

Possible Nanoelectronic Implementation of Neuromorphic Networks

Özgür Türel, Ibrahim Muckra, and Konstantin Likharev

Department of Physics and Astronomy, Stony Brook University

Stony Brook, NY 11794-3800, U.S.A.

Ozgur.Turel@sunysb.edu and klicharev@notes.cc.sunysb.edu

Abstract—Neuromorphic networks of high connectivity may be implemented using CMOS circuits as cell bodies, nanowires as axons and dendrites, and self-assembled single-molecule latching switches as synapses. The integration scale of such “CrossNet” circuits of acceptable size ($\sim 30 \times 30 \text{ cm}^2$) may be comparable with that of the mammal’s cerebral cortex (up to 10^{10} neurons), despite the quasi-2D structure of the artificial networks. At the same time, the speed of information processing and network adaptation may be about 6 orders of magnitude higher than that of the brain, at high but manageable power consumption $\sim 100 \text{ W/cm}^2$. We present an overview of the hardware implementation prospects, and possible strategies of CrossNet training. Two suggested Hopfield-mode training methods have been verified on numerical models of the networks. The results are consistent with our estimates of the maximum network capacity with an account for finite interconnect locality.

I. INTRODUCTION

The recent spectacular demonstration of single-molecule, single-electron transistors by several groups [1-3] gives every hope for the implementation, in a not very distant future, of hybrid “CMOL” circuits [4] of unprecedented density. Such a circuit (Fig. 1) would combine a level of advanced (e.g., 45-nm-node [5]) CMOS devices fabricated by the usual lithographic patterning, a few layers of parallel nanowire arrays formed, e.g., by nanoimprinting, and finally molecular devices that would self-assemble on the wires from solution. The CMOL concept allows to combine advantages of its nanoscale components (e.g., reliability of CMOS circuits and

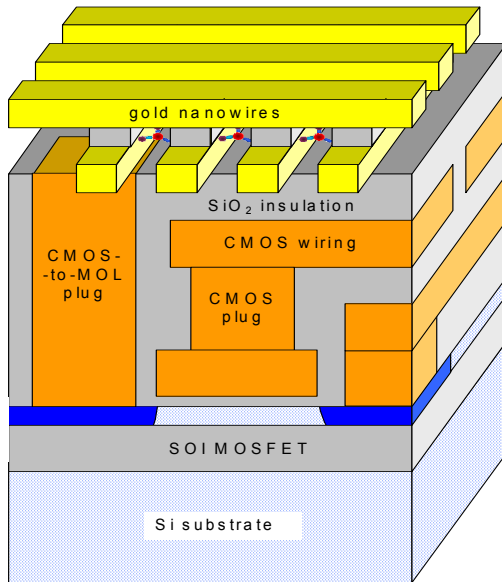


Fig. 1. The general concept of a hybrid CMOS/nanowire/molecular (CMOL) circuit.

miniscule footprint of molecular devices) and patterning techniques (virtually complete flexibility of the usual lithography and potentially very low cost of nanoimprinting and chemically-directed self-assembly - see, e.g., Ref. 6). Thus the original dream of artificial neuromorphic networks with areal density comparable with that of the brain ($\sim 10^7$ neural cells per cm^2), operating at much higher speed, may be revisited, now on a firm technological basis.

However, this development calls for novel network architectures that would take into account hardware-imposed limitations, first of all, the quasi-2D structure of CMOL circuits. The first challenge here is to sustain the cell connectivity comparable to that ($\sim 10^4$) of the cerebral cortex. Earlier, we proposed [7-9] a family of Distributed Crossbar Networks (CrossNets) that can provide such high connectivity, retaining the ultimate density of molecular devices (single-electron latching switches) working as synapses. In this architecture, the synapses are located at any crossing of nanowires (playing the roles of axons and dendrites), while much more sparse neural cell bodies are implemented using CMOS circuits within the fire-rate approach. The main problem of the original CrossNet concept was the two-terminal character of the latching switches that did not allow, in particular, the implementation of the Hebb training rule. The goal of this work is to show that virtually the same devices, but with three terminals, may be used for the effective implementation of the Hebb rule, and as a result, for training of CrossNets as Hopfield networks. The prospects for globally-supervised continuous-mode training of Hebbian CrossNets as image classifiers are also discussed.

II. HEBBIAN CROSSNETS

Our “Hebbian” synapse is composed of four three-terminal devices each consisting of a single-electron transistor (SET) coupled electrostatically to a single electron trap (Fig. 2a). The transistor connects two nanowires (modeling an axon and a dendrite, respectively). The effective voltage V applied to the trap is contributed equally by two axons, one of them providing input signal for the target cell j' , and another one carrying output signal of this cell:

$$V = (V_j + V_{j'})/2 = (V_0/2) \times (\pm x_j \pm x_{j'}), \quad (\text{EQ 1})$$

where x_j are normalized output (axonic) signals of the cells ($-1 < x_j < 1$), and signs depend on the axon polarity (see below).

Just as in our earlier device [7, 8], if the voltage is low, the trap in equilibrium has no extra electrons, and its total electric charge $Q = -ne$ is zero. As a result, the transistor remains in the Coulomb blockade state, and input and output wires are essentially disconnected. If V is increased beyond a certain threshold V_{inj} (which should be lower than the Coulomb blockade threshold voltage V_t of the transistor), one electron is injected into the trap: $n \rightarrow 1$. In this charge state the Cou-

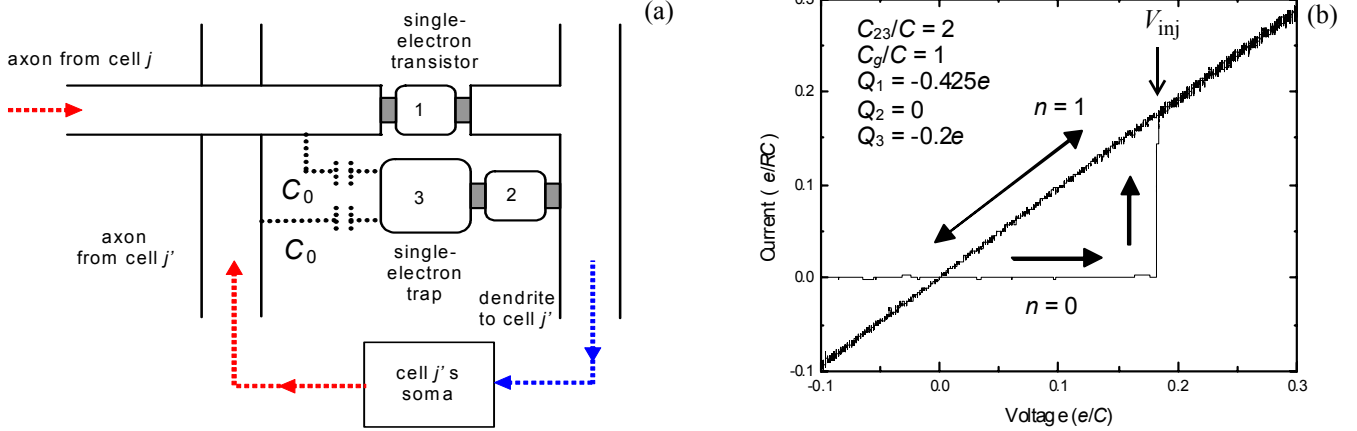


Fig. 2. (a) Schematics of the “Hebbian” (three-terminal) single-electron latching device (four of which constitute an “Hebbian” synapse - Fig. 4) and (b) results of its modeling within the “orthodox” theory of single-electron tunneling.

lomb blockade of the transistor is lifted, making the wires connected by a finite resistance (for a symmetric SET, close to the tunnel resistance R of a single barrier) at any V . However, if the node activity (voltage V) is low for a long time, eventually either the high-order quantum process (“co-tunneling”) or the unavoidable thermal fluctuations kick the trapped electron out of the trap, and the transistor closes, disconnecting the wires.

The results of Monte Carlo modeling such switching are shown in Fig. 2b. It shows that if the device parameters are set properly, the device may be modeled, at the circuit level, just as a resistor R (in the connected state) or the infinite resistance (in the disconnected state). We have to emphasize, however, that the synapse connection is not deterministic but rather probabilistic. (This is the main feature of all single-electron devices.) If voltage V does not approach V_{inj} too closely, the probability p of the trap being filled (and hence the synapse connected) may be described by the master equation

$$dp/dt = \Gamma_i(1-p) - \Gamma_e p, \quad (\text{EQ 2})$$

where the switching rates may be, with a good precision, described by the Arrhenius law:

$$\Gamma_i = \Gamma_0 \exp\{(V-S)/T\}, \quad \Gamma_e = \Gamma_0 \exp\{-V+S/T\} \quad (\text{EQ 3})$$

where T is the effective temperature (in voltage units), and S is a shift that may be changed, e.g., by a voltage applied to the trap island via an additional gate.

Figure 3 shows a possible single-molecular implementation of the latching switch [9]. Acceptor groups play the role of single-electron islands, while oligo-phenylene ethynylene (OPE) chains work as tunnel barriers. The function of thiolate groups at the end of the chains, is to work as “alligator clips” providing self-attachment of the molecules from solution to gold nanowires, so that the molecule will bridge a gap between them, if its width is close to the length of the molecule. The synthesis and experimental study of such single-molecular devices by our Stony-Brook centered collaboration is in progress.

III. HEBB RULE IMPLEMENTATION

Equation (1) shows that a single “Hebbian” latching device connection depends on the sum (or difference) of neural signals rather than on their product, as necessary for the implementation of the classical Hebb rule - see, e.g., Ref. 10. Moreover, the connection is generally random, with only their

probability dynamics depending on neural activity. However, the strong nonlinearity of Eq. (3) does allow to approximate the Hebb rule in a group of four latching switches - see Fig. 4. As this figure shows, these devices differ by two factors:

(i) the signs of signals sent to two axonic wires that host the device - see Eq. (1),

(ii) the signs of the somatic amplifier inputs fed by the device through the corresponding dendritic wire. The rule is that the sign of the dendrite input is same as the sign of the axon voltage of the destination neuron.

To understand how this rule works, consider the simplest case of constant and strong signals ($x_j = \pm 1$). The solution of Eqs. (2), (3) at $t \rightarrow \infty$ is $p = (1/2)\{1 + \tanh[(V-S)/T]\}$. If shift S satisfies the condition

$$T \ll S \ll V_0, \quad (\text{EQ 4})$$

then only one of four devices (the one having $\pm x_j \pm x_{j'} = +2$) will have the connection probability p very close to 1, while the rest three will be almost certainly disconnected ($p \approx 0$). Hence the signal sent to the target cell is virtually deterministic and proportional to $(V_0/R) \times \text{sgn}(x_j x_{j'})$; that may be interpreted as the synaptic weight

$$w_{j,j'} = \text{sgn}(x_j x_{j'}), \quad (\text{EQ 5})$$

Therefore the “clipped” Hebb rule is recovered.

For the sake of symmetry, in CrossNets, two closely located somatic cells are connected by four 4-synapse groups: two groups in one direction (Fig. 4) and two similar groups in the feedback direction. Combining all these connections for all the signal directions, we may see that all synaptic connections naturally fall into similar square “plaquettes”, with 32 latching devices each - see Fig. 5a. Each CrossNet is essentially a field of these plaquettes, and various CrossNet species differ only by the method of insertion of sparse somatic cells into this field. CrossNet is described by a system of equations of motion for (normalized) dendritic voltages x_j :

$$\left(\tau \frac{\partial}{\partial t} + 1\right)x_j = f\left(g \sum_j w_{j,j'} x_{j'}\right), \quad (\text{EQ 6})$$

where $g = GR_L/R$ is the effective somatic gain, $f(x)$ is an activation function ($|f(x)|_{\max} = 1$, $df/dx|_{x=0} = 1$), while $\tau = 2MC_0R_L$ is the effective time constant. (We assume that the load resistance, see Fig. 4b-c, is low: $R_L \ll R/M$.) The summation in Eq. (6) is over neurons j' that are connected to neuron j .

Most our results so far have been obtained for the Inclined

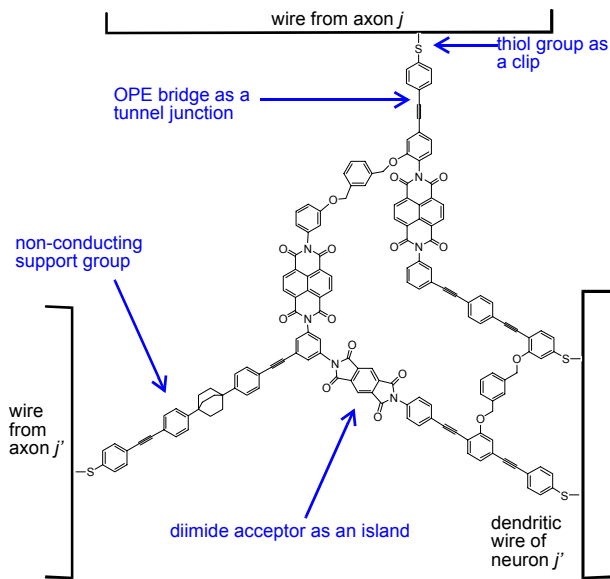


Fig. 3. Possible molecular implementation of the “Hebbian” synapse [9].

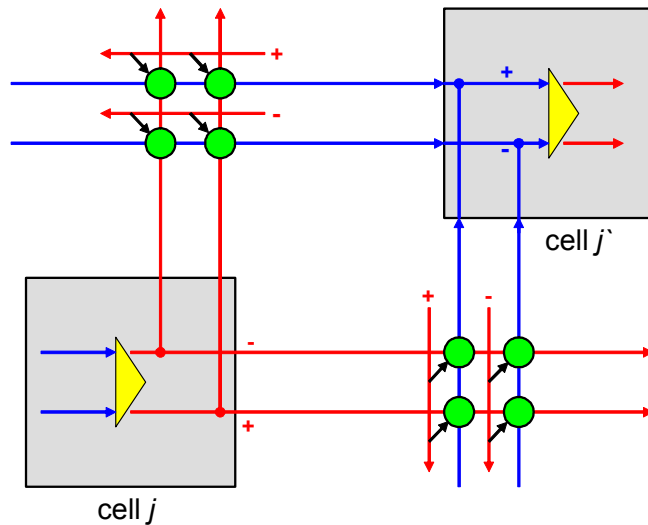


Fig. 4. Feed forward from one cell to another in CrossNet arrays. Red lines show axonic wires, blue dendritic wires. Green dots show molecular switches (Fig. 2), four of which constitute a “Hebbian” synapse, and gray squares show somatic cells. The axons shown by short lines come from the corresponding outputs of the target cell j' .

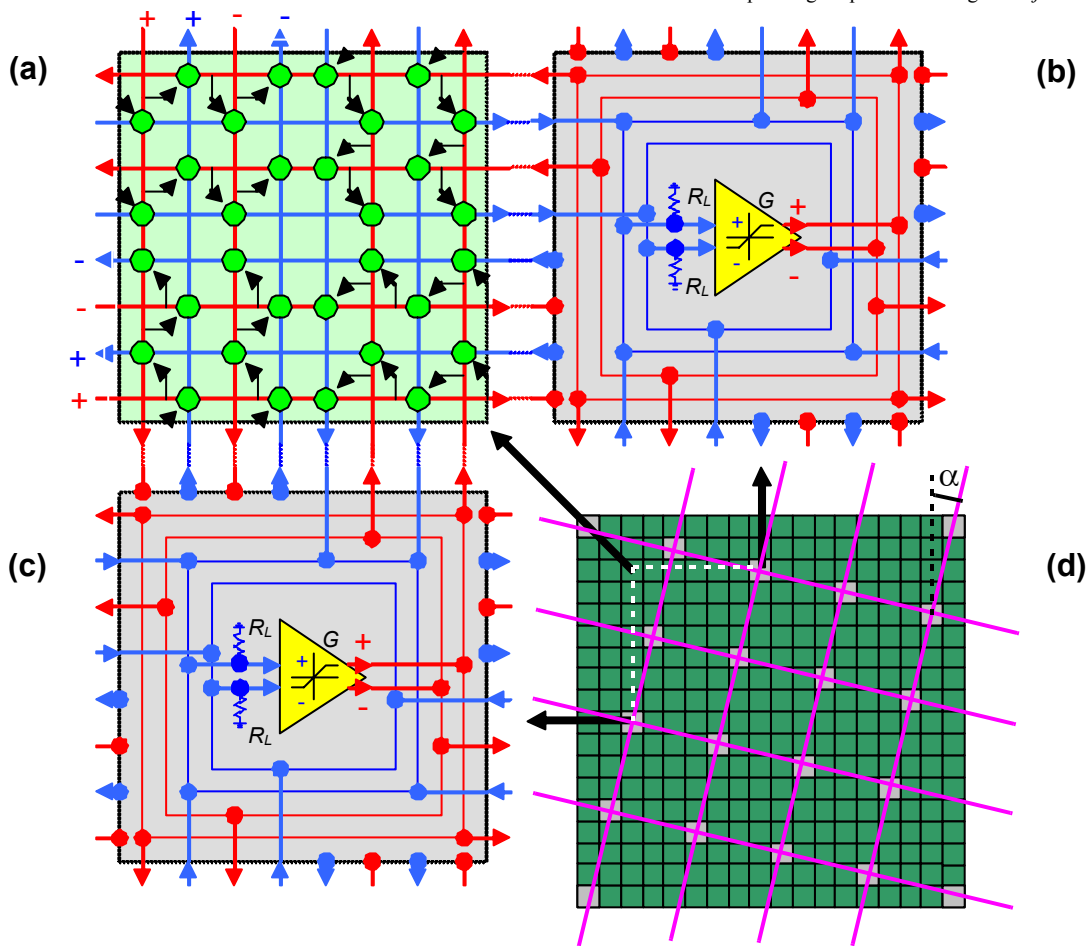


Fig. 5. CrossNet: synaptic plaquettes (green cells) and somas (gray cells) and their connection with axonic (red) and dendritic (blue) wires. Solid points on the somatic cell borders denote open-circuit terminations of the dendritic and axonic wires; due to these terminations the neural cells in the same row or column of the InBar do not interact. The bottom right panel shows the most explored CrossNet architecture, the Inclined Crossbar (“InBar”).

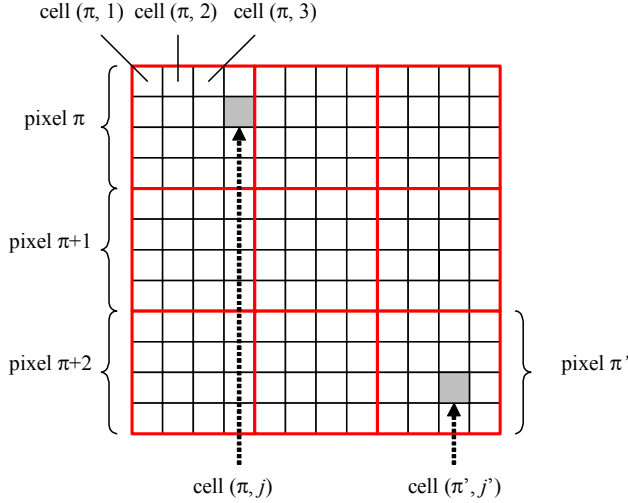


Fig. 6. Cell numbering in the pixelization method description

Distributed Crossbar (“InBar”) topology – Fig. 5d. In this network, the somas are located on a square lattice that is inclined relative to the synaptic plaquette lattice. Because of this incline, axons and dendrites that belong to a certain neuron reach all cells within the distance of $M = 1/(\tan\alpha)^2$ plaquettes in each direction. As a result, each neuron is connected to $4M$ other neurons.

The most prominent alternative to InBar is the Randomly Distributed Crossbar (“RandBar”) [8]. In that topology, the somatic cell insertion is random. This placement has two major implications:

(i) Lengths of axonic and dendritic wires are random, obeying the Poisson distribution. The mean length of these wires (expressed in synaptic plaquettes) is M , which is just the ratio of the number of synaptic plaquettes to the number of somatic cells. (The latter relation is valid for InBar as well.) As a result, RandBar features (rare) very long connections, while in InBar all connections are of a fixed Manhattan length (M).

(ii) Since the insertion of a somatic cell interrupts incoming axonic wires and outgoing dendritic wires, two neural cells of RandBar may be connected by either two or one (or none) 4-device groups each way, while in InBar the number is either two (for closer cells) or none (for distant cells). As a result, synaptic weights $w_{j,j'}$ in RandBar may be asymmetric even at the quasi-deterministic synapse settling described above.

IV. HOPFIELD-MODE TRAINING

In order to comprehend the basic properties of CrossNets, we have started with their training (using the model described above) as Hopfield networks. The basic limitations imposed by our hardware in this context are as follows:

(i) An individual synapse cannot be accessed directly, and external signals may be only fed into the somatic bodies of the neural cells. (This is especially convenient in InBar where each soma may sit of the crossing of two global wires - see purple lines in Fig. 5d.)

(ii) The cell connections are local, with each cell reaching $4M$ others (for RandBar, on the average). According to the previous results on diluted Hopfield networks [10], one can expect the maximal number of stored images P to scale as M .

(iii) Even this maximum is not easy to reach, because of the probabilistic character of synaptic connections. In fact,

any deviation of the synapse connection probability p from 0 or 1 plays the role of additional noise that sums up with the “noise” of undesirable images [10, 11].

Still, we have managed to perform the Hopfield training by two methods.

A. Pixelization

In this method we divide the system into *pixels*, each with P neurons, P being the number of patterns we want to teach. Each neuron can be identified by two numbers, the pixel number $0 \leq \pi < N/P$ and the neuron number $0 \leq i < P$ representing the position of neuron in the pixel π - Fig. 6. Then each pair of cells, say (π, i) and (π', i') is fed, in turns, by intensive external signals $V_{(\pi, i)} = V_0 \times x_{\pi}^{[i-i']}$ and $V_{(\pi', i')} = V_0 \times x_{\pi'}^{[i'-i']}$, where x_{π}^p is the π -th pixel of the p -th image, and square parenthesis mean mod P . If conditions (4) are satisfied, the synapse connections are virtually deterministic ($p = 0$ or 1), and correspond to the synaptic weights

$$w_{(\pi, i), (\pi', i')} = x_{\pi}^{[i-i']} x_{\pi'}^{[i-i']}. \quad (\text{EQ } 7)$$

Our numerical experiments using *Njal* computer cluster [12] have confirmed that this training really work if the number P of images is not too high. An estimate of P_{\max} can be obtained in the following way: Let all neurons in the same pixel be the same potential initially, corresponding to the pattern that we want to recover: $x_{\pi}^i(t=0) = x_{\pi}^0$ for all i . Then Eqs. (6) and (7) give

$$\begin{aligned} \left(\tau \frac{\partial}{\partial t} + 1\right) x_{\pi} \Big|_{t=0} &= g \sum_{\pi'} x_{\pi'}^0 \sum_p x_{\pi}^p x_{\pi'}^p \\ &= g \sum_{\pi'} \sum_{p \neq 0} x_{\pi}^p x_{\pi'}^p x_{\pi}^0 + 4x_{\pi}^0 M/P. \end{aligned} \quad (\text{EQ } 8)$$

The first term in the last equation is a random walk of unit step, with $4M$ terms; therefore it has Gaussian probability distribution with mean 0 and variance $4M$. If we want only $\varepsilon N/P$ of the pixels to be wrong, then

$$\begin{aligned} \varepsilon &= \text{Prob} \left(\sum_{\pi' \neq 0} (x_{\pi}^p x_{\pi'}^p x_{\pi}^0) / x_{\pi}^0 + (4M)/P < 0 \right) \\ &= \int_{-\infty}^{-M/P} \frac{\exp\{-x^2/(8M)\}}{\sqrt{8\pi M}} dx = \frac{1}{2} - \frac{1}{2} \text{erf}\left(\frac{\sqrt{2M}}{P}\right). \end{aligned} \quad (\text{EQ } 9)$$

From here we get $P_{\max} \approx [2/f(\varepsilon)]M^{1/2}$, where $f(\varepsilon)$ is the solution to the equation $\varepsilon = \{1 - \text{erf}[f(\varepsilon)]\}/2$. For tolerance $\varepsilon = 10^{-2}$, $P_{\max} = \sqrt{\alpha_1(4M)}$ with $\alpha_1 = 0.18$ for large M . One can see that for this training method the network capacity is low (proportional to $M^{1/2}$). Our numerical simulation results are in accord with this estimate.

B. Externally Summed Voltages

In this method, we again address each neuron pair individually, setting all axonic voltages to zero, except of the neurons in the selected pair, i and j :

$$V_i = V_0 \sum_p x_i^p x_j^p, \quad V_j = V_0. \quad (\text{EQ } 10)$$

In this case, according to Eq. 5, the synaptic weights are:

1. For both teaching methods, more than one neuron pair can be trained at a time, but the condition is that there is no synaptic connections between the pairs.

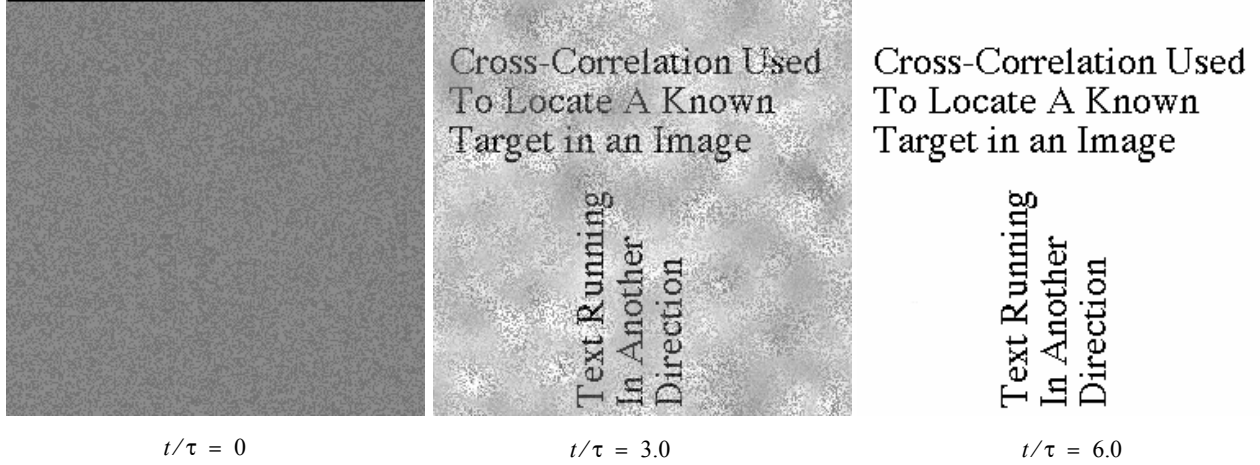


Fig. 7. Image restoration by a 256×256 pixel InBar, that had been taught 3 images using Method B. The initial state was obtained by flipping 40% of randomly selected pixels of one of the taught images, that was identical to the final image shown in the right frame.

$$w_{i,j} = \text{sgn} \left(\sum_p x_i^p x_j^p \right). \quad (\text{EQ 11})$$

This is the well known clipped-synapse Hopfield network [13], but with the sum over just $4M$ neighbors connected to each cell, rather than over all N cells of a completely connected network. To estimate the network capacity, we need to calculate the probability that the total input of neuron i , h_i , has the same sign as pattern at that neuron, x_i^0 . Let's introduce variable Z defined as $Z = h_i x_i^0$; then

$$Z = \sum_j \text{sgn} \left(x_i^0 x_j^0 \sum_p x_i^p x_j^p \right) = \sum_j \text{sgn} \left(1 + \sum_{p \neq 0} x_i^p x_j^p \right) \quad (\text{EQ 12})$$

where we have used Eq. (11) and the fact that x_i^0 is either 1 or -1. Summation inside the signum can be considered as a random walk with $P-1$ terms with step size 1. Therefore it has a probability distribution of a Gaussian of mean zero and variance $P-1$. This leads to the fact that probability of sgn to be 1 is $[1 + \text{erf}\{1/\sqrt{2(P-1)}\}]/2$. Now consider the summation over signums which has $4M$ terms. This is again a random walk with step size 1, but this time it is biased since step 1 has a higher probability than step -1. Then, probability distribution of Z is a Gaussian with variance $4M$ and mean $4M(p_{+1} - p_{-1}) = 4M \text{erf}\{1/\sqrt{2(P-1)}\}$:

$$P(Z) = \frac{1}{\sqrt{8\pi M}} \exp \left\{ - \left[Z - 4M \text{erf} \left(\frac{1}{\sqrt{2(P-1)}} \right) \right]^2 / (8M) \right\} \quad (\text{EQ 13})$$

Integrating $P(Z)$ over $-\infty$ to 0 we find ratio of wrong bits ϵ :

$$\epsilon = \frac{1}{2} - \frac{1}{2} \text{erf} \left[\sqrt{2M} \text{erf} \left\{ \frac{1}{\sqrt{2(P-1)}} \right\} \right] \quad (\text{EQ 14})$$

Hence $P_{\max} \approx 4M/\pi^2(\epsilon)$. For tolerance level $\epsilon = 10^{-2}$, one recovers $P_{\max} = \alpha_2(4M)$ with $\alpha_2 = 0.118$ for large M . This is close to the standard estimate of the Hopfield network capacity, with the connectivity parameter playing the role of the total number of cell. It is interesting that if applied to a fully connected network of N cells, this method would give $P_{\max} \approx 0.118N$ instead of the exact result $P_{\max} = 0.10N$ [10,13].

One can see that P_{\max} for the second teaching method is much higher than that for the first method. Again, the results of our numerical simulations agree with this estimate. For example, Fig. 7 shows three snapshots of the image restora-

tion by a 256×256 pixel InBar with the connectivity parameter $M = 64$, that have been taught three black-and-white images. In this particular case, the convergence to the original image is perfect.

V. CROSSNETS AS CONTINUOUSLY RUNNING NETWORKS

Even though Hopfield networks are very interesting (because of their analytic tractability) and may be useful as associative memories, they cannot perform more advanced functions, starting from image classification [10, 11]. This is why it is imperative to explore CrossNet operation as a continuously running neural network, still with due respect to the hardware imposed limitations listed in the previous section. The main additional factor here is CrossNets' deep recurrency; generally this property can lead to system latch-up in one of its "spin-glass" states, making the effective processing of information impossible.

Fortunately, in CrossNets the latch-up problem may be overcome because of the sign symmetry of its synaptic weights. Let the synaptic connections be random, with the

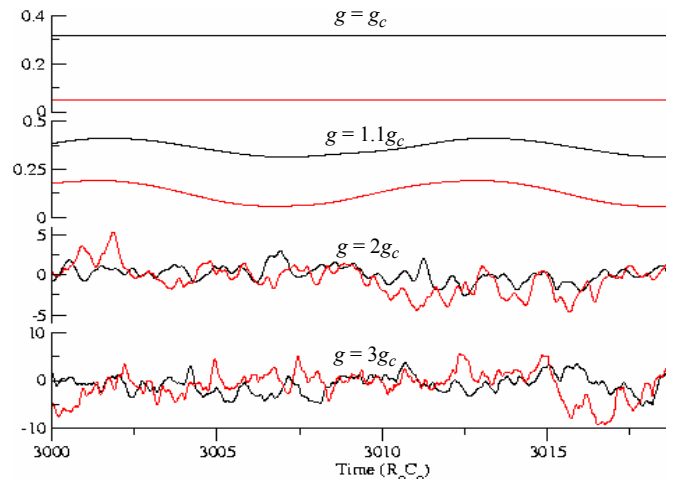


Fig. 8. Sample dendritic voltages of a system of 2500 neurons with $M=16$ for two neurons in RandBar topology. The top plot is where gain of the amplifiers g is exactly equal to the critical gain g_c . As gain is increased, first neurons starts oscillating in a few modes (second plot from top) and becomes completely chaotic as $g=3g_c$ (the bottom plot).

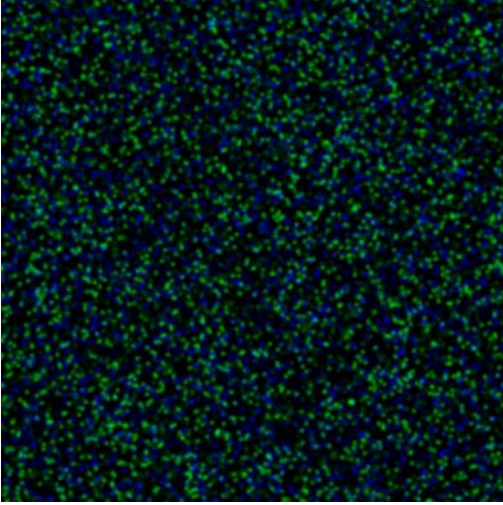


Fig. 9. A snapshot of the 2500 neuron InBar with $M=16$ at gain $g=1.5g_c$. Green and blue regions represent positive and negative axonic voltages, respectively.

same connection probability $p = 1/2$ for all synapses. (This is easy to sustain by setting shift S to zero, and making the somatic cell saturation level relatively low, $T \ll V_0 \ll V_{inj}$). In this case the average synaptic weight $w_{ij} = 0$. This means that for any closed signal loop passing through several neurons, there is an equal chance to have a total positive feedback and total negative feedback, at zero frequency. Loops with positive feedback, of course, tend to form local dc latch-ups if the somatic cell gain g is high enough. Equation (6) show that at a finite frequency ω , each intercell connection introduces an additional phase shift $\Delta\phi = \tan^{-1}(\omega\tau)$. Therefore, at sufficient gain, self-excitation will take place in the negative-feedback loops as well, but at a finite frequency $\omega \sim 1/\tau$. At high connectivity, $M \gg 1$, oscillations in various loops interfere in the same cells, and due to the somatic amplifier nonlinearity, interact with each other and with the local dc latch-ups. At relatively large g this interaction leads to chaotic dynamics [7,8]. This effect is illustrated in Fig. 8. Notice that near the critical value (g_c) of the gain, local dc latch-ups develop first. At the further increase of gain, the system develops quasi-sinusoidal ac oscillations. At larger gain, more and more oscillations appear, and at $g \sim 2g_c$ the process is almost completely chaotic. Figure 9 shows that the spatial distribution of neural activity is also virtually random.

In the light of this property, a possible method of continuous network training may be as follows. Suppose, in a CrossNet we designate a relatively small number $O \sim N/M \ll N$ neurons as output cells, and a larger number $I (O \ll I \ll N)$ neurons as input cells, leaving all other cells to play the role of the hidden “layer” (though this term is hardly applicable to a deeply recurrent network such as CrossNet). Setting the initial shift S to zero, we ensure that all synapses are random, so that increasing g above g_c , we get into the chaotic oscillation regime. Now let us impose the first image of the training set on input neurons. As soon as the output signals match (or correlate considerably with) our desired output, we increase shift S to $\sim V_0$ for a short time $\Delta t \sim 1/\Gamma_0$, allowing a partial synaptic adaptation. Repeating this procedure for all patterns, and possibly repeating the same training several times, we should arrive at the system with fixed synaptic weights (where the chaos disappears) that hopefully will provide effective image classification.

There are many open questions about the applicability of this method. Most importantly, since the available phase space is quite large compared to the space of the desired outputs, it might take impractically long time for the system to learn. The numerical and analytical study of this problem is the first priority of our future work.

VI. DISCUSSION

Due to potentially low cost of chemically-directed self-assembly of single-molecule devices, CrossNets based on the CMOL technology may become the first artificial neuromorphic networks with the areal density comparable to that of the cerebral cortex ($\sim 10^7$ neurons per cm^2), operating at much higher speed at acceptable power consumption. If created and trained to perform high-quality image classification and more advanced tasks, these networks will probably be able to self-evolve (and self-evolve fast!) in the process of their interaction with the environment. Such a development would have an unparalleled impact on information technology.

ACKNOWLEDGMENTS

Fruitful discussions with P. Adams, J. Barhen, V. Protopopescu, and T. J. Sejnowski are gratefully acknowledged. A. Mayr has kindly allowed the use of Fig. 4 before the publication of [9]. The work was supported in part by ARDA via ONR, DOE (directly and via ORNL), and NSF. Most numerical calculations have been carried out on Njal computer cluster that was acquired with a grant from DoD's DURIP program via A FOSR.

REFERENCES

- [1] H. Park *et al.*, “Nanomechanical Oscillation in a Single C_{60} Transistor”, *Nature* **407**, 57 (2000).
- [2] N. B. Zhitenev, H. Meng, and Z. Bao, “Conductance of Small Molecular Junctions”, *Phys. Rev. Lett.* **88**, 226801 (2002).
- [3] J. Park *et al.*, “Coulomb Blockade and the Kondo Effect in Single-Atom Transistors”, *Nature* **417**, 722 (2002).
- [4] K. Likharev, “Electronics Below 10 nm”, accepted for publication in: *Giga and Nano Challenges in Microelectronics*, to be published by Elsevier, Amsterdam (2003); preprint available on the Web at rsfq1.physics.sunysb.edu/~likharev/nano/NanoGiga010603.pdf.
- [5] The International Technology Roadmap for Semiconductors, 2001 Edition, 2002 Update, available on the Web at public.itrs.net.
- [6] J. H. Fendler, “Chemical self-assembly for electronics applications”, *Chemistry of Materials* **13**, 3196 (2001).
- [7] S. Fölling, Ö. Türel and K. K. Likharev, in *Proc. of the 2001 Int. Joint Conf. on Neural Networks (IEEE/Omnipress 2001)*, p. 216.
- [8] Ö. Türel and K. K. Likharev, *Int. J. of Circuit Theory and Appl.* **31**, 37 (2003).
- [9] K. K. Likharev, A. Mayr, I. Muckra and Ö. Türel, in *Proc. of the 6th Engineering Foundation Conf. on Molecular-Scale Electronics (Key West, FL, December 2002)*, to be published by the New York Academy of Sciences, New York (2003).
- [10] J. Hertz, A. Krogh and R. G. Palmer, *Introduction to the Theory of Neural Computation* (Perseus, Cambridge, MA 1991).
- [11] P. Dayan and L. F. Abbott, *Theoretical Neuroscience* (MIT Press, Cambridge, MA 2001).
- [12] See Web site njal.physics.sunysb.edu.
- [13] J. L. van Hemmen and R. Kühn, *Phys. Rev. Lett.* **57**, 913 (1986).