A model for short-term and long-term learning in continuous-time recurrent neural networks

Mattias Wahde
Department of Applied Mechanics
Chalmers University of Technology
412 96 Göteborg, Sweden
mattias.wahde@chalmers.se

Abstract: A biologically inspired computational model for learning in continuous-time recurrent neural networks is introduced and described. The model includes both short-term learning, dependent on neural activity, and long-term learning, dependent on synaptic tagging and artificial gene regulation. Even though many aspects of learning remain to be included in the model, it is shown that, in its present state, the model can reproduce important aspects of fundamental forms of learning such as habituation and sensitization.

1 INTRODUCTION

Learning, i.e. the modification of behavior based on previous experience, is a crucial property of many living systems. The storage of information during learning can be either short-term or long-term; even relatively simple biological organisms are capable of both forms of memory, something that has been elucidated in detailed analyses of organisms such as *Aplysia* (see [5] for a review) and *C. Elegans* [10].

Two of the most basic forms of learning are *habituation* and *sensitization*. Habituation refers to the gradual extinction of the behavioral response to a repeated, neutral stimulus, such as a light touch. This fundamental type of learning, which has been found in a wide range of biological organisms [10], allows an animal to ignore harmless stimuli and thus to focus on other, more relevant stimuli.

Sensitization, on the other hand, refers to an increased behavioral response to a neutral stimulus following the application of an aversive stimulus (such as, for example, an electric shock).

The giant sea snail *Aplysia* is capable of both short-term and long-term habituation and sensitization [8, 7, 5]. Moreover, due to the large size of its neurons, it is a suitable model system for detailed molecular analyses. As a protective mechanism, *Aplysia* will rapidly withdraw its gill (its respiratory organ) upon a light touch to a piece of skin that partly covers the gill. If the animal is touched repeatedly, its response to the touch will gradually become weaker, that is, the animal will display habituation. By contrast, if a mild electric shock is applied to the animal's tail, it will exhibit sensitization, such that its subsequent response to a light touch will instead *increase* in magnitude.

In a series of studies, Kandel and colleagues (see [5] for a review) were able to demonstrate that short-term sensitization in *Aplysia* involves the modulatory neurotransmitter serotonin, which is released in response to a sensitizing stimulus. Serotonin, in turn, unleashes a complex chain of events, the net result of which is an increase in the level of neurotransmitter released at the affected synapses. It was also found that, unlike short-term learning, long-term learning requires *anatomical* modifications in the animal's neurons, i.e. the growth of new synapses. In the case of sensitization in *Aplysia*, repeated application of the neurotransmitter serotonin at a synapse initiates a cascade of gene regulation, leading, after a complex series of steps, ultimately to synaptic growth. Moreover, even though the gene regulation of course involves the nucleus of the neural cell, rather than affecting all synaptic connections made by the cell, the ensuing synaptic growth is synapse-specific, only affecting synapses that have been *tagged* (a process that also involves serotonin, in the case of *Aplysia*).

Similarly, short-term habituation in *Aplysia* has been found to involve changes in neurotransmitter release at synapses, whereas long-term habituation requires anatomical changes [2, 5].

Analytical models of several basic forms of learning (such as habituation, sensitization, and also *classical conditioning*) have been considered by several authors, in some cases using detailed biophysical models (see e.g. [3]), in other cases focusing on phenomenological models [13, 15, 14, 9]. In most cases, rather than being applied as general computational tools, models of basic learning have mainly been focused on describing learning phenomena in biological organisms. However, some authors have considered the evolution of learning rules for artificial neural networks (see e.g. [12, 4]). In addition, computational models of genetic regulatory networks have been introduced as well. For example, in connection with artificial neural networks, several developmental models have been considered (such as Kitano's *grammar encoding* [6]), in which neural networks are generated using genotype-to-phenotype mappings of varying degrees of complexity; see [4] for a review.

Even though several authors have considered both short-term and long-term aspects of basic learning and memory formation, the models introduced thus far do not (to the author's knowledge) explicitly include a model of gene regulation to account for dynamic long-term memory formation (an exception is a model proposed by the author and investigated in [16]).

The aim of this paper is to introduce and describe a model combining short-term and long-term learning in fully recurrent continuous-time neural networks, using the concepts of artificial gene regulation and synaptic tagging in connection with long-term memory formation.

2 MODEL

Consider a fully recurrent, continuous-time neural network containing n neurons. The output x_i of neuron i ($i = 1, \ldots, n$) is given by the first-order differential equation

$$\tau_i \dot{x}_i + x_i = \sigma_1 \left(\sum_{j=1}^{n} w_{ij} x_j + w_{i0} + I_i \right), \qquad (1)$$

where τ_i is a time constant and w_{ij} are the (connection) weights, corresponding to synapses in biological neural networks. I_i denotes the (external) input signal (if any) to neuron i. Note that the neurons are *not* arranged in layers; any neuron may be connected to any other neuron (including itself). The activation function $\sigma_1(z)$ is here taken as

$$\sigma_1(z) = \begin{cases} 1 - \mathrm{e}^{-cz} & \text{if } z \geq 0 \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

where c is a constant, typically set to 1. With this activation function, the output of any neuron is restricted to the range $[0, 1]$. Note that $\sigma_1(z)$ is not differentiable at $z = 0$. However, as illustrated in Sect. 3, Example 3, the constants of the learning model (described below) can be set using stochastic optimization algorithms such as genetic algorithms or particle swarm optimization, rather than gradient descent (or other methods involving derivatives), implying that the activation functions need not be differentiable.

For any given set of input signals, the detailed dynamics of a neural network is determined by its weights and time constants. In the model presented here, the network weights will vary with time, and will therefore be referred to as *parameters*, rather than constants. By contrast, quantities that remain unchanged during evaluation of a neural network (e.g. τ_i) will be referred to as *constants*. In this model, the weights w_{ij} ($i = 1, \ldots, n$, $j = 0, \ldots, n$) are obtained as

$$w_{ij} = w_{\max} \sigma_2 \left(\nu_{ij}^{\mathrm{stm}} + \nu_{ij}^{\mathrm{ltm}} \right), \qquad (3)$$

where ν_{ij}^{stm} and ν_{ij}^{ltm} (both restricted to $[-1, 1]$) denote the short-term and long-term parts of the weight, respectively. The activation function $\sigma_2(z)$ is given by

$$\sigma_2(z) = \tanh cz, \qquad (4)$$

where c is a constant, normally set to 1. The constant $w_{\max}$ determines the (asymptotic) range of the weights w_{ij}. The short-term memory dynamics is given by

$$\tau_{ij}^{\mathrm{stm}} \dot{\nu}_{ij}^{\mathrm{stm}} + \nu_{ij}^{\mathrm{stm}} = \sigma_2 \left(\sum_{k=1}^{n} \alpha_{ijk} x_k \right), \qquad (5)$$

where α_{ijk} are constants (henceforth referred to as *modulatory weights*), and

$$\tau_{ij}^{\mathrm{stm}} = \begin{cases} \tau_{ij}^{\mathrm{stm}+} & \text{if } \mathrm{sgn}(\dot{\nu}_{ij}^{\mathrm{stm}}) = \mathrm{sgn}(\nu_{ij}^{\mathrm{stm}}) \\ \tau_{ij}^{\mathrm{stm}-} & \text{otherwise} \end{cases} \qquad (6)$$

Thus, each short-term weight ν_{ij}^{stm} is associated with two time constants, so that the time scale for rising weight magnitudes differs from the time scale of falling weight magnitudes. The variation in the short-term weights is a result of neural activity. Thus, during initialization (prior to integration of the network equations), all short-term weights are set to zero, and the long-term weights ν_{ij}^{ltm} are obtained from the (given) initial weights w_{ij} as

$$\nu_{ij}^{\mathrm{ltm}} = \sigma_2^{-1} \left(\frac{w_{ij}}{w_{\max}} \right). \qquad (7)$$

Since ν_{ij}^{ltm} should be restricted to the range $[-1, 1]$ (see Eq. (11) below), the initial values of the weights w_{ij} should be sufficiently small, relative to $w_{\max}$, so as to generate values within that range. This implies no restriction: The value of $w_{\max}$ can always be set (at initialization) to accomodate any desired initial weight values.

During integration of the network equations, the short-term weights change according to Eq. (5) resulting, in turn, in a modulation of the synaptic weights w_{ij} according to Eq. (3). This modulation is equivalent to the short-term dynamics found in biological neurons, in which changes in the amount of neurotransmitter released at a synapse result in a change in synaptic efficacy.

In addition to the short-term dynamics, the model also includes long-term dynamics that, as in the biological counterpart (see Sect. 1), depends on synaptic tagging and gene regulation. Thus, in the model, the variation in the short-term weights ν_{ij}^{stm} results in an artificial marker substance, denoted s_{ij}, being deposited at the synapse in question, according to

$$\tau_{ij}^{\mathrm{s}} \dot{s}_{ij} + s_{ij} = \sigma_2 \left(\beta_{ij} \nu_{ij}^{\mathrm{stm}} \right), \qquad (8)$$

where β_{ij} are constants. The time constants τ_{ij}^{s} are given by

$$\tau_{ij}^{\mathrm{s}} = \begin{cases} \tau_{ij}^{\mathrm{s}+} & \text{if } \mathrm{sgn}(\dot{s}_{ij}) = \mathrm{sgn}(s_{ij}), \\ \tau_{ij}^{\mathrm{s}-} & \text{otherwise} \end{cases} \qquad (9)$$

In other words, synapses undergoing changes are *tagged*. Note that the marker substance s_{ij} takes values in the range $[-1, 1]$, and should therefore (in a biological analogy) be interpreted as a net effect of several marker substances, rather than as a simple concentration of a single marker substance. The presence of marker substance at the synapses of neuron i triggers expression of an artificial gene g_i according to

$$\tau_i^{\mathrm{g}} \dot{g}_i + g_i = \sigma_1 \left(\sum_{j=0}^{n} \gamma_{ij} |s_{ij}| + \Gamma_i \right), \qquad (10)$$

where τ_i^{g}, γ_{ij}, and Γ_i are constants. In this model, gene expression levels take values in the range $[0, 1]$. For any given gene, the parameter Γ_i determines the level of input required for the gene to be activated (i.e. to obtain expression levels above 0). It should be noted that, as in biological neurons, rather than being synapse-specific, the genes are properties of the cell (neuron) as a whole. However, in order to be useful, the long-term dynamics must, like the short-term dynamics, be synapse-specific. Thus, the long-term weights ν_{ij}^{ltm} vary according to

$$\tau_{ij}^{\mathrm{ltm}} \dot{\nu}_{ij}^{\mathrm{ltm}} = (1 - |\nu_{ij}^{\mathrm{ltm}}|) \sigma_2 \left(\delta_{ij} s_{ij} g_i \right), \qquad (11)$$

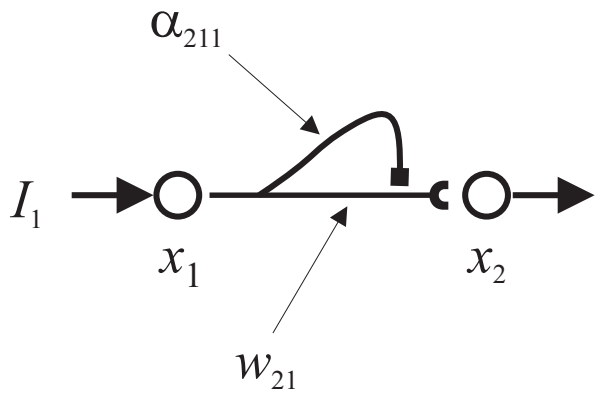

Fig. 1: The network used in Example 1 (habituation). The modulatory connection (specified by the modulatory weight α_{211}) is represented by the line ending with a filled square.

where τ_{ij}^{ltm} and δ_{ij} are constants.

This completes the model. During initialization, the long-term weights ν_{ij}^{ltm} are determined as described above, whereas the short-term weights (ν_{ij}^{stm}), the marker substance levels (s_{ij}), and the gene expression levels (g_i), and the neuron activation levels (x_i) are all set to zero. During network integration, the short-term weights are updated according to Eq. (5). The marker substance levels and the gene expression levels are integrated according to Eqs. (8) and (10), respectively, and the long-term weights are obtained from Eq. (11). The weights w_{ij} (to be employed in the next time step of the integration, using Eq. (1)) are then computed from Eq. (3).

The full specification of a network requires that a large number of constants should be set, namely τ_i, $w_{\max}$, $\tau_{ij}^{\text{stm}+}$, $\tau_{ij}^{\text{stm}-}$, α_{ijk}, $\tau_{ij}^{\text{s}+}$, $\tau_{ij}^{\text{s}-}$, β_{ij}, τ_i^{g}, γ_{ij}, Γ_i, τ_{ij}^{ltm} and δ_{ij} (neglecting the parameters c in the activation functions that, as mentioned above, are normally set to 1). In addition, the initial weight values w_{ij} must also be set, so that the initial values of ν_{ij}^{ltm} can be computed. Thus, the dynamic flexibility of the model comes at a price. However, as illustrated below, in some cases only a few of the available constants need take non-zero values.

3 RESULTS

The learning model will now be illustrated by means of three simple examples, namely habituation (Example 1), habituation *and* sensitization (Example 2), and general signal following (Example 3). In all cases, the network equations were integrated numerically using a time step of 0.01 s.

3.1 Example 1: Habituation

The first example considered here will be *habituation*, both short-term and long-term. (Here, *long-term* refers to any learning lasting more than a few seconds). The network used is shown in Fig. 1. As can be seen, the network contains only two neurons, marked x_1 and x_2. A single input signal (I_1) enters the network via neuron 1 that, in turn, is connected to neuron 2 through the only non-zero connection weight w_{21}. In addition, the network contains a modulatory weight denoted α_{211}. Thus, for this simple network the full set of equations becomes

$$\tau_1 \dot{x}_1 + x_1 = \sigma_1 (I_1) , \tag{12}$$

Table 1: CONSTANTS USED IN THE NETWORK ILLUSTRATING HABITUATION (EXAMPLE 1).

Neuron dynamics:			
τ_1	0.050	τ_2	0.050
w_{21} (initial)	2.000	$w_{\max}$	5.000
Short-term memory dynamics:			
$\tau_{21}^{\text{stm}+}$	0.500	$\tau_{21}^{\text{stm}-}$	3.000
α_{211}	-1.000		
Long-term memory dynamics:			
$\tau_{21}^{\text{s}+}$	5.000	$\tau_{21}^{\text{s}-}$	5.000
β_{21}	1.000		
τ_2^{g}	5.000		
γ_{21}	1.000	$\Gamma 2$	-0.100
τ_{21}^{ltm}	10.000		
δ_{21}	2.000		

$$\tau_2 \dot{x}_2 + x_2 = \sigma_1 (w_{21} x_1) , \tag{13}$$

$$\tau_{21}^{\text{stm}} \dot{\nu}_{21}^{\text{stm}} + \nu_{21}^{\text{stm}} = \sigma_2 (\alpha_{211} x_1) , \tag{14}$$

$$\tau_{21}^{\text{s}} \dot{s}_{21} + s_{21} = \sigma_2 \left(\beta_{21} \nu_{21}^{\text{stm}}\right) , \tag{15}$$

$$\tau_2^{\text{g}} \dot{g}_2 + g_2 = \sigma_1 (\gamma_{21} |s_{21}| + \Gamma_2) \tag{16}$$

and

$$\tau_{21}^{\text{ltm}} \dot{\nu}_{21}^{\text{ltm}} = \left(1 - |\nu_{21}^{\text{ltm}}|\right) \sigma_2 (\delta_{21} s_{21} g_2) . \tag{17}$$

Note that, for simplicity, no bias term (w_{20}) has been included in the input to neuron 2, since it is not needed in this example. Note also that any gene regulation involving neuron 1 can be ignored, as it neither has any bias term nor receives input from any other *neuron*.

In order to specify the network, the *initial* value of the weight w_{21} must be set, as well as the constant α_{211}. Furthermore, the constants $w_{\max}$, $\beta_{21}, \gamma_{21}, \Gamma_2$, and δ_{21} must be set, as well as the time constants $\tau_1, \tau_2, \tau_{21}^{\text{stm}+}, \tau_{21}^{\text{stm}-}, \tau_{21}^{\text{s}+}, \tau_{21}^{\text{s}-}, \tau_2^{\text{g}}$, and τ_{21}^{ltm}. A set of suitable values of the constants (for habituation) is given in Table 1. In this case, the constants were found through simple trial-and-error experimentation.

Fig. 2 shows the resulting dynamics. The input signal I_1, shown in the top panel, starts with 10 equally spaced pulses, of equal magnitude, followed by a period of around 3.5 s during which no input pulses are applied. Next, a new sequence of equally spaced input pulses, of the same magnitude as before, is applied.

The second panel (from the top) shows the output (x_2): Following a strong initial response, the output is gradually weakened, as a result of short-term depression of the connection weight w_{21}. Furthermore, the weight w_{21} is gradually tagged with a marker substance s_{21}, here taking negative values, as shown in the third panel of the figure. The presence of the marker substance eventually triggers expression of the gene g_2, which, in turn, triggers long-term modification of the weight w_{21}.

The bottom panel of Fig. 2 shows the variation of w_{21}, resulting from the combination of short-term (ν_{21}^{stm}) and long-term (ν_{21}^{ltm}) weight variations (not shown). As can be seen in this panel, as soon as the habituating stimulus disappears, w_{21} stages a recovery that, however, only becomes incomplete, as a result of the *long-term* depression of the weight. Thus, when the second pulse train arrives, the initial response x_2 is weaker than the initial response to the first pulse train. Furthermore, after around 16.5

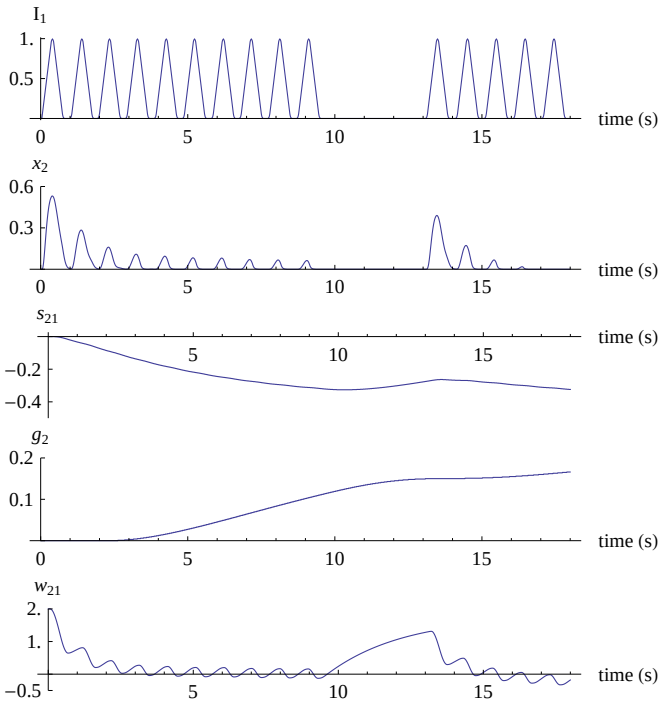

Fig. 2: The results obtained for Example 1 (habituation). From top to bottom, the panels show the variation (with time) of I_1, x_2, s_{21}, g_2, and w_{21}. The gradual extinction of the response (x_2) as a result of habituation can be seen in the second panel from the top. For a complete description of the figure, see the main text. Note that, for clarity, different scales have been used on the vertical axes of the different panels.

s, w_{21} becomes consistently negative so that, in fact, the response is completely extinguished: A case of complete habituation.

3.2 Example 2: Sensitization and habituation

As a second example, sensitization *and* habituation will be considered together. A network capable of exhibiting both features is shown in Fig. 3. Since the network extends the one considered in Example 1 above, the enumeration of the neurons (and constants) present in the previous network has been kept, to facilitate comparison between the two networks.

In this example, the network takes two inputs, denoted I_1 and I_3. As before, I_1 represents a neutral stimulus; thus, the proper response of the network will be to habituate, giving progressively smaller response (x_2) to pulses entering the network in the form of the signal I_1. By contrast, I_3 represents an aversive stimulus (such as an electric shock, as commonly used in *Aplysia* studies [5]). The desired response is one of sensitization, such that, following the application of a pulse in I_3, the response of the network to a neutral stimulus (I_1) should *increase*.

For the network shown in Fig. 3, the equation for x_1 is the same as above; see Eq. (12). Neglecting the bias terms again, the equation for x_2 changes to

$$\tau_2 \dot{x}_2 + x_2 = \sigma_1 \left(w_{21} x_1 + w_{23} x_3 \right), \tag{18}$$

whereas x_3 is obtained from

$$\tau_3 \dot{x}_3 + x_3 = \sigma_1 \left(I_3 \right). \tag{19}$$

The short-term memory dynamics now takes the form

$$\tau_{21}^{\text{stm}} \dot{\nu}_{21}^{\text{stm}} + \nu_{21}^{\text{stm}} = \sigma_2 \left(\alpha_{211} x_1 + \alpha_{213} x_3 \right), \tag{20}$$

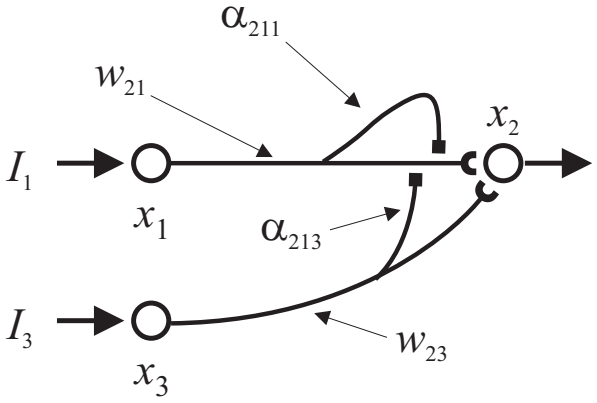

Fig. 3: The network used in Example 2 (sensitization and habituation).

whereas the equations related to long-term memory, Eqs. (15)-(17), remain unchanged (since $\nu_{23}^{\text{stm}} \equiv 0$).

Again, the constants were set using a process of trial-and-error, starting from the constants used in habituation, given in Table 1. The detailed constant values thus obtained will not be given here; suffice it to say that α_{211} should be negative and α_{213} should be positive. Fig. 4 shows the resulting dynamics for one suitable set of constants. In this experiment, a single pulse of the neutral stimulus (I_1, top panel) was first applied, giving a response (x_2, third panel) with a maximum of around 0.3. Next, the aversive stimulus (I_3, second panel) was applied five times (with equal magnitude each time). The aversive stimuli gave rise to strong output pulses, with a maximum of around 0.7. Next, after a resting period of around 5 s, a single neutral stimulus was again applied at $t \approx 15$ s. As can be seen in the third panel of the figure, the resulting output pulse x_2 reached a maximum of around 0.55, i.e. higher than the original response (with a maximum of around 0.3), indicating sensitization. Next, starting at $t \approx 18$ s, a train of five neutral pulses was applied. The response to the first pulse reached a maximum of around 0.35, indicating a lingering sensitization to the aversive stimuli applied earlier. However, at this point, the repeated neutral stimuli gave rise to (short-term) habituation, so that the response to the last four pulses (ending at $t \approx 23$ s) was gradually extinguished. After another resting period of around 3.5 s, an additional set of three neutral stimuli was applied, this time resulting in an initial output pulse with a maximum of around 0.28, showing that habituation now had eliminated the long-term sensitization. Again, with repeated application of the neutral stimulus, the response x_2 was rapidly extinguished.

The variation of the weight w_{21} is shown in the bottom panel of Fig. 4. As can be seen, the aversive stimuli generated a rapid increase in the weight, followed by a decay towards a plateau slightly higher than the initial weight value, due to an increase in the long-term part (ν_{21}^{ltm}) of the weight (not shown). Subsequent neutral stimuli then began to reduce w_{21}. The initial depression of the weight was of a short-term nature. However, towards the end of the experiment (around $t \approx 29$ s), habituation had started affecting the long-term part of the weight as well. Had the experiment been extended, the long-term part would gradually have fallen towards negative values, completely reversing the sensitizing effects of the aversive

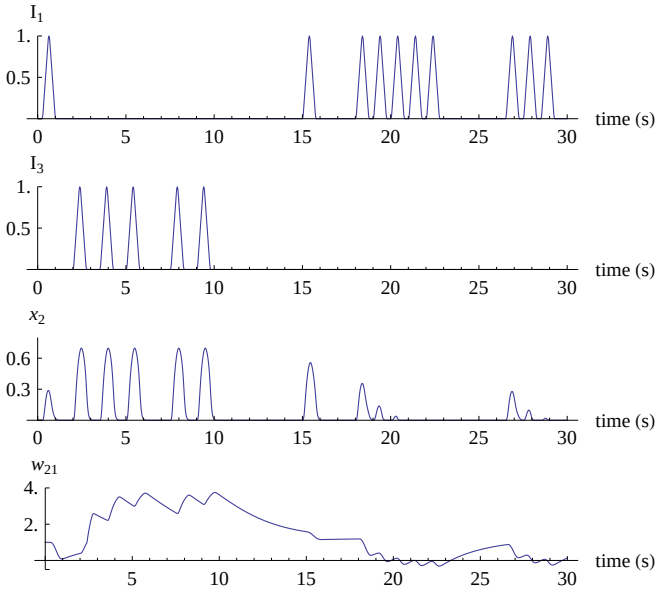

Fig. 4: The results obtained for Example 2 (sensitization and habituation). From top to bottom, the panels show the variation (with time) of the variables I_1 (the neutral stimulus), I_3 (the aversive stimulus), x_2, and w_{21}. Following an initial application of the neutral stimulus, a sequence of aversive stimuli are applied, leading to sensitization, as evidenced by the increased response (x_2, third panel) to subsequent neutral stimuli. For a complete description of the figure, see the main text. Note that, for clarity, different scales have been used on the vertical axes of the different panels.

stimuli applied from $t \approx 2$ s to $t \approx 10$ s. It should be noted that the network thus could handle both short-term and long-term memory dynamics for sensitization *and* habituation.

3.3 Example 3: Signal following

As a final example, a simple instance of signal following will be considered. In this example, a single slowly oscillating, input signal (I_1) was used. This pre-specified input signal normally varied sinusoidally between 0.2 and 0.8. However, one one occasion, at $t \approx 10$ s, the signal dropped to 0.05, and remained constant for around 2.5 s. The input signal then again oscillated between 0.2 and 0.8 for another 5.5 s. The (somewhat arbitrary) desired output (taken as x_2) was as follows: Until the input signal dropped to 0.05, the output x_2 was supposed to mimic the input signal as closely as possible; that is, the *desired* output x_2 was then equal to I_1. However, once the input signal had dropped to 0.05, the desired output signal was changed to $1 - I_1$.

Thus, the network was required to know that a drop to 0.05 signalled profound changes in the desired output, requiring modification of the network weights. Of course, during the first 10 s of the experiment, the network could easily obtain a good result with *constant* weights and a small time constant τ_2. However, after the desired output changed to $1 - I_1$, the network needed to quickly modify its connection weights to handle the new situation.

Before proceeding with the example, one should note that this experiment, of course, represents a toy problem: The simple signal following considered here can be solved without neural networks (using instead, for example, methods from classical control theory). Thus, the example is intended merely as an illustration of the learning

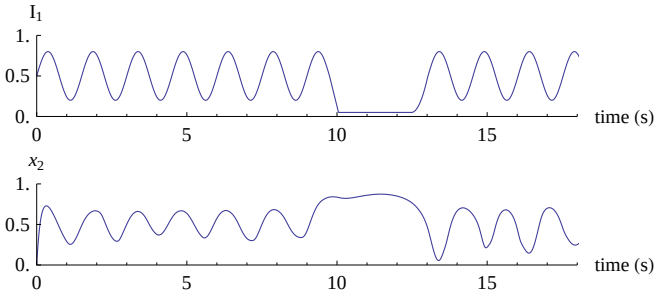

Fig. 5: The top panel shows the input signal used in Example 3 (signal following). The bottom panel shows the output (x_2) for the best network found by the GA.

model.

In order to train the network (i.e. to set the *constants* to appropriate values), a genetic algorithm (GA) was used. The network constants were encoded in a genome consisting of 11 real-valued chromosomes. The constants encoded by the chromosomes (denoted C_k, $k = 1, \ldots 11$) were: The initial weights w_{ij} (C_1, $n(n+1)$ constants), α_{ijk} (C_2, $n^2(n+1)$ constants), β_{ij} (C_3, $n(n+1)$ constants), γ_{ij} (C_4, $n(n+1)$ constants), Γ_i (C_5, n constants), δ_{ij} (C_6, $n(n+1)$ constants), τ_i (C_7, n constants), $\tau_{ij}^{\mathrm{stm}+}$ and $\tau_{ij}^{\mathrm{stm}-}$ (C_8, $2n(n+1)$ constants), τ_{ij}^{ltm} (C_9, $n(n+1)$ constants), $\tau_{ij}^{\mathrm{s}+}$ and $\tau_{ij}^{\mathrm{s}-}$ (C_{10}, $2n(n+1)$ constants), and τ_g (C_{11}, n constants). The error measure was taken as the root mean square error between the actual output signal (x_2) and the desired output signal (i.e. I_1 for the first 10 s of the evaluation, and $1 - I_1$ thereafter). Except for the division of the genome into 11 chromosomes, a fairly standard genetic algorithm was used, with tournament selection, followed by crossover (with a certain probability) and mutation (either full range mutation or creep mutation). Crossover was applied within chromosomes, i.e. with one crossover point for each chromosome. Elitism was used as well, such that a single copy of the best individual in generation G was transferred unchanged to generation $G + 1$.

The task of the GA was thus to set the constants of the network, so as to minimize the error over the evaluation. For any given individual (network), the parameters (e.g. the network weights) varied during the evalution, according to the equations given in Sect. 2.

Several evolutionary runs were made, with different network sizes (number of neurons). For this example, the best results were obtained using networks with seven neurons. Fig. 5 shows the input signal (I_1, top panel) and the output signal (x_2) of the best seven-neuron network found by the GA.

Note that the network started with $x_2 = 0$, whereas the input signal I_1 was equal to 0.5 at the beginning of the evaluation. As can be seen in the figure, after an initial transient, the network managed to follow the oscillation of the input signal rather well, albeit with somewhat smaller amplitude. At around $t = 10$ s, I_1 dropped to 0.05, and the desired output signal changed to $1 - I_1$. The output promptly rose to a value of around 0.85 (i.e. slightly below the optimal value of 0.95). As the input signal then began oscillating again, at around $t = 12.5$ s, the network generated an out-of-phase output, as desired. Interestingly, some genes (particularly gene 4, affecting the connection weights of neuron 4) showed

marked up-regulation, starting at around $t = 10$ s, indicating that the network rewired itself to cope with the change in the desired output.

Thus, the GA successfully discovered a network capable of solving the problem. However, before drawing far-reaching conclusions, one should note that adaptation to a given input signal can, of course, be achieved without the network actually learning to follow an *arbitrary* input signal. A more stringent test of the model's capabilities would require more extensive testing, using a variety of input signals. Furthermore, one should note that the evolution of the network discussed above was, in fact, quite slow, requiring a few thousand generations of the GA (with a population size of 30) to reach the result shown in Fig. 5.

4 DISCUSSION AND CONCLUSION

Even though the model of learning, introduced in Sect. 2 above, can reproduce several important aspects of learning in biological organisms, it can (and should) be improved in various ways.

For example, the activity-dependent short-term learning, described in Eq. (5), results in a continuous variation of the short-term part (ν_{ij}^{stm}) of the network weights, except in those (uncommon) situations where the input remains constant for extended periods of time, in which case ν_{ij}^{stm} asymptotically approaches a constant. Continuous modification of the network weights is not always desirable; for instance, in Example 3 above, the network would not really need to modify its weights to track the signal during the first 10 s of the experiment. This raises the issue of *meta-plasticity*, i.e. the modulation of learning [1] itself. This aspect could be introduced in the model by turning the modulatory weights (α_{ijk}) into tunable parameters that only take non-zero values under certain circumstances, when learning is allowed. Of course, the question of determining *when* to allow learning must then also be addressed.

Another important issue concerns the gene regulation involved in long-term learning. So far, this model only features positive regulation, such that long-term learning becomes possible when the artificial gene g_i (associated with neuron i) is expressed. However, in biological organisms, it is known that learning also involves memory-suppressor genes [5] that must be down-regulated for long-term learning to occur. Such genes may be involved in determining *which* events, if any, should be stored in long-term memory.

An equally important question concerns the reward signals involved in learning. In the examples presented in Sect. 3 above, the desired network output was known *a priori* in all cases (and in the first two cases even the network structure was known at the outset). In real-world learning situations, such as those encountered by biological organisms, no detailed error signal is readily available, in most cases. Behavioral learning studies indicate that, in biological organisms, learning occurs when there is a discrepancy between predicted reward and actual reward [11]. Thus, in order to achieve general learning, one must also consider (the evolution of) both reward signals and internal prediction of rewards.

In conclusion, the model for learning introduced here is capable of both short-term and long-term memory formation, and can furthermore reproduce some important aspects of habituation and sensitization, two of the most fundamental forms of learning in biological organisms. The issues related to improvements and refinements of the model are currently under consideration by the author.

References

[1] Abraham, W.C., Metaplasticity: tuning synapses and networks for plasticity, *Nature Rev. Neurosci.*, **9**, pp. 387-399, 2008.

[2] Bailey, C.H. and Chen, M., Long-term memory in *Aplysia* modulates the total number of varicosities of single identified sensory neurons, *Proc. Natl. Acad. Sci. USA*, **85**, pp. 2373-2377, 1988.

[3] Ciaccia, P., Maio, D., and Vacca, G.P. An analytical short- and long-term memory model of presynaptic plasticity, *Biol. Cybern.*, **67**, pp. 335-345, 1992.

[4] Floreano, D,. Dürr, P., and Mattiussi, C., Neuroevolution: from architectures to learning, *Evol. Intel.*, **1**, pp. 47-62, 2008.

[5] Kandel, E.R., The molecular biology of memory storage, *Bioscience reports*, **21**, pp. 565-611, 2001.

[6] Kitano, H., Designing neural networks by genetic algorithms using graph generation system, *Complex Syst. J.*, **4**, pp. 461-476, 1990.

[7] Pinsker, H.M., Hening, W.A., Carew, T.J., and Kandel, E.R., Long-term sensitization of a defensive withdrawal reflex in *Aplysia*, *Science*, **182**, pp. 1039-1042, 1973.

[8] Pinsker, H.M., Kupfermann, I., Castelucci, V., and Kandel, E.R., Habituation and dishabituation of the gill-withdrawal reflex in *Aplysia*, *Science*, **167**, pp. 1740-1742, 1970.

[9] del Rosal, E., Alonso, L., Moreno, R., Vázquez, M., and Santacreu, J., Simulation of habituation to simple and multiple stimuli, *Behavioural Processes*, **73**, pp. 272-277, 2006.

[10] Rose, J.K. and Rankin, C.H., Analyses of habituation in *Caenorhabditis elegans*, *Learning and memory*, **8**, pp. 63-69, 2001.

[11] Schultz, W., Multiple reward signals in the brain, *Nature Rev. Neurosci.*, **1**, pp. 199-207, 2000.

[12] Soltoggio, A., Dürr, P., Mattiussi, C., and Floreano, D., Evolving neuromodulatory topologies for reinforcement learning-like problems, In: Angeline, P. *et al.* (Eds.) *Proc. of the 2007 IEEE Congress on Evolutionary Computation*, 2007.

[13] Stanley, J.C., Computer simulation of a model of habituation, *Nature*, **261**, pp. 146-148, 1976.

[14] Wang, D.L., A neural model of synaptic plasticity underlying short-term and long-term habituation, *Adapt. Behav.*, **2**, pp. 111-129, 1993.

[15] Wang, D.L. and Arbib, M.A., Modeling the dishabituation hierarchy: The role of the primordial hippocampus, *Biol. Cybern.*, **67**, pp. 535-544, 1992.

[16] Zhong, H., A computational model for basic learning and memory formation in *Aplysia*, Master thesis, Chalmers University of Technology, 2007.