Neuromorphic Electronic Systems

CARVER MEAD

Invited Paper

Biological information-processing systems operate on completely different principles from those with which most engineers are familiar. For many problems, particularly those in which the input data are ill-conditioned and the computation can be specified in a relative manner, biological solutions are many orders of magnitude more effective than those we have been able to implement using digital methods. This advantage can be attributed principally to the use of elementary physical phenomena as computational primitives, and to the representation of information by the relative values of analog signals, rather than by the absolute values of digital signals. This approach requires adaptive techniques to mitigate the effects of component differences. This kind of adaptation leads naturally to systems that learn about their environment. Large-scale adaptive analog systems are more robust to component degredation and failure than are more conventional systems, and they use far less power. For this reason, adaptive analog technology can be expected to utilize the full potential of wafer-scale silicon fabrication.

Two Technologies

Historically, the cost of computation has been directly related to the energy used in that computation. Today's electronic wristwatch does far more computation than the Eniac did when it was built. It is not the computation itself that costs—it is the energy consumed, and the system overhead required to supply that energy and to get rid of the heat: the boxes, the connectors, the circuit boards, the power supply, the fans, all of the superstructure that makes the system work. As the technology has evolved, it has always moved in the direction of lower energy per unit computation. That trend took us from vacuum tubes to transisitors, and from transistors to integrated circuits. It was the force behind the transition from n-MOS to CMOS technology that happened less than ten years ago. Today, it still is pushing us down to submicron sizes in semiconductor technology.

So it pays to look at just how much capability the nervous system has in computation. There is a myth that the nervous system is slow, is built out of slimy stuff, uses ions instead of electrons, and is therefore ineffective. When the Whirlwind computer was first built back at M.I.T., they made a movie about it, which was called "Faster than Thought." The Whirwind did less computation than your wristwatch does. We have evolved by a factor of about 10 million in the cost of computation since the Whirlwind. Yet we still cannot begin to do the simplest computations that can be done by the brains of insects, let alone handle the tasks routinely performed by the brains of humans. So we have finally come to the point where we can see what is difficult and what is easy. Multiplying numbers to balance a bank account is not that difficult. What is difficult is processing the poorly conditioned sensory information that comes in through the lens of an eye or through the eardrum.

A typical microprocessor does about 10 million operations/s, and uses about 1 W. In round numbers, it cost us about 10^{-7} J to do one operation, the way we do it today, on a single chip. If we go off the chip to the box level, a whole computer uses about 10^{-5} J/operation. A whole computer is thus about two orders of magnitude less efficient than is a single chip.

Back in the late 1960's we analyzed what would limit the electronic device technology as we know it; those calculations have held up quite well to the present [1]. The standard integrated-circuit fabrication processes available today allow us to build transistors that have minimum dimensions of about 1 μ (10^{-6} m). By ten years from now, we will have reduced these dimensions by another factor of 10, and we will be getting close to the fundamental physical limits: if we make the devices any smaller, they will stop working. It is conceiveable that a whole new class of devices will be invented—devices that are not subject to the same limitations. But certainly the ones we have thought of up to now—including the superconducting ones—will not make our circuits more than about two orders of magnitude more dense than those we have today. The factor of 100 in density translates rather directly into a similar factor in computation efficiency. So the ultimate silicon technology that we can envision today will dissipate on the order of 10^{-9} J of energy for each operation at the single chip level, and will consume a factor of 100–1000 more energy at the box level.

We can compare these numbers to the energy requirements of computing in the brain. There are about 10^{16} synapases in the brain. A nerve pulse arrives at each synapse about ten times/s, on average. So in rough numbers, the brain accomplishes 10^{16} complex operations/s. The power dissipation of the brain is a few watts, so each operation costs only 10^{6} J. The brain is a factor of 1 billion more efficient than our present digital technology, and a factor of

Manuscript received February 1, 1990; revised March 23, 1990.
The author is with the Department of Computer Science, California Institute of Technology, Pasadena, CA 91125.
IEEE Log Number 9039181.

0018-9219/90/1000-1629$01.00 © 1990 IEEE

10 million more efficient than the best digital technology that we can imagine.

From the first integrated circuit in 1959 until today, the cost of computation has improved by a factor about 1 million. We can count on an additional factor of 100 before fundamental limitations are encountered. At that point, a state-of-the-art digital system will still require 10 MW to process information at the rate that it is processed by a single human brain. The unavoidable conclusion, which I reached about ten years ago, is that we have something fundamental to learn from the brain about a new and much more effective form of computation. Even the simplest brains of the simplest animals are awesome computational instruments. They do computations we do not know how to do, in ways we do not understand.

We might think that this big disparity in the effectiveness of computation has to do with the fact that, down at the device level, the nerve membrane is actually working with single molecules. Perhaps manipulating single molecules is fundamentally more efficient than is using the continuum physics with which we build transistors. If that conjecture were true, we would have no hope that our silicon technology would ever compete with the nervous system. In fact, however, the conjecture is false. Nerve membranes use *populations* of channels, rather than individual channels, to change their conductances, in much the same way that transistors use populations of electrons rather than single electrons. It is certainly true that a single channel can exhibit much more complex behaviors than can a single electron in the active region of a transistor, but these channels are used in large populations, not in isolation.

We can compare the two technologies by asking how much energy is dissipated in charging up the gate of a transistor from a 0 to a 1. We might imagine that a transistor would compute a function that is loosely comparable to synaptic operation. In today's technology, it takes about 10^{-13} J to charge up the gate of a single minimum-size transistor. In ten years, the number will be about 10^{-15} J—within shooting range of the kind of efficiency realized by nervous systems. So the disparity between the efficiency of computation in the nervous system and that in a computer is primarily attributable not to the individual device requirements, but rather to the way the devices are used in the system.

Where Did the Energy Go?

Where did all the energy go? There is a factor of 1 million unaccounted for between what it costs to make a transistor work and what is required to do an operation the way we do it in a digital computer. There are two primary causes of energy waste in the digital systems we build today.

1) We lose a factor of about 100 because, the way we build digital hardware, the capacitance of the gate is only a very small fraction of capacitance of the node. The node is mostly wire, so we spend most of our energy charging up the wires and not the gate.

2) We use far more than one transistor to do an operation; in a typical implementation, we switch about 10 000 transistors to do one operation.

So altogether it costs 1 million times as much energy to make what we call an operation in a digital machine as it costs to operate a single transistor.

I do not believe that there is any magic in the nervous system—that there is a mysterious fluid in there that is not defined, some phenomenon that is orders of magnitude more effective than anything we can ever imagine. There is nothing that is done in the nervous system that we cannot emulate with electronics if we understand the principles of neural information processing. I have spent the last decade trying to understand enough about how it works to be able to build systems that work in a similar way; I have had modest success, as I shall describe.

So there are two big opportunities. The first factor-of-100 opportunity, which can be done with either digital or analog technology, is to make alogrithms more local, so that we do not have to ship the data all over the place. That is a big win—we have built digital chips that way, and have achieved a factor of between 10 and 100 reduction in power dissipation. That still leaves the factor of 10^4, which is the difference between making a digital operation out of bunches of AND and OR gates, and using the physics of the device to do the operation.

Evolution has made a lot of inventions, as it evolved the nervous system. I think of systems as divided into three somewhat arbitrarily levels. There is at the bottom the *elementary functions*, then the *representation of information*, and at the top the *organizing principles*. All three levels must work together; all three are very different from those we use in human-engineered systems. Furthermore, the nervous system is not accompanied by a manual explaining the principles of operation. The blueprints and the early prototypes were thrown away a long time ago. Now we are stuck with an artifact, so we must try to reverse engineer it.

Let us consider the primitive operations and representations in the nervous system, and contrast them with their counterparts in a digital system. As we think back, many of us remember being confused when we were first learning about digital design. First, we decide on the information representation. There is only one kind of information, and that is the bit: It is either a 1 or a 0. We also decide the elementary operations we allow, usually AND, OR, and NOT or their equivalents. We start by confining ourselves to an incredibly impoverished world, and out of that, we try to build something that makes sense. The miracle is that we can do it! But we pay the factor of 10^4 for taking all the beautiful phyics that is built into those transistors, mashing it down into a 1 or a 0, and then painfully building it back up, with AND and OR gates to reinvent the multiply. We then string together those multiplications and additions to get more complex operations—those that are useful in a system we wish to build.

Computation Primitives

What kind of computation primitives are implemented by the device physics we have available in nervous tissue or in a silicon integrated circuit? In both cases, the state variables are analog, represented by an electrical charge. In the nervous system, there are state variables represented by chemical concentrations as well. To build a nervous system or a computer, we must be able to make specific connections. A particular output is connected to certain inputs and not to others. To achieve that kind of specificity, we must be able to isolate one signal on a single electrical node, with minimum coupling to other nodes. In both electronics and the nervous system, that isolation is achieved by building an energy barrier, so that we can put some charge on

an electrical node somewhere, and it does not leak over to some other node nearby. In the nervous system, that energy barrier is built by the difference in the dielectric constant between fat and aqueous solutions. In electonics, it is built by the difference in the bandgap between silicon and silicon dioxide.

We do basic aggregation of information using the conservation of change. We can dump current onto an electrical node at any location, and it all ends up as charge on the node. Kirchhoff's law implements a distributed addition, and the capacitance of the node integrates the current into the node with respect to time.

In nervous tissue, ions are in thermal equilibrium with their surroundings, and hence their energies are Boltzmann distributed. This distribution, together with the presence of energy barriers, computes a current that is an exponential function of the barrier energy. If we modulate the barrier with an applied voltage, the current will be an exponential function of that voltage. That principle is used to create active devices (those that produce gain or amplification in signal level), both in the nervous system and in electronics. In addition to providing gain, an individual transistor computes a complex nonlinear function of its control and channel voltages. That function is not directly comparable to the functions that synapses evaluate using their presynaptic and postsynaptic potentials, but a few transistors can be connected strategically to compute remarkably competent synaptic functions.

Fig. 1(a) and (b) shows the current through a nerve membrane as a function of the voltage across the membrane. A plot of the current out of a synapse as the function of the voltage across the presynaptic membrane is shown in (c). The nervous system uses, as its basic operation, a current that increases exponentially with voltage. The channel current in a transistor as a function of the gate voltage is shown in (d). The current increases exponentially over many orders of magnitude, and then becomes limited by space charge, which reduces the dependence to the familiar quadratic. Note that this curve is hauntingly similar to others in the same figure. What class of computations can be implemented efficiently using expontential functions as primitives? Analog electronic circuits are an ideal way to explore this question.

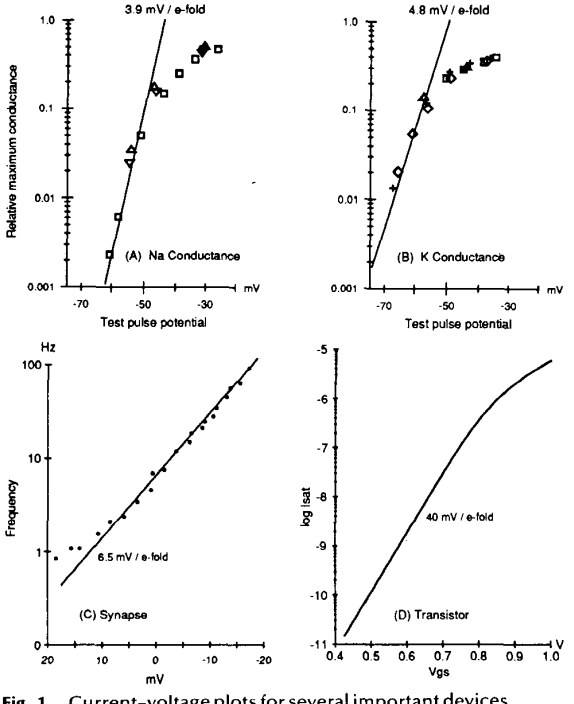


Fig. 1. Current-voltage plots for several important devices, each showing the ubiquitous exponential characteristic. Curves *A* and *B* show the behavior of populations of active ion channels in nerve membrane. Curve *C* illustrates the exponential dependence of the arrival rate of packets of the neurotransmitter at the postsynaptic membrane on the presynaptic membrane potential. Curve *D* shows the saturation current of a MOS transistor as a function of gate voltage.

Most important, the nervous system contains mechanisms for long-term learning and memory. All higher animals undergo permanent changes in their brains as a result of life experiences. Neurobiologists have identified at least one mechanism for these permanent changes, and are actively pursuing others. In microelectronics, we can store a certain quantity of charge on a floating polysilicon node, and that charge will be retained indefinitely. The floating node is completely surrounded by high-quality silicon dioxide—the world's most effective known insulator. We can sense the charge by making the floating node the gate of an ordinary MOS transistor. This mechanism has been used since 1971 for storing digital information in EPROM's and similar devices, but there is nothing inherently digital about the charge itself. Analog memory comes as a natural consequence of this near-perfect charge-storage mechanism. A silicon retina that does a rudimentary form of learning and long-term memory is described in the next section [2]. This system uses ultraviolet light to move charge through the oxide, onto or off the floating node. Tunneling to and from the floating node is used in commercial EEPROM devices. Several hot-electron mechanisms also have been employed to transfer charge through the oxide. The ability to learn and retain analog information for long periods is thus a natural consequence of the structures created by modern silicon processing technology.

The fact that we can build devices that implement the same basic operations as those the nervous system uses leads to the inevitable conclusion that we should be able to build entire systems based on the organizing principles used by the nervous system. I will refer to these systems generically as *neuromorphic systems*. We start by letting the device physics define our elementary operations. These functions provide a rich set of computational primitives, each a direct result of fundamental physical principles. They are not the operations out of which we are accustomed to building computers, but in many ways, they are much more interesting. They are more interesting than AND and OR. They are more interesting than multiplication and addition. But they are very different. If we try to fight them, to turn them into something with which we are familiar, we end up making a mess. So the real trick is to invent a representation that takes advantage of the inherent capabilities of the medium, such as the abilities to generate exponentials, to do integration with respect to time, and to implement a zero-cost addition using Kirchhoff's law. These are powerful primitives; using the nervous system as a guide, we will attempt to find a natural way to integrate them into an overall system-design strategy.

Retinal Computation

I shall use two examples from the evolution of silicon retinas to illustrate a number of physical principles that can be used to implement computation primitives. These examples also serve to introduce general principles of neural computation, and to show how these principles can be applied to realize effective systems in analog electronic integrated-circuit technology.

In 1868, Ernst Mach [3] described the operation performed by the retina in the following terms.

The illumination of a retinal point will, in proportion to the difference between this illumination and the average of the illumination on neighboring points, appear brighter or darker, respectively, depending on whether the illumination of it is above or below the average. The weight of the retinal points in this average is to be thought of as rapidly decreasing with distance from the particular point considered.

For many years, biologists have assembled evidence about the detailed mechanism by which this computation is accomplished. The neural machinery that performs this first step in the chain of visual processing is located in the outer plexiform layer of the retina, just under the photoreceptors. The lateral spread of information at the outer plexiform layer is mediated by a two-dimensional network of cells coupled by resistive connections. The voltage at every point in the network represents a spatially weighted average of the photoreceptor inputs. The farther away an input is from a point in the network, the less weight it is given. The weighting function decreases in a generally exponential manner with distance.

Using this biological evidence as a guide, Mahowald [4], [5] reported a silicon model of the computation described by Mach. In the silicon retina, each node in the network is linked to its six neighbors with resistive elements to form a hexagonal array, as shown in Fig. 2. A single bias circuit

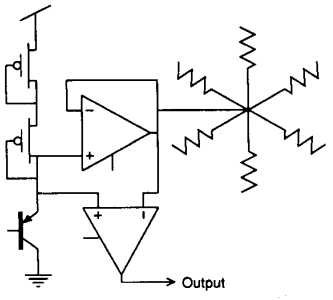


Fig. 2. Schematic of pixel from the Mahowald retina. The output is the difference between the potential of the local receptor and that of the resistive network. The network computes a weighted average over neighboring pixels.

associated with each node controls the strength of the six associated resistive connections. Each photoreceptor acts as a voltage input that drives the corresponding node of the resistive network through a conductance. A transconductance amplifier is used to implement a unidirectional conductance so the photoreceptor acts an effective voltage source. No current can be drawn from the output node of the photoreceptor because the amplifier input is connected to only the gate of a transistor.

The resistive network computes a spatially weighted average of photoreceptor inputs. The spatial scale of the weighting function is determined by the product of the lateral resistance and the conductance coupling the photoreceptors into the network. Varying the conductance of the transconductance amplifier or the strength of the resistors changes the space constant of the network, and thus changes the effective area over which signals are averaged.

From an engineering point of view, the primary function of the computation performed by a silicon retina is to provide an automatic gain control that extends the useful operating range of the system. It is essential that a sensory system be sensitive to changes in its input, no matter what the viewing conditions. The structure executing this level-normalization operation performs many other functions as well, such as computing the contrast ratio and enhancing edges in the image. Thus, the mechanisms responsible for keeping the system operating over an enormous range of image intensity have important consequences with regard to the representation of data.

The image enhancement performed by the retina was also described by Mach.

Let us call the intensity of illumunation $u = f(x, y)$. The brightness sensation v of the corresponding retinal point is given by

$$v = u - m\left(\frac{d^2u}{dx^2} + \frac{d^2u}{dy^2}\right)$$

where m is a constant. If the expression in parentheses is positive, then the sensation of brightness is reduced; in the opposite case, it is increased. Thus, v is not only influenced by u, but also its second differential quotients.

The image-enhancement property described by Mach is a result of the receptive field of the retinal computation, which shows an antagonistic center-surround response. This behavior is a result of the interaction of the photoreceptors, the resistive network, and the output amplifier. A transconductance amplifier provides a conductance through which the resistive network is driven towards the photoreceptor potential. A second amplifier senses the voltage difference across that conductance, and generates an output proportional to the difference between the photoreceptor potential and the network potential at that location. The output thus represents the difference between a center intensity and a weighted average of the intensities of surrounding points in the image.

The center-surround computation sometimes is referred to as a Laplacian filter, which has been used widely in computer vision systems. This computation, which can be approximated by a difference in Gaussians, has been used to help computers localize objects; this kind of enhancement is effective because discontinuities in intensity frequently correspond to object edges. Both of these mathematical forms express, in an analytically tractable way, the computation that occurs as a natural result of an efficient physical implementation of local normalization of the signal level.

In addition to its role in gain control and spatial filtering, the retina sharpens the time response of the system as an intrinsic part of its analog computation. Effective temporal processing requires that the time scale of the computation be matched to the time scale of external events. The temporal response of the silicon retina depends on the prop-

erties of the horizontal network. The voltage stored on the capacitance of the resistive network is the temporally as well as spatially averaged output of the photoreceptors. Because the capacitance of the horizontal network is driven by a finite conductance, its response weights its input by an amount that decreases exponentially into the past. The time constant of integration is set by the bias voltages of the wide-range amplifier and of the resistors. The time constant can be varied independently of the space constant, which depends on only the difference between these bias voltages, rather than on their absolute magnitude. The output of the retinal computation is thus the difference between the immediate local intensity and the spatially and temporally smoothed image. It therefore enhances both the first temporal and second spatial derivatives of the image.

Adaptive Retina

The Mahowald retina has given us a very realistic real-time model that shows essentially all of the perceptually interesting properties of early vision systems, including several well-known optical illusions such as Mach bands. One problem with the circuit is its sensitivity to transistor offset voltages. Under uniform illumination, the output is a random pattern reflecting the properties of individual transistors, no two of which are the same. Of course, biological retinas have precisely the same problem. No two receptors have the same sensitivity, and no two synapses have the same strength. The problem in wetware is even more acute than it is in silicon. It is also clear that biological systems use adaptive mechanisms to compensate for their lack of precision. The resulting system performance is well beyond that of our most advanced engineering marvels. Once we understand the principles of adaptation, we can incorporate them into our silicon retina.

All of our analog chips are fabricated in silicon-gate CMOS technology [6]. If no metal contact is made to the gate of a particular transistor, that gate will be completely surrounded by silicon dioxide. Any change parked on such a floating gate will remain for eons. The first floating-gate experiments of which I am aware were performed at Fairchild Research Laboratories in the mid-1960's. The first product to represent data by charges stored on a floating gate was reported in 1971 [7]. In this device, which today is called an EPROM, electrons are placed on the gate by an avalanche breakdown of the drain junction of the transistor. This injection can be done selectively, one junction at a time. Electrons can be removed by ultraviolet light incident on the chip. This so-called erase operation is performed on all devices simultaneously. In 1985, Glasser reported a circuit in which either a binary **1** or a binary **0** could be stored selectively in each location of a floating-gate digital memory [8]. The essential insight contributed by Glasser's work was that there is no fundamental asymetry to the current flowing through a thin layer of oxide. Electrons are excited into the conduction band of the oxide from both electrodes. The direction of current flow is determined primarily by the direction of the electric field in the oxide. In other words, the application of ultraviolet illumination to a capacitor with a silicon-dioxide dielectric has the effect of shunting the capacitor with a very small leakage conductance. With no illumination, the leakage conductance is effectively zero. The leakage conductance present during ultraviolet illumination thus provides a mechanism for adapting the charge on a float gate.

Frank Werblin suggested that the Mahowald retina might benefit from the known feedback connections from the resistive network to the photoreceptor circuit. A pixel incorporating a simplified version of this suggestion is shown in Fig. 3 [2]. In this circuit, the output node is the

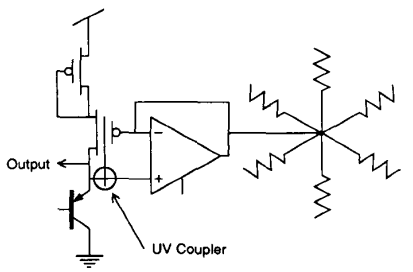


Fig. 3. Schematic of a pixel that performs a function similar to that of the Mahowald retina, but can be adapted with ultraviolet light to correct for output variations among pixels. This form of adaptation is the simplest form of learning. More sophisticated learning paradigms can be evolved directly from this structure.

emitter of the phototransistor. The current out of this node is thus set by the local incident-light intensity. The current into the output node is set by the potential on the resistive network, and hence by the weighted average of the light intensity in the neighborhood. The difference between these two currents is converted into a voltage by the effective resistance of the output node, determined primarily by the Early effect. The advantage of this circuit is that small differences between center intensity and surround intensity are translated into large output voltages, but the large dynamic range of operation is preserved. Retinas fabricated with this pixel show high gain, and operate properly over many orders of magnitude in illumination. The transconductance amplifier has a hyperbolic-tangent relationship between the output current and the input differential voltage. For proper operation, the conductance formed by this amplifier must be considerably smaller than that of the resistive network node. For that reason, when a local output node voltage is very different from the local network voltage, the amplifier saturates and supplies a fixed current to the node. The arrangement thus creates a center-surround response only slightly different from that of the Mahowald retina.

To reduce the effect of transistor offset voltages, we make use of ultraviolet adaptation to the floating gate that has been interposed between the resistive network and the pull-up transistor for the output node. The network is capacitively coupled to the floating node. The current into the output node is thus controlled by the voltage on the network, with an offset determined by the charge stored on the floating node. There is a region where the floating node overlaps the emitter of the phototransistor, shown inside the dark circle in Fig. 3. The entire chip is covered by second-level metal, except for openings over the phototransistors. The only way in which ultraviolet light can affect the floating gate is by interchanging electrons with the output

node. If the output node is high, the floating gate will be charged high, thereby decreasing the current into the output node. If the output node is low, the floating gate will be charged low, thereby increasing the current into the output node. The feedback occasioned by ultraviolet illumination is thus negative, driving all output nodes toward the same potential.

Adaptation and Learning

The adaptive retina is a simple example of a general computation paradigm. We can view the function of a particular part of the nervous system as making a prediction about the spatial and temporal porperties of the world. In the case of the retina, these predictions are the simple assertions that the image has no second spatial derivative and no first temporal derivative. If the image does not conform to these predictions, the difference between expectation and experience is sent upward to be processed at higher levels. A block diagram of the essential structure is shown in Fig. 4.

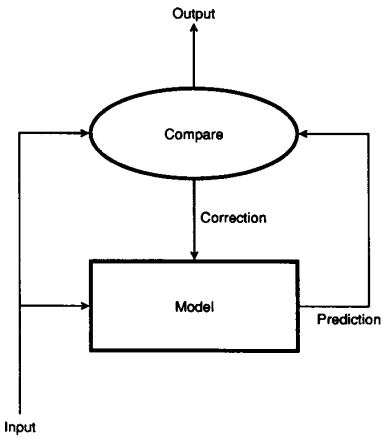


Fig. 4. Conceptual arrangement of a single level of a neural processing system. The computation consists of a prediction of the input, and a comparison of that prediction to the actual input. When the model accurately predicts the input, no information is passed to the next level, and no correction is made to the model. When the model fails to predict the input, the difference is used to correct the model. Random differences will cause a continued small "random walk" of the model parameters around that required for correct prediction. Systematic differences will cause the model to center itself over the true behavior of the input. Most routine events are filtered out to low level, reserving the capabilities of higher centers for genuinely interesting events.

The box labeled "model" is a predictor, perhaps a crude one; in the case of the retina, the model is the resistive network. We give the predictor the input over time, and it computes what is likely to happen next, just before the actual input arrives. Then, when that input materializes, it is compared to the prediction. If the two values are the same, no new information is produced; the system already knew what was about to happen. What happened is what was expected; therefore, no information is sent up to the next level of processing. But when something unexpected has occurred, there is a difference, and that difference is transferred on up to the next level to be interpreted. If we repeat this operation at each level of the nervous system, the information will be of higher quality at each subsequent level because we process only the information that could not be predicted at lower levels.

Learning in this kind of system is provided by the adaptation feedback from the comparator to the model. If the model is making predictions that are systematically different from what happens in nature, the ongoing corrections based on the individual differences will cause the model to learn what actually happens, as well as can be captured at its level of representation. It is only those events that are truly random, or that cannot be predicted from this level and therefore appear random, that will cancel out over all experience. The system parameters will undergo a local random walk, but will stay nearly centered on the average of what nature is providing as input. The retina is presented with a wide variety of scenes; it sees white edges and black edges. But every pixel in the retina sees the same intensity, averaged over time. Corrections towards this average constantly correct differences in photoreceptor sensitivity and variation in the properties of individual neurons and synapses. All other information is passed up to higher levels. Even this simple level of prediction removes a great deal of meaningless detail from the image, and provides a higher level of representation for the next level of discrimination.

That a system composed of many levels organized along the lines of Fig. 4 can compute truly awesome results is perhaps not surprising: each level is equipped with a model of the world, as represented by the information passed up from lower levels. All lower level processing may, from the point of view of a given level, be considered preprocessing. The most important property of this kind of system is that the same mechanism that adapts out errors and mismatches in its individual components also enables the system to build its own models through continued exposure to information coming in from the world. Although this particular example of the adaptive retina learns only a simple model, it illustrates a much more general principle: this kind of system is *self-organizing* in the most profound sense.

Nueral Silicon

Over the past eight years, we have designed, fabricated, and evaluated hundreds of test chips and several dozen complete system-level designs. All these adaptive analog chips were fabricated using standard, commercially available CMOS processing, provided to us under the auspices of DARPA's MOSIS fabrication service. These designs include control systems, motor-pattern generators, retina chips that track bright spots in an image, retina chips that focus images on themselves, and retina chips that perform gain control, motion sensing, and image enhancement. We have made multiscale retinas that give several levels of resolution, stereo-vision chips that see depth, and chips that segment images. A wide variety of systems has been designed to process auditory input; most of them are based on a biologically sensible model of the cochlea. There are monaural chips that decompose sound into its component features, binaural chips that compute horizontal and vertical localization of sound sources, and Seehear chips that convert a visual image into an auditory image—one where moving objects produce sound localized in the direction of the object.

This variety of experiments gives us a feeling for how far we have progressed on the quest for the nine order of magnitude biological advantage. The retina described in the preceding section is a typical example; it contains about 10^5 devices, performs the equivalent of about 10^8 operations/s, and consumes about 10^{-3} W of power. This and other chips using the same techniques thus perform each operation at a cost of only about 10^{-11} J compared to about 10^{-7} J/operation for a digital design using the same technology, and with 10^6 J/operation for the brain. We are still five orders of magnitude away from the efficiency of the brain, but four orders of magnitude ahead of that realized with digital techniques. The real question is how well the adaptive analog approach can take advantage of future advances in silicon fabrication. My prediction is that adaptive analog techniques can utilize the potential of advanced silicon fabrication more fully than can any other approach that has been proposed. Today (1990), a typical 6 in diameter wafer contains about 10^8 devices, partitioned into several hundred chips. After fabrication, the chips are cut apart and are put into packages. Several hundred of these packages are placed on a circuit board, which forms interconnections among them.

Why not just interconnect the chips on the wafer where they started, and dispense with all the extra fuss, bother, and expense? Many attempts by many groups to make a digital wafer-scale technology have met with abysmal failure. There are two basic reasons why wafer-scale integration is very difficult. First, a typical digital chip will fail if even a single transistor or wire on the chip is defective. Second, the power dissipated by several hundred chips of circuitry is over 100 W, and getting rid of all that heat is a major packaging problem. Together, these two problems have prevented even the largest computer companies from deploying wafer-scale systems successfully. The low-power dissipation of adaptive analog systems eliminates the packaging problem; wafers can be mounted on edge, and normal air convection will adequately remove the few hundred milliwatts of heat dissipated per wafer. Due to the robustness of the neural representation, the failure of a few components per square centimeter will not materially affect the performance of the system: its adaptive nature will allow the system simply to learn to ignore these inputs because they convey no information. In one or two decades, I believe we will have 10^{10} devices on a wafer, connected as a complete adaptive analog system. We will be able to extract information from connections made around the periphery of the wafer, while processing takes place in massively parallel form over the entire surface of the wafer. Each wafer operating in this manner will be capable of approximately 10^{13} operations/s. At that time, we will still not understand nearly as much about the brain as we do about the technology.

Scaling Laws

The possibility of wafer-scale integration naturally raises the question of the relative advantage conveyed by a three-dimensional neural structure over a two-dimensional one. Both approaches have been pursued in the evolution of animal brains so the question is of great interest in biology as well. Let us take the point of view that whatever we are going to build will be a space-filling structure. If it is a sheet, it will have neurons throughout the whole plane; if it is a volume, neurons will occupy the whole volume. If we allow every wire from every neuron to be as long as the dimensions of the entire structure, we will obviously get an explosion in the size of the structure as the number of neurons increases. The brain has not done that. If we compare our brain to a rat brain, we are not noticeably less efficient in our use of wiring resources. So the brain has evolved a *mostly local* wiring strategy to keep the scaling from getting out of hand. What are the requirements of a structure that keep the fraction of its resources devoted to wire from exploding as it is made larger? If the structure did not scale, a large brain would be all wire and would have no room for the computation.

First, let us consider the two-dimensional case. For the purpose of analysis, we can imagine that the width W of each wire is independent of the wire's length L, and that the probability that a wire of length between L and $L + dL$ is dedicated to each neuron is $p(L)\,dL$. The expected area of such a wire is the $WL\,p(L)\,dL$. The entire plane, of length and width L_{max}, is covered with neurons, such that there is one neuron per area A. Although the wires from many neurons overlap, the total wire from any given neuron must fit in area A. We can integrate the areas of the wires of all lengths associated with a given neuron, assuming that the shortest wire is of unit length:

$$\int_1^{L\max} WL\,p(L)\,dL = A.$$

The question is then: What are the bounds on the form of $p(L)$ such that the area A required for each neuron does not grow explosively as L_{max} becomes large? We can easily see that if $p(L) = 1/L^2$, the area A grows as the logarithm of L_{max}—a quite reasonable behavior. If $p(L)$ did not decrease at least this fast with increasing L_{max}, the human brain would be much more dominated by wire than it is, compared to the brain of a rat or a bat. From this argument, I conclude that the nervous system is organized such that, on the average, the number of wires decreases no more slowly than the inverse square of the wire's length.

We can repeat the analysis for a three-dimensional neural structure of extent L_{max}, in which each neuron occupies volume V. Each wire has a cross-sectional area S, and thus has an expected volume $SL\,p(L)$. As before, the total wire associated with each neuron must fit in volume V:

$$\int_1^{L\max} SL\,p(L)\,dL = V.$$

So the three-dimensional structure must follow the same scaling law as its two-dimensional counterpart. If we build a space-filling structure, the third dimension allows us to contact more neurons, but it does not change the basic scaling rule. The number of wires must decrease with wire length in the same way in both two and three dimensions.

The cortex of the human brain, if it is stretched out, is about 1 m/side, and 1 mm thick. About half of that millimeter is wire (white matter), and the other half is computing machinery (gray matter). This basically two-dimensional strategy won out over the three-dimensional strategies used by more primitive animals, apparently because it could evolve more easily: new areas of cortex could arise in the natural course of evolution, and some of them would be

retained in the genome if they conveyed a competitive advantage on their owners. This result gives us hope that a neural structure comprising many two-dimensional areas, such as those we can make on silicon wafers, can be made into a truly useful, massively parallel, adaptive computing system.

Conclusion

Biological information-processing systems operate on completely different principles from those with which engineers are familiar. For many problems, particularly those in which the input data are ill-conditioned and the computation can be specified in a relative manner, biological solutions are many orders of magnitude more effective than those we have been able to implement using digital methods. I have shown that this advantage can be attributed principally to the use of elementary physical phenomena as computational primitives, and to the representation of information by the relative values of analog signals, rather than by the absolute values of digital signals. I have argued that this approach requires adaptive techniques to correct for differences between nominally identical components, and that this adaptive capability leads naturally to systems that learn about their environment. Although the adaptive analog systems build up to the present time are rudimentary, they have demonstrated important principles as a prerequisite to undertaking projects of much larger scope. Perhaps the most intriguing result of these experiments has been the suggestion that adaptive analog systems are 100 times more efficient in their use of silicon, and they use 10 000 times less power than comparable digital systems. It is also clear that these systems are more robust to component degradation and failure than are more conventional systems. I have also argued that the basic two-dimensional limitation of silicon technology is not a serious limitation in exploiting the potential of neuromorphic systems. For these reasons, I expect large-scale adaptive analog technology to permit the full utilization of the enormous, heretofore unrealized, potential of wafer-scale silicon fabrication.

References

[1] B. Hoeneisen and C. A. Mead, "Fundamental limitations in microelectronics—I. MOS technology," *Solid-State Electron.*, vol. 15, pp. 819-829, 1972.

[2] C. Mead, "Adaptive retina," in *Analog VLSI Implementation of Neural Systems*, C. Mead and M. Ismail, Eds. Boston, MA: Kluwer, 1989, pp. 239-246.

[3] F. Ratliff, *Mach Bands: Quantitative Studies on Neural Networks in the Retina*. San Francisco, CA: Holden-Day, 1965, pp. 253-332.

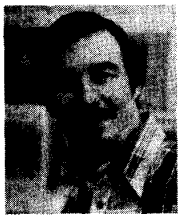
[4] M. A. Mahowald and C. A. Mead, "A silicon model of early visual processing," *Neural Networks*, vol. 1, pp. 91-97, 1988.

[5] ——, "Silicon retina," in C. A. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989, pp. 257-278.

[6] C. A. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.

[7] D. Frohman-Bentchkowsky, "Memory behavior in a floating-gate avalanche-injection MOS (FAMOS) structure," *Appl. Phys. Lett.*, vol. 18, pp. 332-334, Apr. 1971.

[8] L. A. Glasser, "A UV write-enabled PROM," in *Proc. 1985 Chapel Hill Conf. VLSI*, H. Fuchs, Ed. Rockville, MD: Computer Science Press, 1985, pp. 61-65.



Carver A. Mead is Gordon and Betty Moore Professor of Computer Science at the California Institute of Technology, Pasadena, where he has taught for more than 30 years. He has contributed in the fields of solid-state electronics and the management of complexity in the design of very-large-scale integrated circuits, and has been active in the development of innovative design methodologies for VLSI. He has written, with Lynn Conway, the standard text for VLSI design, *Introduction to VLSI Systems*. His recent work is concerned with modeling neuronal structures, such as the retina and the cochlea, using analog VLSI systems. His new book on this topic, *Analog VLSI and Neural Systems*, has recently been published by Addison-Wesley.

Dr. Mead is a member of the National Academy of Sciences, the National Academy of Engineering, a foreign member of the Royal Swedish Academy of Engineering Sciences, a Fellow of the American Physical Society, and a Life Fellow of the Franklin Institute. He is also the recipient of a number of awards, including the Centennial Medal of the IEEE.