

# Is a biological temporal learning rule compatible with learning Synfire chains?

David C. Sterratt

Institute for Adaptive and Neural Computation, Division of Informatics,  
University of Edinburgh, 2 Buccleuch Place, Edinburgh EH8 9LW, Scotland.  
Email: dcs@cogsci.ed.ac.uk

## Abstract

I investigate how a biologically realistic temporal learning rule [11] and the neuronal firing threshold jointly determine the recall speed of a synfire chain trained by sequential activation of its nodes. Numerical analysis of an idealised system of discrete Spike Response Model neurons [6] yields the relationship between threshold and speed of recall, in particular showing that recall is not possible at all speeds and that recall may not be possible at the speed at which the chain was trained. A continuous approximation to the discrete system is analytically more tractable but does not reflect the stability of the system accurately.

## 1 Introduction

The timing of spikes is important in both coding by neurons and learning at synapses. This paper investigates whether one form of temporal coding — synfire chains — and some recent experimental evidence about temporal learning contradict or support each other.

A synfire chain [1] requires a neuronal group in which a sequence of firings of smaller pools (known as *nodes*) of neurons can be stored by weighted connections that diverge from the neurons of one node and converge on the neurons of the next. The set of neurons in a particular node is random. A neuron can belong to more than one node and the neurons of one node need not be located close to each other. When certain conditions are fulfilled activity can propagate down the chain synchronously, i.e. with neurons of the same pool being activated almost simultaneously, and the different nodes being activated in the sequence defined by their connections. Abeles [1] named this mode of prop-

agation *synfire activity* and called the chains of neurons that support it *synfire chains*. For synfire activity to occur the number of neurons in each group must be sufficiently large, and the variances of the connection weights, the delays between adjacent nodes and the thresholds of the neurons must all be sufficiently small. There are a number of hypotheses about the information processing role of synfire chains [3, 10].

The unsupervised learning of synfire chains has been investigated [7]. This paper concentrates on supervised learning, in which the neurons of the nodes of the synfire chain are activated in sequence. Such training might occur in the hippocampus when presented with sequences of input stimuli or in the motor cortex which might be presented with sequences from another part of the brain such as the basal ganglia. Gerstner *et al.* [6] have shown how spatiotemporal patterns (a general case of synfire chains) may be stored in a network of spiking neurons by setting its weights with a temporal learning rule. The resulting connections form a generalised synfire chain in which there may be connections between non-adjacent nodes. Recall of the spatiotemporal patterns will be affected by parameters other than the weights. Each neuron's threshold determines the speed of recall, defined as the number of nodes activated per unit time, and its refractory properties determine whether a node re-excites after being recalled.

The learning evidence from mammalian cortical pyramidal [8] and frog retinotectal [11] synapses suggests that the delay between the spiking of a synapse's pre- and post-synaptic neurons determines its potentiation or depression. When a pre-synaptic spike arrives shortly (within about 20 ms) before a post-synaptic one, the synapse is potentiated and when the converse is true it is depressed.

This threshold-speed relationship has been previously investigated in simulation [2] and by treating neural tissue as a continuum of Spike Response Model (SRM) [9] and Integrate-and-fire (IF) model [4] neurons with connections corresponding to idealised temporal learning rules. In section 2 I present an analysis of a network of discrete SRM neurons and confirm that under appropriate conditions it can be approximated by continuous equations similar to those derived in the continuum case [9]. In section 3 application of both analyses to a temporal learning rule suggested by recent experimental work [11] confirms that the discrete analysis is important at low speeds and that recall may occur faster than training. I discuss the implications of these results for learning synfire chains and suggest a possible extension in section 4.

## 2 Analysis

To derive the speed of propagation for a given threshold I assume an infinitely large network of noiseless neurons in which an infinitely long synfire chain with non-overlapping nodes is to be learned. The network is trained by activating the neurons corresponding to each node in turn at regular intervals  $1/v_t$ , where  $v_t$  is the *training speed* of propagation.

The connection from neuron  $j$  to neuron  $i$  has a synaptic efficacy (weight)  $J_{ij}$ , an axonal spike propagation delay  $\Delta_a$ , assumed to be the same for all connections, and a delay to postsynaptic spikes backpropagating through the dendrites  $\Delta_d$ , which is also connection-invariant. The network is trained by altering the weights according to a temporal learning rule. It has been proposed [6] that near-simultaneous arrival of neurotransmitter at a synapse due to a presynaptic spike and a backpropagated spike from the soma of the postsynaptic neuron causes synaptic potentiation. The amount of potentiation  $\delta J_{ij}$  caused by an arbitrary pair of pre- and postsynaptic spikes depends on the difference between their arrival times at the synapse. If  $t_i$  is the firing time of the postsynaptic neuron and  $t_j$  the firing time of the presynaptic neuron,

$$\delta J_{ij} \propto v((t_i + \Delta_d) - (t_j + \Delta_a)) \quad (1)$$

where the *potentiation function*  $v(t)$  quantifies the amount of potentiation or depression for a delay  $t$ . We can sum all the small changes  $\delta J_{ij}$  over a period of activity in the network to obtain  $J_{ij}$ . The exact mechanism of potentiation or depression is unimportant for this paper; what is

important is that we can measure the potentiation function experimentally. Previous analysis has used idealised Gaussian, rectangular or exponential functions. In this paper I use the experimentally-inspired function [11] depicted in figure 1.

The analysis in this paper uses the Spike Response Model [6] as (a) it is a generalised version of a common model of neurons, the un gated IF model [5]; (b) it can approximate gated IF models [4, 5]; and (c) it is analytically tractable, offering general results which may be solved for arbitrary biological functions and constants.

Each SRM neuron  $i$  has a time-varying membrane potential  $h_i(t)$ , which is the sum of the *synaptic potential*  $h_i^s(t)$  and the *refractory potential*  $h_i^r(t)$ :  $h_i(t) = h_i^s(t) + h_i^r(t)$ . When the membrane potential crosses the threshold  $\theta$  (assumed here to be the same for all neurons), the neuron fires.

The synaptic potential results from the other neurons of the network firing at times  $\{t_i^f\}$ , where  $t_i^f$  denotes the  $f$ th last firing of the  $i$ th neuron. When a spike from the soma of  $j$  arrives at a synapse, with a delay of  $\Delta_a$  after it was triggered, it causes an excitatory postsynaptic potential (EPSP) in  $i$ . We can make the simplifying assumption that only the last firing time  $t_j := t_j^1$  is important, in which case the synaptic potential of  $i$  is

$$h_i^s(t) = \sum_j J_{ij} \epsilon_j(t - t_j - \Delta_a). \quad (2)$$

A typical choice for the EPSP function is the alpha function  $\epsilon(t) = (t/\tau) \exp(1 - t/\tau) \Theta(t)$ , where  $\Theta(t)$  is the Heaviside (step) function and  $\tau$  is the EPSP time constant.

A refractory function is triggered when neuron  $i$  fires and is thereafter added to the refractory potential of  $i$ . It simulates the absolute refractory period with an infinitely negative potential followed by any after-potentials of the neuron. For a given time course of the synaptic potential, the form of the refractory potential determines whether a neuron re-excites after firing.

I assume that the network learns noiselessly according to (1), so that: there is no variance in the the trained connection strength between a pair of nodes; all neurons are at the same resting potential immediately before firing; and the refractory potential ensures neurons fire only once. The refractory potential can therefore be ignored in the following derivations. For the

above reasons, each neuron in a node experiences the same membrane potential, so a node can be treated as a single neuron (with appropriately scaled weights). Consequently, from now on I refer to nodes as though they were neurons. At the end of the training protocol the weight from node  $j$  to node  $i$  will be:

$$J_{ij} = v \left( \frac{i-j}{v_t} - \Delta_a + \Delta_d \right). \quad (3)$$

Although the weights determine the order of recall of nodes, the magnitude of the threshold determines whether and how fast the nodes may be recalled for a given set of weights. If it is too high no units will ever fire; if it is too low the units will be firing at the maximum frequency allowed by their refractory period. For values between these extremes, the threshold has the effect of controlling the speed of propagation since each unit fires when the membrane potential crosses the threshold while rising. The membrane potential will cross a lower threshold sooner, causing activity to propagate faster.

Whilst physically the threshold determines the recall speed, mathematically we can calculate the threshold as an explicit function of speed. In the following analysis I consider one node and assume that all previous nodes have fired regularly at speed  $v$ , either because of synfire recall, or maybe caused by some forcing mechanism. Without loss of generality, I specify the recall firing times to be  $t_i = i/v$  and consider the membrane potential of node 0, which should fire at time 0. To ensure that the node fires at the correct time the threshold is set equal to the membrane potential at the time the node should fire, i.e.,

$$h_0(0; v) = \theta(v), \quad (4)$$

where  $v$  is a parameter of the membrane potential and threshold. By substituting (3) in (2) it should be possible to solve this equation. However, not all these solutions necessarily correspond to stable recall as some of the solutions may occur when the membrane potential passes through the threshold on its falling edge. In this case the node will have already fired and so that particular speed is not stable. The condition

$$\dot{h}_0(0; v) > 0 \quad (5)$$

weeds out the unstable speeds.

We can write down the membrane potential of an arbitrary node  $i$  at time  $t$  by substituting

the recall firing times in (2) to give

$$h_i(t) = \sum_{j=-\infty}^{\infty} J_{ij} \epsilon(t - \Delta_a - j/v), \quad (6)$$

and find its derivative at the zeroth node

$$\dot{h}_0(t) = \sum_{j=-\infty}^{\infty} J_{0j} \dot{\epsilon}(t - \Delta_a - j/v). \quad (7)$$

We can approximate the discrete equations above by continuous ones in the case where the reciprocal of the speed is much less than the EPSP time constant,  $1 \ll v\tau$ , and the reciprocal of the training speed is much less than the characteristic time  $\tau_p$  of the potentiation function,  $1 \ll v_t\tau_p$ . This is because in effect we are numerically integrating the weight and EPSP terms in equations (6) and (7) with step size  $1/(v\tau)$  and  $1/(v_t\tau_p)$ . With  $\tau = 1$  ms the first condition is not fulfilled for  $v < 2 \times 10^3$  s<sup>-1</sup>. Although this means that the continuous analysis is not strictly applicable, it still gives a rough idea of the behaviour we can expect from the system and gives insights into its scaling properties.

Each neuron is now represented by a position on the real number line. We replace the discrete weights  $J_{ij}$  by a continuous weight function defined between positions  $x$  and  $x'$ ,  $J(x, x')$ . In fact this will depend only on the displacement between positions,  $x - x'$ . Applying the learning rule (1) leads to an equivalent of (3) and it is easy to show that the continuous equivalents of (6) and (7) are

$$h(x, t) = \int_{-\infty}^{\infty} J(x - x') \epsilon(t - \Delta_a - x'/v) dx' \quad (8)$$

and

$$\dot{h}(0, 0) = \int_{-\infty}^{\infty} J(-x') \dot{\epsilon}(-\Delta_a - x'/v) dx'. \quad (9)$$

These equations are essentially the same as those found previously [9]. They yield analytic solutions for some EPSP and potentiation functions, in which case we can find an expression for the threshold-speed relationship.

### 3 Application to Realistic Potentiation Function

The experimentally-based potentiation function [11] displayed in figure 1 plots percentage change in the strength of *excitatory* synapses

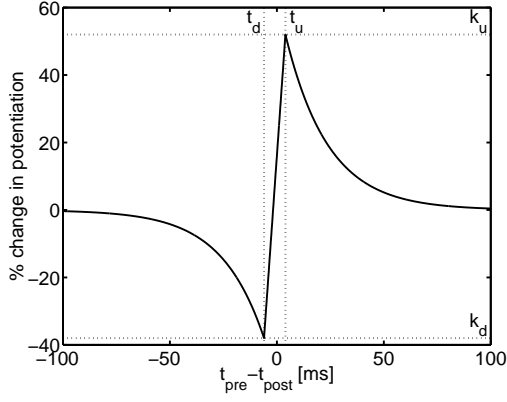


Figure 1: The potentiation function derived from experiment. See section 3 for explanation of constants.

against time delay between pre- and post-synaptic spikes. To treat this function as an absolute synaptic weight and substitute it into (3) I assume that each connection initially comprises balanced inhibitory and excitatory components. Training the excitatory connections with the potentiation function results in effectively excitatory and inhibitory connections according to its potentiating and depressing parts.

The experimental function comprises potentiating and depressing exponential sections with a time constant of  $\tau_p = 20$  ms joined by a linear portion. The constants used in this paper are roughly the same as the curves fitted in [11], although the linear section is not very well defined. The maximum potentiation  $k_u = 52$  occurs at a delay of  $t_u = 4$  ms and the maximum depression  $k_d = -38$  occurs at a delay of  $t_d = -6$  ms.

For clarity, however, I initially consider a simpler version of the experimental function where  $t_u = t_d = 0$  ms, and  $k_u = 1$  and  $k_d = -1$ . There is therefore a discontinuity at  $t = 0$  ms. In this case the analytical calculation reveals that for  $v/v_t > (\Delta_a - \Delta_d)/\Delta_a$ ,

$$\theta(v) = \frac{k_u v \tau e^{(1 - \Delta_a (\frac{v}{v_t} - 1) - \Delta_d)}}{\left(\frac{v}{v_t} \frac{\tau}{\tau_p} + 1\right)^2} \quad (10)$$

and

$$\dot{h}(0) = \frac{k_u v e^{(1 - \Delta_a (\frac{v}{v_t} - 1) - \Delta_d)}}{\left(\frac{v}{v_t} \frac{\tau}{\tau_p} + 1\right)^2} \left(\frac{v}{v_t} \frac{\tau}{\tau_p}\right). \quad (11)$$

This restriction ensures that only the potentiating part of the curve is included in the weights

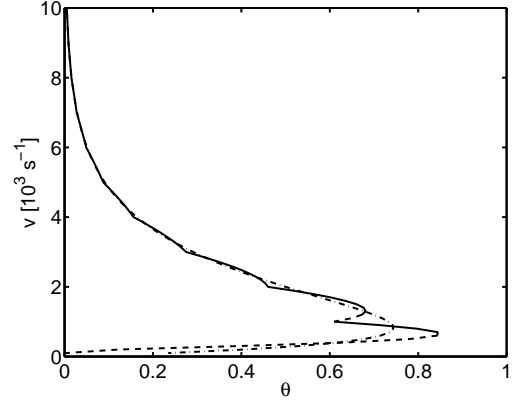


Figure 2: Speed-threshold relationship in a simple case where  $t_u = t_d = 0$  ms,  $k_u = k_d = 1$  and  $v_t = 0.1 \times 10^3 \text{ s}^{-1}$ . Discrete analysis: solid part of curve shows stable solutions and dashed part unstable ones. Continuous analysis: dash-dotted curve shows stable solutions; there are no unstable ones.

that cause the node to fire. For the rest of this paper, the delays  $\Delta_a = \Delta_d = 1$  ms and the training speed  $v_t = 0.1 \times 10^3 \text{ s}^{-1}$  are assumed, so the restriction is  $v/v_t > 0$ .

Figure 2 shows the discrete and continuous threshold-speed relationships determined by substituting the potentiation function in (6), (7), (8) and (9) and solving the resulting equations for a training speed  $v_t = 0.1 \times 10^3 \text{ s}^{-1}$ . The different analyses are in good agreement at higher speeds, though both the speed and stability for a given threshold are markedly different for lower speeds, where the condition  $v\tau \ll 1$  breaks down. The cusps in the discrete curve occur when  $v = j/\Delta_a$ , where  $j$  is an integer, and are due to the effect of a node being able to see the activity of an extra node at these points.

As expected, propagation is only possible for a limited range of thresholds and for higher speeds the speed always increases as the threshold decreases. We might not have expected the speed to increase as the threshold is increased at lower speeds as it does in the continuous analysis. At lower speeds, the synaptic potential reaches the threshold almost at its peak whereas at higher speed solutions for the same threshold the synaptic potential continues to rise well after the node has fired.

A lower bound to the speed in the discrete case is  $1/(\Delta_a + \tau)$ , because if the last node has fired longer than  $\Delta_a + \tau$  ago all it sees

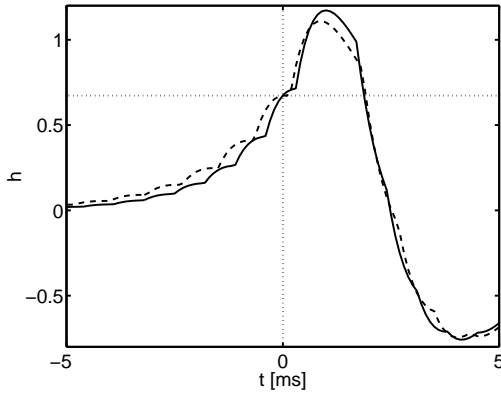


Figure 3: Time course of membrane potential for a neuron in the discrete case which fires at  $t = 0$  showing how the stable propagation can occur at two speeds for the same threshold ( $\theta = 0.6728$ ). Solid curve  $v = 1.2 \times 10^3 \text{ s}^{-1}$ ; dashed curve  $v = 1.4237 \times 10^3 \text{ s}^{-1}$ .

is the summed tails of several EPSPs. In fact, even if all other nodes have been forced to fire at a speed less than this limit, a node allowed to fire freely fires on the rising part of the EPSPs, sooner than it “ought” to. This causes the next node to fire after a shorter time interval, and the speed of recall will gradually converge on the stable value for that threshold. By contrast, in the continuous case the neuron will be seeing EPSPs from neurons that have fired only  $\tau$  ago, and consequently the rising parts of the EPSP functions will balance out the tails that it also sees. We can also see this in equation (11) where it is impossible for the derivative to be less than zero.

One of unstable speeds is the training speed itself. Consequently a chain whose connections were learned at  $v_t$  will be recalled at much higher speed.

There are some values of the threshold (in the approximate region  $[0.6, 0.65]$  for the discrete analysis in figure 2) which allow two propagation speeds. Figure 3 shows two possible time courses of membrane potential for the same threshold for a discrete calculation and in a continuous analysis.

From (10) we can see that it is not just the ratio of  $v$  to  $v_t$  that determines the threshold as one might have initially thought, but that the speed scales the threshold. We can also see that we must consider  $v$  and  $\tau_p$  separately in contrast to previous work [4], where the speed

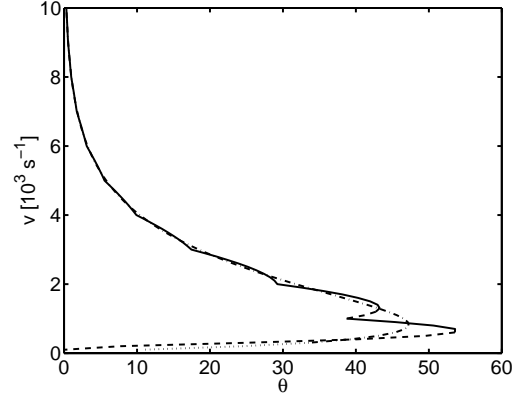


Figure 4: Plot of  $v$  against  $\theta$  for  $\tau = \Delta_a = \Delta_d = 1$  and  $v_t = 0.1 \times 10^3 \text{ s}^{-1}$  where the weights have been learned with the potentiation function shown in figure 1. Discrete analysis: solid part of curve shows stable solutions and dashed part unstable ones. Continuous analysis: dash-dotted curve shows stable solutions and dotted part unstable ones.

scales with the space constant of the weights to a given unit, and therefore with the time constant of the potentiation function. This is because we have used an asymmetric learning function rather than a symmetric one as considered previously.

I have solved the analytic threshold and stability equations for the potentiation function with experimentally-determined constants, though presenting the solutions would not be particularly illuminating. The solutions are shown graphically, along with the discrete speed-threshold relationship in figure 4. It is broadly similar to the simpler case, although there are some values of  $\theta$  for which the continuous case would be unstable.

## 4 Discussion

In this paper I have derived equations for the threshold-speed relationship of a generalised synfire chain comprising discrete units trained by a temporal learning rule, derived a continuous approximation to it and applied both analyses to a realistic potentiation function.

The discrete analysis is important as it corresponds better to physical reality than does the continuous approximation to it. On the basis of the continuous approximation alone, the instability of activity at lower speeds would not be predicted. Inclusion of the training speed

as an explicit parameter does not simply lead to a scaled version of the previous continuous analysis [4, 9] and leads to the surprising conclusion that for low training speeds (less than  $(\Delta_a + \tau)^{-1}$ ) the recall speed of chains will be higher than the speed at which they were trained. This result raises the possibility that temporal sequences may be trained more slowly than they can be recalled. This possibility could be of significance in learning motor sequences, where if nodes represent actions a slow sequence of movements could be learned and then speeded up [cf. 9]. The effective threshold could be modified dynamically by an external signal that modifies neuronal background activity, thereby causing chains to speed up or slow down and possibly allowing two chains to synchronise their activities, as suggested in some theories of synfire processing [3].

The principle differences between the biologically realistic potentiation function [11] considered in this paper and the others (e.g. Gaussian, rectangular) considered previously [4, 9] are (a) the longer time constant (20 ms as opposed to 1 ms) and (b) the depression occurring when presynaptic spikes follow postsynaptic ones. The shape of the speed threshold curve remains virtually the same as with a Gaussian curve, though the precise crossover point of the potentiation function affects it. The time course of the membrane potential is similar, though with inhibition after a node has fired. This will tend to turn off a mode more quickly than it would had it been trained on a Gaussian learning function, and therefore affects the likelihood of a node re-exciting after firing. Due to the asymmetric learning function the speed does not scale with the time constant of the potentiation function, as in previous work [4].

An interesting extension to the work would be to investigate whether the more realistic temporal learning rule allows more successful unsupervised learning of synfire chains than has been achieved previously [7].

To conclude, the answer to the question posed in the title of this paper is that the more realistic learning rule is compatible with synfire chains, but that the speeds of training and recall may be quite different.

**Acknowledgements** I am grateful for the financial support of the Caledonian Research Foundation, discussions with Bruce Graham, David Willshaw and Thomas Wennekers, and Will Lowe and Gert Westermann's proof reading.

## References

- [1] Abeles, M. (1991). *Corticonics: neural circuits of the cerebral cortex*. Cambridge University Press, Cambridge.
- [2] Arnoldi, H. M. R. and Brauer, W. (1996). Synchronization without oscillatory neurons. *Biological Cybernetics*, **74**, 209–223.
- [3] Bienenstock, E. (1995). A model of neo-cortex. *Network — Computation in Neural Systems*, **6**, 179–224.
- [4] Ermentrout, B. (1998). The analysis of synaptically generated travelling waves. *Journal of Computational Neuroscience*, **5**, 191–208.
- [5] Gerstner, W. (1998). Spiking neurons. In W. Maass and C. M. Bishop, editors, *Pulsed Neural Networks*, pages 3–54. MIT Press, Reading, Massachusetts.
- [6] Gerstner, W., Ritz, R., and van Hemmen, J. L. (1993). Why spikes? Hebbian learning of time-resolved excitation patterns. *Biological Cybernetics*, **69**, 503–515.
- [7] Hertz, J. and Prügel-Bennett, A. (1996). Learning short synfire chains by self-organization. *Network — Computation in Neural Systems*, **7**, 357–363.
- [8] Markram, H., Lübke, J., Frotscher, M., and Sakmann, B. (1997). Regulation of synaptic efficiency by coincidence of postsynaptic APs and EPSPs. *Science*, **275**, 213–215.
- [9] Wennekers, T. and Palm, G. (1996). Controlling the speed of synfire chains. In C. v. d. Malsburg, W. v. Seelen, J. C. Vorbrüggen, and B. Sendhoff, editors, *Proceedings of the ICANN 1996*, volume 1112 of *Lecture Notes in Computer Science*, Bochum, Germany. Springer-Verlag.
- [10] Wennekers, T. and Palm, G. (1999). Cell assemblies, associative memory and temporal structure in brain signals. To appear in R. Miller, editor, *Time and the Brain*. Harwood Academic Publishers.
- [11] Zhang, L. I., Tao, H. W., Holt, C. E., Harris, W. A., and Poo, M. M. (1998). A critical window for cooperation and competition among developing retinotectal synapses. *Nature*, **395**, 37–44.