Book Review

Computing with Connections

W. DANIEL HILLIS. **The Connection Machine.** Cambridge,
MA: MIT Press, 1985. Pp. xi + 190. $22.50.

Reviewed by TERRENCE J. SEJNOWSKI

W. Daniel Hillis is President of Thinking Machines Corporation, a company that he co-founded. He obtained his Ph. D. in Computer Science at MIT in 1985.

The reviewer, Terrence J. Sejnoski, received a Ph. D. in physics from Princeton University in 1978. He is currently an Associate Professor of Biophysics, Biology, and Electrical Engineering and Computer Science at the Johns Hopkins University. His primary research interest is how information is represented, transformed, and learned, by parallel networks of neurons.

In 1981 the VLSI revolution was already in full swing when graduate student W. Daniel Hillis wrote MIT AI Memo No. 646 on the "Connection Machine (Computer Architecture for the New Wave)." The memo outlined a new type of parallel computer with millions of processing units that was "designed for symbol manipulation, not number crunching." Other computer architects were linking together tens to hundreds of off-the-shelf microprocessors; Hillis suggested instead building a much finer-grain architecture based on many more custom-built processors, each with only a few thousand bits of memory and capable of performing only simple logical operations. This design was inspired by parallel architectures for artificial intelligence, particularly Scott Fahlman's NETL (1979), that could implement fast set intersection in semantic networks.

The book under review is Hillis' doctoral dissertation, published just four years after the AI Memo. It describes both the design and implementation of a 65,536 processor Connection Machine^{TM}, a computer that is now manufactured by Thinking Machines Corporation, a company Hillis co-founded. Curiously, the original motivation for the Connection Machine—semantic networks in artificial intelligence—remains unimplemented. The Connection Machine has found other uses as a general-purpose parallel processor suitable for a wide variety of problems, many unanticipated when the Connection Machine was conceived. In particular, the recent work on connectionist models in artificial intelligence (Feldman & Ballard,

Please address all communications and requests for reprints to Dr. Terrence J. Sejnowski, Department of Biophysics, Johns Hopkins University, Baltimore, MD 21218.

0022-2496/87 $3.00

1982) and the parallel distributed processing models in cognitive science (Rumelhart & McClelland, 1986) could greatly benefit from the enormous potential for computation provided by the extensible hardware design of the Connection Machine.

In this review I will focus on the parts of the book that are most relevant to the goal expressed by Hillis in the introduction: "1.1 We Would Like to Make a Thinking Machine." Even the technical parts of the book, however, are highly accessible to a general scientific audience and many of the general design features are explained in a clear, informal style, perhaps influenced by Disney World where the first draft of the book was written. In addition to hardware considerations, the book also has chapters on data structures and algorithms, such as sorting, that are appropriate for the architecture.

Parallel Computing

Computation requires getting the right piece of information to the right place at the right time, and doing the right thing to it once it gets there. The key to the Connection Machine is how the information gets there. In a conventional digital computer, both the program and the data are stored in the memory and single items are sent to the central processing unit sequentially. This feature characterizes the von Neumann architecture and was fixed in the early days of digital computers when memory was slow and expensive. The rate of processing is now sufficiently high in the fastest computers that the speed of light limits the amount of information that can be transmitted sequentially through the connection between the memory and the central processing unit: this is the so-called von Neumann bottleneck.

Numerous designs have been proposed to open up the bottleneck. For example, the Bolt, Beranek, and Newman Butterfly^{TM} and the Intel Cosmic Cube^{TM} have multiple computers that communicate with each other through a switching network. The Connection Machine is more similar in its basic architecture to the Massively Parallel Processor (MPP) built by Goodyear for NASA in 1983, intended primarily for processing satellite images (Potter, 1985), and the Distributed Array Processor (DAP), built even earlier in England, about one-fourth the size of the MPP. The MPP has 16,000 paired memories and processors which communicate with their nearest neighbors on a two-dimensional grid. It executes single instructions on multiple data (SIMD).

The Connection Machine is also a SIMD architecture but differs from the MPP in two respects: First, the processors in the Connection Machine are at the corners of an n-dimensional hypercube (so that each has a direct connection to n neighbors). Only three pages are devoted to a "Tour of the Topology Zoo," an important topic that deserved more discussion. Second, the processors can send packets of information to any other processor by paths that are automatically routed through other processors. Routing is much slower than a direct connection, but allows more flexibility for some applications. The design of the router, based on probabilistic algorithms (Valiant, 1982), is probably the most impressive technical achievement in the Connection Machine and the details remain a trade secret of the Thinking Machines Corporation.

In principle, a conventional digital computer could simulate any parallel design, given world enough and time. In practice, the motivation for research into algorithms is highly architecture dependent, because the only way to fully test an algorithm is to implement it and measure its performance. Applications that have been successfully implemented on the Connection Machine include stereo and motion algorithms in computer vision, VLSI circuit design using simulated annealing, hydrodynamic simulations using cellular automata, and document retrieval based on key words. New parallel algorithms are being discovered for problems, such as the inversion of sparse matrices, that were previously overlooked, in part because they are inefficient on the von Neumann architecture. Even rule-based expert systems can be implemented in parallel (Blelloch, 1986).

Debugging programs on a parallel machine can be a nightmare unless good programming tools are available. Fortunately, the Connection Machine can be programmed in *Lisp, a version of Common Lisp to which parallel instructions have been added. It is possible in this environment for a moderately skilled programmer to bring up new applications in a few days. CmLisp, a more powerful version that "defines the Connection Machine" (p. 47) is not yet released.

Many of the knowledge representations and search algorithms that are commonly used in artificial intelligence have been optimized for the von Neumann architecture. Similarly, the model of computation based on logic that led to the von Neumann architecture has served as a model for human reasoning in cognitive science (Pylyshyn, 1984). The recent availability of parallel hardware makes apparent the extent to which cognitive science and artificial intelligence have been shaped by hardware that is based on sequential symbol processing. If it is no longer necessary to make a virtue of the von Neumann bottleneck, how should symbols be distributed over a million processors to take best advantage of the increased processing power of a parallel architecture? Tinkering with fundamental assumptions like this one may lead to new computational models for artificial intelligence, cognitive science, and neuroscience (Churchland, 1986).

BRAINS

Parallel machines may also have significance for the exploration of the human brain, by far the most sophisticated parallel computer in existence. In the brain there are over 100 billion neurons and 100 trillion synaptic connections between neurons, but more than 90% of the brain's volume is composed of axons and dendrites; that is, the brain is mostly connections. These biological wires are poor conductors compared with copper wire. Axons transmit information at a maximum rate of a few hundred bits per second and at speeds of at most a few hundred meters per second. Moreover, neurons are relatively simple processing units compared with digital computers, capable only of adding and multiplying analog signals with low precision. How is the brain, using processors with a cycle time measured in milliseconds, able to retrieve information in less than a second? Even more impressive is our ability in an equally short time to recognize objects in images and plan limb movements. The brain can perform in a few hundred cycles what digital machines cannot now perform in many millions of cycles.

Hillis compares digital computers with brains on p. 3. It is difficult to compare their processing powers because we do not yet understand the principles of computation in the brain. Hillis compares the maximum switching rate of gates in a computer (a billion transistors switching a billion times per second) and the maximum rate of firing of all the neurons in the brain (100 billion neurons firing at a thousand times a second). However, this is not the right measure since switching events by themselves are only one part of performing a computation, and both the brain and the digital computer would burn out if all their components were to start switching at their maximum rates for even a short time. Two more realistic measures of performance are the average processing power, measured in operations per second, and useful communications bandwidth, measured in bits per second.

The processing units in the current generation Connection Machine are bit-sliced processors with a one microsecond cycle time and 4,000 bits of memory each. A processor can add two numbers with 8 bits of accuracy in 8 cycles, and can multiply two numbers with the same accuracy in 64 cycles. Thus, the 65,536 processor Connection Machine can perform a maximum of about one billion 8-bit multiplications per second. The total communications bandwidth between processing units in a Connection Machine is about 10 billion bits per second, but the I/O bandwidth for communicating between the Connection Machine and its host computer is only 500 million bits per second. Only a fraction of the maximum processing power of the Connection Machine may be achieved on a particular problem unless a highly efficient algorithm is found that maps well onto the architecture of the Connection Machine and all of the information needed to solve the problem is resident. Thus, if the data exceeds 32 MB (the total memory capacity of a 65,536 Connection Machine) then the I/O bandwidth may be rate-limiting.

Firing at a maximum rate of a few hundred spikes per second, a neuron can convey only a few bits per second via its average rate of firing, but it can communicate by direct connections with thousands of other neurons. Hence, the *average* communications bandwidth used by the brain in moment to moment computation is about

$$(10^{11}\ \text{neurons})(5 \times 10^{3}\ \text{connections/neuron})(2\ \text{bits/connection/sec}) \approx 10^{15}\ \text{bits/sec}.$$

This is about a 10^5 times greater bandwidth than the current generation Connection Machine. It is significant that the brain can make effective use of this bandwidth; each synapse between neurons can perform a low-precision addition or multiplication (depending on the type of synapse). Hence, the *average* processing rate in the brain is at least 10^{15} operations per second. This estimate represents the minimal amount of digital computation that must be done to simulate neural operations in real time. It is a lower bound since we have not taken fully into

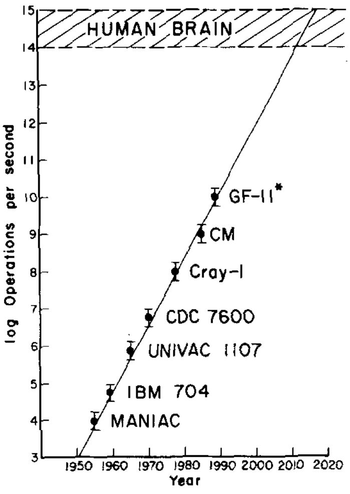

FIG. 1. Graph of computing power, measured in operations per second, for the largest general pupose digital computers as a function of time. Operations vary from simple boolean evaluations to 64 bit floating point arithmetic and vary in their execution times. Different problems require different mixtures of operations, so the error bars indicate the approximate range of the effective computing power. The Connection Machine (CM) is described in the review. The GF-11 is an experimental machine under development at IBM. A lower bound for the equivalent computational power needed to simulate the synaptic activity in the human brain is given in the text and drawn as a horizontal dashed region at the top of the graph. In primates, the visual system uses about 20–40% of the total processing power.

account the analog operations that occur in dendritic trees. Many of the operations in the brain are analog and could be simulated much more efficiently with analog technology (Mead, 1987).

The cost of computing has decreased by a factor of about 10 every 5 years over the last 35 years (Fig. 1). If this continues, then it will only take about 25 more years (2015) before processing power comparable to that in the brain can be purchased for $3 Million, approximately the current cost of the Connection Machine. David Waltz (personal communication) independently arrived at a similar conclusion taking into account the cost of memory, communications, and processing. It is very unlikely, however, that this goal can be achieved with the current technology: new technologies, perhaps based on optical computing, are needed.

CONNECTIONEST MODELS

At about the same time that Hillis was designing massively parallel hardware, massively parallel algorithms were being explored by others who were inspired by the massively parallel architecture of the brain. At a meeting held in 1979 at San Diego, researchers from artificial intelligence, electrical engineering, cognitive psychology, and neuroscience met to explore parallel models of associative memory (Hinton and Anderson, 1981). The common assumption was that many relatively simple processing units, similar to neurons, could be connected together to solve complex computational problems. This approach could be called computing with connections, or connectionist computing (Feldman & Ballard, 1982), and it shows promise as a link between cognitive science (Rumelhart & McClelland, 1986) and neuroscience (Hopfield & Tank, 1986; Sejnowski & Churchland, 1987).

Connectionist models are highly nonlinear and highly difficult to analyze, except in special cases (Ackley, Hinton, & Sejnowski, 1985; Cohen & Grossberg, 1983; Hopfield, 1982). As a consequence, much of the recent research has been empirical. Conventional computers can be used to simulate a connectionist network model with a time penalty that scales with the number of connections in the network. A parallel implementation of a network model would greatly speed up the design and exploration of these models. However, the increase in speed is generally accompanied by a decrease in the flexibility of the hardware. In the extreme case, a VLSI device can be designed with a fixed set of weights that can run a particular network, but no other network, a million times a second. A special purpose network simulator that could handle a wider range of network architectures would be preferred (Bailey & Hammerstrom, 1986; Hecht-Nielsen & Smith, 1986).

All of the currently available general-purpose parallel machines, including the Connection Machine, can be adapted to run connectionist network models even though they were not designed with network models in mind. For coarse-grain architectures, such as the BBN Butterfly and the Intel Cosmic Cube, the processing units can handle many units in the network. The fine-grain architecture of the Connection Machine maps better onto connectionist networks if the processing units are used to represent the connections themselves (Blelloch & Rosenberg, 1987). Which of these parallel machines is the most useful for developing connectionist models will depend as much on the ease with which they can be programmed as on the details of the hardware designs.

Physics of Computation

The last chapter of the book is entitled "New Computer Architectures and Their Relationship of Physics, or Why Computer Science is No Good." Theory in computer science tends to be based on existing hardware, and most existing computers use the von Neumann architecture. The chapter begins appropriately with a quote from von Neumann, who was well aware of the need for a broader science of computation. Von Neumann himself contributed not just to the development of the sequential architecture named after him, but as well to the foundations of cellular automata (von Neumann, 1963), an early parallel architecture that has only recently been exploited (Wolfram, 1983).

Hillis turns to physics in search of models on which to base a new science of parallel computation: "If the universe is a computing machine, then we know of at least some computing machines that have elegant laws." Are there other models of computation that are closer to physics than current theory in computer science? As Hillis points out, the physics of macroscopic systems depends on the law of large numbers, and he speculates that a similar "law of large systems" may someday be found for computing systems based on many simple processing units. The brain is another obvious model of computation in nature, one whose principles we do not yet understand, but whose existence makes it more plausible that Hillis is thinking in the right direction.

Conclusion

The Connection Machine is available today for use by scientists who want to study problems that are amenable to a massively parallel fine-grain computer architecture. New fields of computational science may grow out of the early explorations that the Connection Machine and other parallel computers are making possible. There are already computational branches of physics that depend on supercomputers as much as high-energy experimentalists depend on particle accelerators. Someday there may be branches of psychology and neuroscience that depend on massively parallel computers.

Hillis has produced a lively book as well as a lively machine (with a little help from his friends). It is hoped that the entrepreneurial spirit that built the Connection Machine does not interfere with the scientific spirit that guided its design. The fate of the Connection Machine depends as much on the vagaries of the marketplace, where it now must complete with many other parallel designs, as on the virtues of the architecture itself. The next generation Connection Machine, already in production, will have more memory and more processing power. Parallel computing has an exciting future.

Acknowledgments

I am grateful to Guy Blelloch, Geoffrey Hinton, Charles Rosenberg, David Touretzky, and David Waltz for helpful discussions on parallel architectures.

References

Ackley, D. H., Hinton, G. E., & Sejnowski, T. J. (1983). A learning algorithm for Boltzmann machines. *Cognitive Science*, **9**, 147–169.

Bailey, J. & Hammerstrom, D. (1986). *How to make a billion connections.* Technical Report of the Oregon Graduate Center, *CS/E*–86–007.

Blelloch, G. E. (1986). CIS: A massively concurrent rule-based system. *Proceedings of Fifth National Conference of the American Association for Artificial Intelligence, August* 11–15, 1986, **2**, 735–741.

Blelloch, G. E., & Rosenberg, C. R. (1987). Network learning on the connection machine. *Proceedings of the International Joint Conference on Artificial Intelligence.*

Churchland, P. S. (1986). *Neurophilosophy: Toward a unified science of the mind/brain.* Cambridge, MA: MIT Press.

COHEN, M. A., & GROSSBERG, S. (1983). Absolute stability of global pattern formation and parallel memory storage by competitive neural network. *IEEE Transactions on Systems, Man and Cybernetics*, **13**, 815–826.

FAHLMAN, S. E. (1979). *NETL: A system for representing and using real-world knowledge.* Cambridge, MA: MIT Press.

FELDMAN, J. A., & BALLARD, D. H. (1982). Connectionest models and their properties. *Cognitive Science*, **6**, 205–254.

HECHT-NIELSEN, R., & SMITH, C. A. (1985). *DARPA ADAPT Program Mark IV ADAPT Processor.* TRW Technical Report.

HINTON, G. E., & ANDERSON, J. A. (1981). *Parallel models of associative memory.* Hillsdale, NJ: Erlbaum.

HOPFIELD, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA*, **79**, 2554–2558.

HOPFIELD, J. J., AND TANK, D. W. (1986). Computing with neural circuits: A model. *Science*, **233**, 625–633.

MEAD, C. (1987). *Analog VLSI and Neural Systems.* Reading: Addison-Wesley.

POTTER, J. L. (1985). *The massively-parallel processor.* Cambridge, MA: MIT Press.

PYLYSHYN, Z. W. (1984). *Computation and cognition.* Cambridge, MA: MIT Press.

RUMELHART, D. E., & MCCLELLAND, J. L. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition.* Cambridge, MA: MIT Press.

SEJNOWSKI, T. J., & Churchland, P. S. (1987). Computational Neuroscience (in preparation).

VALIANT, L. G. (1982). A scheme for fast parallel communication. *SIAM Journal of Computing*, **11**, 350–361.

VON NEUMANN, J. (1963). The general and logical theory of automata. In A. H. Taub (Ed.), *J. von Neumann, collected works* (Vol. pp. 5, 288).

WOLFRAM, S. (1983). Statistical mechanics of cellular automata. *Reviews of Modern Physics*, **55**, 601–644.