We Could Build an Artificial Brain Right Now

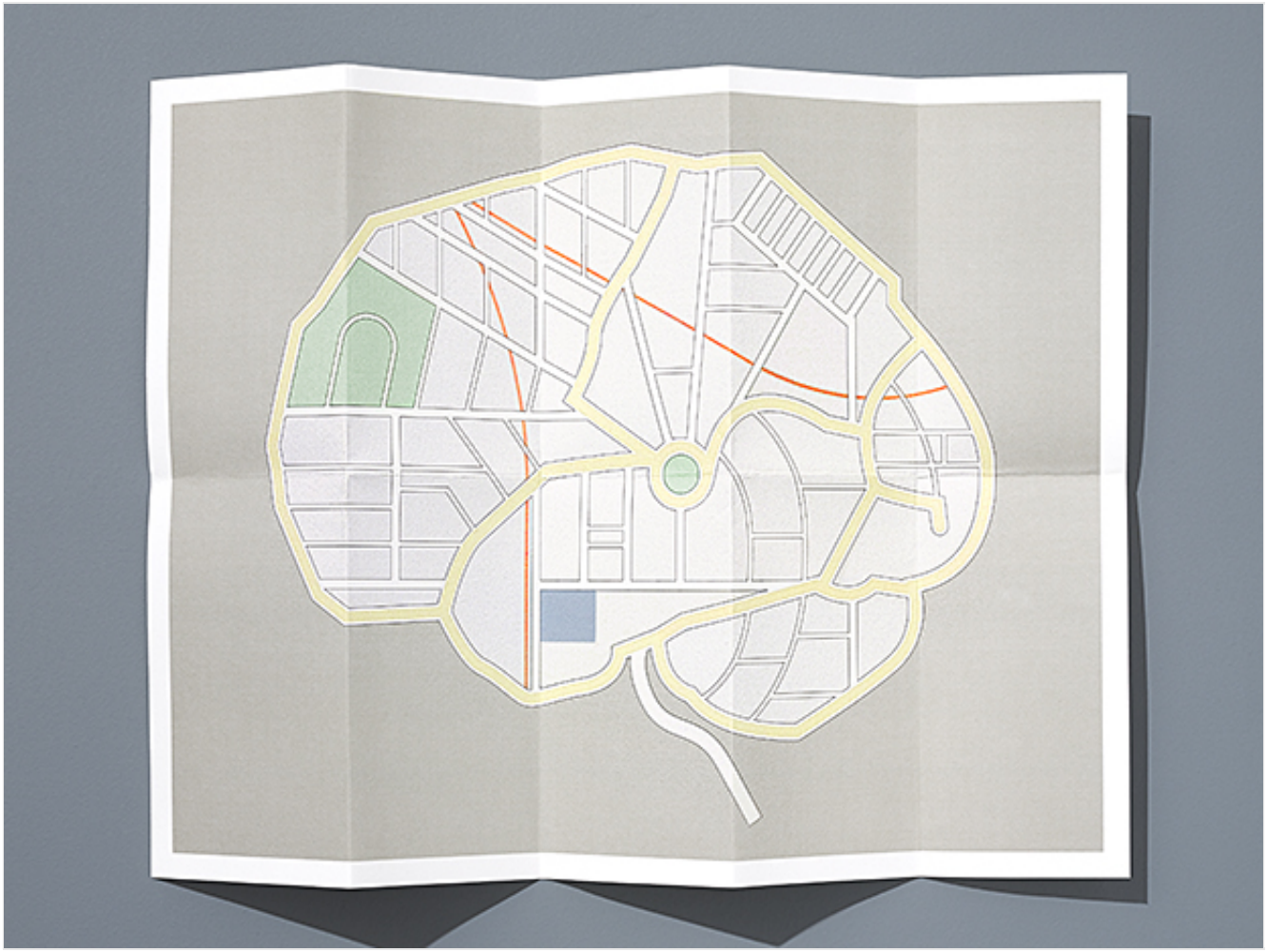

Photo: Dan Saelinger

Brain-inspired computing is having a moment. Artificial neural network algorithms like deep learning, which are very loosely based on the way the human brain operates, now allow digital computers to perform such extraordinary feats as translating language, hunting for subtle patterns in huge amounts of data, and beating the best human players at Go.

But even as engineers continue to push this mighty computing strategy, the energy efficiency of digital computing is fast approaching its limits. Our data centers and supercomputers already draw megawatts—some 2 percent of the electricity consumed in the United States goes to data centers alone.

The human brain, by contrast, runs quite well on about 20 watts, which represents the power produced by just a fraction of the food a person eats each day. If we want to keep improving computing, we will need our computers to become more like our brains.

Hence the recent focus on neuromorphic technology, which promises to move computing beyond simple neural networks and toward circuits that operate more like the brain's neurons and synapses do. The development of such physical brainlike circuitry is actually pretty far along. Work at my lab and others around the world over the past 35 years has led to artificial neural components like synapses and dendrites that respond to and produce electrical signals much like the real thing.

So, what would it take to integrate these building blocks into a brain-scale computer? In 2013, Bo Marr, a former graduate student of mine at Georgia Tech, and I looked at the best engineering and neuroscience knowledge of the time and concluded that it should be possible to build a silicon version of the human cerebral cortex with the transistor technology then in production. What's more, the resulting machine would take up less than a cubic meter of space and consume less than 100 watts, not too far from the human brain.

That is not to say creating such a computer would be easy. The system we envisioned would still require a few billion dollars to design and build, including some significant packaging innovations to make it compact. There is also the question of how we would program and train the computer. Neuromorphic researchers are still struggling to understand how to make thousands of artificial neurons work together and how to translate brainlike activity into useful engineering applications.

Still, the fact that we can envision such a system means that we may not be far off from smaller-scale chips that could be used in portable and wearable electronics. These gadgets demand low power consumption, and so a highly energy-efficient neuromorphic chip—even if it takes on only a subset of computational tasks, such as signal processing—could be revolutionary. Existing capabilities, like speech recognition, could be extended to handle noisy environments. We could even imagine future smartphones conducting real-time language translation between you and the person you're talking to. Think of it this way: In the 40 years since the first signal-processing integrated circuits, Moore's Law has improved energy efficiency by roughly a factor of 1,000. The most brainlike neuromorphic chips could dwarf such improvements, potentially driving down power consumption by another factor of 100 million. That would bring computations that would otherwise need a data center to the palm of your hand.

The ultimate brainlike machine will be one in which we build analogues for all the essential functional components of the brain: the synapses, which connect neurons and allow them to receive and respond to signals; the dendrites, which combine and perform local computations on those incoming signals; and the core, or soma, region of each neuron, which integrates inputs from the dendrites and transmits its output on the axon.

Simple versions of all these basic components have already been implemented in silicon. The starting point for such work is the same metal-oxide-semiconductor field-effect transistor, or MOSFET, that is used by the billions to build the logic circuitry in modern digital processors.

These devices have a lot in common with neurons. Neurons operate using voltage-controlled barriers, and their electrical and chemical activity depends primarily on channels in which ions move between the interior and

exterior of the cell—a smooth, analog process that involves a steady buildup or decline instead of a simple on-off operation.

MOSFETs are also voltage controlled and operate by the movement of individual units of charge. And when MOSFETs are operated in the "subthreshold" mode, below the voltage threshold used to digitally switch between on and off, the amount of current flowing through the device is very small—less than a thousandth of what is seen in the typical switching of digital logic gates.

The notion that subthreshold transistor physics could be used to build brainlike circuitry originated with Carver Mead of Caltech, who helped revolutionize the field of very-large-scale circuit design in the 1970s. Mead pointed out that chip designers fail to take advantage of a lot of interesting behavior—and thus information—when they use transistors only for digital logic. The process, he wrote in 1990 [PDF], essentially involves "taking all the beautiful physics that is built into…transistors, mashing it down to a 1 or 0, and then painfully building it back up with AND and OR gates to reinvent the multiply." A more "physical" or "physics-based" computer could execute more computations per unit energy than its digital counterpart. Mead predicted such a computer would take up significantly less space as well.

In the intervening years, neuromorphic engineers have made all the basic building blocks of the brain out of silicon with a great deal of biological fidelity. The neuron's dendrite, axon, and soma components can all be fabricated from standard transistors and other circuit elements. In 2005, for example, Ethan Farquhar, then a Ph.D. candidate, and I created a neuron circuit using a set of six MOSFETs and a handful of capacitors. Our model generated electrical pulses that very closely matched those in the

soma part of a squid neuron, a long-standing experimental subject. What's more, our circuit accomplished this feat with similar current levels and energy consumption to those in the squid's brain. If we had instead used analog circuits to model the equations neuroscientists have developed to describe that behavior, we'd need on the order of 10 times as many transistors. Performing those calculations with a digital computer would require even more space.

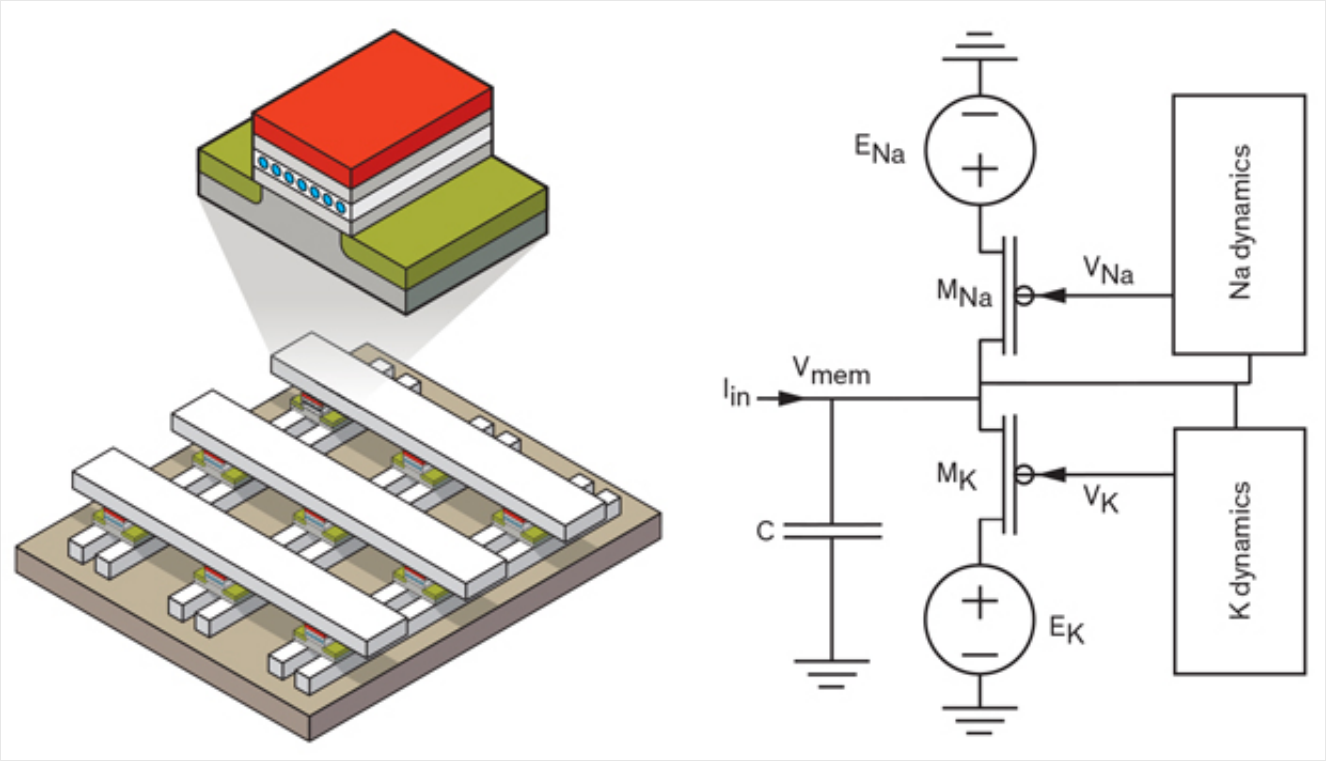

Illustration: James Provost

Synapses and Soma: The floating-gate transistor [top left], which can store differing amounts of charge, can be used to build a "crossbar" array of artificial synapses [bottom left]. Electronic versions of other neuron components, such as the soma region [right], can be made from standard transistors and other circuit components.

Emulating synapses is a little trickier. A device that behaves like a synapse must have the ability to remember what state it is in, respond in a particular way to an incoming signal, and adapt its response over time.

There are a number of potential approaches to building synapses. The most

mature one by far is the single-transistor learning synapse (STLS), a device that my colleagues and I at Caltech worked on in the 1990s while I was a graduate student studying under Mead.

We first presented the STLS in 1994, and it became an important tool for engineers who were building modern analog circuitry, such as physical neural networks. In neural networks, each node in the network has a weight associated with it, and those weights determine how data from different nodes are combined. The STLS was the first device that could hold a variety of different weights and be reprogrammed on the fly. The device is also nonvolatile, which means that it remembers its state even when not in use—a capability that significantly reduces how much energy it needs.

The STLS is a type of floating-gate transistor, a device that is used to build memory cells in flash memory. In an ordinary MOSFET, a gate controls the flow of electricity through a current-carrying channel. A floating-gate transistor has a second gate that sits between this electrical gate and the channel. This floating gate is not directly connected to ground or any other component. Thanks to that electrical isolation, which is enhanced by high-quality silicon-insulator interfaces, charges remain in the floating gate for a long time. The floating gate can take on many different amounts of charge and so have many different levels of electrical response, an essential requisite for creating an artificial synapse capable of varying its response to stimuli.

My colleagues and I used the STLS to demonstrate the first crossbar network, a computational model currently popular with nanodevice researchers. In this two-dimensional array, devices sit at the intersection of input lines running north-south and output lines running east-west. This configuration is useful because it lets you program the connection strength

of each “synapse” individually, without disturbing the other elements in the array.

Thanks in part to a recent Defense Advanced Research Projects Agency program called SyNAPSE, the neuromorphic engineering field has seen a surge of research into artificial synapses built from nanodevices such as memristors, resistive RAM, and phase-change memories (as well as floating-gate devices). But it will be hard for these new artificial synapses to improve on our two-decade-old floating-gate arrays. Memristors and other novel memories come with programming challenges; some have device architectures that make it difficult to target a single specific device in a crossbar array. Others need a dedicated transistor in order to be programmed, adding significantly to their footprint. Because floating-gate memory is programmable over a wide range of values, it can be more easily fine-tuned to compensate for manufacturing variation from device to device than can many nanodevices. A number of neuromorphic research groups that tried integrating nanodevices into their designs have recently come around to using floating-gate devices.

So how will we put all these brainlike components together? In the human brain, of course, neurons and synapses are intermingled. Neuromorphic chip designers must take a more integrated approach as well, with all neural components on the same chip, tightly mixed together. This is not the case in many neuromorphic labs today: To make research projects more manageable, different building blocks may be placed in different areas. Synapses, for example, may be relegated to an off-chip array. Connections may be routed through another chip called a field-programmable gate array, or FPGA.

But as we scale up neuromorphic systems, we’ll need to take care that we

don’t replicate the arrangement in today’s computers, which lose a significant amount of energy driving bits back and forth between logic, memory, and storage. Today, a computer can easily consume 10 times the energy to move the data needed for a multiple-accumulate operation—a common signal-processing computation—as it does to perform the calculation.

The brain, by contrast, minimizes the energy cost of communication by keeping operations highly local. The memory elements of the brain, such as synaptic strengths, are mixed in with the neural components that integrate signals. And the brain’s “wires”—the dendrites and axons that extend from neurons to transmit, respectively, incoming signals and outgoing pulses—are generally fairly short relative to the size of the brain, so they don’t require large amounts of energy to sustain a signal. From anatomical data, we know that more than 90 percent of neurons connect with only their nearest 1,000 or so neighbors.

Another big question for the builders of brainlike chips and computers is the algorithms we will run on them. Even a loosely brain-inspired system can have a big advantage over digital systems. In 2004, for example, my group used floating-gate devices to perform multiplications for signal processing with just 1/1,000 the energy and 1/100 the area of a comparable digital system. In the years since, other researchers and my group have successfully demonstrated neuromorphic approaches to many other kinds of signal-processing calculations.

But the brain is still 100,000 times as efficient as the systems in these demonstrations. That’s because while our current neuromorphic technology takes advantage of the neuronlike physics of transistors, it doesn’t consider the algorithms the brain uses to perform its operations.

Today, we are just beginning to discover these physical algorithms—that is, the processes that will allow brainlike chips to operate with more brainlike efficiency. Four years ago, my research group used silicon somas, synapses, and dendrites to perform a word-spotting algorithm that identifies words in a speech waveform. This physical algorithm exhibited a thousandfold improvement in energy efficiency over predicted analog signal processing. Eventually, by lowering the amount of voltage supplied to the chips and using smaller transistors, researchers should be able to build chips that parallel the brain in efficiency for a range of computations.

When I started in neuromorphic research 30 years ago, everyone believed tremendous opportunities would arise from designing systems that are more like the brain. And indeed, entire industries are now being built around brain-inspired AI and deep learning, with applications that promise to transform—among other things—our mobile devices, our financial institutions, and how we interact in public spaces.

And yet, these applications rely only slightly on what we know about how the brain actually works. The next 30 years will undoubtedly see the incorporation of more such knowledge. We already have much of the basic hardware we need to accomplish this neuroscience-to-computing translation. But we must develop a better understanding of how that hardware should behave—and what computational schemes will yield the greatest real-world benefits.

Consider this a call to action. We have come pretty far with a very loose model of how the brain works. But neuroscience could lead to far more sophisticated brainlike computers. And what greater feat could there be than using our own brains to learn how to build new ones?

This article appears in the June 2017 print issue as "A Road Map for the Artificial Brain."

About the Author

Jennifer Hasler is a professor of electrical and computer engineering at the Georgia Institute of Technology.