INTERNATIONAL JOURNAL OF CIRCUIT THEORY AND APPLICATIONS
Int. J. Circ. Theor. Appl. 2004; **32**:277–302 (DOI: 10.1002/cta.282)

Neuromorphic architectures for nanoelectronic circuits

Özgür Türel, Jung Hoon Lee, Xiaolong Ma and Konstantin K. Likharev*,†

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

SUMMARY

This paper reviews recent important results in the development of neuromorphic network architectures (‘CrossNets’) for future hybrid semiconductor/nanodevice-integrated circuits. In particular, we have shown that despite the hardware-imposed limitations, a simple weight import procedure allows the CrossNets using simple two-terminal nanodevices to perform functions (such as image recognition and pattern classification) that had been earlier demonstrated in neural networks with continuous, deterministic synaptic weights. Moreover, CrossNets can also be trained to work as classifiers by the faster error-backpropagation method, despite the absence of a layered structure typical for the usual neural networks. Finally, one more method, ‘global reinforcement’, may be suitable for training CrossNets to perform not only the pattern classification, but also more intellectual tasks. A demonstration of such training would open a way towards artificial cerebral-cortex-scale networks capable of advanced information processing (and possibly self-development) at a speed several orders of magnitude higher than that of their biological prototypes. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: nanoelectronics; single-electron devices; nanowires; CMOS; hybrid circuits; neuromorphic networks; fuzzy synapses; crossbar arrays; self-development; adaptation

1. INTRODUCTION

VLSI circuits with sub-10-nm device features could provide enormous benefits for all information technologies, including information storage, processing, and transfer [1]. However, recent results [2, 3] indicate that the current VLSI paradigm (based on a combination of lithographic patterning, CMOS circuits, and Boolean logic) can hardly be extended into this region. The main reason is that at gate length below 10 nm, the sensitivity of parameters (most importantly, the gate voltage threshold) of silicon field-effect transistors to inevitable fabrication spreads grows exponentially. As a result, the gate length should be controlled with a few-angstrom accuracy, far beyond even the long-term expectations of the semiconductor industry [1]. Even if such accuracy can be technically implemented using sophisticated

*Correspondence to: K. K. Likharev, Stony Brook University, Stony Brook, NY 11794-3800, U.S.A.
†E-mail: klikharev@notes.cc.sunysb.edu

Contract/grant sponsor: Division of Computer-Communication Research, U.S. National Science Foundation
Contract/grant sponsor: U.S. Air Force Office of Scientific Research

Received 11 April 2004
Revised 2 June 2004

patterning technologies, this would send the fabrication facilities costs (growing exponentially even now) skyrocketing, and lead to the end of Moore's Law some time during the next decade.

Some alternative nanodevice concepts, for example quantum interference devices [4] or single-electronics [5], offer some potential advantages over MOSFETs, including a broader choice of possible materials. Unfortunately, the minimum features of these devices (e.g. the single-electron transistor island size) for room-temperature operation should be below ~ 1 nm [3, 5]. Since the relative accuracy of their definition has to be of the order of 10%, the absolute accuracy should be of the order of an angstrom or less, again far too small for the current and even realistically envisioned lithographic techniques.

This is why there is a rapidly growing consensus that the impending crisis of the microelectronics progress may only be resolved by a radical paradigm shift from lithography to the 'bottom–up' fabrication. In the latter approach, the smallest active devices should be formed in some special way (for example, synthesized chemically), ensuring their fundamental reproducibility. An example of such unit is a specially designed and synthesized molecule comprising of a few tens or hundreds of atoms.

Unfortunately, integrated circuits consisting of molecular-size devices alone are hardly viable, because of their limited functionality. For example, voltage gain of a 1-nm-scale transistor, based on any known physical effect (e.g. the field effect, quantum interference, or single-electron charging), cannot exceed one, i.e. the level necessary for the operation of virtually any active analog or digital circuit [3]. This is why the only plausible way toward high-performance nanoelectronic circuits is to integrate nanoscale (e.g. molecular) devices, with the connecting nanowires, on the top of CMOS chips whose field-effect transistors would provide the circuit with the necessary additional functionality, in particular high voltage gain. The practical implementation of such hybrid integration, of course, faces several hard challenges, in particular that of interfacing the nanowires (whose half-pitch can eventually reach a few nanometers) with cruder, lithographically-defined CMOS-level wiring. We believe that the recent suggestion of a specific species of CMOS/nanodevice hybrids, called 'CMOL' (standing for CMOS/MOLecular circuits) [6] has opened an efficient way for the solution of the interfacing problem.[‡]

A CMOL circuit (Figure 1) would combine an advanced CMOS subsystem with two, mutually perpendicular, arrays of parallel nanowires and similar nanodevices formed at each crosspoint of the nanowires. The reason for this topology is that parallel nanowire arrays may be fabricated by several high-resolution patterning technologies, such as nanoimprint [9] or interference lithography [10]. These novel technologies cannot be used for patterning of arbitrary integrated circuits, in particular because they lack an adequate layer alignment accuracy, but the crosspoint topology does not require such alignment. This approach, of course, requires a nanodevice formation process that also does not need lithographic patterning. An example of such process is the chemically directed self-assembly of pre-synthesized molecules from solution [11, 12].

In contrast with the earlier suggestions of crossbar-like hybrid circuits [7, 8], in CMOL chips the interface between the CMOS and nanowire/nanodevice subsystems is provided by pins that

[‡] Previously suggested solutions of the problem (for a recent review, see, e.g. References [7, 8]) do not seem technologically feasible.

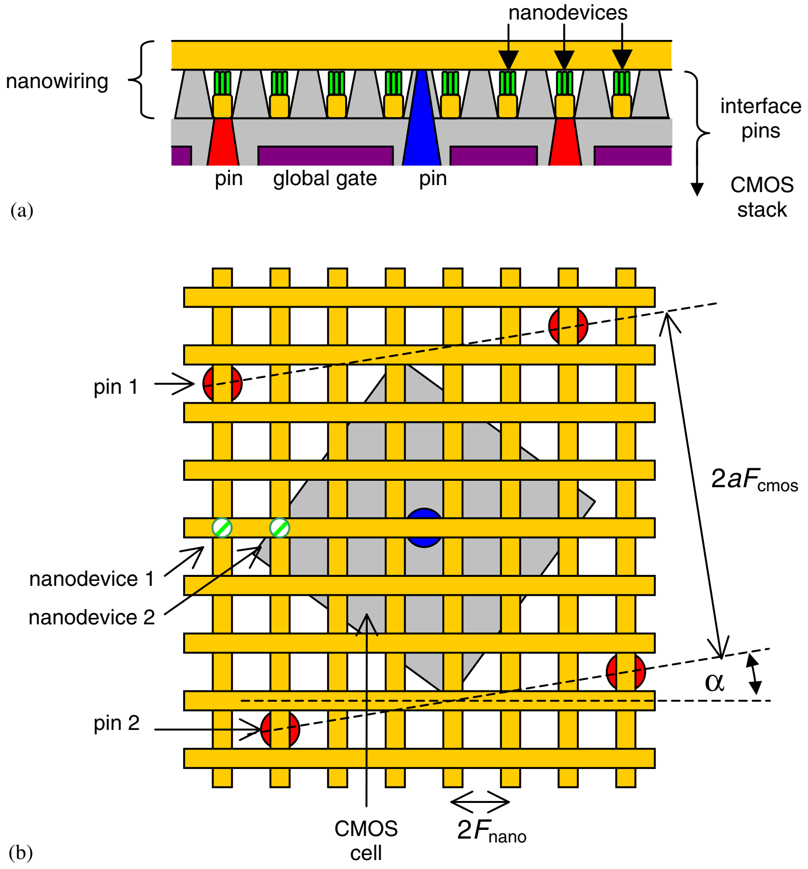

Figure 1. General structure of a CMOL circuit: (a) side view and (b) top view (schematically). For the sake of clarity, panel (b) shows only one CMOS cell (serving the central interface pin) and two nanodevices (actually, a similar device has to be formed at each nanowire crosspoint).

are distributed all over the circuit area. The interface pins are of two types (providing contacts to the lower and higher level of nanowiring); pins of each type are located on a square lattice of period $2aF_{\mathrm{CMOL}}$ that is inclined by a small angle $\alpha = \arctan(F_{\mathrm{nano}}/aF_{\mathrm{CMOS}}) \ll 1$ relative to the nanowire arrays. (Parameter a is defined by the area $A = 2a^2F^2_{\mathrm{CMOS}}$ of the CMOS cell serving each pin.) This trick allows an individual access to each nanowire crosspoint even if the ratio $F_{\mathrm{nano}}/aF_{\mathrm{CMOL}}$ is very small. For example, if the CMOS system applies, via the pin shown in blue in Figure 1(b), voltage V_{h} to the corresponding horizontal nanowire, and voltage $-V_{\mathrm{v}}$ (through red pin 1) to the vertical nanowire shown leftmost, then nanodevice 1 will be biased with larger voltage $(V_{\mathrm{h}} + V_{\mathrm{v}})$ than any other device. For non-linear nanodevices with a sharp threshold voltage V_{t} (within the range $V_{\mathrm{h}}, V_{\mathrm{v}} < V_{\mathrm{t}} < V_{\mathrm{t}} + V_{\mathrm{v}}$), such selection allows the activation of a single device of the whole array. By moving the bias $-V_{\mathrm{v}}$ from pin 1 to, e.g. pin 2 (Figure 1(b)) we may alternatively select nanodevice 2, etc. Note that the distance between the individually selected nanodevices may be as small as $2F_{\text{nano}}$, i.e. much less that the CMOS wiring pitch $2F_{\text{CMOS}}$.

The CMOL approach may enable, in future, an unprecedented density of useful devices. The only fundamental physical limitation here is the direct quantum tunneling between the nanowires; it limits the half-pitch F_{nano} at the level of the order of 3 nm and hence the nanodevice density at approximately $10^{12}\,\text{cm}^2$. Moreover, since the density of CMOS devices may be much lower than that number, the total fabrication costs of CMOL chips may be quite acceptable. The development of this technology (especially the nanodevice formation, e.g. the molecular self-assembly) will certainly require a major industrial effort and substantial time period, probably not less than 10–15 years. However, this timing may be still acceptable to prevent the impending crisis of Moore's Law.

The main architectural challenge for CMOL circuits is that even after the anticipated extensive development, the bottom–up approach to fabrication will hardly allow nanodevice formation with yield approaching 100%. This is why CMOL circuits seem more suitable for the implementation of such defect-tolerant circuits as embedded and stand-alone memories [7, 8, 13] and neuromorphic networks [14–17], rather than Boolean logic circuits for which ways to provide such tolerance still have to be found [18]. The prospects for CMOL memories have been addressed in our recent paper [13]. The goal of this work was to review the current status of the development of neuromorphic networks ('CrossNets') compatible with CMOL technology.

Section 2 reviews the basic structure of CrossNets and major challenges to their training. Section 3 describes one possible approach to the training, the synaptic weight import. In Section 4, we illustrate the application of this strategy to the Hopfield-mode operation of recurrent CrossNets of 'InBar' variety. (In the same section we demonstrate that at least in this operation mode the CrossNets may be highly defect-tolerant.) The application of the same approach to feedforward CrossNets with a different ('FlossBar') topology is discussed in Section 5, while in Section 6 we discuss its applicability to feedforward InBars. In Section 7, we show that CrossNets can be also trained by direct error backpropagation method, without the need for a software precursor network. The bottom line at that point is that CrossNets may be taught to perform virtually any function that artificial neural networks have ever been used for. In Section 8, we describe a possible alternative CrossNet training technique ('global reinforcement') that may also be used for more complex information processing tasks. Finally, in Conclusion (Section 9) we discuss possible performance of CMOL CrossNet networks and prospects for large-scale, hierarchical systems for advanced information processing, based on such networks.

2. CROSSNETS

We have proposed [14–17] a family of neuromorphic circuits, called Distributed Crossbar Networks ('CrossNets'), whose topology is uniquely suitable for CMOL implementation. Like most artificial neural networks explored earlier (see, e.g. References [19–22]), each CrossNet consists of the following components:

(i) Neural cell bodies ('somas') are relatively sparse and hence may be implemented in the CMOS subsystem. Most our results so far have been received within the simplest

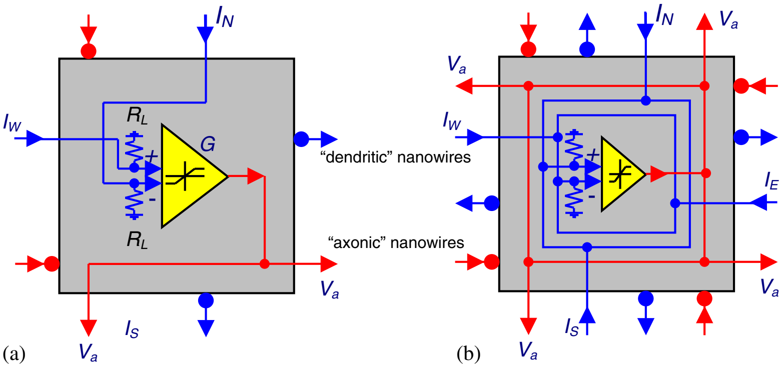

Figure 2. Structure of neural cell bodies (somas) of: (a) feedforward; and (b) recurrent CrossNets in the operation mode. Low input resistances R_L are used to keep all input ('dendritic') voltages $V_{\rm d} = R_L \Sigma_i I_i$ well below the output ('axonic') voltage $V_{\rm a}$, for any possible values of net input currents I_i, thus preventing undesirable anti-Hebbian effects [14]. G is the voltage gain of the somatic amplifier at the linear part of its transfer ('activation') function $f(x)$—see Equation (5) below. Bold points show open-circuit terminations of nanowires, that do not allow somas to interact in bypass of synapses (see below).

'firing rate' model, in which somas operate just as a differential amplifier with a nonlinear saturation ('activation') function (Figure 2).

(ii) 'Axons' and 'dendrites' are implemented as physically similar, straight segments of mutually perpendicular metallic nanowires (Figure 1). Somatic load resistances R_L (Figure 2) keep all dendritic wire voltages $V_{\rm d}$ much lower than axonic voltages $V_{\rm a}$. Estimates show that wire resistances may be negligible in comparison with nanodevice resistances, even in the open state (see below). On the contrary, capacitance of the wires cannot be neglected and (in combination with R_L) determines the CrossNet operation speed.

(iii) Synapses, each comprising one or several similar nanodevices, are formed at crosspoints between axonic and dendritic nanowires (Figure 1). In the light of the recent spectacular demonstration of single-molecule single-electron devices by several groups [23–27], they seem to be the most attractive option for synapse implementation.

Figure 3(a) shows the schematics of the simplest single-electron device, latching switch, that has functionality sufficient for CrossNet operation and allows a natural molecular implementation (Figure 3(b)). The device is essentially a combination of two well-known single-electron devices: the transistor and the 'box' [5].[§] If the applied voltage $V = V_{\rm a} - V_{\rm d}$ is low, the box island in equilibrium has no extra electrons ($n = 0$), and its total electric charge $Q = -ne$ is zero. As a result, the transistor is in the closed ('Coulomb-blockade') state, and input and output wires are essentially disconnected. If V is increased beyond a certain threshold value

[§] This is a simplified version of the device suggested by our group earlier [28]. It may be also considered as a two-terminal version of the four-terminal device discussed in Reference [29]. (Multi-terminal devices look hardly practical for any 'bottom–up' implementation, for example the molecular self-assembly.)

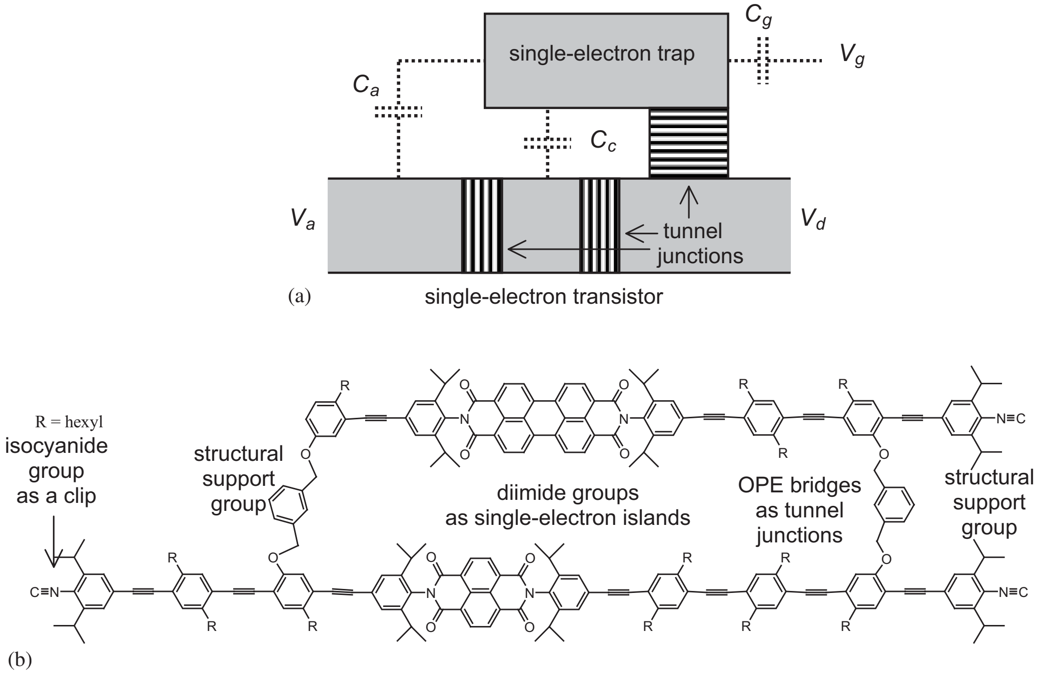

Figure 3. (a) Schematics; and (b) possible molecular implementation of a two-terminal single-electron latching switch. The tunnel barrier connecting the box island is substantially thicker than those embedding the transistor island, so that the rate of tunneling to and from the box is much lower. V_g is the voltage applied to the (quasi-) global gate (Figure 1(a)). Figure 3(b) is courtesy of Prof. A. Mayr (SBU/Chemistry).

V_{inj}, the electrostatic potential of the trap island is sufficiently increased and one electron tunnels into the box (through a thicker barrier than those of the single-electron transistor): $n \to 1$. This change of box charge affects, through the coupling capacitance C_c, the potential of the transistor island, and lifts the Coulomb blockade; as a result, the transistor connects the nanowires with a finite resistance R_0. (For a symmetric transistor, R_0 is close to the tunnel resistance of a single tunnel junction of the transistor [5].) If V stays above V_{inj}, this connected state is sustained indefinitely; however, if the synaptic activity $V(t)$ remains low for a long time, eventually thermal fluctuations will kick the trapped electron out of the box, and the transistor will close, disconnecting the wires. (Such disconnection may be forced to happen much faster by making the applied voltage V sufficiently negative.) Thus the device works as an adaptive binary-weight, analog-signal synapse.

Figure 4(a) shows the general topology of CMOL CrossNets on the examples of the simplest feedforward (a) and recurrent (b, c) networks. Any pair of cells may be connected, in one direction, by maximum two synapses leading to different somatic amplifier inputs, so that the net synaptic weight w_{jk} may take any of three values. (They may be normalized to -1, 0, and $+1$). Note that the real area of the somatic CMOS cell (shown by the light-grey square in Figure 4(a)) may be much larger than that of the interface pin area of that cell (darker-grey square); the former area is only limited by the distance between the adjacent somas.

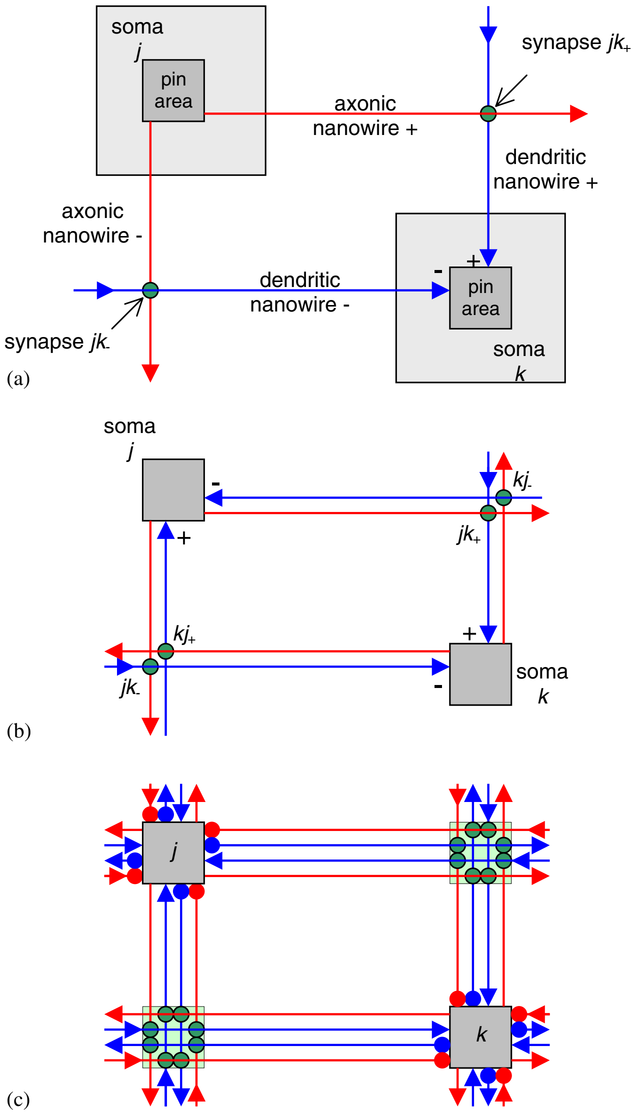

Figure 4. Cell connections in the simplest (a) feedforward; and (b, c) recurrent CrossNets [14–17]. The lines show ‘axonic’, and ‘dendritic’ nanowires. Dark-grey squares are interfaces between nanowires and CMOS-based cell bodies (somas), while light-grey squares in panel (a) show the somatic cells as a whole. (For the sake of clarity, the latter areas are not shown in the following figures.) Signs show the somatic amplifier input polarities. Circles denote nanodevices (latching switches) forming elementary synapses. For clarity, panels (a) and (b) show only the synapses connecting one couple of cells (j and k), while panel (c) shows all the nanowires connected to these two cells and all the synapses located at the crosspoints of these wires. All the synapses are located within two imaginary square ‘plaquettes’.

This distance also determines the most important topological parameter of a CrossNet, its *connectivity* that is defined as the number of cells 'directly' (via a single synapse) connected to any given cell. The mechanism of this limitation is shown in Figure 4(c): any axon running into a somatic cell is open-circuit terminated (bold red points); so is any dendritic wire starting at a somatic cell (bold blue points). These terminations do not allow cell connections bypassing synapses, and set finite lengths for axons and dendrites and hence a finite connectivity: the farther are the somatic cells, the longer are the nanowire segments, and the more synapses they contact, providing connections to more cells. This is probably the most important feature of the CrossNet topology: it ensures arbitrary connectivity (that may be, for example, as high as 10^4, the number typical for the biological cortical networks with their quasi-3D structure [30, 31]) in essentially 2D integrated circuits such as CMOL. (The price for the increase of connectivity is the operation speed-to-power tradeoff and noise immunity—see Section 9 below.)[¶]

While the somatic cell *density* in CrossNets is very important since it determines the network connectivity, the particular *location* of the cells is not too crucial (say, may be completely random [14, 15][||]) and may be directed by the convenience of either hardware implementation, or training, or both. In this review we will discuss only two particular structures.

(i) The simplest CrossNet, the so-called FlossBar (Figure 5(a)), in its feedforward version is essentially a flavor of multilayer perceptrons [19–22], with quasi-local connectivity. Thus, the study of FlossBars allows a natural comparison of CrossNets with traditional artificial neural networks (typically implemented in software running on usual digital computers).

(ii) In the so-called Inclined Crossbar (or just 'InBar', see Figure 5(b)), somatic cell pin areas are located on a square lattice that is inclined by a (small) angle α relatively to the axonic/dendritic nanowire array.[**] This geometry is more natural for CMOL implementation, because each somatic cell may have the same shape.

Preliminary estimates ([14–16]; see also Section 9 below) has shown that CMOL CrossNets may combine very high density (considerably higher than the areal density of biological synapses in the mammal cerebral cortex [30, 31]) with a peak performance at least several orders of magnitude higher than those of the human brain [30, 31], modern and realistically envisioned digital microprocessors [1], and artificial neural networks implemented on either usual serial computers or special CMOS chips [34–37]. However, the peak performance

[¶] Note two other properties of CrossBar architectures, that are crucial for CMOL implementation of such circuits:

(i) The networks use similar nanodevices that are formed at all crosspoints between axonic and dendritic nanowires (Figure 4(c)). Moreover, their operation is not disturbed by additional nanodevices formed at axonic/axonic and dendritic/dendritic crosspoints. Indeed, the former devices just lead to additional power dissipation, while the latter devices are always closed because the smallness of all dendritic voltages. Hence, CrossNets may work with nanodevices formed at ALL nanowire crosspoints.

(ii) Due to similarity of all nanowires and nanodevices, CMOL CrossNet tolerates an almost arbitrary shift between them and CMOS subsystem.

[||] Such randomness provides a small amount of very long interconnects, that has some implications [32] for statistical properties of the networks—see, e.g. Reference [33]. However, so far we have not found possible practical advantages of such random CrossNets ('RandBars' [14, 15]) over more convenient InBars (see below).

[**] There is a substantial parallel between the incline angles α shown in Figures 1(b) and 5(b). This analogy makes InBar arrays especially natural for CMOL implementation.

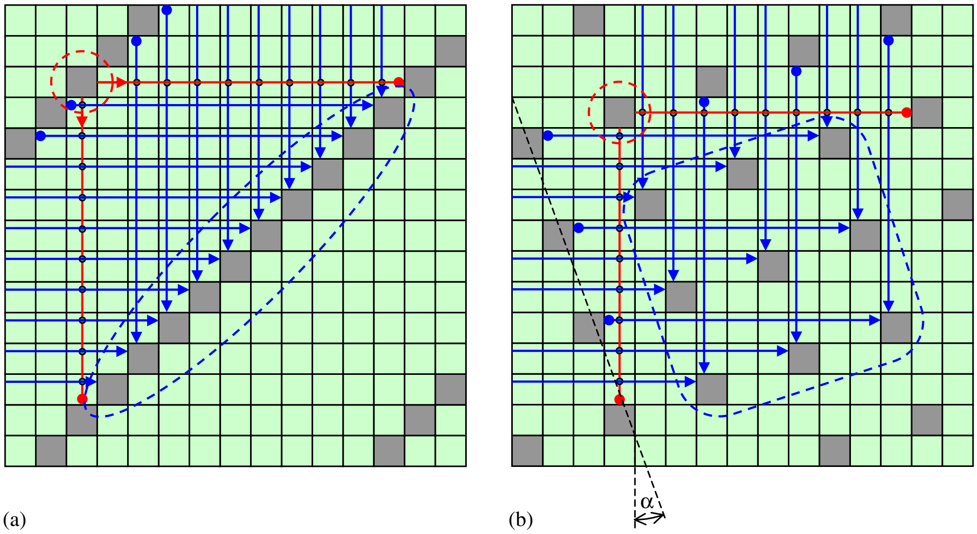

Figure 5. Two particular CrossNet species: (a) FlossBar; and (b) InBar. For clarity, the figures show only the axons, dendrites, and synapses providing connections between one soma (indicated by the dashed circle) and its recipients (inside the dashed oval), for the feedforward case. For FlossBars, the number M of direct recipients is always even (in Figure 5(a), equal to 10), while for InBars M is always the square of an integer number $M^{1/2} = 1/\tan\alpha$, where α is the angle of incline of the square lattice of somatic cells relative to the nanowire arrays. (In Figure 5(b), $M = 9$.) In recurrent CrossNets (Figure 2(b)), the cell connectivity is four times higher (equals $4M$).

advantage of CrossNets make sense only if these networks may be trained to perform efficiently at least the functions demonstrated earlier with the software-implemented artificial neural networks (including notably pattern classification [19–22, 38]) and hopefully more intelligent tasks.

Such training faces several hardware-imposed challenges:

(i) CrossNets use continuous (analog) signals, but the synaptic weights are binary, if only one latching switch for synapse is used.

(ii) The only way to reach for any particular synapse in order to turn it on or off is through the voltage $V = V_a - V_d$ applied between the two corresponding nanowires. Since each of these wires is also connected to many other switches, special caution is necessary to avoid undesirable ‘disturb’ effects.

(iii) Processes of turning single-electron latches on and off are statistical rather than dynamical [5], so that the applied voltage V can only control probability rates $\Gamma_{\uparrow\downarrow}$ of these random events.[††] Fortunately, these rates are very strong functions of V, close to the Arrhenius law

$$\Gamma_{\uparrow\downarrow} = \Gamma_0 \exp\{\pm\beta(V - S)\} \qquad (1)$$

[††] In the terms of neural network literature, CrossNets are ‘fuzzy’ systems—see, e.g. Reference [39].

where $\beta \equiv e/k_B T$, T is the effective temperature, while S is a shift parameter that depends on the switch design, and may be changed by applying voltage to a special global gate electrode (Figure 1(a)).[‡‡] Since voltage V may easily be made much higher than $k_B T/e$ (e.g. $\sim$500 vs $\sim$30 mV, respectively), the degree of randomness ('fuzziness') of switching may be restricted if necessary. For example, if Γ_0 is sufficiently low (so that $\Gamma_0 t \ll 1$, where t is the characteristic time of network operation), Equation (1) ensures that the latch turns on as soon as V exceeds the effective threshold voltage $V_+ \approx S + (k_B T/e)\ln(1/\Gamma_0 t)$, and turns off at $V < V_- \approx S - (k_B T/e)\ln(1/\Gamma_0 t)$. Due to this fact, the last of the three problems listed above is apparently the least serious one.

Until a few months ago, our work on overcoming these challenges had been focused on CrossNets with three-terminal latching switches. In particular, we have shown [15, 16] how CrossNets of a specific ('InBar') variety, based on such switches, can be used as Hopfield networks, e.g. for recognition of corrupted images. However, the practical implementation of three-terminal devices, especially self-assembly of three-terminal molecules, would present an enormous technological challenge. The placement of two-terminal devices is much easier, and for single devices it has already been demonstrated by several groups—see, e.g. References [12, 23–27]. Recently, we have shown [17] that CrossNets with two-terminal switches may have at least similar functionality, so that in this review we will discuss only this case.

3. SYNAPTIC WEIGHT IMPORT

The first CrossNet teaching procedure that allows to overcome the problems listed above is the synaptic weight import. First, a 'precursor' artificial neural network with continuous synaptic weights (say, implemented on usual computers) that is homomorphic to a CMOL CrossNet, is trained using one of existing methods [19–22]. Then the synaptic weights w_{jk} are transferred to the CrossNet, with some 'clipping' (rounding) due to the binary nature of the elementary synapses.

For the weight import operation, all latching switches are first reset to their off state.[§§] Now we can use the flexibility of the CMOS circuitry to reconfigure all somatic cells from the 'operation' configuration (Figure 2) to an 'import configuration'. The fact that somatic cells of FlossBar and InBar are located on rectangular lattices (Figure 5), allows the external teacher system to select, via CMOS-level wiring, any particular somatic cell, just like it is done in the usual semiconductor memories—see, e.g. Reference [40]. Each selected soma applies 'write enable' negative voltages with amplitude $V_0 \approx (2/3)V_t$ to its dendritic wires, and 'data' voltages $V_a = \pm V_0$, with the sign corresponding to the desirable w_{jk}, to all axonic wires. As has been explained during the discussion of Figure 1(b) above, if this procedure is carried out with a pair of cells k and j connected directly by a synapse, the net voltage

[‡‡] Such gate may be made 'quasi-global' (i.e., partitioned into sub-gates, each controlling all nanodevices belonging to a particular CMOS cell) by its patterning at the CMOS level.

[§§] This may be done, e.g. by raising shift S well above $k_B T/e$ for a short time by applying a short pulse to the global gate (Figures 1(a), 3(a)). For the sake of simplicity, we will assume that $S = 0$ during all the following operations described in this section, though such exact setting is not really necessary. In particular, in this case the turn-on/off voltages are equal and opposite: $V_- = -V_t$, $V_+ = +V_t$, where $V_t \equiv (k_B T/e)\ln(1/\Gamma_0 t)$.

$V = V_a - V_d$ applied to this synapse becomes close to $+(4/3)V_t$, i.e. beyond the threshold V_t, and the latches are reliably turned on. At the same time, all the 'half-selected' nanodevices devices, connected to only one of the activated nanowires, experience a net voltage close to $\pm(2/3)V_t$ and hence remain in the initial off state.

After all the necessary synapses have been properly set, the somatic cells are reconfigured back into the operational configuration (Figure 2) and the system is provided with appropriate input signals and/or initial conditions, and is allowed to evolve. In this 'operation' mode the activation function of somatic amplifiers limit all axonic voltages to $|V_a| \leqslant V_s < V_t$, while load resistances R_L (Figure 2) keep all the dendritic voltages even lower, so that all net voltages V are kept below V_t. Equation (1) shows the in this case synaptic weights do not change (with high probability) during the operation stage.

4. RECURRENT INBAR AS A HOPFIELD NETWORK

Let us illustrate this training strategy on a simple example of a recurrent CrossNet working as a Hopfield network. As we have shown before [14, 15], this operation may be performed efficiently not only by a network with global connectivity [19–22], but also by a network with quasi-local connectivity such as InBar (Figure 5(b)), provided that the connectivity parameter M is sufficiently high in comparison with number P of stored patterns. Moreover, the Hopfield function may be achieved with ternary synaptic weights set in accordance with the 'clipped Hebbian rule' [41]:

$$w_{jk} = w_{kj} = \operatorname{sgn} \sum_{p=1}^{P} \xi_j^{(p)} \xi_k^{(p)} \tag{2}$$

where $\xi_j^{(p)}$ is the jth pixel of the pth stored pattern. Figure 6 shows the procedure of importing the externally calculated synaptic weights (2) into a recurrent InBar. (Actually, shown is just one of $4MN^{1/2}$ steps of this process, where N is the total number of cells in the InBar array). At each step, a specific set of control and data voltages applied to CMOS wires[¶¶] by external tutor forces the somatic cells to apply:

(i) 'data' voltages $\pm V_0$ to all four axons connected to each somatic cell of one quasi-horizontal row of the InBar (shown red in Figure 6), and
(ii) mutually opposite, data-independent 'write enable' voltages $\pm V_0$ to two dendrites connected to each $2M^{1/2}$th cell of another row, separated by a vertical distance less than or equal to $M^{1/2}$ from the 'axonic' row. (Figure 6 shows, in blue, just one of these cells.)

The sign of axonic voltages is controlled by 'data' wires and follows rule (2). For example, if $w_{jk} > 0$ and the axonic voltage is positive, the net voltage V applied to the synapse connecting the activated axonic wire and the negatively activated vertical dendrite exceeds V_t, turning the latch on. Figure 4(b) shows that in the operation mode the selected cells are only becoming connected via one synapse jk_+, so the corresponding synaptic weight $w_{jk} = +1$ as required by Equation (2). In the case of the opposite sign of the data, only synapse jk_- is turned on, providing for $w_{jk} = -1$ in the operation mode. (One of synapses kj_+ and kj_- is

[¶¶] A somatic cell (reconfigured for the import mode) is activated by either 'semi-select' voltages applied to two control wires or by a (larger) 'full select' voltage applied to one of the wires.

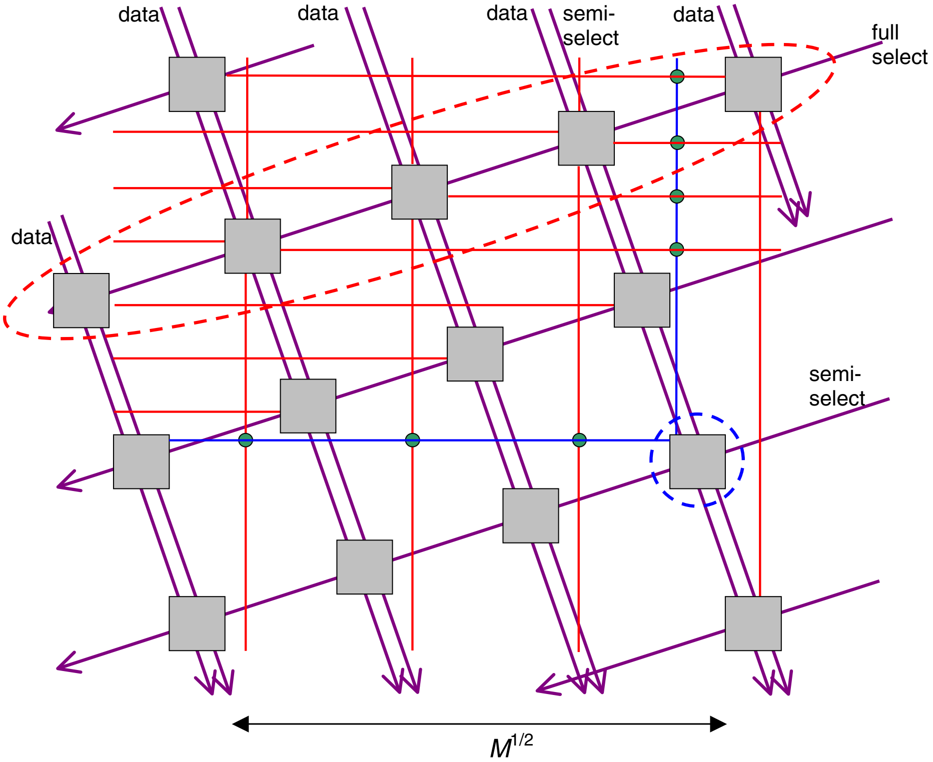

Figure 6. Teaching the recurrent InBar to operate as a Hopfield network. The somas selected for axon activation are marked by a dashed oval, while that with activated dendrites are in a dashed circle. Bold lines show CMOS-level wires carrying control and data signals from the external tutor. For clarity, the figure shows only the synapses being turned on at this particular weight import step, and only the nanowires activated at this step.

turned on when the control signals turn the 'red' soma into 'blue' one and vice versa.) Thus the import procedure allows the external system to set all synaptic weights to values (2).

As has been shown in References [14, 15], recurrent InBar with such weights operates as a Hopfield network with capacity

$$P_{\max} \approx (4/\pi)M/\mu^2(\varepsilon) \tag{3a}$$

where $\mu(\varepsilon)$ is the solution to the transcendent equation

$$2\varepsilon = 1 - \operatorname{erf}[\mu(\varepsilon)] \tag{3b}$$

erf is the error function, and ε is the average fraction of wrong pixels. (For a reasonable value $\varepsilon = 1\%$, $\mu(\varepsilon) \approx 1.64$ and $P_{\max} \approx 0.47M$. This capacity is by only $\sim$30% less than that of a global Hopfield network with $4M$ cells and continuous synaptic weights [19–22].)

We have checked this approximate analytical result by numerical simulation of recurrent InBars on usual computers (in particular, our supercomputer cluster *Njal*, see Reference [42]).

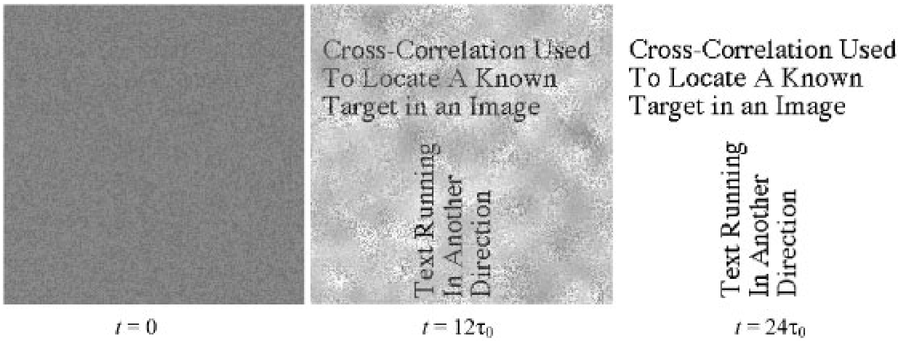

Figure 7. The process of recall of one of three trained black-and-white images by a recurrent InBar-type CrossNet with 256×256 neural cells, binary synapses, and connectivity $M=64$. The initial image (left panel) was obtained from the trained image (identical to the one shown in the right panel) by flipping 40% of randomly selected pixels.

At the modelling, the network evolution follows the usual fire-rate equations [19–22],

$$8MC_0\frac{\mathrm{d}U_k}{\mathrm{d}t}=\frac{V_\mathrm{s}}{R}\sum_{\pm}\sum_{\substack{j=-M\\(j\neq 0)}}^{M}(\pm w_{jk}^{(\pm)})V_j-\frac{U_k}{R_L},\quad V_j=f(GU_j/V_\mathrm{s})\tag{4}$$

describing electric recharging of two dendrites connected to the kth soma (Figure 2(b)) by currents through $4M$ synapses connected to these nanowires.[|||] Here C_0 is the capacitance of nanowires per elementary synapse (so that capacitance per synaptic plaquette is $4C_0$), $\pm V_\mathrm{s}$ are the saturation values of the axonic voltages V_j, and $f(x)$ is the normalized activation function describing this saturation:

$$f(x)=\begin{cases}x, & |x|\ll 1\\ \mathrm{sgn}(x), & |x|\gg 1\end{cases}\tag{5}$$

These calculations have confirmed the analytical result (3) and have also shown that the pattern restoration (‘image recognition’) is very fast. For example, Figure 7 shows the result of the restoration of one of three black-and-white images initially taught to an InBar with $M=64$. The original image was spoiled initially by flipping 40% of randomly selected pixels, and then given to the CrossNet as an initial condition. In this case, the final restoration is not only perfect, but also achieved in just a few characteristic time units $\tau_0\equiv MR_LC_0\lesssim R_0C_0$ of Equation (4). In a CMOL CrossNet with realistic parameters, τ_0 may be as low as a few nanoseconds—see Section 9 below.

We have also studied CrossNet defect tolerance of this operation mode, using both an (approximate) analytical theory and numerical modelling. For example, Figure 8 shows results for a 3744-neuron InBar with $M=25$. It is remarkable how resilient the network may be, if

[|||] Equations (4) are exact if resistances of nanowires, output resistances of somatic amplifiers, and the product MR_L are all much lower than open-state resistance R_0 of the synaptic nanodevice. This assumption may be readily satisfied in practical CMOL circuits.

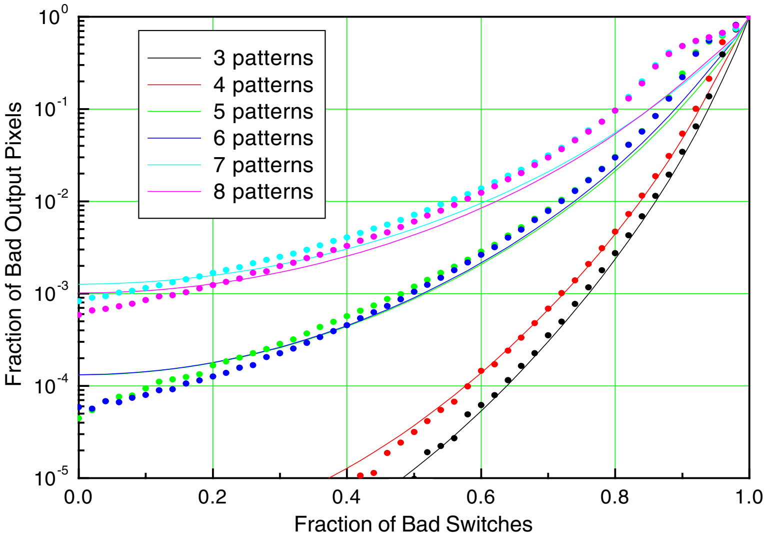

Figure 8. Defect tolerance of a recurrent InBar with connectivity parameter $M = 25$, operating in the Hopfield mode. Lines show the results of an approximate analytical theory, while dots those of a numerical experiment.

the number of stored patterns P is not too close to $P_{\max} \approx 8$. For example, for P as high as 4 (i.e. close to one half of the network capacity), the network functioned very reasonably (with 99% fidelity) even in the case when approximately 85% (!) of randomly selected synaptic switches had been disconnected. The defect level of this order may be quite expected at the initial stage of CMOL circuit development, so that Hopfield CrossNets may be an interesting test application of this emerging technology.***

5. FEEDFORWARD FLOSSBAR AS A MULTILAYER PERCEPTRON

It is well known that practical application of Hopfield networks is rather limited. Many more applications (most notably, pattern classification) have been demonstrated for perceptrons with one or more hidden layers [19–22, 38]. Some CrossNet species, e.g. feedforward FlossBars (Figure 5(a)) are directly suitable for the use as layered perceptrons. Unfortunately, the information loss at synapse clipping may affect the performance of such networks as pattern classifiers more seriously than for the Hopfield networks. For example, Figure 9 shows the results of our calculations of the average error of 'simple' perceptrons (with no hidden layers), as well as multilayer perceptrons with one–three hidden layers, induced by synapse 'clipping', i.e. rounding of the initially continuous weight to the closest of L equally spaced quantization levels. The error has been calculated by considering a perceptron with randomly generated

*** Suggested memory applications of CMOL chips are substantially less defect tolerant, requiring less than 1% of bad nanodevices [13].

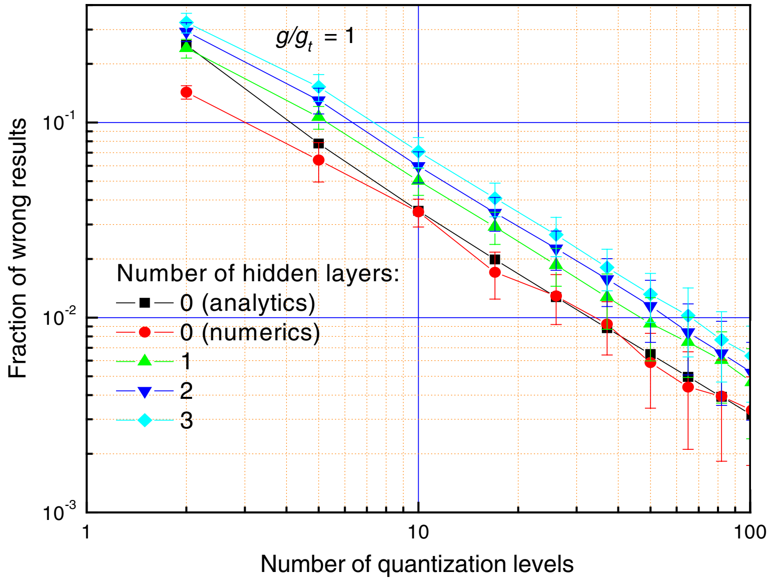

Figure 9. Output error of few-layer perceptrons, with $M = 100$ neurons on each layer, induced by synaptic weight rounding to L discrete values, as a function of number of quantization levels L. Straight line shows results of an approximate analytical theory for the simple perceptron (with no hidden layers). These results are only valid if the effective somatic gain $g \equiv GR_L/R_0$ is close to its threshold value $g_t = M^{-1/2}$ corresponding to signal propagation from input to output without attenuation. (The numerical results are for $g/g_t = 1$.)

continuous weights as perfect, then clipping them, and calculating the output difference.[†††] One can see that for the original CrossNets (Figure 4) with ternary synapses ($L = 3$) the error may be above 20%, unacceptable for most applications. At the same time, an increase of L to a modest value (say, ∼30) reduced the clipping-induced errors to 1–2%, that is almost negligible in comparison with typical errors of existing pattern classifiers [19–22, 38].

Such multi-valued synapses, with $L = 2n^2 + 1$ where n is an integer, may be readily implemented by replacing each latching switch (Figure 3) with a square array of $n \times n$ such switches (Figure 10). In the operation mode, all n axonic wires are fed with the same voltage, while the resulting currents flowing into n dendritic wires are just summed up at the somatic load resistance R_L. As a result, the net output (post-synaptic) signal from two arrays (Figure 4(a)) is proportional to $w = (l_+ - l_-)/n^2$, where $l_\pm$ are the numbers of latches turned on in each array $(0 \leqslant l_\pm \leqslant n^2)$.

In order to fix the desirable value of $l_\pm$ in each array during the weight import mode, both axonic and dendritic are fed with graded voltages:

$$V_{\mathrm{a}}^{(i)} = V_{\mathrm{w}} + A(i/n - 1/2), \quad V_d^{(i')} = \pm [V_{\mathrm{t}} + A(i'/n - 1/2)] \tag{6}$$

[†††] This procedure ignores the opportunity to retrain the clipped-weight network, that still has to be explored.

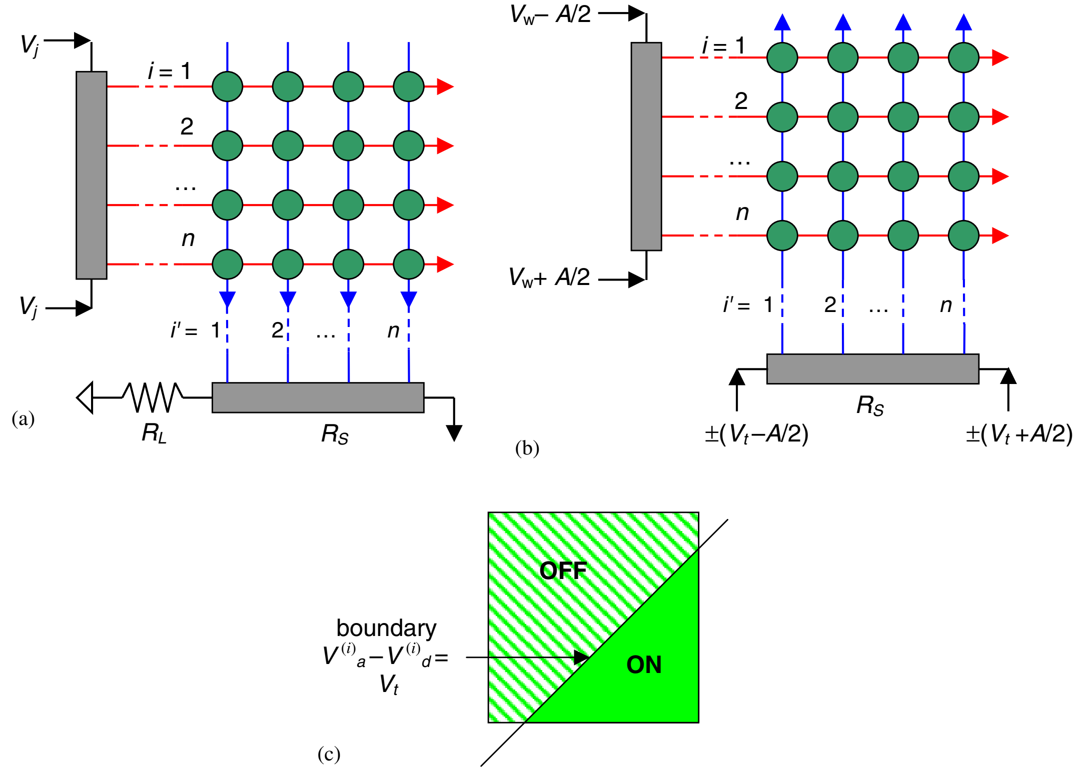

Figure 10. A half of the composite synapse for providing $L = 2n^2 + 1$ discrete levels of the weight in (a) operation and (b) weight import modes. The dark-grey rectangles are resistive metallic strips (with the total resistance $R_S \div R_L$) serving as soma/nanowire interfaces. Plate (c) shows (schematically) the boundary between the domains of two possible states of elementary synapses.

where i $(1 \leqslant i \leqslant n)$ is the nanowire number, the voltage spread A is slightly lower than V_t, and sign of V_d is, as before, opposite for horizontal and vertical dendrites.[‡‡‡] This creates a gradient of the net voltage $V^{(i)} \equiv V_a^{(i)} - V_d^{(i)}$ applied to each switch, and hence a domain of switches being turned on (Figure 10(c)). The boundary of this domain, defined by the equation $V^{(i)} = V_t$, and thus the total number l of latches turned on, depends on the average axonic voltage $V_w(|V_w| < V_t)$, which carries information about the desired (now continuous) synaptic weight.

Figure 11 shows that the procedure of synaptic weight import into a feedforward FlossBar is even simpler than that for the InBar recurrent network (Figure 6). At each time step, the external 'select' signals activate:

(i) both dendrites of each Mth cell of one row of somas, and
(ii) both axons of each cell of the previous row.

As a result, importing all synaptic weights of one layer takes M steps.

[‡‡‡] The necessary voltage gradients may be readily generated, e.g. by passing current along simple resistive strips (marked as R_s in Figure 10) serving as contacts for axonic and dendritic nanowires.

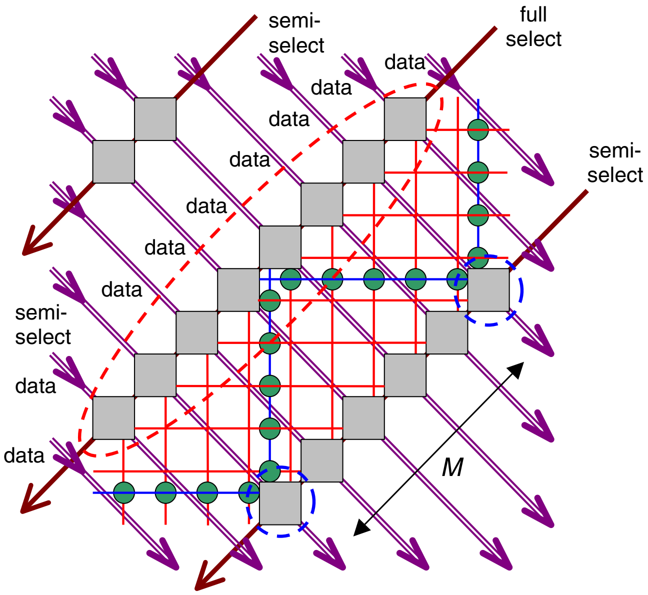

Figure 11. Importing synaptic weights into the feedforward FlossBar (for $M=4$). The notation is similar to that in Figure 6.

6. FEEDFORWARD INBAR AS AN INTERLEAVED PERCEPTRON

Generally, InBar structure is preferable to FlossBar for the CMOS subsystem implementation, because all CMOS somatic cells may have nearly square shape. In this context, we have carried out a preliminary study of feedforward InBars (Figure 5(b)) with quasi-continuous (L-level) synaptic weights as perceptrons (Figure 12).[§§§] For this, we have first generated a set of 'teacher' perceptrons (either a one-hidden-layer perceptron or an InBar) with the input and output vectors of the same size as the InBar under study, and random synaptic weights (distributed uniformly within the range $-1<w_{jk}<+1$). Then, each teacher has been repeatedly used to generate the model output from a random vector of binary input signals. This set of related input and output vector pairs has been separated into a training set and a test set. The former set had been used for teaching of continuous-weight 'student' InBars (each to be used later as the precursor for a CMOL InBar) by the standard error backpropagation method; after that the test set has been used for the evaluation of prediction ability of the student networks, i.e. their quality as pattern classifiers. As a reference, we have also applied it to student perceptrons with one hidden layer and the same number of input, hidden, and output cells as the InBars under study.

[§§§] Note that these networks cannot be reduced to the usual multilayered perceptron: each cell of a feedforward InBar (say, cell 1 in Figure 12) sends its output signal to M cells within an square-shaped area, including cells 2 and 3. Cell 2, in turn, sends signals to cells within a similar area that includes, in particular, cell 3. Hence, InBar cannot be partitioned to layers, and we prefer to call it an 'interleaved' network.

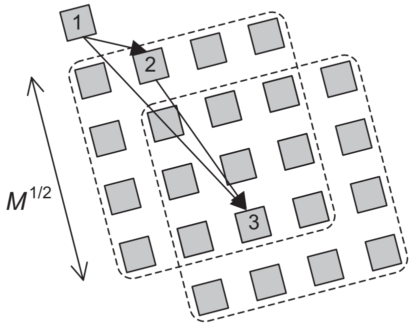

Figure 12. Feedforward InBar as an interleaved perceptron. (For explanation, see the text.)

Figure 13 shows a typical result of such calculation for perceptrons with 36 input cells, 36 hidden cells, and 6 output cells (whose binary output vector belongs to one of $2^6 = 64$ classes). The results show that InBars may be trained to operate as pattern classifiers reasonably well.¶¶¶ Thus, the student InBars may be used as precursors for a hardware (e.g. CMOL) InBar operating with a reasonable accuracy.

7. DIRECT TRAINING OF CROSSNETS BY ERROR BACKPROPAGATION

To summarize the three last sections, the synaptic weight import procedure allows CrossNets to perform, with very small loss of fidelity, the functions of Hopfield networks (pattern recognition, i.e. associative memory), multilayer perceptrons (e.g. pattern classification), and very probably any other function demonstrated for any fire-rate-model neural network (either feedforward or recurrent), provided that the synaptic weights have been calculated externally. For some cases, for example the Hopfield network, such calculation does not present much problem—see Equation (2). However, in some cases (e.g. classification problems) the weight calculation may only be performed by training a homomorphic precursor network with continuous weights. Since we are speaking about very large networks (see Section 9 below), such training may take very long time if performed on the usual sequential computers. For some applications with limited input vector size (say, handwritten character recognition), the training period duration may be quite acceptable. However, in other cases (e.g. recognition of large-size patterns, such as detailed optical images) the precursor network training may require impracticable computer resources.

One possible solution of this problem is direct training of CrossNets with multi-level (quasi-continuous) synapses (Figure 10) by error backpropagation. We have developed a method for such training, that requires doubling the number of nanowires and synapses connecting each

¶¶¶ The fact that layered perceptrons may be trained substantially better (see black square points in Figure 13), on a subset of tasks (namely, those generated by similar perceptrons) does not mean that their general prediction ability is better than that of the InBars. For example, on problems generated by InBar teachers, layered perceptrons perform not better than InBars (green triangles). At this moment it is not clear how would these networks compare for real-life classification problems; we plan to address this issue in near future.

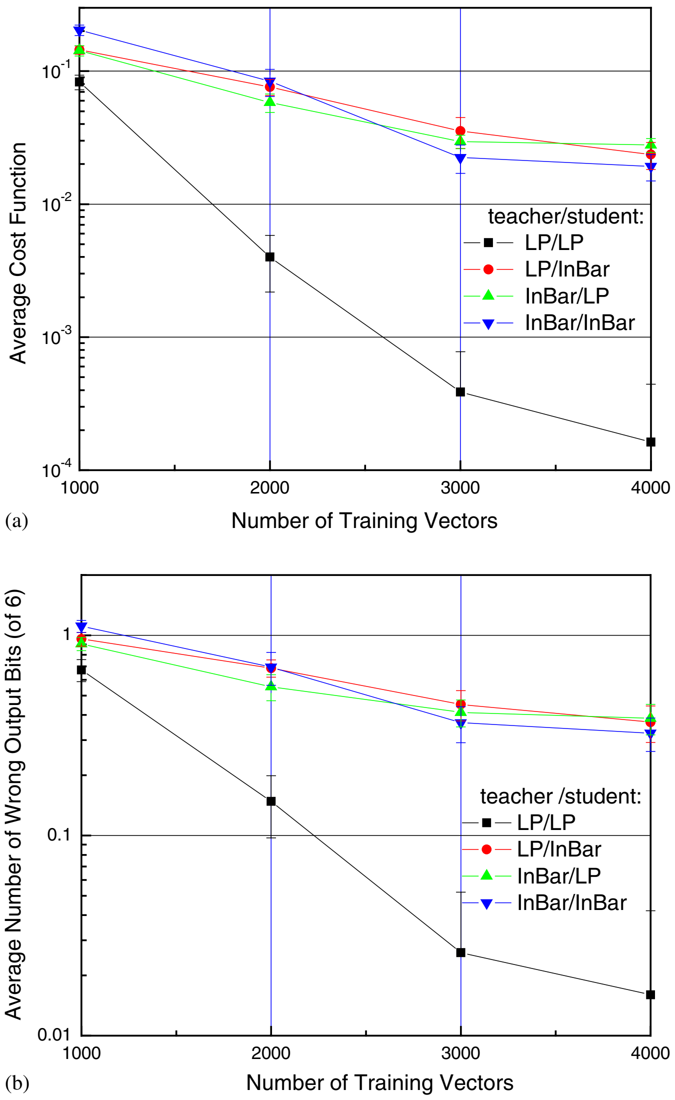

Figure 13. Results of training of feedforward InBars with $M = 36$, and layered perceptron (LP) networks with one hidden layer (each with 36 input cells, 36 hidden cells and 6 output cells) using the error backpropagation, after 2000 iterations ('epochs'). For each teacher/student combination, the results are averaged over 3 teachers, 5 students, and 100 test vectors. Before training, the average cost function is close to 1, while the average number of wrong bits is 3. For networks of both types, the normalized somatic cell gain g/g_t is close to 2.

pair of somas (Figure 14). In such 'dual-rail' networks, each axonic and dendritic signal is carried by two wires, with opposite polarity. The reason for this hardware doubling is that it naturally forms four-synapse groups (Figure 14(b)) that have Hebbian properties. Indeed, using Equation (1) for switching rates, it is straightforward to show that the change of the

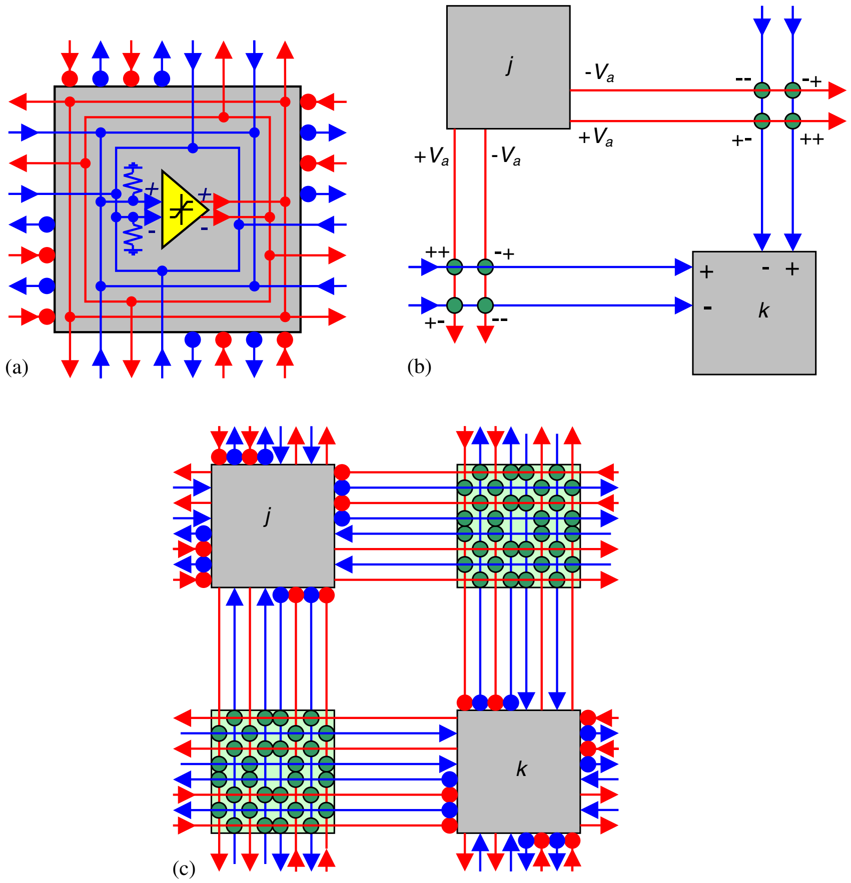

Figure 14. CrossNet with the double number of synapses: (a) soma schematics in the operation mode (cf. Figure 2); (b) cell coupling (shown for one way only—cf. Figure 4(a)); and (c) the same picture as in (b), but showing all the wires connected to each of the two cells. Each crosspoint circle denotes either a single nanodevice or a multi-device array (Figure 10).

average net synaptic weight $w = w_{++} + w_{--} - w_{+-} - w_{-+}$ of the four-synapse group obeys equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\langle w\rangle = -4\Gamma_0 \sinh(\beta S)\sinh(\beta V_\mathrm{a})\sinh(\beta V_\mathrm{d}) \tag{7}$$

provided that $\langle w\rangle \ll 1$. At relatively small axonic and dendritic voltages, and a negative shift S, this equation corresponds to the classical Hebb rule [19–22], $\mathrm{d}\langle w_{jk}\rangle/\mathrm{d}t \propto V_j \times V_k$, if the dendritic voltage of the kth (postsynaptic) cell is made, at least temporarily, relatively large and proportional to its output (axonic) voltage.[‖‖‖]

[‖‖‖] The non-linearity of Equation (7) at larger signal values may actually be beneficial for the Hebb rule enforcement.

This property allows one to implement the network training process by its time-multiplexing into the following three steps that are repeated periodically:

(i) At the first, 'operation' stage, the somas are configured as shown in Figure 14(a), and the network provides the usual feedforward signal propagation described by Equation (4). Since in this mode the dendritic voltages are low, rate (7) of synaptic weight change is negligible, at any moderate value of shift S.

(ii) At the second, 'error-backpropagation' stage, each somatic amplifier is reconfigured to provide linear amplification of the error signal with voltage gain proportional to $f'(U_{\rm f})$, where $U_{\rm f}$ is the final value of the dendritic voltage at the first step. In contrast to the first stage, the amplifier is now fed with axonic voltage and applies its output signal to dendritic nanowires. The outputs of the network (now essentially turned into its input) are fed by an error signals proportional to $(V_{\rm out} - V_{\rm tar})$, where $V_{\rm out}$ are output signals of the network at the first stage, while $V_{\rm tar}$ are the components of target value of the output vector. As a result, error signals ε are propagated back through the network in accordance with the well-known rule of error backpropagation [19–22]. Since at this stage the *axonic* voltages are low, and synapses still do not switch.

(iii) With the developed error signals still applied to dendrite nanowires, axonic nanowires are now fed, for a short time interval Δt, with voltages $V_{\rm f} = f(GU_{\rm f}/V_{\rm s})$. (This requires a storage of signals $U_{\rm f}$ during the second step of the process; this temporary storage may be achieved by using just one additional capacitor per soma.) At this 'weight adjustment' stage, the amplitude of both axonic and dendritic voltages may be considerable, leading to much higher probability of synapse switching. In accordance with Equation (7), at $S<0$ the average change rate of synaptic weight w_{jk} is proportional to $V_j\varepsilon_k\Delta t$, thus implementing the backpropagation method algorithm.

8. GLOBAL REINFORCEMENT

A possible alternative way to train a CrossBar without the external precursor was suggested by our group earlier [14, 15] for three-terminal devices. Here we describe the modification of this method that would work for CrossNets with more realistic two-terminal devices [17], plus some novel results.

The initial idea of this approach has been based on the fact of chaotic excitation of recurrent CrossBars with differential dendritic signals (Figures 2, 4 and 14), at sufficiently large effective gain of somatic cells ($g > g_{\rm t} \propto M^{-1/2}$) [14, 15]. One may say that in this regime the system walks randomly though the multi-dimensional phase space of all possible values of V_j. Now, let input signals be inserted into some of the cells, and outputs picked up from a smaller subset of cells. The system is allowed to evolve freely, but this evolution is periodically interrupted, for brief time intervals Δt, by the application of somatic output voltages V_k of each cell back to its input dendritic wires. Simultaneously, the tutor applies to all synapses a global shift S corresponding to its satisfaction with the system output at this particular instant: $S<0$ if the network output is correct and $S>0$ if it is not. This operation results is a small change of average synaptic weights $\langle w_{jk}\rangle$, that is described by Equation (7) with the replacement of $V_{\rm a}$ for V_j and $V_{\rm d}$ for V_k, thus implementing the Hebb rule if the system output is correct, and the anti-Hebb rule if it is incorrect. It had been our hope that the repeated application of this

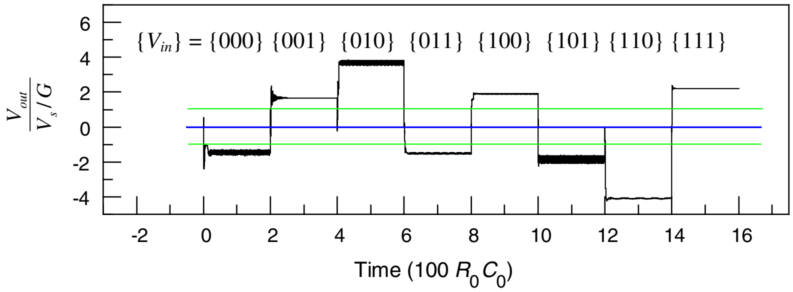

Figure 15. A preliminary result of reinforcement training: the input signal of the output cell of a recurrent InBar with quasi-continuous doubled synapses (Figure 14) trained to calculate parity of three binary inputs. (Any output signal value above the top horizontal line means binary unity, while that below the bottom line is binary zero). System parameters: $N=612$, $M=16$, $n=10$, $R_L/(R/Mn^2)=0.1$, $\Gamma_0/R_0C_0=5\times10^{-6}$, $g=1$, $V_s/T=10$, $S_{\max}/T=\ln 100\approx0.46$, $\Delta t/R_0C_0=10^{-3}$.

procedure would increase the probability of the system's eventual return to the 'good' regions of the phase space, possibly with the eventual quenching of the chaotic dynamics.

By the moment of submission of this paper, we have just started numerical experiments with this procedure, and obtained only initial results, some of which have turned out to be rather unexpected. Namely, recurrent InBar CrossNets exhibit some learning ability (Figure 15) when external signals are so strong that they suppress the chaotic dynamics (due to the activation function non-linearity) even before the training has been started! However, the trained state of the system is not quite stable: if the training procedure is not stopped, the network may walk away from the trained state. We are currently working on the interpretation of these observations, and are experimenting with the initially suggested chaotic regime.

9. DISCUSSION

Our results so far may be summarized as follows. There are several ways, including the import of externally calculated synaptic weight values (see Sections 3–5 above), error backpropagation (Section 6) and probably also global reinforcement (Section 7), that enable CrossNet circuits to perform most (all?) information processing functions that had been demonstrated with artificial neural networks. The significance of this result is that the CMOL implementation may allow CrossNets to have much higher performance that their biological prototypes and artificial predecessors.

Indeed, let us estimate possible CrossNet parameters.[****] The most fundamental limitation on the half-pitch F_{nano} (Figure 1) comes from quantum-mechanical tunneling between nanowires. If the wires are separated by vacuum, the corresponding specific leakage conductance becomes uncomfortably large ($\sim10^{-12}\Omega^{-1}\text{m}^{-1}$) only at $F_{\text{nano}}=1.5$nm; however, since realistic insulation materials (SiO_2, etc.) provide somewhat lower tunnel barriers,

[****] These are slightly more realistic estimates than those published previously [14, 15].

let us use a more conservative value $F_{\text{nano}} = 3$ nm.[††††] With the typical specific capacitance of 3×10^{-10} F/m = 0.3 aF/nm [13], this gives nanowire capacitance $C_0 \approx 1$ aF per working elementary synapse, because the corresponding segment has length $4F_{\text{nano}}$ (see Figures 4(c), 14(c)). The CrossNet operation speed is determined by the time constant τ_0 of dendrite nanowire capacitance recharging through resistances of open nanodevices. (Since both the relevant conductance and capacitance increase similarly with M and n,

$$\tau_0 \approx R_0 C_0 \tag{8}$$

Small load resistances $R_L \lesssim R_0/M$ decrease τ_0, however, this speed-up cannot be over-exploited because of noise immunity concerns—see below.) For example, the time of image recovery shown in Figure 7 is approximately $20\tau_0$, the time of pattern classification shown in Figure 15 is close to one τ_0, etc.

The possibilities of reduction of R_0, and hence τ_0, are limited mostly by acceptable power dissipation per unit area, that is close to $V_s^2/(2F_{\text{nano}})^2R_0$. For room-temperature operation, the voltage scale $V_0 \approx V_t$ should be of the order of at least 30 $k_BT/e \approx 1$ V to avoid thermally induced errors [3, 6]. With our number for F_{nano}, and a relatively high but acceptable power consumption of 100 W/cm^2, we get $R_0 \approx 10^{10}\ \Omega$ (which is a very realistic value for single-molecule single-electron devices like one shown in Figure 3).[‡‡‡‡] With this number, τ_0 is as small as $\sim$10 ns. This means that the CrossNet speed may be approximately six orders of magnitude higher than that of the cerebral cortex circuitry![30,31] Even scaling R_0 up by a factor of 100 to bring power consumption to a more comfortable level of 1 W/cm^2, would still leave us at least a four-orders-of-magnitude speed advantage.

These estimates make us believe that even relatively small CrossNet chips may revolutionize the pattern classification field. For example, assuming that the number of layers of FlossBar (or 'quasi-layers' of InBar) necessary for efficient image classification scales as N/M,[§§§§] a 1-cm^2 CMOL CrossNet chip with $N \approx 10^7$, $M = 10^2$ could classify high-resolution (a-few-megapixel) optical images in $\sim 10^5\tau_0 \approx 1$ ms. Such chips may be very important, for example, for security systems, fabrication quality control, etc.

If the hopes for mass production of CMOL chips materialize, more ambitious goals might be pursued. Imagine a cerebral-cortex-scale CrossNet-based system with $\sim$10^{10} neurons and 10^{15} synapses. With the parameters cited above, it would require an approximately 30×30 cm^2 silicon substrate[¶¶¶¶] and, at power consumption of the order of 1 W/cm^2 (provided by making R_0 of the order of $10^{12}\ \Omega$), operate at least ten thousand times faster than its biological

[††††] Note that this value corresponds to 10^{12} elementary synapses per cm^2, so that for $4M = 10^4$ and $n = 4$ the areal density of neural cells is close to 2×10^7 cm^{-2}. Both numbers are higher than those for the human cerebral cortex [30, 31], despite the fact that the quasi-2D CMOL circuits have to compete with quasi-3D cerebral cortex.

[‡‡‡‡] The large value of R_0, and hence the smallness of current $I \sim V_s/R_0 \sim 10^{-10}$ A through each elementary synapse, may make one worry about the noise immunity of the CrossNet operation. A simple analysis shows that the main contribution to the current fluctuations is provided by shot noise of the nanodevices, that may be estimated as $I_N \sim (eI/\pi\tau_0)^{1/2}$. Assuming that the noise does not affect CrossNets properties at $I_N \ll I$ (this hypothesis still needs to be checked), we get an R_0-independent condition $V_s \gg e/C_0$. For the values of V_s and C_0 listed above, this condition is well satisfied.

[§§§§] This scaling still has to be checked.

[¶¶¶¶] In digital electronics, such a 'superchip' would be impracticable because of vanishing fabrication yield; however, large defect tolerance of neuromorphic networks (see, e.g. Figure 8 and its discussion above) makes this option much more plausible.

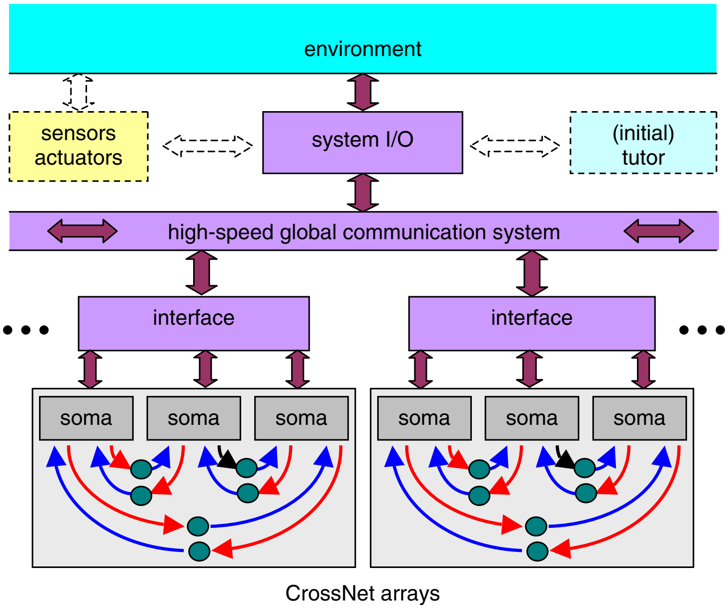

Figure 16. General structure of a possible hierarchical neuromorphic system based on CrossNet arrays and high-speed global communication system.

prototype ($\tau_0 \sim 1$ μs). Such large-scale system would of course require a hierarchical organization involving, at least, the means of fast signal transfer over long distances. Fortunately, for the InBar-type CrossNet with its regular location of somatic cell interfaces (Figure 5(b)), such communication subsystem is easy to organize (Figures 16 and 17).[‖‖‖]

Unfortunately, neurobiology is still very far from teaching us how to train such system for performing higher intellectual functions such as self-awareness (consciousness) and reasoning. However, one may hope that, after a period of initial training by a dedicated external tutor, the system would be able to learn directly from its interaction with the environment (Figure 16). Such self-development could repeat, at much higher speed, the natural evolution of the human cerebral cortex, and may even extend it. Any success along these lines would have a strong impact not only on information technology, but also on society as a whole.

However, it is necessary to emphasize that even in the best case the development of neuromorphic CMOL circuits and systems will require a very substantial effort. Hardware-wise, the most challenging problem is the development of nanodevice fabrication techniques with substantial yield. From the architecture standpoint, the most urgent problem we face is the verification and improvement of the global reinforcement training method (Section 8), including the effects of chaos and noise (see Footnote[‡‡‡‡] above) on this procedure.

[‖‖‖] In order to minimize communication distances in 2D CrossNets, such communication system may have, for example, the so-called X layout (Figure 17(a)) that immediately yields system geometries reminiscent of the brain (Figure 17(b)).

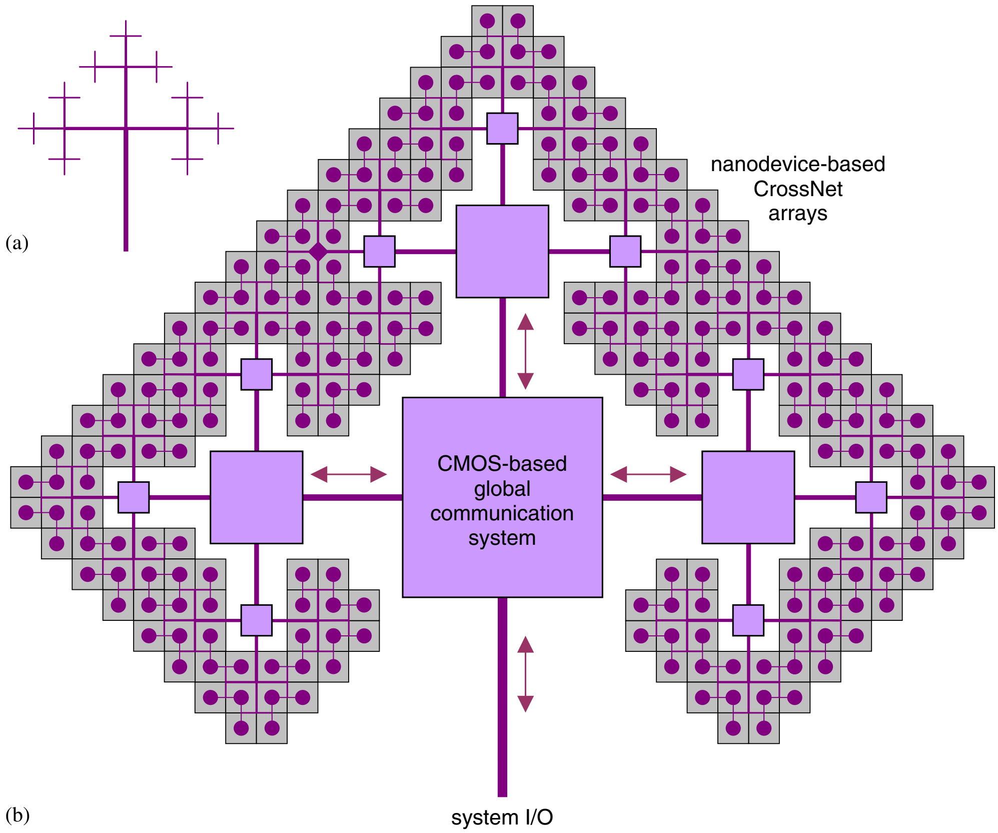

Figure 17. X layout for the global communication network: (a) general structure and (b) possible 2D geometry of a hierarchical neuromorphic system (Figure 16) using such layout.

ACKNOWLEDGEMENTS

The molecular design shown in Figure 3(b) belongs to Prof. Andreas Mayr (SBU/Chemistry). Useful discussions with P. Adams, J. Barhen, S. M. Sherman and V. Protopopescu are gratefully acknowledged.

REFERENCES

1. *International Technology Roadmap for Semiconductors* (2003 edn). Available online at http//public.itrs.net/.
2. Frank DJ *et al.* Device scaling limits of Si MOSFETs and their application dependencies. *Proceedings of IEEE* 2001; **89**(3):259–288.
3. Likharev KK. Electronics below 10 nm. In *Nano and Giga Challenges in Microelectronics*, Greer J *et al.* (ed.). Elsevier: Amsterdam, 2003; 27–68.
4. Capasso F (ed.). *Physics of Quantum Electron Devices*. Springer: Berlin, 1990.
5. Likharev KK. Single-electron devices and their applications. *Proceedings of IEEE* 1999; **87**(4):606–632.
6. Likharev KK. CMOL: a new concept for nanoelectronics. Invited talk at the *12th International Symposium on Nanostructures Physics and Technology*, St. Petersburg, Russia, June 2003. Available online at http://rsfq1.physics.sunysb.edu/~likharev/nano/SPb.pdf.
7. Stan MR, Franzon PD, Goldstein SC, Lach JC, Ziegler MM. Molecular electronics: from devices and interconnects to circuits and architecture. *Proceedings of IEEE* 2003; **91**(11):1940–1957.
8. Chen Y *et al.* Nanoscale molecular-switch crossbar circuits. *Nanotechnology* 2003; **14**(4):462–468.

9. Zankovych S *et al*. Nanoimprint lithography: challenges and prospects. *Nanotechnology* 2001; **12**(2):91–95.
10. Brueck SRJ *et al*. There are no limits to optical lithography. In *International Trends in Optics*. Guenther A. (ed.). SPIE Press: Bellingham, WA, 2002; 85–109.
11. Fendler JH. Chemical self-assembly for electronics applications. *Chemistry of Materials* 2001; **13**(10): 3196–3210.
12. Tour J. *Molecular Electronics*. World Scientific: Singapore, 2003.
13. Strukov DB, Likharev KK. Prospects for nanoelectronics memories. *Nanotechnology* 2004, submitted.
14. Türel Ö, Likharev KK. CrossNets: possible neuromorphic networks based on nanoscale components. *International Journal of Circuit Theory and Applications* 2003; **31**(1):37–53.
15. Likharev K, Mayr A, Muckra I, Türel Ö. CrossNets: high-performance neuromorphic architectures for CMOL circuits. *Annals of the New York Academy of Sciences* 2003; **1006**:146–163.
16. Türel Ö, Muckra I, Likharev K. Possible nanoelectronic implementation of α neuromorphic networks. In *Proceedings of the 2003 International Joint Conference on Neural Networks*. International Neural Network Society: Mount Royal, NY, 2003; 365–370.
17. Türel Ö, Lee JH, Ma X, Likharev KK. Nanoelectronic neuromorphic networks (CrossNets): new results. Invited as a plenary talk at the *2004 International Joint Conference on Neural Networks*, Budapest, Hungary, July 2004; 389–394.
18. Nikolic K, Sadek A, Forshaw M. Fault-tolerant techniques for nanocomputers. *Nanotechnology* 2002; **13**(3): 357–362.
19. Hertz J, Krogh A, Palmer RG. *Introduction to the Theory of Neural Computation*. Perseus: Cambridge, MA, 1991.
20. Fausett L. *Fundamentals of Neural Networks*. Prentice-Hall: Upper Saddle River, NJ, 1994.
21. Hassoun MH. *Fundamentals of Artificial Neural Networks*. MIT Press: Cambridge, MA, 1995.
22. Haykin S. *Neural Networks*. Prentice-Hall: Upper Saddle River, NJ, 1999.
23. Park H *et al*. Nanomechanical oscillations in a single-C_{60} transistor. *Nature* 2000; **407**(6800):57–60.
24. Gubin SP *et al*. Molecular clusters as building blocks for nanoelectronics: the first demonstration of a cluster single-electron tunneling transistor at room temperature. *Nanotechnology* 2002; **13**(2):185–194.
25. Zhitenev NB, Meng H, Bao Z. Conductance of small molecular junctions. *Physical Review Letters* 2002; **88**(22):226801:1–4.
26. Park J *et al*. Coulomb blockade and the Kondo effect in single-atom transistors. *Nature* 2002; **417**(6890): 722–725.
27. Kubatkin S *et al*. Single-electron transistor of a single organic molecule with access to several redox states. *Nature* 2003; **425**(6959):698–701.
28. Fölling S, Türel Ö, Likharev KK. Single-electron latching switches as nanoscale synapses. In *Proceedings of the 2001 International Joint Conference on Neural Networks.* International Neural Network Society: Mount Royal, NY, 2001; 216–221.
29. Heath JR, Kuekes PK, Snider GS, Williams RS. A defect-tolerant computer architecture: opportunities for nanotechnology. *Science* 1998; **280**(5370):1716–1721.
30. Purves D *et al. Neuroscience*. Sinauer: Sunderland, MA, 1997.
31. Braitenberg V, Schüz A. *Cortex Statistics and Geometry of Neuronal Connectivity* (2nd edn). Springer: Berlin, 1998.
32. Türel Ö, Muckra I, Likharev KK. Statistics of distributed crossbar networks. Paper in preparation.
33. Albert R, Barabási A. Statistical mechanics of complex networks. *Reviews of Modern Physics* 2002; **74**(1): 47–97.
34. Mead C. *Analog VLSI and Neural Systems*. Addison-Wesley: Reading, MA, 1989.
35. Hammerstrom D, Rehfuss S. Neurocomputing hardware—present and future. *Artificial Intelligence Review* 1993; **7**(5):285–300.
36. Zaghloul ME, Meador JL, Newcomb RW. *Silicon Implementation of Pulse Coded Neural Networks*. Kluwer: Boston, 1994.
37. Chua LO. *CNN*: *A Paradigm for Complexity*. World Scientific: Singapore, 1998.
38. Bishop CM. *Neural Networks for Pattern Recognition*. Oxford University Press: Oxford, UK, 1995.
39. Chen CH (ed.). *Fuzzy Logic and Neural Network Handbook*. McGraw-Hill: New York, 1996.
40. Prince B. *Semiconductor Memories* (2nd edn). Wiley: Chichester, UK, 1991.
41. van Hemmen JL, Kühn R. Nonlinear neural networks. *Physical Review Letters* 1986; **57**(7):913–916.
42. Web site http://njal.physics.sunysb.edu/