a solution to the inverse-controller problem in an unsupervised behavioural paradigm. submitted to Neural Comp.

Rao, R. P. and Sejnowski, T. J. (2001). Spike-timing-dependent hebbian plasticity as temporal difference learning. *Neural Comp.*, 13:2221–2237.

Roberts, P. D. (1999). Temporally asymmetric learning rules: I. differential hebbian learning. *J. of Comput. Neurosci.*, 7(3):235–246.

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.

Shepherd, G. M., editor (1990). *The synaptic organisation of the brain.* Oxford University Press, New York.

Song, S. and Abbott, L. (2001). Column and map development and cortical re-mapping through spike-timing dependent plasticity. *Neuron*, 32:339–350.

Song, S., Miller, K. D., and Abbott, L. F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nature Neuroscience*, 3:919–926.

Stewart, J. L. (1960). *Fundamentals of signal theory.* Mc Graw-Hill, New York.

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. *Psychol. Review*, 88:135–170.

Sutton, R. and Barto, A. (1982). Simulation of anticipatory responses in classical conditioning by a neuron-like adaptive element. *Behav. Brain. Res.*, 4(3):221–235.

Traub, R. D. (1999). *Fast Oscillations in Cortical Circuits.* MIT Press, Cambridge.

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

Verschure, P. F. and Pfeifer, R. (1992). Categorization, representations, and the dynamics of system-environment interaction: a case study in autonomous systems. In Roitblat, H., Meyer, J., and Wilson, S., editors, *Proceedings of the Second International Conference on Simulation of Adaptive behaviour*, pages 210–217, Cambridge. MIT press.

von Foerster, H. (1960). On self-organizing systems and their environments. In Yovits, M. and Cameron, S., editors, *Self-Organizing Systems*, pages 31–50. Pergamon Press, London.

Widrow, G. and Hoff, M. (1960). Adaptive switching circuits. *IRE WESCON Convention Record*, 4:96–104.

Wolpert, D. M. and Ghahramani, Z. (2000). Computational principles of movement neuroscience. *Nature Neuroscience supplement*, 3:1212–1217.

Xie, X. and Seung, S. (2000). Spike-based learning rules and stabilization of persistent neural activity. *Advances in Neural Information Processing Systems*, 12:199–208.

Zhang, L. I., Tao, H. W., Holt, C. E., Harris, W. A., and Poo, M.-m. (1998). A critical window for cooperation and competition among developing retinotectal synapses. *Nature*, 395:37–44.