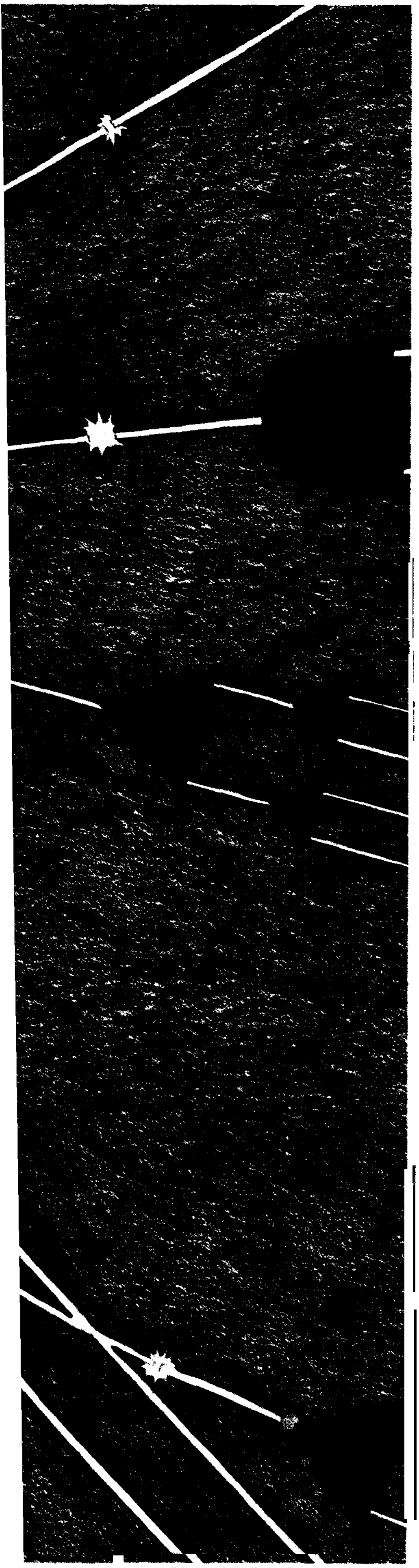

Illustration by Lisa Ann Hebert

FEATURE ARTICLE

A Neural Network Approach to Artificial Intelligence

Using *a Neural Network* for dealing *with Real- World Data*

by Christopher Ciarcia

When you think about modern digital computers, you see fast, dedicated machines that tear through their programmed algorithms at terrific speeds. What you don't see are flexible, adaptable devices that can quickly react to changing circumstances. Unfortunately, the complex demands of many modern applications cry out for flexible solutions rather than mechanical brute force. What we're really looking for, when you get right down to it, is a brain just like ours that can be harnessed to a particular task. Since brains working away in bubbling chemical stews are still the stuff of horror films, however, what can we do to solve advanced application problems?

We can create neural networks of our own! We can emulate our own self-learning, interactive awareness, by creating an artificial neural network (ANN) that reproduces the major components of our own central nervous system. To provide the best possible emulation, our ANN should include sections that correspond to the cerebral hemisphere, which handles sensor processing, abstract thought, and gross motor control; the diencephalon, which is composed of the thalamus forinformationexchangebetweenthe cerebralcortexand therestof thebrain, and the hypothalamus for regulation of autonomic and endocrine systems; the cerebellum for fine motor control; the brainstem, which acts as an interface for the spinal cord and input for hearing, balance, and taste; and the spinal cord, which is the biological analog of a computer bus connecting to our peripheral nervous system.

We know that our human brain works, so why not create a computer neural network that mimics our own brain's vast web of interconnected neuronic structure? All we need to do is study how our brain's estimated 10 billion neurons and its 10^{14} interconnections are configured and reproduce that structure on our home PC. Simple, isn't it? Just map the brain's complexity onto your computer.

For better or for worse, even if your computer is a Cray X/MI' you won't be able to fully duplicate the complexity of a human brain. What you can do, however, is gain some insight into the "thinking" process while modeling a system that is capable of some real work.

By simulating the basic features of our brain's individual processing units (PU), "neurons" with their associated decisionand learningrules,and structuring "neuron nets" with interconnectionsthatemulatethe"learned-stored experience" of the brain's synapses, we can construct an Artificial Neural Network which is "a dynamical system with the topology of a directed graph that can carry out informationprocessingbymeansofitsstate response to continuous or initial input (thedecision and learningrules), with thenodesbeingcalledprocessingunits (neurons) and the directed links or information channels where memory resides being called interconnects (synapses)." [1]

Using a computer architecture similar to our brain, we can develop a system that is highly parallel, highly integrated, noise tolerant, has graceful degradation, contains simple processing units, is memory intensive, associative in nature, and taught, not *programmed.*

Don't Call it a Computer

The ANN is distinguishable from theordinarydigitalcomputerbecause of its radical departure from standard internal organization. Most of today's computer designs call for a separation of a computer's memory and its processor, with a communications link in between. While this arrangement provides for tight control, it has speed limitationsduetobusaddressinteractions. Theneural-netapproachavoids this bottleneck by mimicking the brain's own structure by storing experience and memory in the nodal processor's interconnects. Table 1 shows how the two architectures differ.[2]

The ANN is not built like a "normal" computer. Instead, its memory lies within the path between two elements. It is not stored separate from the "CPU," but is considered an intrinsic part of the information processing. This "informational" connectivity between elements also has arbitrary dimensionality. Any linkage configuration is allowed. And most of all, patterns and response rules are generated internally by correlating inputs and outputs, so the system is not programmed, but taught.

The essential components defining the Artificial Neural Network are its "architecture" which controls the flow of information within the net, its "transfer function/decision rule" which determines how input information and interconnection weights are used to calculate an output value, and its "learning rule" which defines how interconnection weights are adjusted while educating the network.

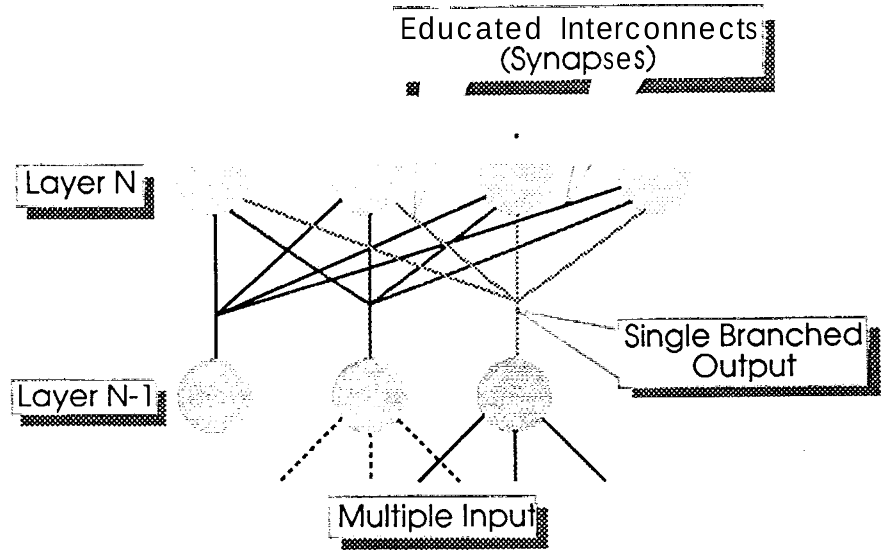

Figure 1 -Each node *at a given level of the neural network can* connect, *with varying* levels of 'weight,' to each of the *nodes at the next level of the network. This* system *emulates the synaptic connections of a biological brain.*

A System of Modified Inputs

A typical processing unit (PU) operationisshowninFigure2. Within the PU, inputs are modified by the interconnection weights. Given a positive input, a positive interconnect weight will be excitatory, a negative interconnect weightwillbeinhibitory, and a zero-valued interconnect weight interrupts the current link. This filtered input is then used by the transfer function to calculate an output value. Then both the input and output information areused by thelearningrule to "teach" or adapt the weights. These modified weights then alter the processing element's future operation by filtering the input data in a new way.

The Transfer Function

The typical transfer function is composed of two parts: an input operator, *f()*, that combines the inputs and interconnection weights to form a single value ready for discriminatory action; and a discrimination function