Explorations in Artificial Life (special issue of AI Expert),
pages 4-8, September, 1995. Miller Freeman.

An Introduction To Artificial Life

Moshe Sipper
Logic Systems Laboratory
Swiss Federal Institute of Technology
IN-Ecublens
CH 1015, Lausanne, Switzerland

Can a machine reproduce? Can software be evolved? How are sophisticated robots built to function in a human environment? Can an ecological system be created within a computer? How do flocks of birds fly?

These are some of the issues confronted by researchers in Artificial Life (ALife), *a young field on the rise that has been gaining acceptance over the past few years.*

Can a machine reproduce? This question was posed by mathematician John von Neumann in the early 1950s and explored by him before his untimely death in 1957. Specifically he asked whether an artificial machine could create a copy of itself, which in turn could create more copies (in analogy to nature).

Von Neumann wished to investigate the *logic* necessary for reproduction. He was not interested, nor did he have the tools, in building a working machine at the bio-chemical or genetic level. Remember that at the time DNA had not yet been discovered as the genetic material in nature.

To conduct a formal investigation of the issue, von Neumann used a model conceived by his colleague, the mathematician Stanislaw Ulam. The model, called *Cellular Automata*, consists of a large grid of cells (similar to a chess board), each possessing a certain color at a given moment (every color represents a specific state). All cells change colors simultaneously such that the color of a cell at the next time step depends only on its color at the current time step and the colors of its four immediate neighbors (north, south, east, and west). The principle that guides color transformations is applied identically to all cells and is referred to as the *rule*. For example, a simple rule for a two-color (black/white) grid sets the color of a cell at the next time step to black if it has an even number of black neighbors, and white if it has an odd number of black neighbors.

A *machine* in the cellular automata model is a collection of cells that can be regarded as operating in unison. For example, if a square configuration of four black cells exists, that appears at each time step one cell to the right, then we say that the square acts as a machine moving right.

Von Neumann used this simple model to describe a universal constructing machine, which can read *assembly instructions* of *any* given machine, and construct that machine accordingly. These instructions are a collection of cells of various colors, as is the new machine after being assembled - indeed, any compound element on the grid is simply a collection of cells.

Von Neumann's universal constructor can build any machine when given the appropriate assembly instructions. If these consist of instructions for building a universal constructor, then the machine can create a duplicate of itself; that is, reproduce. Should we want the offspring to reproduce as well, we must *copy* the assembly instructions and attach them to it. In this manner, von Neumann showed that a reproductive process is possible in artificial machines. The actual proof is quite elaborate and detailed in a book completed posthumously by von Neumann's colleague, Arthur Burks [8]. A much simpler self-replicating structure (though without universal constructing capabilities) was demonstrated by computer scientist Chris Langton, more than three decades later [4].

One of von Neumann's main conclusions was that the reproductive process uses the assembly instructions in two distinct manners: as interpreted code (during actual assembly), and as uninterpreted data (copying of assembly instructions to offspring). During the following decade, when the basic genetic mechanisms began to unfold, it became clear that nature had "adopted" von Neumann's conclusions. The process by which assembly instructions (that is, DNA) are used to create a working machine (that is, proteins), indeed makes dual use of information: as interpreted code and as uninterpreted data. The former is referred to in biology as *translation*, the latter as *transcription*.

Life-As-It-Could-Be

This description demonstrates the underlying approach of ALife. The field draws researchers from different disciplines such as computer science, physics, biology, chemistry, economics, philosophy, and so on. Artificial Life, as defined by Chris Langton, one of its founders, is a field of study devoted to understanding life by attempting to abstract the fundamental dynamical principles underlying biological phenomena, and recreating these dynamics in other physical media, such as computers, making them accessible to new kinds of experimental manipulation and testing [5]. While biological research is essentially *analytic*, trying to break down complex phenomena into their basic components, ALife is *synthetic*, attempting to construct phenomena from their elemental units. As such, ALife complements traditional biological research by exploring new paths in the quest toward understanding the grand, ancient puzzle called life.

The use of the term "artificial" signifies that the systems in question are human-made; that is, the basic components were not created by nature through evolution. However, the higher-level phenomena are completely genuine. The reproductive process detailed by von Neumann is as real as that carried out in nature. The difference is solely in the basic components: artificial cells verses live cells. The fact that veritable phenomena are observed serves as a basis for ALife research - the underlying belief asserts that life is not necessarily carbon-based but can consist of other elements as well. As put forward by Langton, in addition to providing new ways for studying biological phenomena associated with life here on Earth, *life-as-we-know-it*, ALife lets us extend our studies to the larger domain of "bio-logic" of possible life, *life-as-it-could-be* [5].

The issues raised by ALife researchers pertain to existing biological phenomena as well as to complex systems in general. Thus ALife pursues a two-fold goal: increasing our understanding of nature and enhancing our insight into artificial models, thereby providing us with the ability to improve their performance. An example of the first goal is seen in von Neumann's research described earlier. An example of the second goal is John Koza's work involving software development through evolution [3].

Computer programs are written today by humans (programmers). Over the years, we have been witness to a steady rise in software complexity as the tasks computers are set forth to handle become more complicated. The increase in computational power has increased our appetite for more elaborate applications, evident by the onset of such areas as artificial intelligence and neural networks.

The Importance Of Evolution

Koza's method, termed *genetic programming*, is based on John Holland's research on genetic algorithms during the 1970s and 1980s. While a programmer develops a single program, attempting to perfect it as much as possible, genetic programming involves a *population* of programs. The initial population, referred to as the first generation, consists of randomly created programs. The following generations are formed by evolution so that in time the population comes to consist of better (fitter) programs. Each generation is created by applying genetic operators to the previous generation programs. These operators are known in biology as *crossover* and *mutation*. Two parent programs are selected at random, such that a better parent has a higher probability of being selected. The parents reproduce, creating a daughter program consisting of a mixture of the parental genetic material (crossover) along with a small amount of copying errors (mutation). The selection and reproduction process continues until the next generation is formed, in (abstract) analogy to nature: a given generation consists of different creatures (programs) whose chances of survival are in relation to their fitness. The better (fitter) a creature (program), the higher its probability of survival, and in time the population comes to consist of better creatures.

A major issue that must be addressed is what composes a good (fit) program. This question, while highly complex in nature, has a simpler answer in the context of genetic programming - fitness is defined by the programmer, in accordance with the particular problem at hand. For example, if we seek a program for operating a robotic arm that can stack blocks in a specified order, then fitness measures the quality of a particular arrangement.

Note that evolution proceeds without human intervention. After the task is set, an initial population is generated at random (the process is actually not entirely random since the initial population depends to some extent on the task at hand), and evolution treads along until a satisfying solution is found. Koza has successfully applied the genetic programming method to several problems detailed in [3].

An evolutionary method is advantageous not only in solving difficult problems but also in offering better adaptability. Current computer programs in existence today are well known for their "brittleness" - when an unforeseen event occurs, program failure is imminent. This basic problem is one of the major causes of high software development and maintenance costs. Evolution, however, offers the possibility of adaptation to a dynamic environment - when an unforeseen event occurs, the system can evolve; that is, adapt to the new situation, in analogy to nature.

Evolution is central to ALife research. One of the major open problems facing scientists today is the origin of life: How did the first self-replicating organisms appear, an event considered to be a precursor to evolution, leading to the astounding variety of species found on Earth today. Von Neumann dealt with the logic of reproduction more than four decades ago, demonstrating its feasibility in non-carbon-based machines. The underlying conditions necessary for self-reproduction in nature are under intense investigation today. ALife can aid this research by exploring new paths, complementing those of traditional biology.

The Importance Of Emergence

Another process predominating ALife systems is that of *emergence*, where phenomena at a certain level arise from interactions at lower levels. In physical systems, temperature and pressure are examples of emergent phenomena. They occur in large ensembles of molecules and are due to interactions at the molecular level. An individual molecule possesses neither temperature nor pressure, which are higher-level, emergent phenomena.

ALife systems consist of a large collection of simple, basic units whose interesting properties are those that emerge at higher levels. One example is von Neumann's model, where the basic units are grid cells and the observed phenomena involve composite objects consisting of several cells (for example, the universal constructing machine). Another example is Craig Reynolds' work on flocking behavior [7].

Reynolds wished to investigate how flocks of birds fly, without central direction (that is, a leader). He created a virtual bird with basic flight capability, called a "boid". The computerized world was populated with a collection of boids, flying in accordance with the following three rules:

- *Collision Avoidance:* Avoid collisions with nearby flock-mates.
- *Velocity Matching:* Attempt to match velocity with nearby flock-mates.
- *Flock Centering:* Attempt to stay close to nearby flock-mates.

Each boid comprises a basic unit that "sees" only its nearby flock-mates and "flies" according to the three rules.

These three rules served as sufficient basis for the emergence of flocking behavior. The boids flew as a cohesive group, and when obstacles appeared in their way they spontaneously split into two subgroups, without any central guidance, rejoining again after clearing the obstruction. The boids algorithm has been used to produce photorealistic imagery of bat swarms for the feature motion pictures *Batman Returns* and *Cliffhanger*.

Reynolds' model demonstrates the basic architecture of ALife systems - a large number of elemental units, relatively simple, interacting with a small number of nearby neighbors, with no central controller. High-level, emergent phenomena resulting from these low-level interactions are observed. Although Reynolds' boids are artificial, the flocking behavior is as real as that observed in nature (this point was also noted for von Neumann's reproductive process described earlier).

The underlying principles of ALife stand at the core of Rodney Brooks' work [1]. During the past decade, he has been involved in the construction of robots that can function in a (noisy) human environment; for example, traveling in a building and collecting garbage. The robots possess "brains" comprised of a hierarchy of layers, each one performing a more complex function than the one underneath. The first layer handles obstacle avoidance. The second is responsible for wandering behavior; that is, randomly circulating within the environment (room, building). This layer does not concern itself with obstacle avoidance, since this issue is handled by the previous layer. Higher-level layers can subsume the role of lower levels by suppressing their outputs. However, lower levels continue to function as higher levels are added. This method, dubbed the *subsumption architecture*, roughly resembles our own brains where primitive layers handle basic functions (for example, respiration) and high-level layers handle more complex functions (for example, abstract thinking). Brooks' scheme allows incremental construction of robots by adding to existing (operational) layers, thus enabling a sort of robotic evolution.

Each layer consists of behavioral modules that communicate asynchronously with no central controller. For example, the first layer (obstacle avoidance) includes sensory modules, danger-detecting modules, and motor system modules. The operation of the system is reminiscent of Marvin Minsky's approach delineated in his book *The Society of Mind*. He describes the operation of the brain in terms of an ensemble of *agencies*, each responsible for a simple functionality. The agencies communicate among themselves to reach a "decision", and the total (emergent) effect is essentially the operational mind. Brooks envisions a future when robots will aid in our daily lives - for example, a robot housemaid.

Brooks' method for building sophisticated robots demonstrates the ALife approach, which is fundamentally different than that of traditional artificial intelligence (AI). AI employs a top-down methodology where complex behaviors (for example, chess playing) are identified and an attempt is made to build a system that presents all the details of said behavior. ALife operates in a bottom-up manner, starting from simple elemental units, gradually building its way upwards through evolution, emergence, and development.

Another difference between ALife and AI pertains to the issues investigated. Whereas AI has traditionally concentrated on complex human functions such as chess playing, text comprehension, medical diagnosis and so on, ALife concentrates on basic natural behaviors, emphasizing survivability in complex environments. According to Brooks, an examination of the evolution of life on earth reveals that most of the time was spent developing basic intelligence. The elemental faculties evolved enable mobility in a dynamic environment and sensing of the surroundings to a degree sufficient to achieve the necessary maintenance of life and reproduction. The issues dealt with by AI appeared only very recently on the evolutionary scene (a mere few thousands of years) and mostly in humans. This suggests that problem-solving behavior, language, expert knowledge, and reason are all rather simple once the essence of being and reacting is available. The idea is expressed in the title of one of Brooks' papers, *Elephants Don't Play Chess*, suggesting that these animals are nonetheless highly intelligent and able to survive and reproduce in a complex, dynamic environment.

In A Virtual World

Can open-ended evolution be constructed within a computer, proceeding without any human guidance? This issue was addressed by Thomas Ray who devised a virtual world called *Tierra*, consisting of computer programs that can undergo evolution [6]. In contrast to genetic programming where fitness is defined by users, the Tierra creatures (programs) receive no such direction. Rather, they compete for the natural resources of their computerized environment, namely CPU time and memory. Since only a finite amount of these are available, the virtual world's natural resources are limited, as in nature, serving as the basis for competition between creatures.

Ray modeled his system on a relatively late period of earth's evolution known as the Cambrian era, roughly 600 million years ago. The beginning of this period is characterized by the existence of simple, self-replicating organisms, marking the onset of evolution that resulted in the astounding diversity of species found today. For this reason, the era is also referred to as the Cambrian explosion. Ray did not wish to investigate how self-replication is attained, but rather wanted to discover what happens after its appearance on the scene. He inoculated his system with a single, self-replicating organism, called the "Ancestor", which is the only engineered (human-made) creature in Tierra. He then set his system loose, and the results obtained were quite provocative: An entire ecosystem had formed within the Tierra world, including organisms of various sizes, parasites, hyper-parasites, and so on. The parasites, for example, that had evolved are small creatures that use the replication code of larger organisms (such as the ancestor) to self-replicate. In this manner they proliferate rapidly without the need for the excess reproduction code.

Ray argues that open-ended evolution is a highly powerful tool. The human eye, for example, is a superb vision machine "discovered" by evolution through many millions of years. Ray has recently suggested creating a network-wide reserve on the Internet for the digital Tierra creatures. He hopes that by increasing the scale of the system new phenomena may arise that have not been observed on a single computer. Useful programs may appear, analogous to the human eye, which could be extracted and used by us.

ALife And Evolution

The issue of evolution in nature has received renewed attention over the past two decades. Darwin's fundamental theory, while still sound today, is in need of expansion. For example, one well-known principle is that of natural selection, usually regarded as an omnipotent force capable of molding organisms into perfectly adapted creatures. The work of Stuart Kauffman has revealed that other factors can influence evolution besides natural selection [2]. He demonstrated that certain complex systems tend to self-organize; that is, order can arise spontaneously. A major conclusion is that such order constrains evolution, to the point where natural selection cannot divert its course.

Another principle of Darwin's theory is that of gradualism - small phenotypic changes accumulate slowly in a species. Paleontological findings discovered over the years have revealed a different picture - long periods of relative phenotypic stasis, interrupted by short bursts of rapid changes. This phenomenon has been named *punctuated equilibria* by biologist Stephen Jay Gould. While a full explanation does not yet exist, the phenomenon has been recently observed in a number of ALife works, suggesting that it may be inherent in certain evolutionary systems.

ALife offers opportunities for conducting experiments that are extremely complicated in traditional biology or not feasible at all. ALife complements biological research, raising the possibility of joint ventures leading to valuable new scientific discoveries. ALife also holds potential for developing new technologies: software evolution, sophisticated robots, ecological monitoring tools, educational systems, and so on. This exciting discipline combines both scientific endeavor and applied research. Although still young, the field has produced exciting results that hold promise for a bright future.

References

[1] R. A. Brooks. New approaches to robotics. *Science*, 253(5025):1227–1232, September 1991.

[2] S. A. Kauffman. *The Origins of Order.* Oxford University Press, New York, 1993.

[3] J. R. Koza. *Genetic Programming.* The MIT Press, Cambridge, Massachusetts, 1992.

[4] C. G. Langton. Self-reproduction in cellular automata. *Physica D*, 10:135–144, 1984.

[5] C. G. Langton. Preface. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, volume X of *SFI Studies in the Sciences of Complexity*, pages xiii–xviii, Redwood City, CA, 1992. Addison-Wesley.

[6] T. S. Ray. An approach to the synthesis of life. In C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, volume X of *SFI Studies in the Sciences of Complexity*, pages 371–408, Redwood City, CA, 1992. Addison-Wesley.

[7] C. W. Reynolds. Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics*, 21(4):25–34, July 1987.

[8] J. von Neumann. *The Theory of Self-Reproducing Automata.* University of Illinois Press, Illinois, 1966. Edited and completed by A.W. Burks.