Temporally learning floating-gate VLSI synapses

Shih-Chii Liu and Rico Möckel
Institute of Neuroinformatics
University of Zürich and ETH Zürich
Winterthurerstrasse 190, CH-8057 Zürich, Switzerland
Email: shih,moeckel@ini.phys.ethz.ch

***Abstract*— We present a floating-gate synaptic circuit that updates its weight according to the Spike-Timing-Dependent Plasticity (STDP) rule. The weight (or floating-gate voltage) is updated only if the time difference between the pre- and post-synaptic spikes falls within a learning window. The update is implemented through tunneling and injection mechanisms which can be tuned for very long time constants up to seconds. The novelty of this circuit is that the tunneling and injection mechanisms are turned on only when the correlation of the pre and postsynaptic activity is significant. The additional benefit of this non-volatile technology is that synaptic weights can be stored locally on chip. We present experimental results that show the learning and normalization effects from the fabricated circuits.**

I. Introduction

The inclusion of local learning mechanisms into spiking networks allows us to construct systems that can adapt their network connectivity for different tasks. Floating-gate technology provides a natural medium for implementing learning mechanisms and autonomous local adaptation in VLSI networks similar to the adaptive mechanisms in natural systems. This premise has led to the development of single synapse floating-gate transistors that implement long-term nonvolatile analog memory, bidirectional weight updates, and continuous learning from its inputs. These synapse transistors have been used in the implementation of circuits for both supervised and unsupervised learning on continuous-time analog signals [1]-[4], and circuits that compensate for on-chip element differences [1], [2], [5].

In spiking networks, the learning rule of choice is the spike-timing-dependent plasticity (STDP) rule. Networks employing this rule can exhibit temporal pattern recognition, temporal sequence learning, and coincidence detection [6]. Recent implementations of STDP learning in VLSI spiking networks demonstrate the use of these networks in classification and computational tasks [7]-[11]. However, the slowest weight update rate and time constant in most of these implementations is limited to the transistor leakage currents in the chosen fabrication process.

To obtain longer update time constants for the synaptic weights, we describe another implementation of the STDP learning rule which uses floating-gate technology to implement the dynamics of the weight update. The synaptic weight is set by the current flowing through a pFET transistor whose gate is a floating node. The charge on the floating node is removed or added using Fowler-Nordheim electron tunneling and hot-electron injection [1], [12]. The time constant of the weight update can be made long because the tunneling and injection currents can be over 6 orders of magnitude smaller than leakage currents. Pulse-based floating-gate learning circuits have been described in the past [11], [13], however, in this present implementation, the mechanisms (tunneling and injection) that change the floating-gate voltage are not activated with each pre or postsynaptic spike. In addition, the tunneling currents are activated locally for each synapse. The tunneling is also not continuously activated as in most floating-gate learning circuits. Our learning network implements an approximation of the learning rules described in [6] and exhibits the benefits associated with the floating-gate technology, that is, local adaptation, non-volatile storage of the synaptic weights, and the availability of long time constants in the network.

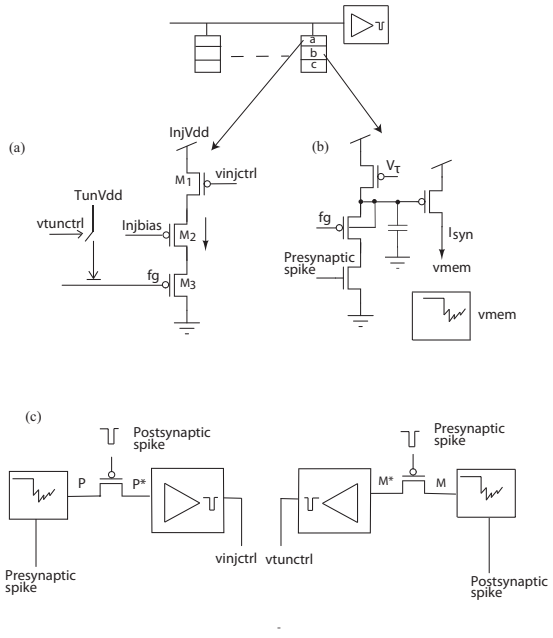

Fig. 1. Circuits to update the floating-gate voltage, synaptic circuit and soma model. (a) The floating-gate node, fg is updated according to control signals generated as shown in (c). (b) A current-mode integrator which is used as the synaptic circuit [9]. The bias V_τ, controls the time constant of the synapse. (c) Circuits for generating the control pulses ($vtunctrl$ and $vinjcntrl$) which turn on the tunneling and injection currents respectively in (a). The circuit that generates P and M uses the same synaptic circuit in (b) except that the fg node is replaced by a fixed bias. The block that generates the control pulses is a soma-like circuit similar to that in Fig. 2 where $vmem$ is now replaced by $P*$ and $M*$.

978-1-4244-1684-4/08/$25.00 ©2008 IEEE

II. Spike-based learning rule

The spike-timing dependent plasticity rule was first described experimentally by [14], [15]. According to this rule, the weight of a synapse is updated according to the relative timing of the pre- and postsynaptic spikes. The amount of the weight change is determined by a temporal learning window following:

$$\Delta W = \begin{cases} f_+(W)\, K^+(\Delta t) & \Delta t > 0 \\ -f_-(W)\, K^-(\Delta t) & \Delta t \leq 0 \end{cases} \quad (1)$$

where $f(W)$ is an updating function and $\Delta t = t_{post} - t_{pre}$. If the presynaptic input spike arrives before the postsynaptic spike, that is, $\Delta t > 0$, the synaptic weight is increased where $K^+(\Delta t) = e^{-\Delta t/\tau_+}$ and τ_+ is the time constant for the potentiation window. Conversely, if the input spike arrives after the postsynaptic spike, the synaptic weight is decreased (or depressed) according to $K^-(\Delta t) = e^{-\Delta t/\tau_-}$ where τ_- is the time constant for the depression window. If $f(W)$ is constant, the weight update is additive and if $f(W)$ is proportional to W, then the weight update is multiplicative.

The update in Equation 1 is computationally expensive if it was applied for every possible pre- and postsynaptic spike pairing in a network simulation. The equation can be simplified as shown in Song et. al [6], where additional variables, P and M are introduced to represent the integrated pre- and postsynaptic activities respectively. The variable M is updated by a fixed amount each time the postsynaptic neuron produces a spike. Conversely, the variable P is updated by a fixed amount each time a synapse receives an input presynaptic spike. These variables decay in time following:

$$\tau_- \frac{dM}{dt} = -M(t) \text{ and } \tau_+ \frac{dP}{dt} = -P(t) \quad (2)$$

To modify the weight, $P(t)$ is sampled at the time of a postsynaptic spike, and $M(t)$ is sampled at the time of a presynaptic spike. This sampled value is then added or subtracted from the synaptic weight respectively.

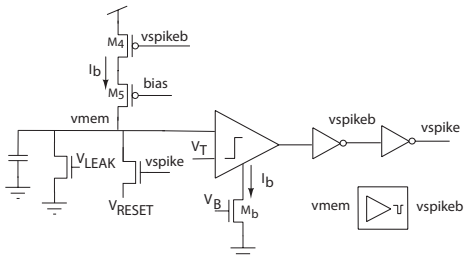

Fig. 2. The soma circuit implements an integrate-and-fire model.

III. Circuits

We describe test results from fabricated circuits consisting of a group of 24 STDP floating-gate synapses connected to a spiking neuron (Fig. 1). The circuits were simulated using a floating-gate SPICE model from [16] before fabrication. This chip was fabricated in a 0.35 μm 4-metal 2-poly CMOS technology. The neuron can also be stimulated by 4 non-plastic excitatory and 4 inhibitory synapses. The floating-gate synapse circuit (approximately 55 μm by 120 μm in area) consists of the circuits for updating the floating-gate voltage fg in Fig. 1(a), the current-mode integrator synaptic circuit in Fig. 1(b), and circuits to generate the control pulses ($vtunctrl$ and $vinjctrl$) that turn on the tunneling and injection currents respectively in Fig. 1(c). The soma is implemented by the integrate-and-fire circuit in Fig. 2.

The dynamics of P and M in Figs. 1(b and c) follows that of Equation 2. Unlike the formulation in Song et al. [6], the sampled P and M values are used to charge up the capacitor of individual soma-like circuits similar to the one in Fig. 2. To determine when the $vinjctrl$ pulse should be generated, the integrated voltage on the soma-like circuit, $P^*(t)$, is compared against a threshold. When the integrated voltage exceeds this threshold, the circuit produces a spike which activates the $vinjctrl$ pulse thus turning on the injection. The injection is carried out by the source-follower pFET in Fig. 1(a) [4], [5]. A similar process occurs on $M(t)$ during a postsynaptic spike. When $M^*(t)$ reaches a threshold, its soma-like circuit generates a $vtunctrl$ pulse thus activating the tunneling through a charge pump circuit. Example $P^*(t)$ and $M^*(t)$ curves are shown in Fig. 3.

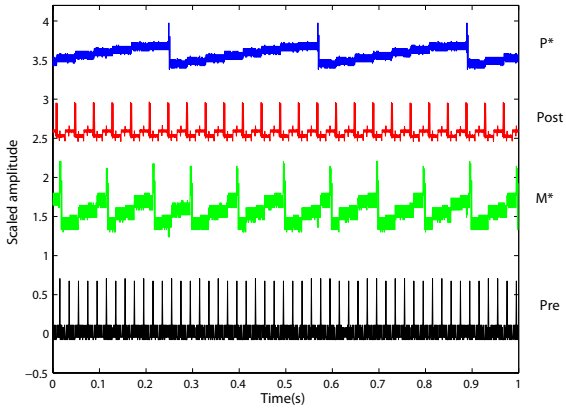

Fig. 3. Timing diagram of M^* and P^* at a learning synapse receiving a regular 50 Hz input spike train (pre) and a postsynaptic rate of 25 Hz (post). $P(t)$ is sampled by every postsynaptic spike onto $P^*(t)$ and $M(t)$ is sampled by every presynaptic spike onto $M^*(t)$. Curves are offset from one another for ease of visibility.

A. Characterization

We measure the effectiveness of the tunneling and injection by quantifying the frequencies of the $vinjctrl$ and $vtunctrl$ pulses in an experiment where a floating-gate learning synapse is stimulated by a regular input spike train and a regular postsynaptic spike output is generated through the stimulation of the neuron by a fixed synapse. We used this measure in quantifying the temporal learning window (see Fig. 4). The time constants, τ_+ and τ_-, in Equation 2 determined the extent of the temporal window and were adjusted so that there is a

net negative area under the learning window, which is needed for stability of the learning system [6].

This update frequency has to be combined with the actual tunneling and injection currents during the pulses to determine the actual weight change. The form of these currents will be analyzed in the next section. In general, the net weight change cannot be predicted from the window because both tunneling and injection pulses occur during the learning process as shown in Fig. 3. The $vinjctrl$ and $vtunctrl$ control pulses are generated during a spike in the P^* and M^* traces respectively.

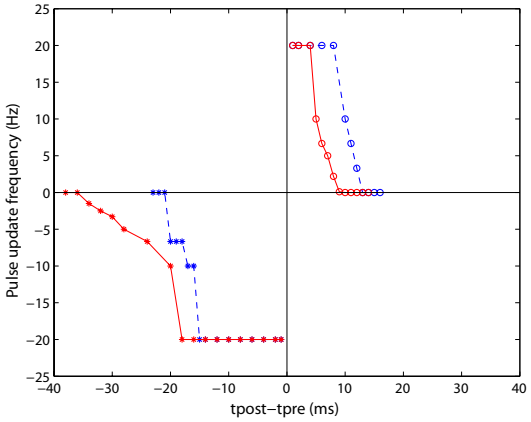

Fig. 4. Frequency of tunneling and injection pulses for a presynaptic input rate of 20 Hz and a postsynaptic rate of 20 Hz. The curves are obtained with various values of τ_+ and τ_-.

B. Weight Update

The weight update in our floating-gate synapse implementation is a multiplicative update because of the dependence of the tunneling and injection currents on the floating-gate voltage and hence the synaptic weight. We show this by using the analysis in [2], [3], [12]. By defining the weight of the floating-gate transistor W as the drain current I_s flowing through this transistor normalized by the drain current I_{so} at a particular operating point, we get $W = I_s/I_{so}$ and the time derivative of W following

$$\frac{dV_{fg}}{dt} = \frac{U_T}{\kappa_p W}\frac{dW}{dt}. \tag{3}$$

The dynamics of the weight decrease for the pFET synapse via the tunneling current is

$$\frac{U_T C_T}{\kappa_p I_{tun0}}\frac{dW}{dt} = -W^{1+(U_T/(\kappa_p V_x))} \tag{4}$$

where U_T is the thermal voltage, C_T is the total capacitance at the floating-gate node, κ is the efficiency of the gate in controlling the channel current, I_{tun0} is the quiescent tunneling current, and V_x is a parameter related to the quiescent tunneling and floating-gate voltage.

The dynamics of the weight increase via the injection current is

$$\frac{U_T C_T}{\kappa_p I_{inj0}}\frac{dW}{dt} = W^{(1+\alpha)} \tag{5}$$

where α is $1-(U_T/V_{inj})$ and V_{inj} is a constant which depends on the injection efficiency of the floating-gate transistor.

C. Normalization and Correlation-Based Learning

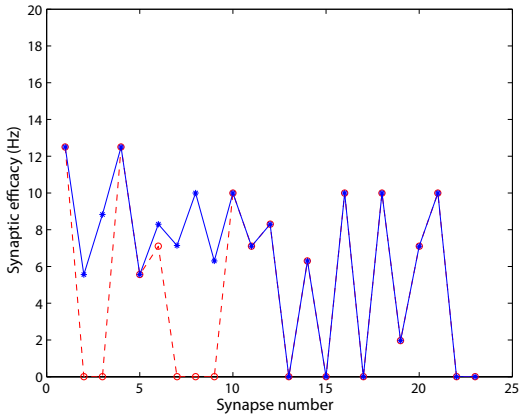

Fig. 5. Effect of input correlations on synaptic weight distribution. Dashed curve: Distribution of synaptic efficacies of 22 synapses in response to uncorrelated 20 Hz input Poisson spike trains. Solid curve: The change in efficacies of 5 silent synapses (2,3,7,8,9) when these synapses along with 3 other finite-weight synapses received 80% correlated 20 Hz input Poisson spike trains while the remaining synapses were stimulated by the uncorrelated Poisson spike trains. Solid and dashed curves are coincident for synapses 11 to 23. The synaptic efficacy was computed by measuring the postsynaptic rate of the neuron when a synapse was solely stimulated by a regular spike train of 50 Hz. TunVdd=6.5V, InjVdd=5.4V.

One feature of a network which incorporates STDP is that synapses with strong temporal correlations are strengthened as expected from a Hebbian-like learning rule. We show this feature in an experiment where we first stimulated a set of 22 floating-gate learning synapses with uncorrelated 20 Hz Poisson spike trains. After approximately 20 minutes, the synaptic distribution settled to the dashed curve in Fig. 5. We then stimulated a subset of 8 synapses (5 are silent) with correlated Poisson input spikes and the remaining synapses with the original uncorrelated Poisson spikes. The weight of the silent synapses increased in response to the new inputs (solid curve in Fig. 5).

Another feature of STDP is that it naturally provides a form of competitive Hebbian learning because of the dependence of the synaptic modification on the spike timing [6]. This is unlike many network models of Hebbian learning which usually have to include constraints to ensure that strong synapses do not grow arbitrarily. The competition with STDP leads to an intrinsic stabilization of weight distribution without requiring a global signal that reflects the state of the synapses. Hence if the output rates increase, the synaptic weights will adjust to keep the output rates stable [6], [17].

This normalization effect has also been described in [11] and we observed the same effect in our floating-gate STDP synapses (Fig. 6). In this experiment, the synapses were driven by a fixed Poisson input rate while we varied the postsynaptic rate through a non-learning synapse. The efficacy of the synaptic weights decreased when the input rate to the non-learning synapse was increased from 20 Hz to 300 Hz.

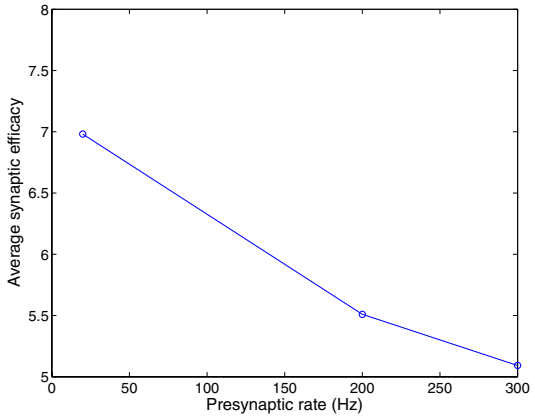

Fig. 6. Distribution of the average synaptic efficacy of synapses which were driven by 10 Hz Poisson spike trains. The synaptic efficacy was determined by measuring the output spike rate when each synapse was stimulated by a 50 Hz regular input spike train. To increase the postsynaptic rate of the neuron, we stimulated a non-learning excitatory synapse with a presynaptic rate of 20 Hz, 200 Hz, and 300 Hz. The average synaptic efficacy decreased with increasing postsynaptic rate. TunVdd=6.5V, InjVdd=5.4V.

IV. Discussion

We present floating-gate synaptic circuits that implement the spike-timing dependent plasticity rule in a learning spiking network. This work shares some circuit similarities with the work of [11] showing the use of floating-gate technology for implementing an STDP-like learning rule. Our circuits differ in that the tunneling and injection currents are not activated for each presynaptic pulse or postsynaptic pulse but but only when the integrated sampled presynaptic or postsynaptic activity exceeds a threshold. This difference means that we do not unnecessarily turn on the injection and tunneling mechanisms which can degrade the gate oxide over time especially if the presynaptic and postsynaptic rates are high. In addition, the tunneling is turned on locally and is not global or continuously activated for all synapses.

This circuit includes the capability for non-volatile local weight storage and slow weight update dynamics which are not present in many VLSI implementations of learning spiking networks. In addition, the multiplicative weight update in this work replaces the additive weight update typically encountered in the non floating-gate spike-based learning networks. The multiplicative rule can provide stabilizing effects on the weight increase or decrease at a synapse. This circuit has intrinsic normalization properties which is an example of homeostasis [18] and the floating-gate technology could potentially implement other homeostatic mechanisms which usually require long time constants [19].

V. Acknowledgments

We acknowledge Seth Bridges for helping with the tunneling and injection circuits in this work; Paul Hasler, and Chris Twigg for multiple discussions on floating-gate topics. We also acknowledge members of the Institute of Neuroinformatics involved in the development of the PCI-AER board, of its drivers, and software library components. This work was partially supported by the IST grant IST-2001-34124.

References

[1] C. Diorio, P. Hasler, B. A. Minch, and C. Mead, "A single-transistor silicon synapse," *IEEE Transactions on Electron Devices*, vol. 43, no. 11, pp. 1972–1980, 1996.

[2] P. Hasler, "Continuous-time feedback in floating-gate MOS circuits," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 56–64, 2001.

[3] C. Diorio, D. Hsu, and M. Figueroa, "Adaptive CMOS: From biological inspiration to systems-on-a-chip," *Proceedings of IEEE*, vol. 90, no. 3, pp. 345–357, 2002.

[4] J. Dugger and P. Hasler, "Supervised learning in a two-input analog floating-gate node," in *Proceedings of the 2004 IEEE International Symposium on Circuits and Systems*, vol. V, May 2004, pp. 756–759, ISCAS 2004, 23 May–26 May.

[5] M. Figueroa, S. Bridges, and C. Diorio, "On-chip compensation of device-mismatch effects in analog VLSI neural networks," in *Advances in Neural Information Processing Systems*, L. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: MIT Press, 2005, vol. 17, pp. 441–448.

[6] S. Song, K. Miller, and L. Abbott, "Competitive Hebbian learning through spike-timing-dependent synaptic plasticity," *Nature Neuroscience*, vol. 3, no. 9, pp. 919–926, 2000.

[7] G. Indiveri, "Neuromorphic bistable VLSI synapses with spike-timing-dependent plasticity," in *Advances in Neural Information Processing Systems*, S. Becker, S. Thrun, and K. Obermayer, Eds. Cambridge, MA: MIT Press, 2003, vol. 15, pp. 1091–1098.

[8] A. Bofill-i Petit and A. F. Murray, "Synchrony detection by analogue VLSI neurons with bimodal STDP synapses," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds. Cambridge, MA: MIT Press, 2004, vol. 16.

[9] J. Arthur and K. Boahen, "Learning in silicon: timing is everything," in *Advances in Neural Information Processing Systems*, Y. Weiss, B. Schölkopf, and J. Platt, Eds. Cambridge, MA: MIT Press, 2006, vol. 18, pp. 75–82.

[10] S. Mitra, S. Fusi, and G. Indiveri, "A VLSI spike-driven dynamic synapse which learns," in *Proceedings of the 2006 IEEE International Symposium on Circuits and Systems*, May 2006, pp. 2777–2780, ISCAS 2006: Island of Kos, Greece, 25 May–28 May.

[11] P. Häfliger and M. Mahowald, "Spike based normalizing Hebbian learning in an analog VLSI artificial neuron," *Analog Integrated Circuits and Signal Processing*, vol. 18: Special issue on Learning in Silicon, no. 2/3, pp. 133–140, February 1999.

[12] P. Hasler, "Foundations of learning in analog VLSI," Ph.D. dissertation, California Institute of Technology, Pasadena, CA, 1997.

[13] A. P. Shon, D. Hsu, and C. Diorio, "Learning spike-based correlations and conditional probabilities," in *Advances in Neural Information Processing Systems*, T. Dietterich, S. Becker, and Z. Ghahramani, Eds. Cambridge, MA: MIT Press, 2002, vol. 14, pp. 1123–1130.

[14] 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*, vol. 18, no. 24, pp. 10 464–10 472, 1998.

[15] H. Markram, J. Lübke, M. Frotscher, and B. Sakmann, "Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs," *Science*, vol. 275, pp. 213–215, 1997.

[16] K. Rahimi, C. Diorio, C. Hernandez, and M. Brockhausen, "A simulation model for floating-gate MOS synapse transistors," in *Proceedings of the 2002 IEEE International Symposium on Circuits and Systems*, vol. I, May 2002, pp. 532–535, ISCAS 2002: Pheonix, USA, 26 May–29 May.

[17] R. Kempter, W. Gerstner, and J. van Hemmen, "Hebbian learning and spiking neurons," *Physical Review E*, vol. 59, no. 4, pp. 4498–4514, 1999.

[18] G. Turrigiano and S. Nelson, "Hebb and homeostasis in neuronal plasticity," *Current Opinion on Neurobiology*, vol. 10, no. 3, pp. 358–364, 2000.

[19] S.-C. Liu and B. Minch, "Silicon synaptic adaptation mechanisms for homeostasis and contrast gain control," *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1497–1503, 2002.