

# Reinforcement learning in cortical networks

Walter Senn<sup>1</sup>, Jean-Pascal Pfister<sup>1</sup>

Article 580-2, *Encyclopedia of Computational Neuroscience*, Springer; submitted August 14, 2014

AUTHOR COPY

## Synonyms

Reward-based learning; Trial-and-error learning; Temporal-Difference (TD) learning; Policy gradient methods

## Definition

Reinforcement learning represents a basic paradigm of learning in artificial intelligence and biology. The paradigm considers an agent (robot, human, animal) that acts in a typically stochastic environment and receives rewards when reaching certain states. The agent's goal is to maximize the expected reward by choosing the optimal action at any given state. In a cortical implementation, the states are defined by sensory stimuli that feed into a neuronal network, and after the network activity is settled, an action is read out. Learning consists in adapting the synaptic connection strengths into and within the neuronal network based on a (typically binary) feedback about the appropriateness of the chosen action. Policy gradient and temporal difference learning are two methods for deriving synaptic plasticity rules that maximize the expected reward in response to the stimuli.

## Detailed Description

Different methods are considered for adapting the synaptic weights  $w$  in order to maximize the expected reward  $\langle R \rangle$ . In general, the weight adaptation has the form

$$\Delta w = R \cdot \text{PI} \quad (1)$$

where  $R = \pm 1$  encodes the reward received upon the chosen action, and PI represents the plasticity induction the synapse was calculating based on the pre- and postsynaptic activity. To prevent a systematic drift of the synaptic weights that is not caused by the co-variation of reward and plasticity induction, either the average reward or the average plasticity induction must vanish,  $\langle R \rangle = 0$  or  $\langle \text{PI} \rangle = 0$ .

Reinforcement learning can be divided in these two, not mutually exclusive, classes of assuming that (A)  $\langle \text{PI} \rangle = 0$  or (B)  $\langle R \rangle = 0$ . The first class encompasses *policy gradient* methods while the other, wider class, encompasses *Temporal Difference* (TD) methods. Policy gradient methods assume less structure as they postulates the required property ( $\langle \text{PI} \rangle = 0$ ) on the same synapse of the action selection module that is adapted by the plasticity. TD methods also involve the adaptation of the internal critique since they have to assure that the required property on the modulation signal ( $\langle R \rangle = 0$ ) holds for each stimulus class separately.

**A) Policy gradient methods** In the simplest biologically plausible form, actions are represented by the activity of a population of neurons. Each neuron in the population is synaptically driven by feedforward input encoding the current stimulus (Fig. 1). The synaptic strengths are adapted according

---

<sup>1</sup>Department of Physiology, University of Bern, Switzerland; senn@pyl.unibe.ch, pfister@pyl.unibe.ch

to the gradient of the expected reward across possible actions. Various learning rules emerge from the different methods of estimating this gradient.

In formal terms, sensory stimuli define an input  $x$ , e.g. a spike train, that is fed to the network characterized by the synaptic strengths  $w$ . This network generates an output  $y$ , say again a spike trains, that depends on the synaptic weights  $w$  and the input  $x$ . Based on  $y$ , an action  $A$  is selected (Fig. 1). The action is eventually rewarded by a typically binary signal  $R(x, A) = \pm 1$  that is fed back to the network where it modulates synaptic plasticity by a global factor. The stimulus choice, the network activity and the action selection may have stochastic components. The probability function  $P_w(A|x)$  that specifies how likely action  $A$  is selected upon input  $x$  is referred to as action policy. A policy gradient method considers the expected reward

$$\langle R \rangle = \sum_{x,A} P(x) P_w(A|x) R(x, A) \quad (2)$$

and adapts the synaptic connection strengths  $w$  along the reward gradient, i.e. such that the expected weight change satisfies  $\langle \Delta w \rangle = \eta \frac{d\langle R \rangle}{dw}$  with some small but positive learning rate  $\eta$  (Williams, 1992).

**Hedonistic synapse** In the current form, a synapse is assumed to estimate how the change in  $w$  affects the likelihood  $P_w(A|x)$ . But this information may not be available at the single synapse. According to the hedonistic synapse model (Seung, 2003), a synapse is only assumed to have access to the very local information of whether there was a release or not ( $r_t = \pm 1$ ) at time  $t$  in response to a presynaptic spike, and  $x_t$  itself encodes the presence or absence of a presynaptic spike. We can then write  $P_w(A|x) = \sum_r P(A|r) P_w(r|x)$ , where  $P_w(r|x)$  is parameterized by the variable  $w$  and the sum is taken across release trains  $r$ . Plugging this into (2) and taking the derivative we obtain

$$\frac{d\langle R \rangle}{dw} = \sum_{r,x,A} P_w(x, r, A) R(x, A) \frac{d}{dw} \log P_w(r|x) . \quad (3)$$

Sampling this gradient leads to updates of  $w$  after taking action  $A$  in response to the presynaptic spike train,

$$\Delta w = \eta R \frac{d}{dw} \log P_w(r|x) . \quad (4)$$

Note that in any case  $\langle \frac{d}{dw} \log P_w(r|x) \rangle_r = \sum_r \frac{d}{dw} P_w(r|x) = 0$  because  $\sum_r P_w(r|x) = 1$  and hence  $\langle \text{PI} \rangle = 0$  as described after Eq. 1.

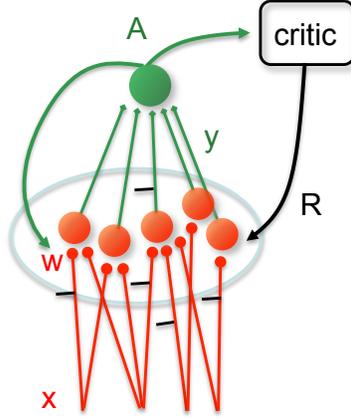
To obtain an online rule from (5) one low-pass filters  $\frac{d}{dw} \log P_w(r_t|x_t)$  to get an eligibility trace that is then multiplied by  $R$  to calculate the synaptic update (Seung, 2003). If for simplicity we consider a single time interval for a presynaptic spike to occur and an action to be taken, the rule (5) becomes

$$\Delta w = \eta R (r - p) x , \quad (5)$$

where  $p = P_w(r = 1|x = 1)$  is probability of release that is parametrized by a sigmoidal function of  $w$ ,  $p = 1/(1 + e^{-w})$ . In the general case, the plasticity induction  $\text{PI} = (r - p) x$  is low-pass filtered with a time constant that reflects the delay of the reward (see *Online learning* below).

**Spike reinforcement** Since there is only little correlation between specific synaptic releases and rewards to actions, a more efficient way to estimate the gradient  $\frac{d\langle R \rangle}{dw}$  is to consider the dependence of  $A$  on  $y$ , as expressed by  $P_w(A|x) = \sum_y P(A|y) P_w(y|x)$ . Plugging this again into (2) and taking the derivative we get for the sampled version

$$\Delta w = \eta R \frac{d}{dw} \log P_w(y|x) . \quad (6)$$



**Figure 1.** Population reinforcement learning adapts the synaptic strengths  $w$  to the population neurons based on 4 ingredients: the presynaptic activity  $x$ , the postsynaptic activity  $y$ , the action  $A$  encoded in the population activity, and the reward signal  $R$  received in response to the action, see Urbanczik and Senn (2009).

To illustrate learning rule (6), we can again interpret  $x$  and  $y$  ( $= \pm 1$ ) to encode the presence or absence of a spike in a given interval, with the probability  $p$  for  $y = 1$  being again a sigmoidal function of  $u = \sum_i w_i x_i$ . Analogously to (5) we obtain the rule  $\Delta w = \eta R (y - p) x$  (Williams, 1992). Alternatively,  $x$  and  $y$  may encode firing rates with  $y = \phi(u) + \xi$ , where  $\phi(u)$  is a non-negative increasing function, and  $\xi$  some Gaussian noise. We then calculate  $\frac{d}{dw} \log P_w(y|x) \propto (y - \phi(u)) \phi'(u) x$ , and the learning rule becomes

$$\Delta w = \eta R (y - \phi(u)) \phi'(u) x \quad (7)$$

This synaptic plasticity rule depends on 3 factors, (i) the reward, (ii) the postsynaptic quantities (here considered as a single factor) and (iii) the presynaptic activity. For spiking neurons, the corresponding learning rule has been introduced by Xie and Seung (2004) and has been further studied by Pfister et al. (2006); Florian (2007); Frémaux et al. (2010).

**Node perturbation** In the framework of spiking neurons the exploration of the neuronal state space is driven by the intrinsic noise present in the individual neuron’s spiking mechanism. A more efficient exploration can be achieved if the noise enters from an external source and can therefore be explicitly tuned. This idea leads to RL based on node perturbation (Fiete and Seung, 2006). In the simple coding scenario considered above, node perturbation is formally equivalent to (7) and, with  $\xi = y - \phi$ , it can be rewritten as

$$\Delta w = \eta R \xi \phi' x. \quad (8)$$

Yet, node perturbation can also be generalized to conductance-based neurons driven by regular ‘student’ input and ‘exploration’ input and, as before, the plasticity induction  $PI = \xi \phi' x$  can again be replaced by a low-pass filtering to comply with delayed reward (Fiete and Seung, 2006).

**Population reinforcement** When assuming that a synapse has access to the downstream information involved in choosing action  $A$ , e.g. via global neuromodulators, the gradient can be well estimated by directly calculating the derivative of (2) with respect to  $w$ . The sampling version of this rule is then

$$\Delta w = \eta R \frac{d}{dw} \log P_w(A|x). \quad (9)$$

The action can itself be binary and e.g. depend on whether the majority of the population neurons did or did not spike in response to the stimulus (Friedrich et al., 2011), or it can be continuous and e.g. depend on the average population firing rate (Friedrich et al., 2014).

To give an example, we consider a population of  $N$  neurons with outputs  $y_i = \phi(u_i) + \xi_i$  for  $i = 1 \dots N$  that are obtained by a sigmoidal function  $\phi(u_i)$  of the weighted input,  $u_i = \sum_j w_{ij}x_j$ , plus some independent Gaussian noise  $\xi_i$  of mean 0. For simplicity we assume  $\phi$  to increase from  $-1$  to  $1$  with  $\phi(0) = 0$  and consider binary actions  $A = 1$  or  $A = -1$  that are taken stochastically based on the population activity  $\mathcal{A} = \frac{1}{\sqrt{N}} \sum_i y_i$ , with  $P(A=1 | \mathcal{A}) = (1 + \tanh 2\mathcal{A})/2$  being a sigmoidal function of  $\mathcal{A}$ . Neglecting again the indices, the learning rule (9) can then be evaluated to (see Friedrich et al. (2014), Eq. 21 therein)

$$\Delta w = \eta R (A - \tanh \mathcal{A}) \phi'(u) x. \quad (10)$$

Note that the formal similarity to (5) and (7) where in the latter the ‘exploration term’ ( $y - \phi(u)$ ) is now replaced by  $(A - \tanh \mathcal{A})$ . The population reinforcement learning rule (10), however, is composed of 4-factors that originate from 4 different consecutive biological processing stages: (i) the presynaptic activity, (ii) a postsynaptic quantity, (iii) the population activity with the corresponding action, and (iv) the reward. Explicit expressions for (7) and (10) in the case of spiking neurons are given in [Spike-Timing Dependent Plasticity, Learning Rules](#).

**Online learning** To obtain online learning rules one considers an ongoing stimulus  $x_t$  that may e.g. define spike trains up to a discrete time step  $t$ . Actions  $A_t$  can be taken each moment  $t$ , and a reward signal  $R_t$  is potentially always present, although it is typically sparse. To express the weight change at each time step  $t$  one considers the so-called synaptic eligibility trace  $e_t$ . In the case of the rule (9) this is defined as a low-pass filtering of the instantaneous plasticity induction  $\text{PI}_t = \frac{d}{dw} \log P_w(A_t | x_t)$ , i.e.  $e_{t+1} = \gamma e_t + \text{PI}_t$ , with some discount factor  $\gamma \in [0, 1)$ . Here, the derivative of the log-likelihood is evaluated as in (10), with the individual factors each being low-pass filtered to obtain a fully online rule. The final rule then becomes

$$\Delta w_t = \eta R_t e_t. \quad (11)$$

By low-pass filtering the corresponding instantaneous plasticity induction terms  $\text{PI}_t$  occurring in (5), (7) and (8) (compare with the general form of  $\Delta w$  given in Eq. 1) one analogously obtains the online version for these rules.

Assuming that the states  $x_t$  are sampled from a Partially Observable Markov Decision Process (POMDP), the online learning rule (11) can be shown to maximize the expected discounted future reward

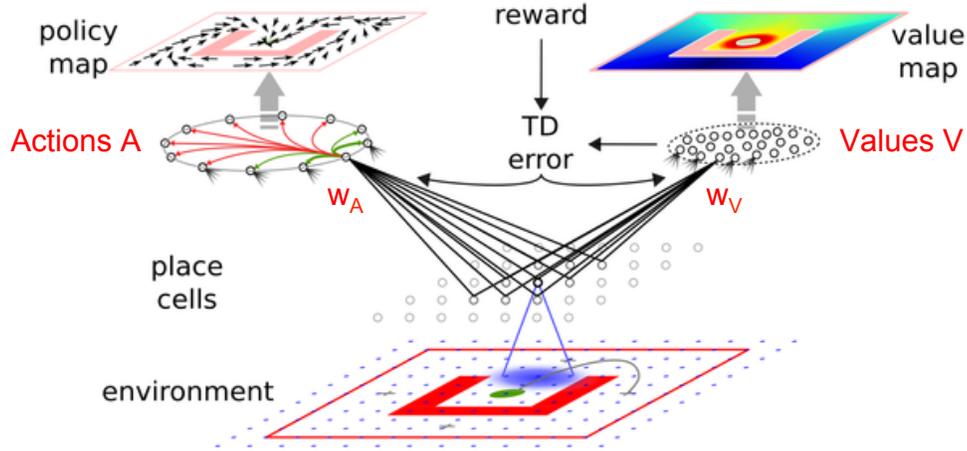
$$V_t = \sum_{k=0}^{\infty} \gamma^k \langle R_{t+k} \rangle \quad (12)$$

for each time step  $t$ , see Baxter and Bartlett (2001), Theorem 5, for the general case, and Friedrich et al. (2011), Supporting Information, for population learning.

**Phenomenological R-STDP models** The gradient-based learning rules discussed so far prevent a systematic weight drift by assuring that the plasticity induction in average vanishes,  $\langle \text{PI} \rangle = 0$ . This was also assumed in reward-modulated spike-timing dependent plasticity (R-STDP, Izhikevich (2007)). Nevertheless, when considering specific (*in vitro*) data on plasticity induction one will typically have  $\langle \text{PI} \rangle \neq 0$  in Eq. 1 (Sjöström and Gerstner, 2010). It is therefore plausible that the reward signal itself in average vanishes. One strategy to achieve this is to subtract the mean reward from the modulating factor such that the learning rule, now reading as  $\Delta w = (R - \langle R \rangle) \cdot \text{PI}$ , is again unbiased. This leads to the concept of actor-critique learning where a stimulus-specific internal critique adapts the global modulation signal of the plasticity induction to prevent drifts even for individual stimuli. When the LTD-part in

the STDP window is suppressed and the remaining R-STDP is bias-corrected, the learning speed for standard association tasks comes close to the one for gradient-based spike reinforcement (Frémaux et al., 2010).

An elegant solution to solve the reward-bias problem is to assume that the internal reward signal is shaped by a temporal kernel that sums up to zero across time,  $\int R_t dt = 0$ , and hence a positive internal reward signal must be followed or preceded by a negative one (Legenstein et al., 2008). What appears as a computational trick is reminiscent to the observed relieve from pain in fruit flies (Tanimoto et al., 2004), or the reward baseline adaptation in rodents (Schultz et al., 1997). Along similar lines, pairing reward with differential Hebbian plasticity was also shown to be bias-free and asymptotically equivalent to temporal difference learning (Kolodziejski et al., 2009).



**Figure 2.** Actor-critic network for navigation. Based on the current position information the action network chooses action  $A$  that moves the agent towards the neighboring position closest to the site of reward delivery (inside the U-shaped obstacle). The critic network calculates the TD error  $\delta_t$  based on the values  $V$  and the reward signal  $R_t$ . The synaptic strengths  $w_A$  and  $w_V$  of the action selection and the value representation network, respectively, are both adapted by a TD learning rule. Figure adapted from Frémaux et al. (2013).

**B) Temporal Difference (TD) learning** TD methods represent a class of learning rules where the modulatory feedback signal on the level of the synapse is again unbiased. These methods consider a learning scenario where an agent navigates through different states to eventually reach a final rewarding state.

The actor-critic version of TD learning is to calculate a value for each state, and to use the value updates to also train the action selection network (Fig. 2). The value of a state  $x$  is defined as the expected discounted future rewards when being in that state at time  $t$  (Sutton and Barto, 1998),

$$V^\pi(x_t) = \sum_{k=0}^{\infty} \gamma^k \langle R_{t+k} \rangle \quad (13)$$

where  $\gamma$  denotes the discounting factor and the expectation  $\langle \cdot \rangle$  is taken over all future actions according to a fixed policy  $\pi$ , see Sutton and Barto (1998). While this definition involves future rewards, an online version would again need to estimate the quantity based on only past experiences. It is therefore interesting to note that (13) can be rewritten in a recursive way as  $V^\pi(x_t) = \langle R_t \rangle + \gamma V^\pi(x_{t+1})$ .

The value function  $V^\pi(x_t)$  is assumed to be represented (and approximated) by the activity  $V_w(x_t)$  of a given neuron, where  $w$  denotes the strength of synapses converging to that neuron and  $x_t$  is the input to the network. The synaptic strengths can be adapted online by gradient descent on the error function  $E = \langle V(x) - V_w(x) \rangle^2$  with respect to  $w$ , where the expectation is over the states  $x$  (Sutton and Barto, 1998; Frémaux et al., 2013). The weight change at time  $t$  is therefore given by

$$\Delta w_t = \eta (V^\pi(x_t) - V_w(x_t)) \frac{dV_w(x_t)}{dw}. \quad (14)$$

Since the true value  $V(x_t)$  is not known to the network it can be approximated as  $V^\pi(x_t) \approx R(t) + \gamma V_w(x_{t+1})$  by using the recursive definition of  $V^\pi(x_t)$ . This leads to the learning rule

$$\Delta w_t = \eta \delta_t \frac{dV_w(x_t)}{dw}, \quad (15)$$

where  $\delta_t$  is called the temporal difference (TD) error and is given by

$$\delta_t = R_t + \gamma V_w(x_{t+1}) - V_w(x_t) \quad (16)$$

If the network that encodes the value  $V_w(x)$  consists of a single neuron, say  $V_w(x) = \phi(u)$  with  $u = wx$ , then the synaptic learning rule can be expressed as

$$\Delta w_t = \eta \delta_t \phi'(u) x_t. \quad (17)$$

As compared to the policy gradient rules above, the TD learning rule (17) is obtained by replacing the reward  $R$  in Eq. 1 with the TD- $\delta$ . Since this  $\delta$  converges to zero during learning, any systematic weight drift is also suppressed.

TD learning in the form of actor-critic has been implemented in spiking neuronal networks (Castro et al., 2009; Potjans et al., 2009; Potjans et al., 2011; Frémaux et al., 2013). In these implementations, separate networks for the value representation and the action selection are considered, and the synaptic strengths  $w$  in both networks are adapted based on the temporal difference  $\delta_t$  (Fig. 2). An alternative to this actor-critic learning is to learn values for state-action pairs ('Q-values') and choose actions based on these values ('SARSA', see Sutton and Barto (1998)). Because the evaluation of a value for a state-action pair in a standard neuronal implementation requires to actually choose that action, however, value evaluation in order to decide for a single next action cannot be implemented in this straightforward form.

**Policy gradient versus TD methods** Both, online policy gradient and TD learning, maximize the discounted future reward as expressed by (12) and (13). But while policy gradient methods do not require specific structures on the network nor the task, TD methods do so. First, TD learning assumes an internal representation of states  $x$ . Second, the definition of the TD-error involves two subsequent states, and hence value learning requires to sample all the corresponding transitions. Third, to assign values to a state, it is implicitly assumed that the history of reaching that state does not influence future rewards. In fact, TD learning assumes that the underlying decision process is Markovian.

Gradient-based learning does not assume an internal representation of states to which values would be assigned. Neither does gradient-based learning assume Markovianity of the underlying decision process for convergence. When this decision process is not Markovian, TD learning can fail in both ways, by either choosing inappropriate actions while correctly estimating values, or by incorrectly estimating the values themselves (Friedrich et al., 2011). Yet, when the decision process is Markovian, TD learning becomes faster than policy gradient learning (Frémaux et al., 2013). Note that a decision process can always be made Markovian by expanding the state-space representation and including hidden states to which again values need to be assigned. To model how the brain can create such hidden states, however, remains a challenge (Dayan and Niv, 2008).

**Cortical implementations** Due to their conceptual simplicity, policy gradient methods may be implemented in any cortical network that is engaged in stimulus-response associations and that receives feedback via some global neuromodulator. TD learning was related to basal ganglia where specific networks were suggested to represent values (Daw et al., 2006; Wunderlich et al., 2012) and dopamine activity was suggested to represent the TD-error  $\delta_t$  (Schultz et al., 1997).

Both, policy gradient and TD learning are considered as model-free, although TD learning makes use of some information about the underlying model. If more information is included such as state-transition probabilities, the learning can again become faster as less sampling is required to explore the reward function. TD learning methods and their extensions have in particular been proven successful in interpreting human cortical activity during decision making tasks (for a review see [Reward-Based Learning, Model-Based and Model-Free](#)).

## Cross-References

Spike-Timing Dependent Plasticity, Learning Rules

Reward-Based Learning, Model-Based and Model-Free

Decision Making Tasks

Basal Ganglia – Decision Making

## References

- Baxter J, Bartlett P (2001) Infinite-horizon policy-gradient estimation. *J. Artif. Intell. Res.* 15:319–350.
- Castro D, Volkinshtein S, Meir R (2009) Temporal difference based actor critic learning: convergence and neural implementation In *In: Advances in neural information processing systems*, Vol. 21, pp. 385 – 392, Cambridge, MA. MIT Press.
- Daw ND, O’Doherty JP, Dayan P, Seymour B, Dolan RJ (2006) Cortical substrates for exploratory decisions in humans. *Nature* 441:876–879.
- Dayan P, Niv Y (2008) Reinforcement learning: the good, the bad and the ugly. *Curr. Opin. Neurobiol.* 18:185–196.
- Fiete IR, Seung HS (2006) Gradient learning in spiking neural networks by dynamic perturbation of conductances. *Phys. Rev. Lett.* 97:048104.
- Florian RV (2007) Reinforcement learning through modulation of spike-timing-dependent synaptic plasticity. *Neural Comput* 19:1468–1502.
- Frémaux N, Sprekeler H, Gerstner W (2010) Functional requirements for reward-modulated spike-timing-dependent plasticity. *J Neurosci* 30:13326–13337.
- Frémaux N, Sprekeler H, Gerstner W (2013) Reinforcement learning using a continuous time actor-critic framework with spiking neurons. *PLoS Comput. Biol.* 9:e1003024.
- Friedrich J, Urbanczik R, Senn W (2011) Spatio-temporal credit assignment in neuronal population learning. *PLoS Comput Biol* 7:e1002092.
- Friedrich J, Urbanczik R, Senn W (2014) Code-specific learning rules improve action selection by populations of spiking neurons. *Int. J. of Neural Systems* 24:1–17.

Izhikevich EM (2007) Solving the distal reward problem through linkage of STDP and dopamine signaling. *Cereb Cortex* 17:2443–2452.

Kolodziejcki C, Porr B, Worgotter F (2009) On the asymptotic equivalence between differential Hebbian and temporal difference learning. *Neural Comput* 21:1173–1202.

Legenstein R, Pecevski D, Maass W (2008) A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. *PLoS Comput. Biol.* 4:e1000180.

Pfister J, Toyoizumi T, Barber D, Gerstner W (2006) Optimal spike-timing-dependent plasticity for precise action potential firing in supervised learning. *Neural Comput* 18:1318–1348.

Potjans W, Diesmann M, Morrison A (2011) An imperfect dopaminergic error signal can drive temporal-difference learning. *PLoS Comput. Biol.* 7:e1001133.

Potjans W, Morrison A, Diesmann M (2009) A spiking neural network model of an actor-critic learning agent. *Neural Comput* 21:301–339.

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

Seung HS (2003) Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. *Neuron* 40:1063 – 1073.

Sjöström J, Gerstner W (2010) Spike-timing dependent plasticity. *Scholarpedia* 5:1362.

Sutton RS, Barto AG (1998) *Reinforcement Learning: An Introduction* MIT Press, Cambridge, MA.

Tanimoto H, Heisenberg M, Gerber B (2004) Experimental psychology: event timing turns punishment to reward. *Nature* 430:983.

Urbanczik R, Senn W (2009) Reinforcement learning in populations of spiking neurons. *Nat Neurosci* 12:250–252.

Williams RJ (1992) Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach Learn* 8:229–256.

Wunderlich K, Dayan P, Dolan RJ (2012) Mapping value based planning and extensively trained choice in the human brain. *Nat. Neurosci.* 15:786–791.

Xie X, Seung HS (2004) Learning in neural networks by reinforcement of irregular spiking. *Phys Rev E Stat Nonlin Soft Matter Phys* 69:041909.