EE 7000
INSTRUCTOR: DR. KAK

PROJECT NUMBER 1

"THE GAME OF LIFE"
SIMULATION AND EXPLORATIONS OF CELLULAR AUTOMATA

BY: N. STEPHAN KINSELLA
MARCH 16, 1988

History of LIFE

The game of LIFE was invented in 1970 by the distinguished British mathematician John Horton Conway, and it was first publicly described by Martin Gardner, in his "Mathematical Games" column in the October 1970 issue of Scientific American. LIFE--a subset of a class of entities called cellular automata--is described by a very simple set of rules.

A programmer supplies an initial state, and from then on the rules determine each next state. In this way, LIFE is similar to neural networks--in that both need only a set of rules (the analog of the weights and interconnections in neural nets) and an initial state to proceed and find further states. Furthermore, the rules of LIFE are very simple; yet the resulting patterns and their interactions are very complex and have intriguing prospects. As Stephen Wolfram, head of the Center for Complex Systems Research at the University of Illinois said, "the most important fact about automata is that they show complex behavior can have a simple cause."

These interesting characteristics, LIFE's similarity to neural networks, and the fact that LIFE can itself be simulated on a specialized neural network makes it worth investigating as a subject in a neural networks class.

The Rules of LIFE

LIFE is defined to exist as discrete cells on an infinite, two-dimensional plane with discrete time. At any discrete moment of time, each cell can be "Alive" or "Dead". (Or "on" or "off"; or a binary "1" or "0".) In such a two-dimensional configuration, each cell has eight neighbors (two horizontal, two vertical, four diagonal).

The rules are simple: at any given time t, a cell counts the number of alive neighbors. If the number of alive neighbors is less than two, the cell will be dead from "starvation". If the number of alive neighbors is more than three, the cell will be dead due to overcrowding. If the cell has exactly three neighbors alive, it will be alive at t+1 no matter what. If the cell has exactly two neighbors alive, it will remain in the same state as it was at time t.

Let A = number of alive neighbors of a cell at time t.

A < 2 ------------ cell is dead at t + 1 from starvation

A > 3 ------------ cell is dead at t + 1 from overcrowding
A = 3 ------------ cell is alive at t + 1
A = 2 ------------ cell is in same state at t + 1 as in t

Representing a LIFE universe on a Neural Network

Each LIFE cell could be represented with a multi-level threshold gate (see below, left). But a simpler way is using a combination of the standard types of threshold gates that we have been using in class to simulate neural networks (see below, right).

The simulation of LIFE

The simulation written in Turbo Pascal on the IBM PC (see program listing at end of report) was used to give a real-time demonstration of the variety and complexity possible in LIFE. The program should also give insight to the possibilities open if a more complex neural net is devised--since such a simple configuration can lead to demonstrable, complex results, a more complex configuration should lead to very complex, possibly useful types of systems.

My program allows simulation of a LIFE plane which is 40 horizontal cells by 25 vertical cells. The program also has built into it various interesting shapes which the user can call up and investigate or observe--such as eaters, gliders, glider guns, and other nontrivial patterns.

Exploring LIFE

From the preceding figure, it is obvious that LIFE is merely a very simplified, specialized subcase of a generalized neural network. But it differs from the typical model we have been studying in class in that feedback is involved (only in the case where A = 2).

Another feature of LIFE of interest to our previous studies is its shift invariance. A shape that is stable at one place on the LIFE plane will be stable at any other place on the plane, since then entire plane is homogenous. The only exception to this arises in a practical simulation on a finite computer or neural net, in which the plane will have to have either "dead" edges or a wrap-around, toroidal geometry.

Not only are there stable patterns in LIFE, as in neural nets, there are stable oscillatory patterns, with period 1, 2, 3, even up to cycles or generations (found so far).

The discovery of a shape called the glider was a turning point. The glider (see below, right) is a 5-celled shape that moves in any diagonal direction across the plane (see my program or Conway's chapter--included). Conway believed that they could be used to make a pattern of light that could compute--just like a computer. A stream of gliders could, Conway reasoned, represent a stream of bits. The problem was, however, to find LIFE's analog of a computer's clock--to emit pulses of gliders. Finally, Bill Gosper, from MIT's AI lab, found the Gosper glider gun (see below--or my program or Conway's chapter--both included). This oscillating, periodic shape spits out gliders in the southeast direction every generations.

Bill Gosper's Glider Gun

Conway went on to show that--with various combinations and collisions of gliders, glider guns, eaters, blocks, ships and other esoteric entities--all the building blocks for a computer exist in LIFE (see Conway's chapter for more detail).

The subject of cellular automata--of which LIFE is a subset--has received much attention. Ed Fredkin, a professor of computer science at MIT, was so interested in LIFE and the phenomenon of cellular automata the he set up a research group called Information Mechanics Group. "A medical researcher might investigate a substance by injecting it into many different animals," he said. "We investigate such physical concepts as the conservation of energy and matter, or the passage of time, by watching how they work in different universes."

In the summer of 1986, a conference on cellular automata was held at MIT. The use of automata as models for exploration and prediction in such diverse fields as economics, neurophysiology, art, pattern recognition, parallel computing, circuit design, evolution and physics was discussed.

The complexity of LIFE and cellular automata in general--and therefore the neural networks which can represent them--has convinced some scientists that automata might even be able to create life. After all, von Neumann has already shown that automata can reproduce (and this can be shown specifically with LIFE as well).

Given a large enough LIFE plane, with a random scattering of the basic shapes--gliders, glider guns, blocks, blinkers, ships, eaters--it is conceivable that a type of evolution would occur. That is, shapes that are unsuccessful at surviving would of course die out. Others better at living--due to their structure, reproductive abilities, and defense mechanisms--would of course tend to dominate regions of the LIFE plane. And when any unexpected event happened--such as a stray glider entering a previously successful pattern--this would be like a mutation occuring.

Of course, as in evolution in our world, most mutations would be harmful. But, due to chance, some would be beneficial. In this way, more efficient and complex "animals" could come to exist--possibly to the point where they are actually intelligent and conscious.

As Conway said, "It is no doubt true that on a large enough scale LIFE would generate living configurations. Genuinely living. Evolving, reproducing, squabbling over territory. Writing Ph.D. theses. On a large enough board there's no doubt in my mind this sort of thing would happen."

A situation could arise in which a simulation of a LIFE plane had progressed so far that <u>conscious</u> <u>life</u> has actually arisen. Then one might wonder whether the engineer running the simulation has the moral right to <u>turn</u> <u>it</u> <u>off</u>--fearing that to do so might possibly be akin to murder. Unfortunately--due to the lamentable dearth of any explicit philosophy whatsoever in the field of engineering, except for a cursory glance at the scientific method--it is not considered to be the role of engineering to answer such questions. Or even, I fear, to ask them.

<u>Further implication of LIFE</u>

The most widely studied (models of) neural nets calculate, find next states, and have states consisting of the "on" or "off" values of all neurons at any given time t. But it is the structure of a neural net that makes it unique--its interconnections, weights, time delays, and thresholds. This does not depend upon the current state of the machine. If a neural net stores information, it is stored in the weights; the actual values of the outputs of the neurons used in different states are merely used to locate or access the stored information.

Although LIFE can be represented easily on a neural net, as already shown, with each cell needing at most three neurons or threshold gates--one primary neuron and two hidden neurons to activate the inhibitory inputs of the primary neuron in the cases of starvation or overcrowding--there are subtle differences with possibly far-reaching effects.

As explained, a neural net stores information in the weights, while using the states of neurons to access or manipulate stored states. But LIFE does somewhat the opposite: since the entire LIFE plane is homogeneous, information is and can only be stored in the states themselves--in the alive and dead states which stabilize to form gliders, glider guns, blocks, and other shapes too complex to describe or perhaps not found yet--and the weights and rules are used to manipulate the states.

Therefore, on a neural net, when a new pattern is stored, the T matrices can be retrained. But when it has been retrained, it is just as if the new pattern had been included in the original set. Also, there is no significant relational quality between stored states (save hamming distance) except that they are stable states. The only unique characteristic about a stored pattern in a typical neural net is its own, distinct pattern.

By considering this problem in light of the afore-mentioned subtle difference of LIFE's inner-workings, we can pose some interesting questions. (One must keep in mind that, whatever is possible with a LIFE universe, is possible on a neural net since it can simulate a LIFE universe, and is by the same reasoning also possible on a normal digital computer--although the neural net would be much faster at the simulation.)

On a LIFE plane, if one could devise a method of storing, retrieving and manipulating data (which is conceivable), it would have to be in some form of gliders and two-dimensional, tangible patterns.

Since these stored memories are two-dimensional--and this is the crucial difference between LIFE's storing method and the typical neural net's--they can have relationships to one another that memories cannot have in a neural net where the memories are really represented by weights.

The two-dimensional locations on the plane, including distance, size and orientation constitute meaningful relationships between stored memories or groups of memories. These relationships could represent: relative importance of memories, alphabetical order, certainty, order of arrival, age of memories, the types of relationships used in expert systems, or any number of relational concepts.

Further investigations of these properties could yield worthwhile results.