Genetic Algorithms and Artificial Life

Ziv Kedem

August 13,1998

This is a summary of an article "Genetic algorithms and Artificial Life", published by Melanie Mitchell from Santa Fe Institute and Stephanie Forrest from the Department of Computer Science at the University of New Mexico.

1 Abstract

Genetic algorithms are computational models of evolution that play a central role in many artificial-life models. This article reviews the history and the scope of research (updated to 1994) of genetic algorithms used in artificial life. It contains examples in which the GA's are used for a variety of applications. It also outlines some open questions for research in GA's.

2 Introduction

Evolution by natural selection is a central idea in biology, and the concept of natural selection has influenced our view of biological systems. This concept is also an important component of A-life, as it provides both a modeling tool and an automated design method for A-life objects. In fact, GA's are the most widely used models of evolution in A-life systems. The GA's can be used both as tools for solving practical problems and as scientific models of evolutionary processes. This article focuses on GA's as modeling tools of biological phenomena and does not discusses topics on GA's as "black boxes" for the design of evolutionary robots and autonomous agents.

3 Overview of Genetic Algorithms

John Holland was the first to describe (in the 1960s) the GA's as they are known today. Holland's book from 1975 presents the GA as an abstraction of biological evolution and gives the theoretical framework for the adaptation under the GA.

Holland's GA gives a generic method of moving from one population of "chromosomes" to a new population using natural selection along with the genetic operators of crossover, mutation and inversion. Each chromosome is a sequence (usually a bit string) representing a member of the population (which can be an organism or a candidate solution to a problem). Each chromosome consists of "genes", which are usually bits.

In each "generation", selection is applied by choosing the chromosomes in the population that will be allowed to reproduce. This is done in Holland's GA by using an explicit fitness function that rates the organisms. The organisms (or corresponding chromosomes, with a better fitness, have a better chance to reproduce than those who are less fit. To the reproducing chromosomes, the GA applies mutation, which is a random change in some genes, crossover, which is an exchange of subparts of two chromosomes, mimicking sexual reproduction in organisms, and inversion, which is the reversal of the order of a contiguous section of the chromosome. Today, inversion is rarely used in GA's. All of these operators are probabilistic and therefore we get a variety of chromosomes in the population.

Holland's GA which is a population-based algorithm based on generations of genetic operators was an important innovation. It has some limitations as the fitness function has to be explicit and in many models, this is not the case, since it is difficult or impossible to determine the fitness of a specific chromosome. However, Holland's GA has an advantage which is its theoretical framework which is based on "schemas". This framework can justify the empirical results of the convergence of the population to "good" chromosomes.

In addition to Holland's generation based algorithm, there are other algorithms with an implicit fitness function. A typical example of this type of algorithms is a set of agents moving on a lattice and interacting in several ways (finding food, eating, mating, trading and talking). The agents die when their stamina is too low (for example losing a fight, or not finding enough food), and reproduce when it is high enough (like after finding a large amount of food). This GA has the capability to describe a wide range of biological models, as in almost every biological model there is no explicit fitness function, but has the disadvantage of the lack of theoretical foundation.

4 Applications of Genetic Algorithms

GA's in their various forms have been applied to many scientific and engineering problems:

- *Optimizations*: GA's have been used in a wide variety of optimization tasks, including numerical optimizations and combinatorial optimization problems (like circuit design and job-shop scheduling).

- *Automatic programming*: GA's have been used to evolve computer programs for specific tasks (usually with special programming languages, for example a variation of the LISP language). They were also used to design computational structures (like cellular automata and sorting networks).

- *Machine and robot learning*: GA's have been used for many machine learning applications. These include classification and prediction tasks, the design of neural networks, and the designing and controlling of robots.

- *Economic models*: GA's have been used to model processes of innovation, the development of bidding strategies and the emergence of economic markets.
- *Immune system models*: GA's have been used to model various aspects of the natural immune system, including somatic mutation during an individual's lifetime and the discovery of multi-gene families during evolutionary time.
- *Ecological models*: GA's have been used to model ecological phenomena such as biological arms race, host-parasite co-evolution, symbiosis and resource flow in ecologies.
- *Population genetics models*: GA's have been used to study questions in population genetics, such as "when is a gene evolutionary viable?".
- *Interactions between evolution and learning*: GA's have been used to study how individual learning and species evolution affect one another.
- *Models of social systems*: GA's have been used to study evolutionary aspects of social systems, such as the evolution of cooperation, communication and the trail-following behavior in ants.

5 Open problems and future directions

There are many open problems in GA's. Some are technical problems (such as questions about genetic operators and chromosomes representations) and some are more general questions. This article addresses the open problems that were relevant to GA's in 1994. Since that time, there were some improvements in these open problems, but in most of them, there is still a great need of improvement.

There is a general problem with GA's and A-life, which is the difficulty to distinguish between a nice simulation of known facts and an experiment that teaches us something new and important. Therefore, establishing adequate criteria for evaluating A-life systems is still an open problem.

Another central problem that is specific to GA's is the problem of representations. The representation as bits is easy to program and to model, but it can be quite far from the corresponding biological model. Several works are being done on this subject, such as representation by LISP programs, or by using a higher-cardinality alphabet (such as real numbers) for the genes.

Another problem is the mappings between the genetic information in the genotype and the phenotype. All of the experiments described in this article were using a direct mapping between the genotype (the genetic DNA information) and the phenotype (the resulting A-life agent). This approach is clearly not scalable to more complex situations and therefore some experiments were made in which the organism "develops" from the genetic material. Improvements in this field are necessary if GA's and A-life are needed to describe complex agents.

The theoretical framework is another problem. Although there is some theory about development in a GA with an explicit fitness function, almost all complex models use an implicit fitness function. There is virtually no theory describing the behavior of the GA under complex circumstances (such as co-evolution).

6 Conclusions

This article is a good introductory article for genetic algorithms and artificial life. It has several demonstrations of how GA's can be used to model many biological phenomena. Although the article covers many aspects of GA's, they are all focused in GA's as a model for biological phenomena, and the aspects of GA's as an optimization tool and a designing tool for A-life is only mentioned briefly. Also, I believe that the article should have included in t least a few sentences, an explanation about the theory that stands behind the GA's, to show that the GA's validity is not entirely experimental.