MULTI-OBJECTIVE LEARNING VIA GENETIC ALGORITHMS

J. David Schaffer
Department of Electrical Engineering

John J, Grefenstette
Department of Computer Science
Vanderbilt University
Nashville, TN 37235

ABSTRACT

Genetic algorithms (GAs) are powerful, general purpose adaptive search techniques which have been used successfully in a variety of learning systems. In the standard formulation, GAs maintain a set of alternative knowledge structures for the task to be learned, and improved knowledge structures are formed through a combination of competition and knowledge sharing among the alternative knowledge structures. In this paper, we extend the GA paradigm by allowing multidimensional feedback concerning the performance of the alternative structures. The modified GA is shown to solve a multiclass pattern discrimination task which could not be solved by the unmodified GA.

1- Introduction

Pattern classification is a central task in flexible systems which incorporate an arsenal of problem solving techniques. For example, a given problem instance may need to be classified in order to decide which problem solving method should be applied. This paper concerns the task of multi-class pattern discrimination in a learning system. As in Mitchell [8], we view learning as a search process. But rather than searching a space of concepts, we consider a learning system which searches a space of production system programs for programs which adequately accomplish the desired pattern classification. The search is accomplished by means of a genetic algorithm (GA). GAs are powerful adaptive search techniques which have been used successfully in a variety of learning systems [3,4,5,12]. In this paper, GAs are extended in order to perform multi-objective learning in a pattern classification domain.

2• Learning via Genetic Search.

This section contains a brief description of GAs. More detailed descriptions are available in the literature [3,6,7,12]. Briefly, GAs may be viewed as adaptive generate-and-test procedures. GAs are adaptive in the sense that the candidate solutions generated reflect and exploit information obtained by earlier tests. A GA maintains a population of knowledge structures (e.g. alternative sets of production rules for a given task) and repeatedly (1) selects structures on the basis of observed performance, and (2) applies idealized genetic operators to the selected structures to construct new structures. For example, one important genetic operator is crossover, by which subsets of rules may be exchanged between two alternative knowledge structures. This results in a sophisticated search in which subsets of rules which contribute to good performance are propagated through the population. Other genetic operators are described in Smith [12]. In this paper, we concentrate on the influence of the performance feedback on step (1) above, the selection of knowledge structures for reproduction.

The characterization of the power and limitations of GAs is an active research area, but preliminary theoretical results are available. For example, the number of structures in the population which contain a given subset of rules can be expected to increase or decrease over time at a rate proportional to the average observed performance of all knowledge structures which contain that set of rules [11]. Thus, all subsets of rules appearing in the population of knowledge structures are explored simultaneously in a near-optimal fashion, a phenomenon which is called implicit parallelism by Holland[7]. Bethke[2] describes some properties of search spaces which may be especially hard for GAs.

Smith[12] implemented a GA-based machine learning system called LS-1. In LS-1, each structure maintained by the GA represents a production system (PS) program. Each PS program is evaluated on the learning task by a critic which assigns a numerical measure of fitness to the evaluated program. When all of the PS programs in the current population have been evaluated, the GA is invoked to construct a new population of PS programs, and the cycle is repeated. (See Figure 1.) LS-1 successfully learned PS programs for maze tasks and for draw poker. Our work extends the LS-1 system in order to achieve multi-objective learning.

3• The Task Domain

Multi-class pattern discrimination was selected as a representative multi-objective learning task. The specific task under investigation was to classify muscle activity patterns for five human gait classes, representing one normal and four abnormal gait types. A training set of 11 test cases was obtained from the literature [1], eaoh training case consisting of a 12-blt string derived from EMG signals from leg muscles while walking. As In LS-1, we used a GA to search a

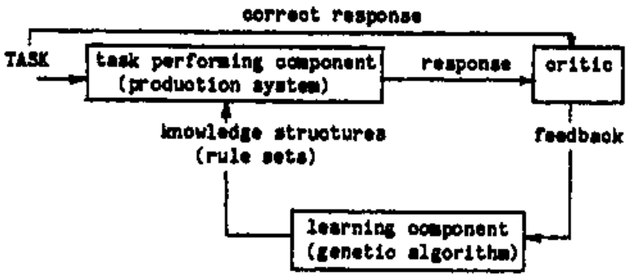

Figure 1. Schematic Diagram of Learning System

space of knowledge structures (i.e., rule sets). The goal of the learning system was to find a knowledge structure which correctly classifies the 11 training cases. The performance of the learning system was measured by the number of knowledge structures tested before obtaining a solution. By choosing subsets of the training cases, we derived 2-class, 3-class, 4-class, and 5-class discrimination problems. Each experiment was started with an initial population of randomly generated knowledge structures, in order to test the power of the GA with no initial knowledge. (Our implementation system[9] does allow the Incorporation of heuristic knowledge into the initial population of knowledge structures.)

4. The Need for Multidimensional Feedbaok

A series of experiments was performed in which a GA using a scalar critic was applied to multiclass discrimination problems. Although the GA could solve 2-class problems, it could not solve the full five-class problem. An analysis of the individual PS programs generated by the GA at various times during search revealed a common pattern. Knowledge of how to recognize a particular class was frequently absent from later PS programs, even when such knowledge was present in earlier programs. The problem is that structures which contain complementary knowledge are forced to compete by a GA using a scalar critic. Consider this simple example: Suppose that the critic measures the fitness of each structure by counting the number of training cases which were correctly classified. Suppose program P1 contains rules which correctly classify classes A and B, and program P2 correctly classifies only instances in class C. If all classes are equally represented in the training set, then program P1 appears to be twice as "fit" as program P2. Since the GA selects programs for reproduction on the basis of the fitness assigned by the critic, P1 will tend to contribute rule subsets to twioe as many new programs in the next population as will P2. As the number of classes increases, specialized knowledge (like that in P2) may tend to suffer extinction, finally resulting in suboptimal performance for the learning system as a whole.

Our solution was to modify the oritic so that a vector of performance measures was computed for eaoh struoture, with one slot in the fitness vector for each class represented in the training set. We then modified the GA so that complementary knowledge structures would share knowledge (through the genetic operators) rather than compete directly.

An interesting question emerges at this point which was never an issue with scalar critics. Where should any punishment for incorrect behavior be applied? Consider a knowledge structure which incorrectly classifies a class A case as class B. By applying the penalty to the A slot of the reward vector, we are punishing the failure to do the right thing. By applying it to the B slot, we are punishing doing the wrong thing. The former strategy was adopted for all subsequent experiments, arguing that class X training cases should contribute, positively or negatively, only to the X slot of the reward vector.

The modification to the basic GA is as follows: instead of selecting structures for reproduction on the basis of a single fitness measure, a portion of each new population is selected on the basis of each slot in the fitness vector. Note that if a structure scores well on several measures, then it will tend to be chosen in several selection phases. However, structures which perform well on even one measure will be given the opportunity to pass along their specialized knowledge. After the selection phases, structures are combined via genetic operator just as in LS-1. As a result, our modified GA performs multiobjective optimization in the space of knowledge structures [10].

5. The Critic

One important property of the critic in a GA-based system is that the fitness reported by the critic must reflect more than just success or failure on the task. Otherwise, the GA is unable to identify promising programs in the early stages when successes are rare, and cannot discriminate the better programs in the later stages when they are plentiful. One source of information of this type is the amount of uncertainty exhibited by a PS program while attempting the task, where uncertainty is defined as the extent to which conflict resolution is required in the decision process. A good critic should not discourage this uncertainty in the early stages but should discourage it in the later stages. Some experiments with different critics of this sort revealed something of the power of the GA to exploit subtle features in the critic. For example, a critic which applied a desirable uncertainty correction, but only rewarded success, was found to yield populations rioh in programs we called specialists. A specialist was a program which achieved a maximum score in one slot of the fitness vector, but zero in all others. It appeared to "know" only one aspect of the task. Unfortunately, these specialists were actually PS programs which were wildly guessing in the sense that they contained an overly general rule which suggested the same classification for every training case. A modified critic, designed to punish this indiscriminate guessing, was found to be too strict. By making risk-taking behavior too dangerous, it led the GA to evolve programs which produced no classification. This strategy at least scores zero, which is better than being excessively punished. Finally, a judicious balance of reward and punishment was achieved by incorporating in the critic a scoring scheme inspired by the Scholastic Aptitude Test (SAT scoring). This scheme led the GA to consistent success on problems involving 2, 3, 4 and 5 classes.

6. Results

The results of the experiments with 2, 3, 4 and 5-class discrimination learning are summarized in table 1. The figures reported for number-of-evaluations-to-solution are averages from several trials.

By successfully learning to solve the 5-class problem, the value of the vector-valued feedback to the learning component was demonstrated. It may be of interest that the only attempt to solve this problem by hand required a non-triviaj. effort and produced a solution program containing 16 rules. Although all solutions evolved by the GA used the full allotment of 11 rules, in all cases many of these rules were not used in the solution of the learning task. (They never fired.) They represented a kind of unexpressed genetic material. The average number of functional rules in the solutions to the 5-class task was 7.5.

TABLE 1. RESULTS OF EXPERIMENTS IN MULTICLASS PATTERN DISCRIMINATION LEARNING WITH LS-2

Number of classes to discriminate	Number of evaluations to learn solution	Maximum number of rules	Length of knowledge structures (bits)
2	1440	4	136
3	5647	6	228
4	15938	8	328
5	44309	11	462

7. Conclusions

The use of a scalar critic limits the usefulness of GAs for multi-objective learning. The scalar critic forces competition between knowledge structures which contain rules for complementary aspects of the learning task. We have shown that GAs can be extended to perform multicriteria learning through the use of a multidimensional feedback mechanism from the critic and a modification of the GA selection procedure. A series of experiments with an implemented system show that GAs are opportunistic learning algorithms, requiring careful design of the critic. In fact, the GA responded in a reasonable manner to the various critics. When the critic is too lax, only rewarding successes, the GA evolves programs which guess wildly to maximize the possibility of success. When the critic is too strict, the GA quickly learns than doing nothing is a good strategy. Only judicious balancing of reward and punishment leads to effective learning.

REFERENCES

1. A. B. Bekey, C. Chang, J. Perry, and M. M. Hoffer, "Pattern recognition of multiple EMG signals applied to the description of human gait," Proc. IEEE Vol. 65(5) (May 1977).

2. A. D. Bethke, Genetic algorithms as funotion optimizers, Ph. D. Thesis, Dept. Computer and Communication Sciences, Univ. of Michigan (1981).

3. K. A. DeJong, "Adaptive system design: a genetic approach," IEEE Trans. Syst., Man, and Cyber. Vol. SMC-10(9), pp.566-574 (Sept. 1980).

4. J. M. Fitzpatrick, J. J. Grefenstette, and D. Van-Gucht, "Image registration by genetio search," Proceedings of IEEE Southeastcon '84, pp.460-464 (April 1984).

5. D. Goldberg, Computer aided gas pipeline operation using genetic algorithms and rule learning, Ph. D. Thesis, Dept. Civil Eng., Univ. of Michigan (1983).

6. J. H. Holland, Adaptation *in* Natural and Artificial Systems, Univ. Michigan Press, Ann Arbor (1975).

7. M. L. Mauldin, "Maintaining diversity in genetic search," Proc. National Conf. on AI, AAAI 84, pp.247-250 (Aug. 1984).

8. T. M. Mitchell, "Generalization as Search," Artificial Intelligence Vol. 18 (1982).

9. J.D. Schaffer, Some experiments in machine learning using vector evaluated genetic algorithms, Ph. D. Thesis, Dept. of Electrical Engineering, Vanderbilt University (Dec. 1984).

10. J.D. Schaffer, "Multiple objective optimization with vector-evaluated genetic algorithms," submitted to Int'1 Conf. on Genetic Algorithms and their Applications, Carnegie-Mellon Univ., Pittsburgh (July 1985).

11, S. F. Smith, A learning system baaed on genetic adaptive algorithms, Ph. D. Thesis, Dept. Computer Science, Univ. of Pittsburgh (1980).

12. S. F. Smith, "Flexible learning of problem solving heuristics through adaptive search," Proc. of 8th IJCAI (Aug. 1983).