Field Computation

A Model of Massively Parallel Computation in Electronic, Optical, Molecular and Biological Systems

Bruce J. MacLennan
Department of Computer Science
Ayres Hall
University of Tennessee
Knoxville, TN 37996-1301
(615)974-4406
CSNET: mclennan@utkcs2.cs.utk.edu

Summary

Most current AI technology has been based on propositionally represented theoretical knowledge. We argue that if AI is to accomplish its goals, especially in the tasks of sensory interpretation and sensorimotor coordination, then it must solve the problem of representing *nonpropositional* knowledge. Biological evidence shows that animals use this knowledge in a way very different from digital computation. This suggests that if these problems are to be solved, then we will need a new breed of computers, which we call *field computers*. Examples of field computers are: neurocomputers, optical computers, molecular computers, and any kind of massively parallel analog computer. We claim that the principle characteristic of all these computers is their *massive parallelism*, but we use this term in a special way. We argue that true massive parallelism comes when the number of processors is so large that it can be considered a continuous quantity. Designing and programming these computers requires a new theory of computation, which includes the notion of a *universal field computer*, that is, a field computer that can emulate any other field computer.

The "New" AI

We argue that AI is moving into a new phase characterized by a broadened understanding of the nature of knowledge, and by the use of new computational paradigms.[1]

A sign of this transition is the growing interest in neurocomputers, optical computers, molecular computers and a new generation of massively parallel analog computers. We

1 The forces behind this movement, especially the need to represent nonpropositional knowledge,are discussed in more detail in: B. J. MacLennan, ``Logic for the New AI," in *Aspects of Artificial Intelligence*, J. H. Fetzer (ed.), D. Reidel, 1988, in press, pp. 163-192.

mention briefly the forces driving the development of this ``new'' AI before introducing the idea of a *field computer*, which is intended to be a comprehensive framework for this new paradigm. The ``old'' AI has been quite successful in performing a number of difficult tasks, such as theorem proving, chess playing, medical diagnosis and oil exploration. These are tasks that have traditionally required human intelligence and considerable specialized knowledge. On the other hand, there is another class of tasks in which the old AI has made slower progress, such as speech understanding, image understanding, and sensorimotor coordination. It is interesting that these tasks apparently require less intelligence and knowledge than do the tasks that have been successfully attacked. Indeed, most of these recalcitrant tasks are performed skillfully by animals endowed with much simpler nervous systems than our own. How is this possible?

It is apparent that animals perform (at least some) cognitive tasks very differently from computers. Neurons are slow devices. The well-known ``Hundred Step Rule''[2] says that there cannot be more than about a hundred sequential processing steps between sensory input and motor output. This suggests that nervous systems perform sensorimotor tasks by relatively shallow, but very wide (i.e., massively parallel) processing. Traditional AI technology depends on the digital computer's ability to do very deep (millions of sequential operations), but narrow (1 to 100 processors) processing. Neurocomputing is an attempt to obtain some of the advantages of the way animals do things by direct emulation of their nervous systems. Optical, molecular and the new analog computers may emulate nervous systems at a more abstract level.

Massive Parallelism

The preceding section suggests that the new AI will augment the traditional deep, narrow computation with shallow, wide computation. That is, the new AI will exploit *massive parallelism*. On one hand, massive parallelism means different things to different people; massive parallelism may begin with a hundred, a thousand, or a million processors. On the other hand, biological evidence suggests that skillful behavior requires a very large number of processors, so many in fact that it is infeasible to treat them individually; they must be treated *en masse*. This has motivated us to propose[3] the following definition of massive parallelism:

> A computational system is *massively parallel* if the number of processing elements is so large that it may conveniently be considered a continuous quantity.

That is, a system is *massively* parallel if the processing elements can be considered a

2 J. A. Feldman and D. H. Ballard, ``Connectionist models and their properties,'' *Cognitive Science*, Vol. 6, pp. 205-254, 1982.

3 B. J. MacLennan, ``Technology-Independent Design of Neurocomputers: The Universal Field Computer,'' *Proceedings of the IEEE First Annual International Conference on Neural Networks*, in press, San Diego, CA, June 21-24, 1987.

continuous mass rather than a *discrete ensemble*.

How large a number is large enough to be considered a continuous quantity? That depends on the purpose at hand. A hundred is probably never large enough; a million is probably always large enough; a thousand or ten thousand may be enough. One of the determining factors will be whether the number is large enough to permit the application of continuous mathematics, which is generally more tractable than discrete mathematics.

We propose this definition of massive parallelism for a number of reasons.
First, as noted above, skillful behavior seems to require significant neural *mass*. Second, we are interested in computers, such as optical computers and molecular computers, for which the number of processing elements *is* effectively continuous. Third, continuous mathematics is generally easier than discrete mathematics. And fourth, we want to encourage a new style of thinking about parallelism. Currently, we try to apply to parallel machines the thought habits we have acquired from thinking about sequential machines. This strategy works fairly well when the degree of parallelism is low, but it will not scale up. One cannot think individually about the10^{20} processors of a molecular computer. Rather than postpone the inevitable, we think that we should begin now to develop a theoretical framework for understanding massively parallel computers. The principal goal of our research is to develop such a theory.

Field Transformation Computers

Our aim then is to develop a way of looking at massive parallelism that encompasses a variety of implementation technologies, including neural networks, optical computers, molecular computers and a new generation of analog computers. What these all have in common is the ability to process in parallel amounts of data so massive as to be considered a continuous quantity. This suggests that we structure our theory around the idea of a *field*, i.e. a continuous (dense) ensemble of data. We have in mind both scalar fields (such as potential fields) and vector fields (such as gravitational fields). Any operation on such a field, either to produce another field or to produce a new state of the field, can be considered massively parallel, since it operates on all the elements of the field in parallel. Indeed, it would not be feasible to serialize the processing of the field; modest degrees of parallelism cannot cope with the large number of field elements.

The goal of our research is the exploration of *field transformation computers*, that is, computers characterized by the ability to perform (in parallel) transformations on scalar and vector fields. Field computers may be designed for special purposes; this has been the case with field computers to date, and we expect it to be the case in the future. In these computers, devices implementing field transformations (such as filters and convolutions) are assembled to solve a small class of problems (e.g., pattern recognition). On the other hand, our experience with digital computation has shown us the value of *general purpose* or *programmable* computers. This architectural feature permits one computer to perform a variety of digital computations, which eliminates the need to construct special purpose devices, and speeds implementation of digital algorithms.

The foregoing observations suggest that general purpose *field* computers will be similarly valuable. In these the connections between field transformation units and field storage units are programmable, thus facilitating their reconnection for a variety of purposes.

We cannot build into a general purpose field computer every transformation we might need. Instead we must choose a set of primitive operations that permit the programming of all others. How can such a set of primitive operations be chosen? How can we be guaranteed that we have provided all the necessary facilities? For digital computers this question is answered in part by computability theory. For example, this theory shows us how to construct a *universal Turing machine*, which, given an appropriate program, can emulate any Turing machine. Although the universal Turing machine is hardly a practical general purpose computer, consideration of it and other universal machines shows us the kinds of facilities a digital computer must have in order to be universal. There follows the hard engineering job of going from the theoretically sufficient architecture to the practically necessary architecture.

Can the same be accomplished for field computers? Is there a *universal field computer* that can emulate any field computer? If there is such a thing, then we can expect that it may form a basis for practical general purpose field computers in much the same way that Turing machines do for digital computers.

Elsewhere[4] we have presented a general theory of field computation and described one notion of a universal field computer. In particular, we have show that with a certain set of built in field transformations we can implement (to a desired degree of accuracy) any field transformation in a very wide class. This is analogous to the result from Turing machine theory: The universal Turing machine allows us to implement (to a desired degree of accuracy) any function in a wide class (now known as the *Turing computable functions*).

The phrase `to a desired degree of accuracy' appears in both of the preceding statements. What does it mean? For the Turing machine it means that a given accuracy (e.g., precision or range of argument) can be achieved by providing a long enough tape. For the digital computer it means that computations are normally performed to a given precision (e.g., the word length), and that finite increments in the desired precision require finite increments in the resources required (e.g., additional registers and memory cells for double and multiple precision results, or stack space for recursion). The case is much the same for the universal field computer. Finite increments in the desired accuracy of a field transformation will require finite increments in the resources used (such as field transformation and storage units).

---

4 MacLennan, ``Technology-Independent Design of Neurocomputers,'' loc cit. See also: B. J. MacLennan, ``Field Computation and Nonpropositional Knowledge,'' Naval Postgraduate School Computer Science Dept. Tech. Rept. NPS52-87-040, Sept. 1987.

There are a number of theoretical bases for a universal field computer. We have investigated designs based on Fourier analysis, interpolation theory and Taylor's theorem, all generalized for field transformations. The previously cited paper presents a design based on Taylor's theorem. There are no doubt as many principles upon which universal field computers can be based as there are bases for universal digital computers (e.g., Turing machines, Markov algorithms, λ calculus, Post productions).

Conclusions

We have argued that AI is moving — and must move — into a new phase that recognizes the role of nonpropositional knowledge in intelligent behavior. We also argued that the ``new'' AI must make use of massive parallelism to achieve its ends. We proposed a definition of massive parallelism, namely that the number of processing elements can be taken as a continuous quantity. We believe that this definition will encourage the development of the necessary theoretical basis for neurocomputers, optical computers, molecular computers, and a new generation of analog computers. We claimed that these computing technologies can be profitably viewed as *field computers*, computers that operate on entire fields of data in parallel. We discussed the importance of general purpose field computers, and related them to universal field computers. Finally, we claimed that a universal field computer could be based on a generalization of Taylor's theorem for field transformations.

Acknowledgements

The research reported herein was supported in part by the Office of Naval Research under contract N00014-87-WR-24037.