Computer Models of Complex Adaptive Systems

Melanie Mitchell
Santa Fe Institute
1399 Hyde Park Road
Santa Fe, New Mexico 87501
U.S.A.

In *New Scientist*, February 13, 1993

The science of computation and the sciences of complexity were born and raised together in this century, brash newcomers in the panoply of more traditional scientific topics and techniques. Their joint birth and mutual unfolding was no coincidence: like astronomy and the telescope, biology and the microscope, or elementary-particle physics and the accelerator, the detailed study of complex systems had to await the proper scientific instrument, and that was (and is) the electronic computer. For systems such as biological cells, brains, human intelligence, the human immune system, natural ecologies, and economies, computer modeling has become the third arena of science, supplementing, or, in some cases, even replacing the more traditional approaches of mathematical theory and direct experimentation. The above are all examples of *complex adaptive systems*, in which complex behavior of the system as a whole emerges from the interaction of large numbers of simple components, and in which the system is able to *adapt*—that is, to automatically improve its performance (according to some measure) over time in response to what has been been encountered previously. The importance of understanding such systems is enormous: many of the most serious challenges facing humanity—e.g., environmental sustainability, economic stability, the control of disease—as well as many of the hardest scientific questions—e.g., the nature of intelligence, the origin of life—will require a deep understanding of complex adaptive systems. However, for such systems, the traditional approaches are often too difficult or even infeasible to apply, which means that computer modeling is often the method of choice for studying such systems.

The relationship is not at all one-sided, however, with computation serving merely as a tool for the scientific exploration of complex systems. Since the advent of computers there has been a strong feedback from natural complex adaptive systems to computer science, with natural systems actually *inspiring* new computational methods for solving technological problems. Ideas from systems such as the brain or biological evolution have inspired some of the newest and most promising computational methods, such as neural networks and genetic algorithms, which are now being applied to previously intractable practical problems including the automatic analysis of complex data, computer security, and the creation of autonomous learning systems, all of which will have tremendous significance for science and technology.

In short, there is a symbiosis in which computer models act as exploratory tools for the study of natural complex adaptive systems, and, in turn, as natural complex adaptive systems become better understood, they often lead to inspiration for new computational technologies. This symbiosis has given rise to a number of interdisciplinary and overlapping fields including “adaptive computation”, “neural computation”, and “artificial life”.

The best way to give a flavor of this symbiosis and of the excitement of these new fields is to survey a few samples of work in this area.

The Genetic Algorithm—Evolution on a Computer

The genetic algorithm is perhaps the most salient example of the melding of the natural and computational worlds. Developed by John Holland in the 1970s and described in his pioneering book “Adaptation in Natural and Artificial Systems”, the GA (as it is called by practitioners) is a simplified version of biological evolution that can run on a computer.

The essential mechanisms of biological evolution are *reproduction with variation* and *selection*. Variation occurs when an organism reproduces—the offspring is not a carbon-copy of its parent, but usually contains its parent’s genes with some random mutations thrown in. When reproduction is sexual, the main source of variation is the *recombination* of the two parents’ genetic material (their *chromosomes*) to produce the offspring. Selection is the process, sometimes known as “survival of the fittest”, in which organisms that are better suited to their environment tend to survive and produce more offspring. These two simple mechanisms (though a bit oversimplified here) are the centerpiece of the Darwinian theory of evolution by natural selection, and are widely believed to account for the existence of highly successful and adapted species of organisms throughout the world.

In a GA, a population of “chromosomes” lives in the memory of a computer (each chromosome corresponds to a single “organism”). In the simplest form of a GA, each chromosome is simply a string of 1’s and 0’s (“bits”) that encodes a solution to some problem—say, a program for constructing a robot. The fitness of a chromosome is determined by testing how good a solution it is to the problem at hand—e.g., how successful the resulting robot is at performing some task. Once the fitness of all chromosomes in the population has been determined, selection and variation take place. The fittest chromosomes are selected to reproduce (i.e., be copied), with variation occurring in the copies through random mutations (some 1’s are changed to 0’s, and vice versa) and “sexual recombination” (two chromosomes together produce offspring by exchanging subsequences of 1’s and 0’s). These offspring, produced by the fittest parents, make up the new population, and now are ready for their fitnesses to be tested and to create their own offspring in this computational version of the unending cycle of life. This process, usually started with a randomly generated initial population, and repeated for many “generations” of offspring, often results in the automatic discovery of excellent solutions to the given problem.

Simple as the GA sounds, variations of this basic idea have been successfully applied to a large number of technological problems, such as optimization and design problems in engineering, automatic evolution of computer programs, and machine and robot learning. Versions of the GA have also been used to simulate evolutionary processes in computer models of complex adaptive systems such as immune systems, economies, and ecologies, as will be described below. The GA, in various forms, has thus played a central role in research on complex adaptive systems.

A Computational Model of the Immune System

One particularly interesting application of the GA is as a tool for modeling evolutionary processes in the immune system. In the immune system, specialized molecules called *antibodies* are produced by white blood cells in order to attack foreign invaders (*antigens*) such as viruses or bacteria. Antibodies can attack only those antigens to which they can chemically bind very strongly, and thus an antibody has to have a very specialized design to successfully attack a given antigen. So how does the immune system know ahead of time what specialized antibodies to make? The answer is, it doesn't; rather, the correct antibodies are generated by an evolutionary process known as *clonal selection*. A very large number of antibodies are generated, and ones that succeed in binding with some strength to an invading antigen are copied extensively through a complex process. In this process, the cells responsible for generating the most successful antibodies (i.e., the ones that bind most strongly to the attacking antigen) are triggered to generate more of the same, but with *variation*—the new antibodies are imperfect copies of the original successful antibody. The same process repeats, with the most successful of the *new* antibodies being amplified in number (with small variations), and over a period of approximately a week, the system usually succeeds in generating a large number of antibodies that are perfectly designed to wipe out the intruder. This process is strikingly similar to the process of variation and selection in biological evolution.

Stephanie Forrest, a computer scientist at the University of New Mexico, and Alan Perelson, an immunologist at the Los Alamos National Laboratory, have joined forces to apply the GA to the computer modeling of the clonal-selection process. In their simplified model, antibodies and antigens are represented as bit strings, and the strength of "binding" between an antibody and an antigen is the number of bits that match in the two strings. (This is, of course, an extremely simplified version of a complex chemical process). The GA starts out with a population of randomly generated antibodies, which is faced with a large population of different antigens. The object is to evolve a population of antibodies that bind sufficiently strongly to individual antigens; a large proportion of the antigen population should be covered in this way.

After an initial random population of antibodies is generated, various subsets of antigens are chosen, and antibodies are selected to reproduce in proportion to how strongly they bind to some antigen in the chosen subset. The most successful antibodies are copied many times, and each is subjected to some small number of mutations. Over time, a set of antibodies is evolved that does a good job of covering the antigen population.

Forrest and Perelson are using this model to study, among other things, how sufficient diversity can be maintained in the population of antibodies. Unlike most other applications of GAs, in which the desired result is to evolve a *single* highly fit string, the antibody population works together cooperatively to cover a diverse and changing set of antigens. Thus the antibody population must remain diverse and adaptable. Forrest and Perelson, along with Robert Smith, a computer scientist at the University of Alabama, and Brenda

Javornik, a graduate student at the University of New Mexico, have developed a new form of the GA that succeeds in maintaining the necessary diversity. This research not only has the potential of leading to a better understanding of such processes in the real immune system, it also is relevant to the many other applications of GAs that require the maintenance of diversity in the population of potential solutions.

Forrest and Perelson are also studying ways to apply their models of immune systems to problems in computer security and computer-virus detection. As computer viruses become more and more sophisticated and hard to detect by traditional means, it may be that the best approach is to give each computer its own "immune system" to ward off such intruders. Though research in this area is currently at a preliminary stage, its significance for computer security could be enormous. This is an excellent example of the feedback between the computer modeling of natural complex adaptive systems and the development of new computational methods based on ideas from such systems.

Artificial Adaptive Agents in Economics

Genetic algorithms have also been applied to the computer modeling of another class of complex adaptive systems—economies. Traditionally, economic structures (e.g., stock markets, local economies, global economies) have been modeled as arising from the actions of economic agents (e.g., individual consumers, small businesses, multinational corporations) who are perfectly rational (i.e., always obey the laws of logic) and who have perfect knowledge of all other agents. Such assumptions are, of course, very unrealistic, but are made in order to make mathematical descriptions more feasible. However, in recent years the introduction of computer modeling into economics has allowed economists to construct and study more realistic models.

One important problem in economics that can be addressed by computer modeling is understanding the formation of efficient markets. A *market* is any place where individuals can trade under specified trading rules (e.g., a local supermarket, where prices are set by sellers according to a number of factors such as supply and demand; or an auction market, where prices are set by competitive bidding on the part of the buyers). A market is *efficient* to the extent that no single buyer or seller in the market could be made better off without some other buyer or seller being made worse off. Market efficiency is thus a kind of global "equilibrium" in which the prices buyers are willing to pay and the prices at which sellers are willing to sell cannot be changed without someone paying more for a good than they feel it is worth or selling it at less than they feel it is worth.

How can such a result emerge from the decentralized, self-interested behavior of individual agents with limited rationality and knowledge? Answering this question is of increasing importance to social and international policy, especially given the recent transitions in a number of countries to free-market policies.

John Miller, an economist and decision scientist at Carnegie-Mellon University, has developed a computer model to study the formation of efficient markets. Miller's artificial economy consists of a number of "artificial adaptive agents", each with its own economic (buying or selling) strategy. These strategies are represented as bit strings, for use by a genetic algorithm. Each agent has some limited amount of information about the value of goods being sold and about how other agents are likely to bid in various circumstances. Each agent's bidding strategy is a rule for deciding, given the agent's limited information, how much to bid on a given good. The "fitness" of an agent depends on its profit over time (i.e., how its sum total of money or goods increases or decreases over time). Thus, the only goal for each agent is to maximize its own profit, and selection in the GA is based only on this criterion. (Of course, the effectiveness of an agent's bidding strategy depends on what strategies all the other agents have, so this is really a problem of *co-evolution* among a large number of individual agents.) Yet even though fitness was based only on individual interest, the GA was able, starting from a population of agents with completely random bidding strategies, to evolve a collection of agents whose bidding strategies, taken together, constitute an optimally efficient market.

Models such as Millers can help economists understand the conditions under which efficient markets can emerge. This is important both for theoretical economics as well as real-world policy-making. It also has implications for non-economic problems, such as the problem of optimal resource allocation in a distributed, decentralized computer network. Artificial economies such as Miller's might be developed to solve resource allocation problems in such a context. This is another example of the potential feedback from computer models of complex adaptive systems to the solution of technological problems.

Modeling the Evolution of Ecologies with Digital Organisms

A well-known and fairly challenging exercise in computer programming is to write a program whose only action is to print out an exact copy of itself. Thomas Ray, an ecologist at the University of Delaware, took this idea a step further, and used a kind of GA to evolve an "ecological community" of self-replicating programs ("digital organisms"). Ray's computational ecology, called "Tierra" (Spanish for "Earth"), begins with a single "ancestor", a self-replicating program written by Ray, which "lives" in one region of computer memory and places copies of itself in other regions. However, to make things interesting (and more realistic), there are occasional errors made in copying—these errors amount to mutations, which pave the way for the familiar principle of variation and selection. The offspring programs themselves self-replicate with mutations and, after several cycles, the computer memory is filled with a population of self-replicating programs, all related to the original ancestor. These programs compete for computer time so that they can be run, and for computer memory in which to place their offspring. As in real evolution, the "genetic material" of programs that are most efficient at self-replication tends to persist, and under this selection pressure, a large array of new strategies for self-replication emerges.

In a single run of Tierra, starting from a single ancestor placed in computer memory, Ray has seen a spectrum of ecological phenomena, such as "parasites"—programs that pirate the self-copying mechanisms of other programs in order to replicate themselves, "hyperparasites"—programs that discover means to thwart the parasites, and so on, with a kind of "arms race" emerging between the increasingly sophisticated parasites and hosts. This is remarkably similar to biological arms races actually seen in nature. The arms races in Tierra are in part responsible for the discovery of new self-replicating programs of increasing efficiency and robustness, far exceeding that of Ray's original ancestor program. All these phenomena emerge from interactions within the system; none was explicitly programmed (or even expected!) by Ray.

Ray has found that a number of ecological phenomena can be studied in great detail in this simple model, and the Tierra system has gained much attention recently in both the ecology and computer-science communities. Ray and his colleagues are currently studying how the automatic discovery of efficient self-replicating programs in Tierra can be harnessed to do useful work—in particular, to automatically evolve efficient programs for parallel computers. This is a third example of the feedback between computer models of complex adaptive systems and new computer technologies,

Conclusion

As is clear from the sampling of research described above, nature-inspired computer-modeling techniques such as genetic algorithms are allowing a new kind of experimental exploration of complex adaptive systems. There are many other examples of such research than the ones given above, including John Holland's Echo model, a unifying framework for modeling such systems. Such models allow the researcher to vary conditions in ways that could never be done by direct experimentation in nature, and to explore the consequences. This has the potential to lead to new insights about the system being modeled. Moreover, ideas from computer models of natural systems are being applied to technological problems, and are in many cases revolutionizing the way computation is being done. This is a worthwhile symbiosis indeed.

Further Reading

S. Forrest and A. S. Perelson, "Genetic Algorithms and the Immune System". In H. Schwefel and R. Maenner (editors), *Parallel Problem-Solving From Nature*. Berlin, 1990. Springer-Verlag (Lecture Notes in Computer Science).

J. H. Holland, "Genetic Algorithms", *Scientific American*, July, 1992.

J. H. Holland, "Adaptation in Natural and Artificial Systems" (second edition). Cambridge, Massachusetts, 1992. MIT Press.

J. H. Holland and J. H. Miller, "Adaptive Artificial Agents in Economic Theory". Santa Fe Institute Working Paper 92-10-051, Santa Fe Institute, Santa Fe, New Mexico.

T. S. Ray, "Is It Alive, or Is It GA?", in R. K. Belew and L. B. Booker, editors, Proceedings of the Fourth International Conference on Genetic Algorithms. San Mateo, California, 1991.

Morgan Kaufmann.