CAM-BRAIN
The Evolutionary Engineering of a Billion Neuron Artificial Brain by 2001 which Grows/Evolves at Electronic Speeds inside a Cellular Automata Machine (CAM)

Hugo de Garis

Brain Builder Group, Evolutionary Systems Department, ATR Human Information Processing Research Laboratories, 2-2 Hikarida, Seika-cho, Soraku-gun, Kansai Science City, Kyoto, 619-02, Japan, tel. + 81 7749 5 1079, fax. + 81 7749 5 1008, degaris@hip.atr.co.jp

Abstract

A "Darwin Machine" is defined to be a piece of hardware that evolves its own architecture at electronic speeds. A Darwin Machine is an example of "Evolutionary Engineering", which in turn is defined to be the building of complex structures and dynamics using evolutionary methods. The author believes that evolutionary techniques will dominate 21st century engineering, especially molecular scale (nanotech) engineering, with its gargantuan number of components and complexities. This paper reports on the second year of an ambitious 8 year research project which aims to implement a type of Darwin Machine in the form of a cellular automata based artificial brain with a billion neurons by 2001, which grows/evolves at (nano-)electronic speeds inside a Cellular Automata Machine - ATR's so-called "CAM-Brain Project". The basic idea is to use cellular automata based neural networks which grow under evolutionary control at (nano-)electronic speeds. The states of the cellular automata (CA) cells and the CA state transition rules can be stored cheaply in gigabytes of RAM. By using state of the art cellular automata machines, e.g. MIT's "CAM8" machine ($40,000), which can update 200 million CA cells a second, it may be technically feasible within a year or so to evolve artificial nervous systems containing a thousand neurons, and within 5 years, a million neurons. By the end of the current research project, i.e. 2001, it should be possible using nano-scale electronics to grow/evolve artificial brains containing a billion neurons and upwards. This is our aim.
Keywords

Evolutionary Engineering, Neural Networks, Genetic Algorithms, Cellular Automata, Cellular Automata Machines (CAMs), Nano-Electronics, Darwin Machines.

1. Introduction

Following on from the abstract above, to understand how CAs can be used to grow/evolve neural networks, imagine a 2D CA trail which is 3 cells wide (e.g. Fig. 2). Down the middle of the trail, send growth signals. When a growth signal hits the end of the trail, it makes the trail extend, or turn left, or right, or split etc., depending upon the nature of the signal. It is the sequence of these signals (fed continuously over time into an initial short trail) that is evolved. This sequence of growth signals is the "chromosome" of a genetic algorithm, and it is this sequence that maps to a cellular automata network. When trails collide, they form "synapses". Once the CA network has been formed in the initial "growth phase", it is later used in a second "neural signaling phase". Neural signals move along CA-based axons and dendrites, and across synapses etc. The CA network is made to behave like a conventional artificial neural network. The outputs of some of the neurons of the complex recurrent networks which result can be used to control complex time dependent behaviors whose fitnesses can be measured. These fitness values can be used to drive the evolution. By growing/evolving thousands of neural net modules and their interconnections in an incremental evolutionary way, it will be possible to build artificial brains. We expect that within a few years, a new field will be established, based partly on this work, called simply "Brain Building". According to the CAM developers at MIT, it is likely that the next generation of CAMs will achieve an increase in performance of the order of thousands, within 5 years. However, to be able to evolve a billion neuron artificial brain by 2001, a "nano-CAM" machine will need to be developed. To this end, we are collaborating with an NTT researcher who has developed a nanoscale electronics device, who wants to combine them to behave like molecular scale cellular automata machines.

In the summer of 1994, a two dimensional CAM-Brain simulation was completed which required 11,000 hand crafted state transition rules. It was successfully applied to the evolution of maximizing the number of synapses, outputting an arbitrary constant neural signal value, outputting a sine wave of a desired arbitrary period and amplitude and to the evolution of a simple artificial retina which could output the vector velocity of a "white line" which "moved" across an array of "detector" neurons. Next, will be to port the 2D simulation to a CM5 supercomputer to pursue more interesting and ambitious
evolutionary experiments. Work on the 3D simulation should be completed early in 1995, and is expected to take about 60,000 rules. The Brain Builder Group of ATR took possession of one of MIT's CAM8 machines in the fall of 1994. An attempt will be made to port the rules of the 3D simulation to this machine. If this is not possible, then a CAM9 machine will be designed specifically for CAM-Brain.

The following paragraphs now present the CAM-Brain Project in more detail. As mentioned in the extended abstract, the "CAM-Brain Project" is an 8 year research project at ATR labs in Kyoto, Japan, which intends to build an artificial brain containing billions of artificial neurons. The complexity of such an artificial brain will make it largely undesignable, so a (directed) evolutionary approach called "evolutionary engineering" is being used. Neural networks based on cellular automata [Codd 1968], can be grown and evolved at electronic speeds inside state of the art cellular automata machines, e.g. MITs "CAM8" machine, which can update 200 million cells per second [Toffoli & Margolus 1990]. Since RAM is cheap, gigabytes of RAM can be used to store the states of the CA cells used to grow the neural networks. CA based neural net modules are evolved in a two phase process. Three cell wide CA trails are grown by sending a sequence of growth signals (extend, turn left, turn right, fork left, fork right, T fork) down the middle of the trail. When an instruction hits the end of the trail it executes its function. This sequence of growth instructions is treated as a chromosome in a Genetic Algorithm [Goldberg 1989] and is evolved. This sequence maps to a CA network. When trails collide, they form "synapses". Once the CA network is grown, it is used as a neural network in a second neural signaling phase. Some of the neural signals can be tapped to control some process and the fitness of the control can be measured. This fitness is used to drive the evolution. Once gigabytes of RAM and electronic evolutionary speeds can be used, genuine brain building, involving millions and later billions of artificial neurons, becomes realistic, and should become concrete within a year or two. The CAM-Brain Project should revolutionize the fields of neural networks and artificial life, and in time help create a new specialty called "Brain Building", with its own conferences and journals.

This paper consists of the following sections. Section 2 describes briefly the idea of "Evolutionary Engineering", of which the CAM-Brain Project is an example. Section 3 describes how neural networks can be based on cellular automata [Codd 1968], and evolved at electronic speeds. Section 4 presents some of the details of CAM-Brain's implementation. Section 5 shows how using cellular automata machines will enable millions of artificial neural circuits to be evolved to form an artificial brain. Section 6 discusses changes needed for the 3D version of CAM-Brain. Section 7 deals with future work, and section 8 summarizes.
2. Evolutionary Engineering

Evolutionary Engineering is defined to be "the art of using evolutionary algorithms (such as genetic algorithms [Goldberg 1989]) to build complex systems." This paper reports on the idea of evolving cellular automata based neural networks at electronic speeds inside cellular automata machines. This idea is a clear example of evolutionary engineering. Evolutionary engineering will be increasingly needed in the future as the number of components in systems grows to gargantuan levels. Today's nano-electronics for example, is researching single electron transistors (SETs) and quantum dots. Probably within a decade or so, humanity will have full blown nanotechnology (molecular scale engineering), which will produce systems with a trillion trillion components [Drexler 1992]. The potential complexities of such systems will be so huge, that designing them will become increasingly impossible. However what is too complex to be humanly designable, might still be buildable, as this paper will show. By using evolutionary techniques (i.e. evolutionary engineering), it is often still possible to build a complex system, even though one does not understand how it functions. This arises from the notion of the "complexity independence" of evolutionary algorithms, i.e. so long as the (scalar) fitness values which drive the evolution keep increasing, the internal complexity of the evolving system is irrelevant. This means that it is possible to successfully evolve systems which function as desired, but which are too complex to be designable. The author believes that this simple idea (i.e. the complexity independence of evolutionary algorithms) will form the basis of most 21st century technologies (dominated by nanotechnology [Drexler 1992]). Thus, evolutionary engineering can "extend the barrier of the buildable", but may not be good science, because its products tend to be black boxes. However, confronted with the complexity of trillion trillion component systems, evolutionary engineering may be the only viable method to build them.

3. Cellular Automata Based Neural Networks

Building an artificial brain containing billions of artificial neurons is probably too complex a task to be humanly designable. The author felt that brain building would be a suitable task for the application of evolutionary engineering techniques. The key ideas are the following. Use evolutionary techniques to evolve neural circuits in some electronic medium, so as to take advantage of electronic speeds. The medium chosen by the author was that of cellular automata (CA) [Codd 1968], using special machines, called "Cellular Automata Machines (CAMs)" , which can update hundreds of millions of CA cells a second [Toffoli & Margolus 1990]. CAMs can be
used to evolve the CA based neural networks at electronic speeds. The states of the cellular automata cells can be stored in RAM, which is cheap, so one can have gigabytes of RAM to store the states of millions of CA cells. This space is large enough to contain an artificial brain. MIT's Information Mechanics Group (Toffoli and Margolus) believe that within a few years it will be technically possible to update a trillion CA cells in about 0.1 nanoseconds [p221, Toffoli & Margolus 1990]. Thus, if CA state transition rules can be found to make CA behave like neural networks, and if such CA based networks prove to be readily evolvable, then a potentially revolutionary new technology becomes possible. The CAM-Brain Project is based on the above ideas and fully intends to build artificial brains before the completion of the project in 2001. The potential is felt to be so great that it is likely that a new specialty will be formed, called "Brain Building".

For the first 18 months of the CAM-Brain Project, the author simulated a two dimensional version of CAM-Brain on a Sparc 10 workstation, hand coding over 11,000 (eleven thousand) CA state transition rules to get the CA-based neural nets to evolve. This work was completed in the summer of 1994. The 2D version was used briefly (before work on the 3D version was started) to undertake some evolutionary tests, whose results will be presented in the next section. The 2D version served only as a feasibility and educational device. Since trails are obliged to collide in 2D, the 2D version was not taken very seriously. Work was begun rather quickly on the more interesting 3D version almost immediately after the 2D version was ready. Proper evolutionary tests will be undertaken once the 3D version is ready, which should be by early 1995.

To begin to understand how cellular automata [Codd 1968] can be used as the basis for the growth and evolution of neural networks, consider Fig. 1 which shows an example of a 2D CA state transition rule, and Fig. 2 which shows a 2D CA trail, 3 cells wide. All cells in a CA system update the state of their cells synchronously. The new state of a given cell depends upon its present state and the states of its nearest neighbors. Down the middle of the 3 cell wide CA trail, move "signal or growth cells" as shown in Fig. 2.

```
Top
Left | Center | Right
Bottom
```

CTRBL -> Cnext  9,18,16,11.5 -> 4

**Fig. 1** A 2D CA State Transition Rule
Fig. 2  Signal Cells Move Along a Cellular Automata Trail

As an example of a state transition rule which makes a signal cell move to the right one square, consider the right hand most signal cell in Fig. 2, which has a state of 5. The cell immediately to its right has a state of 1, which we want to become a 5. Therefore the 2D state transition rule to turn the 1 into a 5 is 1,2,2,2,5→5. These signal or growth cells are used to generate the CA trails, by causing them to extend, turn left or right, split left or right, and Tsplt. When trails collide, they can form synapses. It is the sequence of these signal cells which determines the configuration of the CA trails, thus forming a CA network. It is these CA trails which later are used as neural network trails of axons and dendrites. Neural signals are sent down the middle of these CA trails. Thus there are two major phases in this process. Firstly, the CA trails are grown, using the sequence of signal cells. Secondly, the resulting CA trail network is used as a neural network, whose fitness at controlling some system can be measured and used to evolve the original growth sequence. To make this more explicit, it is the sequence of growth cells which is evolved. By modifying the sequence, one alters the CA network configuration, and hence the fitness of the configuration when it functions as a neural net in the second phase. From a genetic algorithm (GA) point of view, the format of the GA "chromosome" is the sequence of integers which code for the signaling or growth instructions. By mutating and crossing over these integers, one obtains new CA networks, and hence new neural networks. By performing this growth at electronic speeds in CAMs, and in parallel, with one CAM per GA chromosome, and attaching a conventional programmable microprocessor to each CAM to measure the user defined fitness of the CA based neural circuit, one has a means to evolve large numbers of neural modules very quickly. Using CAMs to evolve neural circuits, is an example of a type of machine that the author labels a "Darwin Machine", i.e. one which evolves its own structure or architecture. A related idea of the author concerns the concept of "Evolvable Hardware (EHW)" [de Garis 1993] where the software instructions used to configure programmable logic devices (PLDs) are treated as chromosomes in a Genetic Algorithm [Goldberg 1989]. One then rewrites the circuit for each chromosome.
4. Further Details

This section provides further details on the implementation of the CA based neural networks. There were three kinds of CA trails in CAM-Brain, labeled dendrites, excitatory axons and inhibitory axons, each with their own states. Whenever an axon collided with a dendrite or vice versa, a synapse was formed. When a dendrite hit an excitatory/inhibitory axon or vice versa, an excitatory/inhibitory synapse was formed. An inhibitory synapse reversed the sign of the neural signal value passing through it. An excitatory synapse left the sign unchanged. Neural signal values ranged between -240 and +240 (or their equivalent CA states, ranging from 100 to 580). The value of a neural signal remained unchanged when it was in an axon, but as soon as it crossed a synapse into a dendrite, the signal value (i.e., signal strength) began to drop off linearly with the distance it had to travel to its receiving neuron. Hence the signal strength was proportional to the distance between the synapse and the receiving neuron. Thus the reduction in signal strength acted like a weighting of the signal by the time it reached the neuron. But, this distance is evolvable, hence indirectly, the weighting is evolvable. CAM-Brain is therefore equivalent to a conventional artificial neural network, with its weighted sums of neural signal strengths. However, in CAM-Brain there are time delays as signals flow through the network. When two or three dendrite signals collide, they sum their signal strengths (within saturated upper or lower bounds). It soon became noticeable that there were many many ways in which collisions between CA trails could occur. So many, that the author became increasingly discouraged. It looked as though it would take years of handcoding the CA state transition rules to get CAM-Brain to work. The intention was to have rules which would cover every possible collision possibility. Eventually a decision was made to impose constraints on the ways in which CA trails could grow. The first such constraint was to make the trails grow on a grid of cells or squares (cubes) on a side. This process (called "gridding") sharply reduced the number of collision types. It also had a number of positive side effects. One was that in the neural signaling phase, neural signals arrived synchronously at junction points. One no longer needed to have to handcode rules for phase delays in neural signaling summation. By further imposing that different growth cells advanced the length of the trails by the same number of squares, one could further reduce the number of collision types. With synchrony of growth, synchrony of signaling and gridding, it was possible to cover all possible types of collisions. Nevertheless, it still took over 11000 rules to achieve this goal, and this was only for the 2D version. The 3D version is expected to take about 60,000 rules, but due to the experience gained in working on the 2D version, and to the
creation of certain software productivity tools, the 3D version should be completed with only about 6 months work, i.e. by early 1995.

Considering the fact that the 2D version took 11,000 rules, it is impossible in this short paper to discuss all the many tricks and strategies that were used to get CAM-Brain to work. That would require a book (something the author is thinking seriously about writing). However, some of the tricks will be mentioned here. One was the frequent use of "gating cells", i.e. cells which indicated the direction that dendrite signals should turn at junctions to head towards the receiving neuron. To give these gating cells a directionality, i.e. a "leftness" or "rightness", special marker cells had to be circulated at the last minute, after the circuit had stabilized. Since some trails were longer than others, a sequence of delay cells were sent through the network after the growth cells and before the marker cells. Without the delay cells, it was possible that the marker cells would pass before synapses were formed.

Once the 2D simulation was completed (before the CAM8 was delivered) several brief evolutionary experiments using the 2D version were undertaken. The first, was to see if it would be possible to evolve the number of synapses. Fig. 4 shows the result of an elite chromosome evolved to give a large number of synapses. In this experiment, the number of synapses increased steadily. It evolved successfully. The next experiment was to use the neural signaling to see if an output signal (tapped from the output of one of the neurons) could evolve to give a desired constant value. This evolved perfectly. Next, was to evolve an oscillator of a given arbitrary frequency and amplitude, which did evolve, but slowly (it took a full day). Finally, a simple retina was evolved which output the two component directional velocity of a moving "line" which passed (in various directions) over a grid of 16 "retinal neurons". This also evolved but even more slowly. The need for greater speed is obvious.

The above experiments are only the beginning. The author has already evolved (not using CAs) the weights of recurrent neural networks as controllers of an artificial nervous system for a simulated quadruped artificial creature. Neural modules called "GenNets" [de Garis 1990, 1991, 1994] were evolved to make the creature walk straight, turn left or right, peck at food, and mate. GenNets were also evolved to detect signal frequencies, to generate signal frequencies, to detect signal strengths, and signal strength differences. By using the output of the detector GenNets, it was possible to switch motion behaviors. Each behavior had its own separately evolved GenNet. By switching between a library of GenNets (i.e. their corresponding evolved weights) it was possible to get the artificial creature to behave in interesting ways. It could detect the presence and location of prey, predators and mates and take appropriate action, e.g. orientate, approach, and
eat or mate, or turn away and flee. However, every time the author added another GenNet, the motion of the simulated creature slowed on the screen. The author's dream of being able to give a robot kitten some thousand different behaviors using GenNets, could not be realized on a standard monoprocessor workstation. Something more radical would be needed. Hence the motivation behind the CAM-Brain Project.

5. A Billion Neurons in a Trillion Cell CAM by 2001

Fig. 3 shows some estimated evolution times for 10 chromosomes over 100 generations for a Sparc 10 workstation, a CAM8, and a CAM2001 (i.e. a CAM using the anticipated electronics of the year 2001) for a given application. In the current 2D version of CAM-Brain, implemented on a Sun Sparc 10 workstation, it takes approximately 3.4 minutes to grow a stable cellular automata network consisting of only four neurons. It takes an additional 3.2 minutes to perform the signaling on the grown network, i.e. a total growth-signaling time to measure the fitness of a chromosome of 6.6 minutes. This time scales linearly with the number of artificial neurons in the network. If one uses a population of 10 chromosomes, for 100 generations, the total evolution time (on a Sparc 10) is 100*10*6.6 minutes, i.e., 110 hours, or 4.6 days. This is obviously tediously slow, hence the need to use a CAM. MIT's CAM8 [Toffoli & Margolus 1990] can update 25 million cellular automata cells per second, per hardware module. A CAM8 "box" (of personal computer size) contains eight such modules, and costs about $40,000. Such boxes can be connected blockwise indefinitely, with a linear increase in processing capacity. Assuming an eight module box, how quickly can the above evolution (i.e. 100 generations, with a population size of 10) be performed? With eight modules, 200 million cell updates per second is possible. If one assumes that the 2D CA space in which the evolution takes place is a square of 100 cells on a side, i.e., 10,000 cells, then all of these cells can be (sequentially) updated by the CAM8 box in 50 microseconds. Assuming 1000 CA clock cycles for the growth and signaling, it will take 50 milliseconds to grow and measure the fitness of one chromosome. With a population of 10, and 100 generations, total CAM8 evolution time for a four neuron network will be 50 seconds, i.e. about one minute, which is roughly 8000 times faster. Using the same CAM8 box, and a 3D space of a million cells, i.e. a cube of 100 cells on a side, one could place roughly 40 neurons. The evolution time will be 100 times as long with a single CAM8 box. With 10 boxes, each with a separate microprocessor attached, to measure the fitness of the evolved network, the evolution time would be about eight minutes. Thus for 1000 neurons, the evolution would take about 3.5 hours, quite an acceptable figure. For a million neurons, the evolution time would be nearly five months. This is still a workable figure. Note, of course, that these estimates are lower bounds.
They do not include the necessary human thinking time, and the time needed for sequential, incremental evolution, etc. However, since the CAM-Brain research project will continue until the year 2001, we can anticipate an improvement in the speed and density of electronics over that period. Assuming a continuation of the historical doubling of electronic component density and speed every two years, then over the next eight years, there will be a 16-fold increase in speed and density. Thus the "CAM-2001" box will be able to update at a rate of 200*16*16 million cells per second. To evolve the million neurons above will take roughly 13.6 hours. Thus to evolve a billion neurons, will take about 19 months, again a workable figure. But, if a million neurons can be successfully evolved, it is likely that considerable interest will be focused upon the CAM-Brain approach, so that more and better machines will be devoted to the task, thus reducing the above 19-month figure. For example, with 100 machines, the figure would be about two months. The above estimates are summarized in Figure 3. These estimates raise some tantalizing questions. For example, if it is possible to evolve the connections between a billion artificial neurons in a CAM2001, then what would one want to do with such an artificial nervous system (or artificial brain)? Even evolving a thousand neurons raises the same question.

<table>
<thead>
<tr>
<th>Sparc10</th>
<th>CAM8</th>
<th>CAM8</th>
<th>CAM8</th>
<th>CAM8</th>
<th>CAM2001</th>
<th>CAM2001</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000 CA cells</td>
<td>10000 CA cells</td>
<td>1 million CA cells</td>
<td>25 million CA cells</td>
<td>25 billion CA cells</td>
<td>25 billion CA cells</td>
<td>25 trillion CA cells</td>
</tr>
<tr>
<td>4 neurons</td>
<td>4 neurons</td>
<td>40 neurons</td>
<td>1000 neurons</td>
<td>1 million neurons</td>
<td>1 million neurons</td>
<td>1 billion neurons</td>
</tr>
<tr>
<td>1 Sparc10</td>
<td>1 CAM8</td>
<td>10 CAM8s</td>
<td>10 CAM8s</td>
<td>10 CAM8s</td>
<td>10 CAM2001s</td>
<td>100 CAM2001s</td>
</tr>
<tr>
<td>4.6 days</td>
<td>50 seconds</td>
<td>8 minutes</td>
<td>3.5 hours</td>
<td>5 months</td>
<td>13.6 hours</td>
<td>2 months</td>
</tr>
</tbody>
</table>

**Fig. 3  Evolution Times for Different Machines & CA Cell, Neuron & Machine Numbers**

One of the aims of the CAM-Brain research project is to build an artificial brain which can control 1000 behaviors of a "robot kitten" (i.e. a robot of size and capacities comparable to a kitten) or to control a household "cleaner robot". Presumably it will not be practical to evolve all these behaviors at once. Most likely they will have to be evolved incrementally, i.e., starting off with a very basic behavioral repertoire and then adding (stepwise) new behaviors. In brain circuitry terms, this means that the new neural modules will have to connect up to already established neural circuits. In practice, one can imagine placing neural bodies (somas) external to the established nervous system and then evolving new axonal and dendral connections to it. The CAM-Brain Project hopes to create a new tool to enable serious investigation of the new field of "incremental evolution." This field is still rather virgin territory at the time of
writing. This incremental evolution could benefit from using embryological ideas. For example, single seeder cells could be positioned in the 3D CA space under evolutionary control. Using handcrafted CA "developmental or embryological" rules, these seeder cells could grow into neurons ready to emit dendrites and axons [de Garis 1992]. The CAM-Brain Project, if successful, should also have a major impact on both the field of neural networks and the electronics industry. The traditional preoccupation of most research papers on neural networks is on analysis, but the complexities of CAM-Brain neural circuits, will make such analysis impractical. However, using Evolutionary Engineering, one can at least build/evolve functional systems. The electronics industry will be given a new paradigm, i.e. evolving/growing circuits, rather than designing them. The long term impact of this idea should be significant, both conceptually and financially.

6. The 3D Version

The 3D version is a conceptually simple extension of the 2D version. Instead of 4 neighbors, there are 6 (i.e. North, East, West, South, Top, Bottom). Instead of 6 growth instructions as in the 2D version (i.e. extend, turn left, turn right, split extend left, split extend right, split left right), there are 15 in the 3D version. A 3D CA trail cross section consists of a center cell and 4 neighbor cells, each of different state or color (e.g. red, green, blue, brown). Instead of a turn left instruction being used as in the 2D case, a "turn green" instruction is used in the 3D case. The 15 3D growth instructions are (extend, turn red, turn green, turn blue, turn brown, split extend red, split extend green, split extend blue, split extend brown, split red brown, split red blue, split red green, split brown blue, split brown green, split blue green). A 3D CA rule thus consists of 8 integers of the form CTSENWB-->Cnew. The 3D version will enable dendrites and axons to grow past each other, and hence reach greater distances. The weakness with the 2D version is that collisions in a plane are inevitable, which causes a crowding effect, whereby an axon or dendrite cannot escape from its local environment. This is not the case with the 3D version, which is topologically quite different. A 3D version is essential if one wants to build artificial brains with many interconnected neural modules. The interconnectivity requires long axons/dendrites.

FIG. 5 shows some recent results in 3D simulation. A space of 3D CA cells (48*48*48 cubes) was used. A single short 3D CA trail was allowed to grow to saturate the space. One can already sense the potential complexity of the neural circuits that CAM-Brain will be able to build. In 3D, it is likely that each neuron will have hundreds, maybe thousands of synapses, thus making the circuits highly evolvable due to their smooth fitness landscapes (i.e. if you cut one synapse, the effect is minimal when there are hundreds of them per neuron).
7. Future Work

At the time of writing (January 1995), the author is completing the simulation of the 3D version, working on the many thousands of rules necessary to specify the creation of synapses. So far, 30,000 3D rules have been implemented, and it is quite probable that the figure may go as high as double that. Since each rule is rotated 24 ways (6 ways to place a cube on a surface, then 4 ways to rotate that cube) to cater to all possible orientations of a 3D trail, the actual number of rules placed in the (hashed) rule base will be more than a million. Specifying these rules takes time, and constitutes the bulk of time spent doing CAM-Brain. Hence the immediate future work will be to complete the simulation of the 3D version. Probably, this will be done before the summer of 1995. Next, will be to see if it is possible to port these 3D rules to the MIT CAM8 machine, which is now waiting on the author's desk. If this is not possible, then the CAM8 will serve as a "tutorial" machine upon which to base a newer version (a "CAM9") specifically designed with CAM-Brain's specifications in mind. If it is not possible, then it is likely that a "mini" CAM-Brain will be implemented on CAM8. In other words, in the case of the CAM8, the CAM-Brain model will be adapted (simplified) to the machine. In the case of CAM9, the machine will be adapted (built) according to the (CAM-Brain) model.

Concurrently with this work on trying to port the rules to CAM8, etc, will be an ongoing series of experiments using the 3D simulation software to evolve many kinds of neural modules, plus their interconnections. Just how these modules should be interconnected is an open and interesting question, which should stimulate a lot of research. Once CAM-Brain is ready, the brain builder research community will have a new tool to begin investigating these fascinating issues. The "CAM9" (or whatever new name it will be given so as to distinguish it from MIT's next version of their CAM) will be implemented in state-of-the-art VLSI, probably in collaboration with NTT. However, further down the road, will be the attempt to design a "CAM10" or "CAM2001" based on nanoelectronics. The brain builder group at ATR is collaborating with an NTT researcher who wants to build nano-scale cellular automata machines. With the experience of designing and building a "CAM9", a nanoscale CAM should be "buildable" which will have several orders of magnitude greater performance.

Further research aims are to use CAs to make Hebbian synapses capable of learning. One can also imagine the generation of artificial "embryos" inside a CA machine, by having CA rules which allow an embryological "unfolding" of cell groups, with differentiation, transportation, death, etc resulting in a form of neuro-morphogenesis similar to the way
in which biological brains are built. The author calls this CA based neuro-morphogenetic research project "CAMbryo".

8. Summary

The CAM-Brain Project at ATR, Kyoto, Japan, intends to build/grow/evolve an artificial brain of a billion artificial neurons at (nano-) electronic speeds inside Cellular Automata Machines (CAMs) by the year 2001. Quoting from a paper by Margolus and Toffoli of MIT's Information Mechanics group, "We estimate that, with integrated circuit technology, a machine consisting of a trillion cells and having an update cycle of 100 pico-second for the entire space will be technologically feasible within 10 years" (i.e., by 2000) [Margolus and Toffoli 1990]. In a trillion 3D CA cells (cubes), one can place a billion artificial neurons. Such an artificial nervous system will be too complex to be humanly designable, but it may be possible to evolve it, and incrementally, by adding neural modules to an already functional artificial nervous system. In the summer of 1994, a 2D simulation of CAM-Brain using over 11000 CA state transition rules was completed, and initial tests showed the new system to be evolvable. By early 1995, a 3D simulation will be completed, and the resulting (probably 60,000) 3D CA rules will attempt to be ported to a CAM8 machine from MIT ($40,000) which our lab took possession of in the fall of 1994. If the CAM-Brain Project is successful, it will revolutionize the field of neural networks and artificial life, because it will provide a powerful new tool to evolve artificial brains with billions of neurons, and at electronic speeds. The CAM-Brain Project will thus produce the first Darwin Machine, as defined in the abstract. The author is confident that in time a new specialty will be established, based partly on the ideas behind CAM-Brain. This specialty is called simply "Brain Building". At the time of writing, there are now 3 centers in the world doing work similar to CAM-Brain. The first is at ATR, The second is led by Todd Kaloudis at MIT (email tjk@mit.edu), and the third is in Switzerland at the EPFL under Eduardo Sanchez (email eduardo@lssun.epfl.ch).

References

[de Garis 1990] Hugo de Garis, "Genetic Programming:
Modular Evolution for Darwin Machines," ICNN-90WASH-DC, (Int. Joint Conf. on Neural Networks), January 1990, Washington DC, USA.


Fig. 4 2D CAM-Brain CA Network

Fig. 5 3D CAM-Brain CA Network