Weight vector normalization in an analog VLSI artificial neuron using a backpropagating action potential

Ph. Häfliger and M. Mahowald
Institute of Neuroinformatics UNIZ/ETHZ
Gloriastrasse 32
CH-8006 Zürich
Tel: ++41 1 634 27 99
e-mail: hafliger@ini.phys.ethz.ch

Abstract

At NIPS 96 we introduced the Modified Riccati Rule (MRR), a Hebbian like learning algorithm that uses temporal correlations between pre- and postsynaptic spike to determine changes in synaptic connectivity [1]. Recent physiological experiments in young rat neocortex indicate that the relative timing of single excitatory postsynaptic potentials (EPSP) and postsynaptic action potential (AP) indeed influence the synaptic efficacy [2,3]. Since the backpropagating AP is shared information among all synapses, the synaptic changes could evolve in a coordinated way. One coordinated behaviour, which in the MRR is achieved this way, is weight vector normalization. We introduce a CMOS implementation of the MRR and demonstrate its normalizing property.

1 The MRR in analog VLSI

We have fabricated an analog VLSI (aVLSI) chip, using a 2μm CMOS process, to simulate the MRR in analog hardware. Figure 1 shows a repeatedly used component of the circuit. The so called trigger circuit produces digital pulses of adjustable width given a rising flank as input. Figure 2 describes the soma and figure 3 one learning synapse (of which our silicon neuron has three). Also part of the neuron but not shown is an excitatory and an inhibitory non learning synapse with adjustable weight.

2 Normalization

By normalization, we mean that the length of the weight vector is kept constant. This is one mechanism that prevents the synaptic weights growing infinitly. Such growth is a constant danger in more basic Hebbian learning rules, especially in recurrent networks. On the other hand normalization of the weight vector also prevents the case that all synapses of one neuron become ineffective. In many learning rules this normalization is performed as an

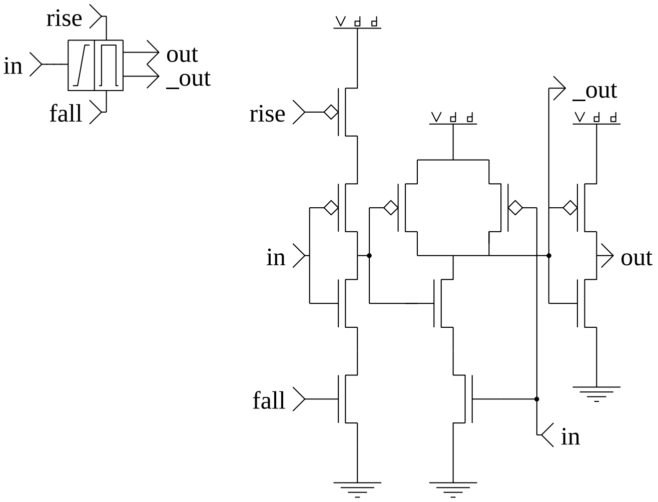

Figure 1: This 'trigger circuit' produces a digital pulse of adjustable size as response to a rising edge. Later in this paper we will use its icon (in the upper left). It is composed of a NAND gate and two inverters. The NAND and one inverter form an AND gate. The input signal is given directly to one input of the AND and delayed and inverted to the other. $fall$ determines that delay and therefore the width of the trigger circuit's output pulse. $rise$ being slightly smaller than Vdd ensures that the falling edge of the signal does not produce a glitch.

extra step after the weights have been adjusted. In other cases, like the Riccati or the related Oja's rule, the normalization is included in the learning rule and the extra step is not required. The MRR is inspired by the Riccati rule and has inherited an approximate weight vector normalization from it and so has our aVLSI implementation.

This is basically achieved by making the weight decrements dependent on the postsynaptic AP and the actual weight, and the increments proportional to the number of presynaptic spikes. If one increases the frequency of the input to all synapses uniformly, then the output frequency will increase proportionally. If the weight increments and decrements were in equilibrium, they will remain so. If we increase the average input frequency at just one synapse, this will cause the weight at this particular synapse to grow. This enhances the average

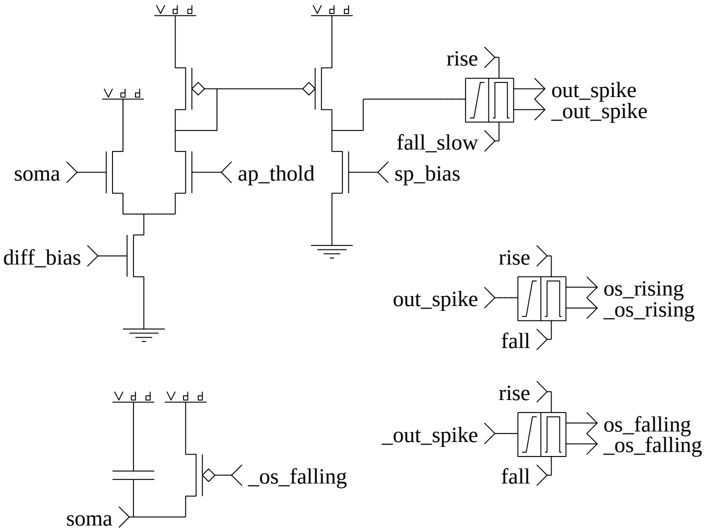

Figure 2: The circuit for the soma: It contains three trigger circuits. The lower two provide timing information to the synapses about the rising and falling of the action potential (*os_rising*, *os_falling*). The third embedded in the top circuit, tuned by *fall_slow* to produce wider pulses, provides the action potential. It is activated as soon as the soma voltage (inverted sense compared to nature) falls below a threshold (*ap_thold*). A differential pair, a current mirror and a simple amplifier change that information into a rising voltage. The rising edge then triggers the trigger circuit. The soma is set back to Vdd by *_os_falling*.

output frequency, which will cause all other synapses that still receive the same amount of input spikes to weaken. This competition among the synapses causes the weight vectors length to remain in a limited range.

On our chip the weight increments are not linearly dependent on the number of input spikes. They would be so if the correlation signal *corr* was simply a non-decaying spike counter and its influence on the weight increments linear rather than exponential. The correlation signals's decay is deliberate, since it makes weight increments dependent on temporal coincidence. It causes the normalization to be less precise, especially when several synapses get inputs that are very similar in frequency (see figures 4 and 5). The exponential relation between *corr* and the weight increments is a more serious problem. By keeping *delta* small, we keep to a small region of that exponential curve so as

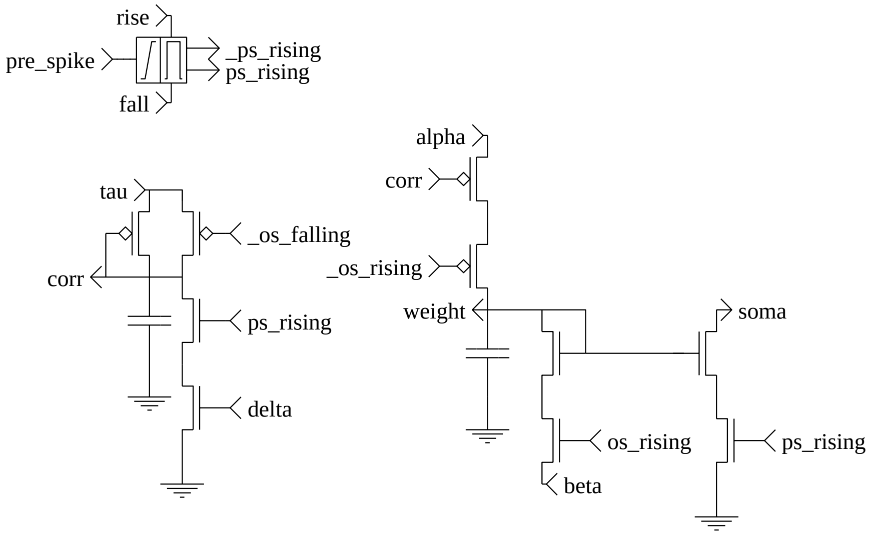

Figure 3: The circuit for one synapse: It first captures an incoming spike with a trigger circuit and translates it to a pulse of well defined width. This pulse opens a gate which allows a current to flow from the soma to ground, the size of the current given by the synapse's *weight* (The sense of the membrane voltage is inverted compared to nature.). It also decreases (increases in the intuitive way) the correlation signal by an amount determined by the *delta*. *alpha* and *beta* set the size of the weight change at an AP-event. A further control on the weight increments and decrements can be achieved by adjusting the length of the *os_rising* pulse with $fall$ (see the circuit for the soma in figure 2). Finally the correlation signal's decay (towards *tau*) and its upper limit is controlled by the parameter *tau*.

to linearize it as much as possible. Still this effect tends to lengthen the weight vector when the input load is concentrated on just a few synapses (figures 4 and 5). This again enlarges the 'valleys' (expected from the theory of the MRR) shown in the figures, where the two input frequencies are similar.

Figures 4 and 5 illustrate test runs on the chip where two synapses received Poisson distributed inputs. Due to fabrication asymmetries, synapse 1 tends to be stronger than synapse 2. That is why the value at the right side of the graph is bigger than at the left side.

3 Conclusion

A backpropagating action potential can be used to obtain coordinated changes in synaptic efficacy among all synapses of one neuron. For example weight

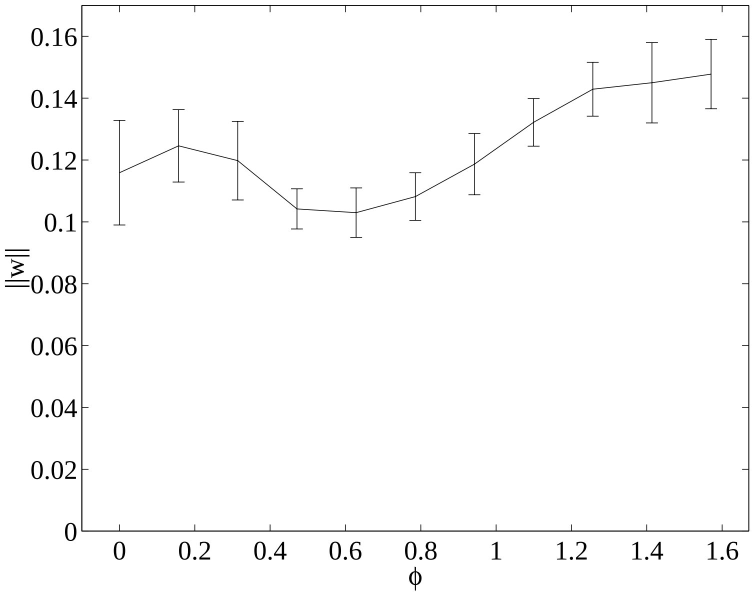

Figure 4: The graph shows the weight vector length on the y-axis for an experiment with our chip using two synapses, both of which get Poisson distributed spike-trains with constant average frequency as input. The squared sum of the average input frequencies was kept constant. The x-axis is an angle ϕ and the input frequencies ν_1 and ν_2 given to the synapses are defined as $\nu_1 = 100sin(\phi) \quad \nu_2 = 100cos(\phi)$. The weights were given time to settle and were then taken as the mean of 50 oscilloscope traces of 200ms each. The bars show the standard deviations in these sets of 50 samples.

vector normalization can be performed this way as demonstrated in an aVLSI artificial learning neuron.

Acknowledgments

We thank the following organizations for their support: SPP Neuroinformatik des Schweizerischen Nationalfonds, Centre Swiss d'Electronique et de Microtechnique, U.S. Office of Naval Research and the Gatsby Charitable Foundation.

1. Ph. Häfliger, M. Mahowald, and L. Watts. A spike based learning neuron in analog VLSI. *Advances in neural information processing systems*, 9,

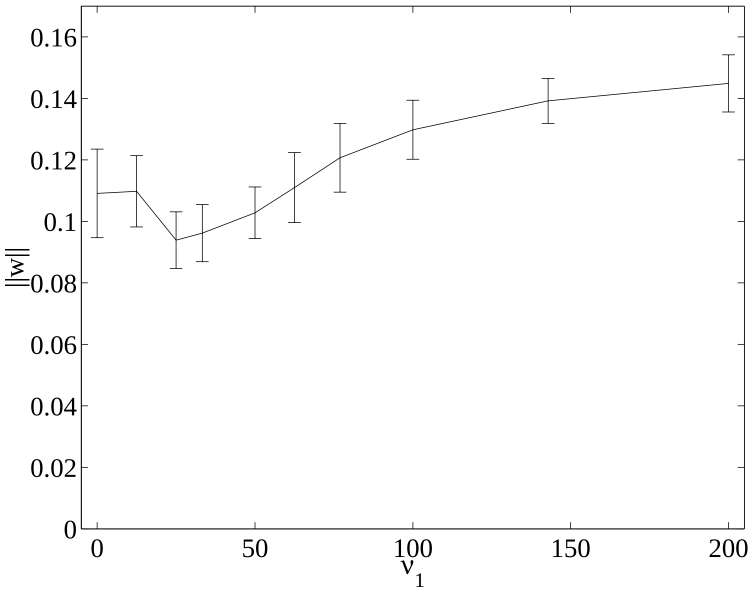

Figure 5: Again (see figure 4) the weight vector length for an experiment with two synapses stimulated with Poisson distributed spike signals is shown. This time the average frequency of synapse 2 is kept constant at 50 Hz. The x-axis shows the average frequency given to the other synapse. The weights were taken as the mean of 200 oscilloscope traces of 200ms each. The bars show the standard deviation in these sets of 200 samples.

1996.

2. M. V. Tsodyks and H. Markram. Redistribution of synaptic efficacy between neocortical pyramidal neurons. *Nature*, 382:807–810, August 1996.
3. H. Markram, J. Lübke, M. Frotscher, and B. Sakmann. Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps. *Science*, 275:213–215, 1997.