A reinforcement learning algorithm for spiking neural networks

Răzvan V. Florian
Center for Cognitive and Neural Studies (Coneural), Cluj-Napoca, Romania
LIRA-Lab, DIST, University of Genoa, Genoa, Italy
Department of Computer Science, Babeş-Bolyai University, Cluj-Napoca, Romania
florian@coneural.org

Abstract

The paper presents a new reinforcement learning mechanism for spiking neural networks. The algorithm is derived for networks of stochastic integrate-and-fire neurons, but it can be also applied to generic spiking neural networks. Learning is achieved by synaptic changes that depend on the firing of pre- and postsynaptic neurons, and that are modulated with a global reinforcement signal. The efficacy of the algorithm is verified in a biologically-inspired experiment, featuring a simulated worm that searches for food. Our model recovers a form of neural plasticity experimentally observed in animals, combining spike-timing-dependent synaptic changes of one sign with non-associative synaptic changes of the opposite sign determined by presynaptic spikes. The model also predicts that the time constant of spike-timing-dependent synaptic changes is equal to the membrane time constant of the neuron, in agreement with experimental observations in the brain. This study also led to the discovery of a biologically-plausible reinforcement learning mechanism that works by modulating spike-timing-dependent plasticity (STDP) with a global reward signal.

1. Introduction

Spiking neural networks [22, 15] are considered to be the third generation of neural networks [20]. It was shown that they have more computational power per neuron than networks from the previous generations (with McCulloch-Pitts neurons or with continuous sigmoidal activation functions) [21]. The main interest for studying spiking neural networks is, however, their close resemblance with biological neural networks. This permits drawing inspiration from experimental neuroscience when designing neural models, and using the knowledge gained from simulations and theoretical analysis of the models to better understand the activity of the brain.

Reinforcement learning algorithms for spiking neural networks are important especially in the context of embodied computational neuroscience, where an agent controlled by a spiking neural networks learns by interacting with an environment. Ideally, the agent should develop its own internal representations of the environment through unsupervised or reinforcement learning, without supervised learning, in order to minimize the biases induced by the human programmer [14].

An existing reinforcement learning algorithm for spiking neural networks works by correlating fluctuations in irregular spiking with a reward signal, in networks composed of neurons firing Poisson spike trains [33]. This algorithm highly depends on the Poisson characteristic of the neurons and needs injecting noise in neurons when using commonly used neural models, such as the integrate-and-fire neuron. It is thus difficult to use in conjunction with these neural models. Also, this learning model presumes that neurons respond instantaneously, by modulating their firing rate, to their input. This partly ignores the memory of the neural membrane potential, an important characteristic of spiking neural models. Another reinforcement learning algorithm that can be used for spiking neural networks works by reinforcing stochastic synaptic transmission [30].

Here, we present a new reinforcement learning algorithm for spiking neural networks. The algorithm is derived analytically for networks of probabilistic (stochastic) integrate-and-fire neurons, and is tested on networks of both probabilistic and deterministic neurons. The learning rule that we propose is local to the synapse, assuming that the reinforcement signal is diffusely distributed into the network: synaptic changes depend on the reinforcement and the activity of the pre- and postsynaptic neurons.

We first present, in Section 2, the derivation of the algorithm. We discuss next the relationship of the proposed algorithm with other similar algorithms (Section 3), and its relevance to neuroscience (Section 4). In Section 5 we describe experiments that validate the learning capabilities of the method. The last Section is dedicated to the conclusions.

Proceedings of the Seventh International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC'05)
0-7695-2453-2/05 $20.00 © 2005 IEEE

2. Derivation of the algorithm

2.1. Analytical derivation

The algorithm we propose is derived as an application of the OLPOMDP reinforcement learning algorithm [9, 8], an online variant of the GPOMDP algorithm [3, 7]. GPOMDP assumes that the interaction of an agent with an environment is a partially observable Markov decision process, and that the agent chooses actions according to a probabilistic policy μ that depends on a vector θ of several real parameters. GPOMDP was derived analytically by maximizing the long-term average of the reward received by the agent. Results related to the convergence of OLPODMP to local maxima have been obtained [2, 23, 24].

We consider a neural network that evolves in discrete time. At each timestep t, a neuron i either fires ($f_i(t) = 1$) with probability $\sigma_i(t)$, or does not fire ($f_i(t) = 0$) with probability $1 - \sigma_i(t)$. The neurons are connected through plastic synapses with efficacies $w_{ij}(t)$, where i is the index of the postsynaptic neuron. The efficacies w_{ij} can be either positive or negative (corresponding to excitatory and inhibitory synapses, respectively). A global reward signal $r(t)$ is broadcast to all synapses.

By considering each neuron i as an independent agent, the firing/non-firing probabilities of the neuron as the policy μ_i of the corresponding agent, the weights w_{ij} of the incoming synapses as the vector θ_i that parameterizes the agent's policy, and the firing states f_j of the presynaptic neurons as the observation of the environment available to the agent, we may apply OLPODMP to the neural network. The result is the following plasticity rules that update the synapses such as to optimize the long term average of the reward received by the network:

$$w_{ij}(t+\delta t) = w_{ij}(t) + \gamma\, r(t+\delta t)\, z_{ij}(t+\delta t) \tag{1}$$

$$z_{ij}(t+\delta t) = \beta\, z_{ij}(t) + \zeta_{ij}(t) \tag{2}$$

$$\zeta_{ij}(t) = \begin{cases} \dfrac{1}{\sigma_i(t)} \dfrac{\partial \sigma_i(t)}{\partial w_{ij}}, & \text{if } f_i(t) = 1 \\ -\dfrac{1}{1-\sigma_i(t)} \dfrac{\partial \sigma_i(t)}{\partial w_{ij}}, & \text{if } f_i(t) = 0, \end{cases} \tag{3}$$

where δt is the duration of a timestep, the learning rate γ is a small constant parameter, z is an eligibility trace, and ζ is a notation for the change of z resulted from the activity in the last timestep. The discount factor β is a parameter that can take values between 0 and 1, and that can also be written as $\beta = \exp(-\delta t/\tau_z)$, where τ_z is a time constant for the exponential decay of z.

Up to now, we have followed a derivation also performed in [4, 5]. However, unlike these studies, which dealt with networks of memoryless binary stochastic units, from now on we will consider here networks of stochastic leaky integrate-and-fire neurons, that evolve in discrete time according to:

$$V_i(t) = V_i(t-\delta t)\, \exp\left(-\delta t/\tau_i\right) + \sum_j w_{ij}(t-\delta t) f_j(t-\delta t) \tag{4}$$

where τ_i is the leakage time constant, and the sum on the right represents the growth of the potential caused by the injection of current during a timestep by the firing of presynaptic neurons. The neuron fires stochastically with probability $\sigma(V_i(t))$. This corresponds to a noisy threshold of the neuron, also called escape noise in spiking neural models [15]. We thus have $\partial\sigma_i(t)/\partial w_{ij} = \partial\sigma_i(t)/\partial V_i \;\; \partial V_i(t)/\partial w_{ij}$. If the neuron fires ($f_i(t) = 1$), the potential is reset to a base value (reset potential), $V_i(t) = V_r$. By expanding Eq. 4 back in time, up to the moment $t - n\,\delta t$ when neuron i has fired most recently, we get

$$V_i(t) = V_r \exp\left(-\frac{n\,\delta t}{\tau_i}\right) + \sum_{k=1}^{n} \left(\exp\left(-\frac{(k-1)\,\delta t}{\tau_i}\right) \sum_j w_{ij}(t-k\,\delta t) f_j(t-k\,\delta t) \right). \tag{5}$$

If we neglect the variation of w_{ij} in the interval between two postsynaptic spikes, an approximation that is justified because the parameter γ and/or the value of z are small, we have:

$$\frac{\partial \sigma_i(t)}{\partial w_{ij}} = \frac{\partial \sigma_i(t)}{\partial V_i} \sum_{k=1}^{n} f_j(t-k\,\delta t)\, \exp\left(-\frac{(k-1)\,\delta t}{\tau_i}\right). \tag{6}$$

By introducing this back in Eq. 3, we have:

$$\zeta_{ij}(t) = \begin{cases} \dfrac{1}{\sigma_i} \dfrac{\partial \sigma_i}{\partial V_i} \displaystyle\sum_{k=1}^{n} f_j(t-k\,\delta t)\, \exp\left(-\frac{(k-1)\,\delta t}{\tau_i}\right), \\ \qquad\qquad \text{if } f_i(t) = 1 \\ -\dfrac{1}{1-\sigma_i} \dfrac{\partial \sigma_i}{\partial V_i} \displaystyle\sum_{k=1}^{n} f_j(t-k\,\delta t)\, \exp\left(-\frac{(k-1)\,\delta t}{\tau_i}\right), \\ \qquad\qquad \text{if } f_i(t) = 0. \end{cases} \tag{7}$$

We see that, when a postsynaptic spike follows one or more presynaptic spikes that were emitted after the previous postsynaptic spike, ζ is positive, as $0 \leq \sigma \leq 1$ because it is a probability and $\partial\sigma/\partial V \geq 0$ because the firing probability increases with higher membrane potential. We have thus a spike-timing-dependent potentiation of z. The depression

of z is non-associative, as each presynaptic spike decreases z continuously until a postsynaptic spike is emitted.

We may choose to model the escape noise σ as a bounded exponential function [15]:

$$\sigma_i(V_i) = \begin{cases} \delta t/\tau_\sigma \exp(\beta_\sigma\ (V_i - \theta_i)), & \text{if less than 1} \\ 1, & \text{otherwise} \end{cases} \quad (8)$$

where θ_i is the threshold potential of the neuron and τ_σ and β_σ are constant positive parameters. For exponential escape noise, $\partial\sigma_i/\partial V_i = \beta_\sigma\ \sigma_i$ and we have:

$$\zeta_{ij}(t) = \begin{cases} \beta_\sigma \sum_{k=1}^{n} f_j(t - k\,\delta t)\ \exp\left(-\dfrac{(k-1)\,\delta t}{\tau_i}\right), \\ \qquad\qquad \text{if } f_i(t) = 1 \\ -\dfrac{\beta_\sigma\ \sigma(t)}{1-\sigma(t)} \sum_{k=1}^{n} f_j(t - k\,\delta t)\ \exp\left(-\dfrac{(k-1)\,\delta t}{\tau_i}\right), \\ \qquad\qquad \text{if } f_i(t) = 0. \end{cases} \quad (9)$$

Hence, a spike pair consisting of a postsynaptic spike emitted at t and a presynaptic spike emitted at $t - k\,\delta t$ leads to a potentiation of z with $\beta_\sigma\ \exp\left(-(k-1)\,\delta t/\tau_i\right)$. We thus have exactly an exponential dependence of the potentiation of z on the relative spike timing.

The timestep δt should be small enough in order to keep σ to values much smaller than 1. If σ is allowed to take values close to 1, ζ may diverge. The parameter τ_σ should be chosen as a function of δt, since, if all parameters of the exponential escape noise function remain constant, the firing probability within a given finite time varies as a function of δt.

The parameters β and γ should be chosen such that $1/(1-\beta)$ and $1/\gamma$ are large compared to the mixing time of the system. The mixing time can be defined rigourously for a Markov process and can be thought of as the time from the occurrence of an action until the effects of that action have died away. However, β cannot be set arbitrarily close to 1, since this induces a large variance in the estimation of the gradient towards the optimum. Thus, there is a bias-variance trade-off in setting β. Detailed information about this is provided in [9, 3, 8, 7, 6, 2].

2.2. Generalization to other neural models

The neural models that are commonly used in simulations are usually deterministic. In this case, a straightforward generalization of the reinforcement learning algorithm is to still use Eq. 9 while considering σ as a constant parameter. Also, if the neural model used is not leaky integrate-and-fire, τ_i in the same equation may characterize only the plasticity mechanism, with no connection to neuronal dynamics. We will show that the algorithm proves in experiments to be effective even after these generalizations.

The algorithm is not constrained by the use of discrete time, and can be easily reformulated for continuous time by taking the limit $\delta t \to 0$.

3. Relationship and comparison to other reinforcement learning algorithms for spiking neural networks

It can be shown that the algorithm proposed here shares a common analytical background with the other two existing reinforcement learning algorithms for spiking neural networks [30, 33].

Seung [30] applies OLPOMDP by considering that a synapse is the agent, instead of the neuron, as we did. The action of the agent is the release of a neurotransmitter vesicle, instead of the spiking of the neuron, and the parameter that is optimized is a parameter that controls the release of the vesicle, instead of the synaptic connections to the neuron. The result is a learning algorithm that is biologically plausible, but for which there exists no experimental evidence yet.

Xie and Seung [33] do not model in detail the integrative characteristics of the neural membrane potential, and consider that neurons respond instantaneously to inputs by changing the firing rate of their Poisson spike train. The study derives a episodic algorithm that is similar to GPOMDP, and extends it to an online algorithm similar to OLPOMDP without any justification. It can be seen that the expression of the eligibility trace, Eq. 13 in [33], is identical, after adapting notation and ignoring the sum due to the episodic nature of the algorithm, to the one in our paper, Eq. 3. By reinterpreting the current-discharge function f in [33] as the escape function σ, and the decaying synaptic current h_{ij} as the contribution of a presynaptic spike to the membrane potential V_i of the neuron that decays due to leakage, we can see that the algorithm of Xie and Seung is mathematically equivalent to the algorithm derived, more accurately, here. However, our different implementation of the mathematical framework permits a straightforward generalization and application to neural models commonly used in simulations, which the Xie and Seung algorithm does not permit because of the inescapable dependence on the Poisson characteristic of the neurons.

The common theoretical background of the three algorithms suggests that their learning performance should be similar.

4. Relevance to neuroscience: Reinforcement learning and spike-timing-dependent plasticity (STDP)

The plasticity rule implied by our algorithm for probabilistic integrate-and-fire neurons with exponential escape noise features an exponential dependence of the potentiation of z on the time interval between pre- and postsynaptic spikes, and a non-associative depression of z dependent on presynaptic spikes, as shown in Section 2. If the reinforcement r is negative, the potentiation of z determines a depression of w, and we thus have spike-timing-dependent depression and non-associative potentiation of the synaptic efficacy. This type of plasticity has been experimentally discovered in the brain of the electric fish [16], and is a particular form of spike-timing-dependent plasticity (STDP).

STDP is the dependence of synaptic changes on the relative timing of pre- and postsynaptic action potentials, a phenomenon that was experimentally observed in a variety of biological neural systems [25, 10, 12]. The typical example of STDP is given by the potentiation of a synapse when the postsynaptic spike follows the presynaptic spike within a time window of a few tens of milliseconds, and the depression of the synapse when the order of the spikes is reversed. The dependence is approximately exponential.

If the reinforcement r is positive, our model recovers the exponential characteristic of the typical spike-timing-dependent potentiation. However, the model predicts non-associative depression in this case, instead of the spike-timing-dependent, associative depression typically observed. In an experiment presented below, we have replaced the non-associative depression of z implied by the derived algorithm with exponential spike-timing depression. We have thus explored the learning properties of modulation of standard STDP by the reinforcement signal. We found that reward-modulated STDP is also effective as a reinforcement learning mechanism. A similar result, published in abstract form, seems to have been found independently [13].

Our model also predicts that the time constant of the spike-time dependent potentiation time window is equal to the membrane time constant of the neuron, τ_i. This is consistent with experimental observations, as the two time constants have the same order of magnitude—a few tens of milliseconds. It was already speculated that the two time constants should be comparable, since this ensures that only those presynaptic spikes that arrive within the temporal range over which a neuron integrates its inputs are potentiated, enforcing the requirement of causality that is a typical characteristic of STDP [1]. The same prediction also results from a theoretical model that considers STDP as a mechanism that reduces variability in the postsynaptic spike train [11].

It is remarkable that our model recovers these features of the brain given that the only ingredients that led to the model are the abstract framework of the partially observable Markov decision process following a parameterized probabilistic policy, and a probabilistic version with exponential escape noise of the integrate-and-fire neuron, the simplest and the most widely used spiking neural model.

Our algorithm implies that STDP is modulated by the reward signal r. This may be implemented in the brain by a neuromodulator. For example, dopamine carries a short-latency reward signal indicating the difference between actual and predicted rewards [28] that fits well our learning model based on continuous reward-modulated plasticity. It is known that dopamine and acetylcholine modulate classical (firing rate dependent) long term potentiation and depression of synapses [29, 17, 32]. Current experimental evidence of modulation of STDP by neuromodulators is limited to the discovery of amplification of spike-time dependent potentiation in hippocampal CA1 pyramidal neurons by the activation of β-adrenergic receptors [19, Fig. 6F]. As speculated before [33], it may be that other studies failed to detect the influence of neuromodulators on STDP because they were performed in vitro, where the reward circuitry may not have worked, and the reward signal may have been fixed to a given value.

Thus, the derivation of this algorithm led to the finding of a reinforcement mechanism that is highly biologically plausible and that may operate in our brains.

5. Experiments and results

5.1. Experimental design

We tested the proposed reinforcement learning mechanisms on a biologically-inspired problem. We simulated an imagined aquatic worm that had one end fixed to a support and had a mouth at the other end (see Fig. 1). There was a source of food near the worm, which diffused a gradient of a chemical substance in the water. The worm perceived this gradient. If the mouth moved to a position with a higher concentration of substance, the worm's neural system received a positive reward; if it moved to a position with a lower concentration, the reward was negative. By learning to maximize the average reward, the worm should have moved its mouth towards the food source.

The design of the experiment was motivated by the following considerations. The learning mechanism that we investigated is biologically plausible, hence we wanted to test it in a biologically-inspired framework. We used an embodied agent because of our belief that computational neuroscience models are best studied in an embodied context ([14]; see also [27]). The size of the neural network that we could simulate was restricted by computational constraints.

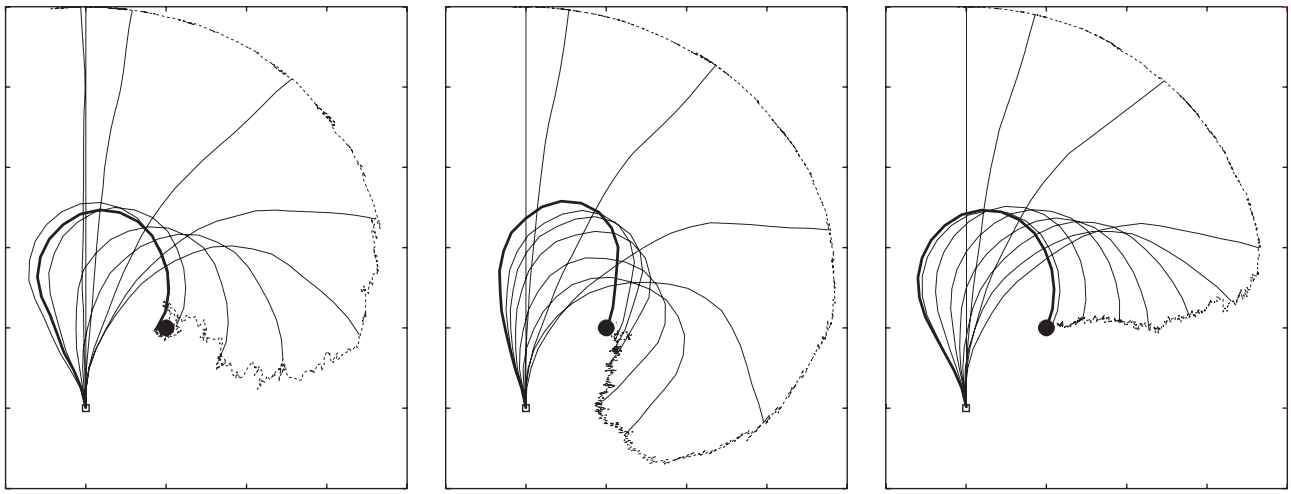

Figure 1. The movement of the worm during the first 30 s of three experiments where the food source is in a difficult to reach position. The dotted line represents the trajectory of the mouth of the worm. The thin lines represent snapshots of the position of the worm, taken every 3 s. The thick line represents the position of the worm at 30 s after the beginning of the experiment. The filled circle indicates the position of the food source, the open square indicates the position of the base of the worm. a) Experiment 1: The worm is controlled by a network of stochastic integrate-and-fire neurons. b) Experiment 2: The worm is controlled by a network of Izhikevich neurons. c) Experiment 3: The worm is controlled by a network of Izhikevich neurons featuring reward-modulated STDP.

Thus the animal we simulated should have been very simple, according to the size of the neural network. In the design of the simulated animal, we followed the principles stated in [26]. While we believe that the setup is of biological relevance, we did not attempt to model a particular real animal.

We simulated the worm as a chain of 20 articulated linear segments in two dimensions. The worm could move each articulation to a relative angle within $\pm\theta_{max} = 25^{\text{o}}$ by commanding two antagonist effectors.

The simulated animal was controlled by a spiking neural network. In some experiments, the network was composed of probabilistic integrate-and-fire neurons, described above. In other experiments, the network was composed of deterministic Izhikevich neurons [18]. The network was composed of a set of input neurons, a set of hidden neurons and a set of motor neurons, with random connectivity. At each time step, the network received a reward $r(t) = 1$ if the mouth moved closer to the food source, or $r(t) = -1$ if the mouth moved farther.

The input neurons conveyed proprioceptive information about the articulations. For each articulation there were 4 corresponding input neurons. The activation of 2 of them was proportional to the angle between the orientation of the articulation and the leftmost possible orientation; the activation of the other 2 corresponded to the angle between the orientation of the articulation and the rightmost possible orientation. The activations were normalized between 0 and 1. The input neurons fired Poisson spike trains, with a firing rate proportional to the activation, between 0 and 50 Hz.

The spikes of motor neurons were converted to effector activations by integrating them with a leaky accumulator of time constant $\tau_e = 2$ s. This is equivalent to performing a weighted estimate of the firing rate using an exponential kernel with the same time constant. The motor activations a evolved according to $a(t) = a(t - \delta t)\ \exp(-\delta t/\tau_e) + (1 - \exp(-1/\nu_e\tau_e))f(t)$, where $f(t)$ indicates whether the motor neuron has fired. The factor that weighed the contribution $f(t)$ of the spikes insured that the activation was normalized to 1 when the neuron fired regularly with frequency $\nu_e = 25$ Hz; the activation was also limited to the interval $[0, 1]$ through hard bounds. The activations of 2 motor neurons were averaged to yield the activation of one effector. There were two antagonist effectors per articulation; if their activations were a_+ and a_-, the articulation was set to a relative angle $(a_+ - a_-)\theta_{max}$. The network was thus composed of 80 input neurons and 80 motor neurons. The network also had 200 hidden neurons (the number of hidden neurons was chosen to be somehow larger than the total number of input and motor neurons, for biological rel-

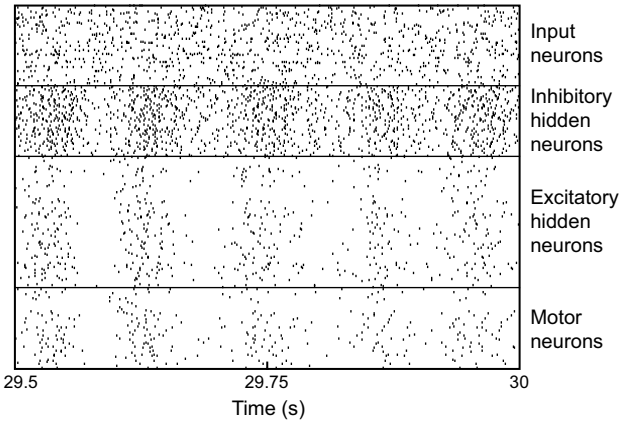

Figure 2. A spikegram representing the activity of the neural network in the last 0.5 s of the movement represented in Fig. 1 b)

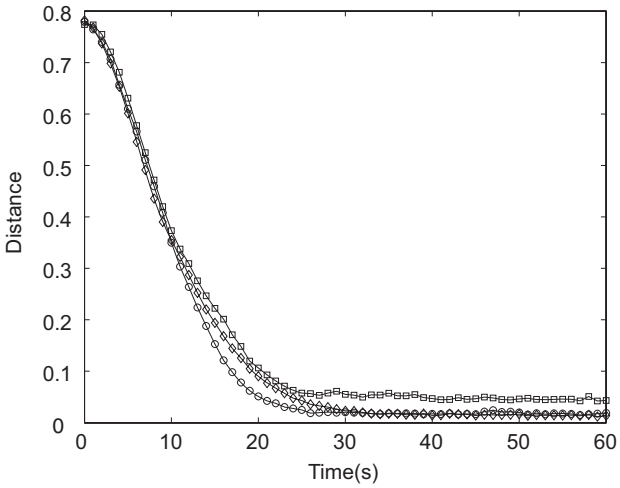

Figure 3. The evolution in time of the average distance between the mouth of the worm and the food source. Squares represent data for experiment 1, circles correspond to experiment 2, diamonds correspond to experiment 3. Data for each curve is extracted from 100 trials; for each trial the food is placed in a random position, in a semicircle with radius of 0.8 times the worm length, centered on the worm's base.

evance). Each neuron in the network sent axons to 15 % of hidden and motor neurons, chosen randomly.

The network was simulated with a time step $\delta t = 1$ ms. Since effector activations were initially set to 0 and bending of an articulation required an imbalance in the activities of the antagonist effectors, the initial position of the worm was vertical.

5.2. Experiment 1: Training a network of probabilistic integrate-and-fire neurons

In the first experiment, the worm was controlled by a network of probabilistic integrate-and-fire neurons, for which the reinforcement learning algorithm was derived. The neurons had a time constant $\tau_i = 20$ ms, reset potential $V_r = 10$ mV, threshold potential $\theta_i = 16$ mV, exponential escape noise with $\tau_\sigma = 20$ ms and $\beta_\sigma = 0.2$ mV^{-1}. The synaptic weights evolved according to Eq. 9, were hard-bounded within $[w_{min}, w_{max}]$ and were initialized with random values at the beginning of the experiment. For synapses from input neurons, $w_{min} = -0.1$ mV, $w_{max} = 1.5$ mV; for the other synapses, $w_{min} = -0.4$ mV, $w_{max} = 1$ mV. There were no axonal delays. The parameters of the learning algorithm were $\tau_z = 5$ ms, $\gamma = 0.025$ mV$^2 \cdot (w_{max} - w_{min})$.

5.3. Experiment 2: Training a network of Izhikevich neurons

In a second experiment, we verified whether the learning algorithm can generalize to networks composed of other types of neurons. Thus, the simulated worm was controlled by a spiking neural network composed of Izhikevich-type neurons [18]. These deterministic model neurons have biologically plausible dynamics, similar to Hodgkin-Huxley-type neurons, but are suitable for large-scale simulation.

The learning algorithm included the generalizations mentioned in Section 2.2. For more biological realism, we also included in the model the dependence of the synaptic current on the membrane potential V_i of the postsynaptic neuron. We considered that synapses are characterized by a positive conductance g_{ij} that depends on presynaptic activity, a positive efficacy w_{ij} hard-bounded between 0 and 1 that was modified during the learning process, according to Eq. 1, and a constant positive strength s_{ij}. At each timestep δt, a postsynaptic potential $s_{ij}\, w_{ij}(t)\, g_{ij}(t)\, (E_{ij} - V_i(t))\, \delta t$ was transmitted by a synapse to the postsynaptic neuron, where E_{ij} is the reversal potential of the synaptic channels. Conductances evolved according to $g_{ij}(t) = g_{ij}(t - \delta t)\, \exp(-\delta t/\tau_g) + f_j(t)$, where $f_j(t)$ is the spike train of the presynaptic neuron and τ_g is a decay time constant. In this case, the sign of the postsynaptic potential is given by $E_{ij} - V_i(t)$ and w is always positive. Hence, we set the parameter γ to be positive for excitatory synapses and negative for inhibitory ones, in order to maintain the same behavior as in the case where w is a signed quantity.

To agree with experimental evidence, we considered the neurons to be either inhibitory or excitatory (Dale's law).

Out of the 200 hidden neurons, 70 were inhibitory and the rest excitatory. The input and motor neurons were also excitatory. Inhibitory neurons were of 'fast spiking' type, the other neurons were 'regular spiking', with parameters from [18]. Axonal delays were chosen randomly between 1 ms and 40 ms. Excitatory synapses had synaptic strength $s = 0.1$ and reversal potential $E = 0$ mV; inhibitory synapses had $s = 0.2$ and $E = -90$ mV. The decay time constant of the conductances was $\tau_g = 5$ ms. The efficacies w were initialized randomly at the beginning of the experiment with values between 0 and 1. The parameters of the learning algorithm were $\tau_i = 20$ ms, $\beta_\sigma = 0.2$, $\sigma = 0.02$, $\tau_z = 5$ ms (we considered that the eligibility trace and the evoked postsynaptic potential evolve on the same time scale); γ was 0.025 for excitatory synapses and -0.025 for inhibitory synapses (the sign changes because we considered that w is always positive).

5.4. Experiment 3: Reinforcement learning through modulated STDP

In a third experiment, we investigated whether we could still have reinforcement learning if we consider that changes of z depend exclusively on the timing between pre- and postsynaptic spikes. In this case, the algorithm is a modulation, through the reinforcement r, of the standard form of STDP. As in previous studies [31, 1], we model STDP as having an exponential dependence on the relative spike timing, and we consider that the effect of different spike pairs is additive. Hence, in this experiment,

$$\zeta_{ij}(t) = A_+ f_i(t) \sum_{k=1}^{\infty} f_j(t - k\,\delta t) \exp\left(-\frac{(k-1)\,\delta t}{\tau_i}\right) - A_- f_j(t) \sum_{k=1}^{\infty} f_i(t - k\,\delta t) \exp\left(-\frac{(k-1)\,\delta t}{\tau_i}\right), \quad (10)$$

where $A_\pm$ are constant parameters, $A_+ = 0.005$, $A_- = 1.05\ A_+$ [31]. In a computer simulation, ζ can be computed by using two variables for each synapse, P^+ that tracks the influence of presynaptic spikes and P^- that tracks the influence of postsynaptic spikes:

$$\begin{aligned} P_{ij}^+(t) &= P_{ij}^+(t - \delta t) \exp(-\delta t/\tau_i) + A_+ f_j(t - \delta t) \\ P_{ij}^-(t) &= P_{ij}^-(t - \delta t) \exp(-\delta t/\tau_i) - A_- f_i(t - \delta t) \\ \zeta_{ij}(t) &= P_{ij}^+(t) f_i(t) + P_{ij}^-(t) f_j(t). \end{aligned} \quad (11)$$

The ζ computed like this was used in conjunction with Eqs. 1 and 2 to compute the synaptic changes. We used a network of Izhikevich neurons, with the same setup as in Experiment 2, except setting $\gamma = 1$ for excitatory synapses and $\gamma = -1$ for inhibitory synapses, since the synaptic changes were already scaled by the $A_\pm$ parameters.

5.5. Results

The simulated worm learned quite fast (in less than one minute of simulated time) to find the food source, in all three experiments. Fig. 1 illustrates typical trajectories of the worm. It was always able to perform the task if food is placed in the accessible area. After approaching the food, the mouth of the worm remained close to it, oscillating around, as illustrated in Fig. 3. Thus, the proposed algorithm is efficient as a reinforcement learning mechanism for networks of probabilistic integrate-and-fire neurons, for which it was designed, as well as for generic spiking neural networks, composed, for example, by Izhikevich neurons. Reward-modulated STDP was as effective for reinforcement learning as the analytically-derived algorithm.

6. Conclusion

In conclusion, we have derived a new reinforcement learning mechanism for spiking neural networks and have tested its efficacy in a biologically-inspired framework. The algorithm was derived for stochastic integrate-and-fire neurons, but we verified that it can be also applied to generic spiking neural networks. Our model recovers the exponential characteristic of the experimentally observed spike-time dependent potentiation, and predicts that the time constant of the spike-time dependent potentiation time window is equal to the membrane time constant of the neuron, in agreement with experimental observations in the brain. The algorithm has also led to the discovery of a reinforcement learning mechanism based on the modulation of spike-timing dependent plasticity, a mechanism that may operate in the brain.

7. Acknowledgments

Part of this work was sponsored by a grant of the University of Genoa and by the RobotCub project of the European Commission (IST-2004-004370). This work was also sponsored by Arxia SRL.

References

[1] L. F. Abbott and S. B. Nelson. Synaptic plasticity: taming the beast. *Nature Neuroscience*, 3:1178–1183, 2000.

[2] P. Bartlett and J. Baxter. Stochastic optimization of controlled partially observable Markov decision processes. In *Proceedings of the 39th IEEE Conference on Decision and Control*, 2000.

[3] P. L. Bartlett and J. Baxter. Direct gradient-based reinforcement learning: I. Gradient estimation algorithms. Technical report, Australian National University, Research School of Information Sciences and Engineering, 1999.

[4] P. L. Bartlett and J. Baxter. Hebbian synaptic modifications in spiking neurons that learn. Technical report, Australian National University, Research School of Information Sciences and Engineering, 1999.

[5] P. L. Bartlett and J. Baxter. A biologically plausible and locally optimal learning algorithm for spiking neurons. http://arp.anu.edu.au/ftp/papers/jon/brains.pdf.gz, 2000.

[6] P. L. Bartlett and J. Baxter. Estimation and approximation bounds for gradient-based reinforcement learning. In *Proceedings of the Thirteenth Annual Conference on Computational Learning Theory*, pages 133–141, San Francisco, CA, 2000. Morgan Kaufmann Publishers.

[7] J. Baxter and P. L. Bartlett. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research*, 15:319–350, 2001.

[8] J. Baxter, P. L. Bartlett, and L. Weaver. Experiments with infinite-horizon, policy-gradient estimation. *Journal of Artificial Intelligence Research*, 15:351–381, 2001.

[9] J. Baxter, L. Weaver, and P. L. Bartlett. Direct gradient-based reinforcement learning: II. Gradient ascent algorithms and experiments. Technical report, Australian National University, Research School of Information Sciences and Engineering, 1999.

[10] G.-Q. Bi and M.-M. Poo. Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, 18(24):10464–10472, 1998.

[11] S. M. Bohte and C. Mozer. Reducing spike train variability: A computational theory of spike-timing dependent plasticity. In L. K. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems*, volume 17, Cambridge, MA, 2004. MIT Press.

[12] Y. Dan and M.-M. Poo. Spike timing-dependent plasticity of neural circuits. *Neuron*, 44:23–30, 2004.

[13] M. A. Farries and A. L. Fairhall. Reinforcement learning with modulated spike timing-dependent plasticity. Programme of Computational and Systems Neuroscience Conference (COSYNE), 2005.

[14] R. V. Florian. Autonomous artificial intelligent agents. Technical Report Coneural-03-01, Center for Cognitive and Neural Studies, Cluj, Romania, 2003.

[15] W. Gerstner and W. M. Kistler. *Spiking neuron models*. Cambridge University Press, Cambridge, UK, 2002.

[16] V. Z. Han, K. Grant, and C. C. Bell. Reversible associative depression and nonassociative potentiation at a parallel fiber synapse. *Neuron*, 27:611–622, 2000.

[17] Y.-Y. Huang, E. Simpson, C. Kellendonk, and E. R. Kandel. Genetic evidence for the bidirectional modulation of synaptic plasticity in the prefrontal cortex by D1 receptors. *Proceedings of the National Academy of Sciences*, 101(9):3236–3241, 2004.

[18] E. M. Izhikevich. Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, 14:1569–1572, 2003.

[19] Y. Lin, M. Min, T. Chiu, and H. Yang. Enhancement of associative long-term potentiation by activation of beta-adrenergic receptors at CA1 synapses in rat hippocampal slices. *Journal of Neuroscience*, 23(10):4173–4181, 2003.

[20] W. Maas. Networks of spiking neurons: the third generation of neural network models. *Neural Networks*, 10:1659–1671, 1997.

[21] W. Maas. Noisy spiking neurons with temporal coding have more computational power than sigmoidal neurons. In M. Mozer, M. I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 211–217. MIT Press, Cambridge, MA, 1997.

[22] W. Maas and C. M. Bishop, editors. *Pulsed neural networks*. MIT Press, Cambridge, MA, 1999.

[23] P. Marbach and J. N. Tsitsiklis. Simulation-based optimization of Markov reward processes. In *Proceedings of the 38th Conference on Decision and Control*, 1999.

[24] P. Marbach and J. N. Tsitsiklis. Approximate gradient methods in policy-space optimization of Markov reward processes. *Discrete Event Dynamic Systems: Theory and Applications*, 13(1–2):111–148, 2000.

[25] H. Markram, J. Lübke, M. Frotscher, and B. Sakmann. Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, 275(5297):213–215, 1997.

[26] R. Pfeifer and C. Scheier. *Understanding intelligence*. MIT Press, Cambridge, MA, 1999.

[27] E. Ruppin. Evolutionary embodied agents: A neuroscience perspective. *Nature Reviews Neuroscience*, 3:132–142, 2002.

[28] W. Schultz. Getting formal with dopamine and reward. *Neuron*, 36:241–263, 2002.

[29] J. K. Seamans and C. R. Yang. The principal features and mechanisms of dopamine modulation in the prefrontal cortex. *Progress in Neurobiology*, 74:1–57, 2004.

[30] H. S. Seung. Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. *Neuron*, 40(6):1063–1073, 2003.

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

[32] C. M. Thiel, K. J. Friston, and R. J. Dolan. Cholinergic modulation of experience-dependent plasticity in human auditory cortex. *Neuron*, 35:567–574, 2002.

[33] X. Xie and H. S. Seung. Learning in neural networks by reinforcement of irregular spiking. *Physical Review E*, 69:041909, 2004.