Co-Adaptive Genetic Algorithms: An Example in Othello Strategy*

Robert E. Smith
Department of Engineering Science and Mechanics
The University of Alabama
Email: rob@comec4.mh.ua.edu

Brian Gray
Department of Computer Science
University of Alabama in Huntsville
Email: bgray@cs.uah.edu

March 22, 1994

Abstract

This paper focuses on *co-adaptive* GAs, where population members are interdependent, and adaptation depends on the evolving population context. Recent research on co-adaptive GAs is reviewed. As an example of co-adaptation, a system for Othello strategy acquisition is presented. Preliminary results illustrate how a co-adaptive GA can continually explore the space of Othello strategies. This exploration yields insight into strategy interactions in Othello. Avenues for future analysis and experimentation with co-adaptive GAs are discussed.

1 Introduction

Typically, genetic algorithms (GAs) are introduced as population-based search procedures whose mechanics are based on those of natural genetics. The typical introduction will include a discussion of a perpetual drive towards the survival-of-the- fittest (selection), coupled with a randomized, but structured, exchange of information (recombination), and low-probability random change to preserve diversity (mutation). The net effect of these operators is a drive towards a single, highly fit population member coming to dominate the GA population. Thus, the effects of the so-called "simple genetic algorithm" are those of an optimization technique. However, when Holland (1975) first introduced GAs, he did so in the broader context of general adaptive plans.

This paper considers co-adaptive GAs, where the fitness values of population members depend on one another. Given these dependencies, a diverse population can be used as an adaptive exploration of a parameter space. As an example of GAs in this adaptive context, a system that explores parameters for Alpha-Beta search (Winston, 1984) in Othello (Rosenbloom, 1982) is presented. Preliminary results indicate how such an exploration can give insight into game playing strategies.

2 GA Adaptation Technology

Baker and Farrell (1992) draw a useful distinction between optimization, adaptation, and learning. Optimization is tuning parameters to suit a particular situation. Adaptation is perpetual, ongoing optimization. Learning is adaptation supplemented with memory.

*To appear in The Proceedings of The Florida Artificial Intelligence Research Symposium, 1994.

The strong convergence of typical GA selection precludes its ongoing adaptation. This effect is illustrated by the failure of simple GAs on time-varying problems (Pettit & Swigger, 1983). To yield more effective ongoing adaptation in the GA, one must at least insure a population that remains diverse. Diversity provides information (schemata) that the GA can recombine and continue the search. Mutation provides GA diversity, but at high rates of application, it can have disruptive effects on the overall search. Diploidy (Smith & Goldberg, 1992) and controlled hypermutation (Cobb & Grefenstette, 1993) have been suggested as intelligent methods for inserting diversity into the GA.

Another method for preserving diversity in the GA has a less temporal character. Sharing (Deb & Goldberg, 1989) was suggested to promote a steady-state population that occupies several peaks in the search space. In sharing, the GA uses an *effective fitness*. f', rather than the original fitness, f, in fitness proportionate selection. The effective fitness is given by

$$f_i' = \frac{f_i}{\sum_{j=1}^{N} Sh(d_{ij})} \ , \tag{1}$$

where d_{ij} is the distance between i and j under a given metric, $Sh(d_{ij})$ is the *sharing function*, given by

$$Sh(d_{ij}) = \begin{cases} 1 & \text{if } d_{ij} = 0 \\ 1 - \left(\frac{d_{ij}}{\sigma_s}\right)^{\alpha} & \text{if } d_{ij} < \sigma_s; \\ 0 & \text{otherwise,} \end{cases} \ , \tag{2}$$

N is the number of individuals in the population, and σ_s and α are parameters.

The net effect of sharing is that as individuals concentrate in one region of the search space, they degrade one another's effective fitness. This causes movement away from this overfilled *niche*. The size of a niche in the search space is limited by the parameter σ_s. As the number of individuals in a niche changes, fitness proportionate selection can dictate a steady state population where all individuals have equal effective fitness. This population remains stable. The effects of sharing counter any genetic drift towards a single individual by degrading that individual's effective fitness. Although sharing was designed for multi-modal GA search, it also preserves population diversity, albeit in a specific fashion. Sharing has particular implications for co-adaptive GAs, which are discussed in the following section.

3 Co-Adaptive GAs

In the previously discussed GA technology, it is assumed that each individual's fitness can be evaluated independently of other individuals in the population. This is the usual assumption for GAs in optimization. In sharing, effective fitness values are explicitly made dependent. However, sometimes individual fitness values are interdependent through some external aspect of the problem at hand. Such applications are referred to as *co-adaptive GAs*.

The most written-about co-adaptive application of GAs is the Michigan-style *learning classifier system* (LCS) (Holland, Holyoak, Nisbett, & Thagard, 1986). Such systems consist of a population where each individual is a production system rule. To complete a meaningful computational task, a diverse set of LCS rules must interact. This interdependence of rules creates the co-adaptive situation. Despite the existence of LCSs since Holland and Reitman's early experiments (1978), analysis of co-adaptive GAs has only begun. In many ways, the complex issues involved in LCSs, including temporal credit assignment, make consideration of co-adaptation in these systems difficult.

Studies point out the similarities between LCSs and immune systems (Farmer, Packard, & Perelson, 1986). In recent studies, GA-based simulations of the immune system illustrated several interesting aspects of co-adaptation (Forrest, Javornik, Smith, & Perelson, 1993; Smith, Forrest, & Perelson, 1993). These simulations did not explicitly include sharing. However, analyses and experimentation illustrated how sharing-like effects emerge from fitness interdependencies. Other results from these simulations show how generalization can also emerge as a natural effect of co-adaptive GAs.

To illustrate the effects of co-adaptation in GAs, the following sections present an example that explores evaluation strategies for Othello.

4 Co-Adaptation and Strategy Acquisition

Consider the task of finding strategies for two-person game playing or combat with GAs. One approach is to have the GA-driven system play a number of games with a player of known proficiency. One example of this approach is the application of an LCS-like system by Dike and Smith (1993) to air-combat maneuvering[1]. Although the results of this approach are promising, they are limited by the fixed nature of the GA's opponent. In effect, the GA system is not learning general strategies for the game at hand, but is instead learning strategies for the player with which it is trained.

One method for exploring a competition of adaptive opponents is to exploit the population basis of a co-adaptive GA. The following sections present an example where a co-adaptive GA evolves a population of competing Othello strategies. These strategies are represented as GA-controlled evaluation functions that serve as bases for Alpha-Beta tree search. Thus, the example also illustrates hybridizing the GA with another, more traditional, search strategy.

5 Othello

Othello (Rosenbloom, 1982) is played on an 8x8 board, with a set of discs that are black on one side and white on the other. The rows of the board are numbered 1 through 8, from top to bottom. The columns are represented by the letters A through H, from left to right. Black begins the game with two discs in the center of the board on the minor diagonal (E4 and D5). White begins the game with two discs in the center of the board on the major diagonal (D4 and E5). Black moves first. A move is made by placing one disc of your color into any legal empty square. A square is legal when placing a disc there will *capture* one or more of the opponent's discs. A row of discs is captured when the row has discs of the opponent's color on both ends. A captured row is turned over, thus changing it to the capturing player's color. Rows can be captured horizontally, vertically, or diagonally. No discs are ever removed from the board. This leaves a maximum of 60 possible moves. No discs are ever captured as a result of being bracketed by newly captured discs. If a player has no legal moves, then he must pass. Play ends when neither side has any legal moves. Usually, this occurs when there are few empty squares. At this point, the number of discs of each color is counted. The winner is the player whose color has the most discs on the board.

6 The Evaluation Function

In this study, Alpha-Beta search (Winston, 1984) will be used to find good Othello moves. The primary function of the GA in this study is to design the evaluation function used in the Alpha-Beta search. The GA will accomplish this through a set of start and stop times (given in terms of move number) for various evaluation function aspects, and a weight for each aspect. These features are described below.

Ultimately, the difference in final scores for the game is what one is attempting to maximize. The score is determined by the number of discs of the given color. Because the difference in scores is the object of the search, it is the only evaluation needed after the depth of the search is sufficient to reach endgame. However, it has been shown that maximizing the score in the beginning and middle portions of the game is actually a rather poor strategy. Therefore, the GA chromosome will include parameters that represent the first move at which the score will be evaluated. Since score is the basic evaluation component, it will receive a fixed weight of one. All other evaluation components will have weights that are under the GA's control. These components include:

Corner Squares cannot be captured once taken. These squares are counted for each side. The player's corner count receives a positive weight, while the opponent's corner count receives a negative evaluation weight.

Corner-Adjacent Squares should not be taken until the corner has been taken. These squares, if occupied, may allow the opponent to occupy the corner. Their count is assigned a negative evaluation weight.

[1] Extensions to this work will include varying, adaptive opponents

Population Size	50
Crossover Rate	0.9
Mutation Rate	0.01
Niche Size (σ_s)	1.9307

Table 1: Co-adaptive GA experiment parameters.

Edge Squares are difficult to capture and aid in capturing squares in the middle of the board. Their count is assigned a positive evaluation weight.

Stable Discs are defined as discs that cannot be captured by the opponent. They are assigned a positive evaluation weight.

Mobility is measured by the number of moves available to a player. This receives a positive weight for the player, and a negative weight for the opponent.

Seven weights are included for the evaluation components above. Each component of the evaluation function is also assigned two other GA-controlled parameters, the start and stop times for these components of the evaluation. The score is definitely needed at the end of the game so a stop time will not be included for that component. Each weight, start, and stop time is assigned three bits. Weights are scaled to range between zero and 3.5. Start times range from four to sixty. Stop times range from zero to fifty-six.

7 The Co-Adaptive Fitness Function

To insert the co-adaptive element into this application, each individual's fitness will be evaluated through competition with another individual in the population. An individual will play a match against a randomly selected population member. Since both sides receive a score for each game, a pair of individuals can be rated at the same time. Each match consists of two games, one with each player going first. The score that comes from the first game will be averaged with the score for the second game to yield the score for the match.

Note that if the GA converges to a single individual in this application, that individual would consistently play against itself, yielding a score of 32 (a tie). This score is relatively low, since it shows no real advantage over the opponent. Therefore, domination of the population by a single individual degrades that individual's fitness. This effect is similar to sharing. However, it was found that literal sharing was also useful for forming an effective, diverse, co-adapted population. Phenotypic sharing was employed (Deb, 1989). The distance metric was Euclidean, but with each parameter scaled to the same range. The niche size parameter σ_s was set to allow for 10 niches in the final population. Details on setting σ_s are available in the literature (Deb, 1989).

8 Results

Figure 1 shows the results of a GA run on the Othello example. Roulette wheel selection, single point crossover, and point mutation are used in this experiment (Goldberg, 1989). Parameters are given in Table 1.

Note that the pattern of the graph is different from that of a usual GA application. The plots do not converge, but instead wander through the space over time. Fitness values here are relative, not absolute. As more highly fit individuals come to dominate the population, they lower overall population fitness. This effect is due to competition of more effective individuals, and sharing. One would expect the population to move through phases where a stable group of effective competitors forms, and is then dominated by a newly emergent set. Due to the extensive computation time required for these experiments, a limited number of generations is presented. However, there seems to be no reason to expect ultimate convergence in these experiments. Rather, the co-adaptive GA is involved in an ongoing exploration of the search space.

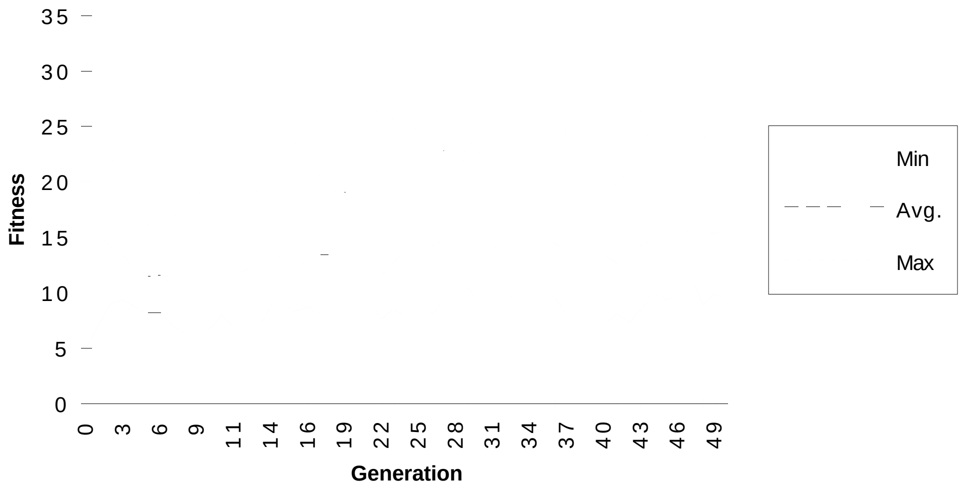

Figure 1: Results from the co-adaptive GA on the Othello example.

Real insight into the results in this application cannot be found in the raw fitness. More information can be drawn from examining the progression of the population. Throughout the run, the population remains diverse. However, groups of individuals with similar characteristics do evolve. Examining these groups shows the ongoing process of individuals dominating, then waning in the population, as new strategies emerge to exploit their predecessor's weaknesses.

The following procedure was used to identify groups in the population. In each generation:

1. Form a sorted list of all population members in decreasing order of *total share value*:

$$\sum_{j=1}^{N} Sh(d_{ij})$$

2. Develop a list of *group leaders*:
   (a) Make the population member with the highest total share value the first group leader.
   (b) Search down the sorted list to find the first population member that has a zero share value with every other group leader. Assign this population member to the group leader list.
   (c) Repeat until no further group leaders are found.
3. For each remaining individual, determine the closest (in terms of d_{ij}) group leader. Assign the individual to this leader's group.

Thus, group leaders are found by creating the set of individuals with highest share value that do not share with each other. In some sense, these group leaders lie at the center of population groups. Groups are formed by finding the group leader to which each individual is closest.

In the experiment presented in Figure 1, the number of groups generally varied between six and ten. In one out of the 51 generations, the number of groups fell to five. In another, the number rose to twelve. Generally when there were more than seven groups, the extra groups were comprised of fewer than four individuals. In most cases, the small groups would quickly disappear. At other times, smaller groups would grow in size and replace established groups.

Early in the run (at generation eight), the generation with the largest share value for an individual appeared. This generation had a total of six groups. All of the groups, except the largest, counted the corners, the near corners, or both. All the groups started counting the score near the middle of the game (except the smallest group). The largest group (with thirteen individuals) started evaluation of the score at move 44 and counted the opponents number of moves between moves 28 and 48 with a weight value of 2.5. The second largest group counted the corners with a weight of 2.5 for most of the game. It also counted the near corner squares and the opponents moves.

A generation chosen from the middle of the run (generation 22), also had six groups in it. Four of the six groups begin counting the score at move 44. The largest group begins counting the score at move 36, with the smallest starting at 12. All of the groups in this generation counted the corners, near corners, or both with fairly strong weights for most of the game. Three out of the six groups count the edge squares with a weight of 0.5 for the middle portion of the game. The largest group (21 individuals) counts the corners, the near corners, its own moves, and the opponents moves. The second largest group counts the corners, the near corners, the edges, the stable squares, and its own moves.

Near the end of the run at generation 46, there are eight groups. The time when the groups start counting scores varies widely. However, the largest group starts counting in the middle and the second largest starts counting at the end of the game. All the groups except one count the corner, the near corner, or both. Half of the groups count the edge squares. Half of the groups count stable squares. All groups but two, count their own moves, the opponents moves, or both. The largest group only counts the near corner squares. The second largest group counts the corners and both players moves.

From these observations, one can glean some general information about Othello strategy. It appears best to start counting the score sometime during the late middle game. Also, counting corners or near corners is generally useful. Counting the moves seems to be the next most useful strategy feature, followed by edge squares. The low number of times the stable disk evaluation was used may be due to the high overhead required to count stable disks. A more optimized version of the stable disk counting routine may reflect more favorably on this strategy feature.

Although these general strategy features are interesting, the more compelling part of these experiments is the ongoing change in strategies over time. Examining the results in these co-evolved populations is a complex, ongoing process. Given the computational time involved in this application, it will take time to obtain more detailed conclusions. However, this automated, adaptive, simulation-based strategy yields a unique, ongoing stream of information on the interplay of strategies.

9 Final Comments

The results presented here are preliminary. However, they point out an interesting new area for GA application. By forming a co-adaptive population, one can gain insight into the nature of a search space. The parameters given in the Othello experiments can provide insight into how Othello evaluation functions interact as strategies. Any arbitrary selection of evaluation functions would (in general) have intrinsic weaknesses. The GA simulations allow one to see these weaknesses, and possibly form better evaluation strategies in a given competitive situation.

Much research remains to be done in co-adaptive GAs. New visualization tools for the complex interactions of co-evolving individuals would be helpful in analyzing results. Also, many general analytical issues remain open. The propagation of schemata between individuals in a co-adaptive GA is clearly an important factor in the population's evolution. In the Othello example, competitors that form share schemata, and thus features from previously formed strategies. Therefore, one must consider how children can be expected to effect the fitness of parents, and vice versa. Such issues have yet to be explored analytically. Foundational work is also necessary on other aspects of co-adaptive GAs, including population sizing, implicit parallelism, and the k-armed bandit (Holland, 1975). Despite analytical gaps that exist, co-adaptive GAs seem to be a natural and promising extension of the GA paradigm.

References

Baker, W. L., & Farrell, J. A. (1992). An introduction to connectionist learning control systems. In *Handbook of intelligent control: Neural, fuzzy, and adaptive approaches* (35–64). New York: Van Nostrand Reinhold.

Cobb, H. G., & Grefenstette, J. J. (1993). Genetic algorithms for tracking changing environments. In S. Forrest (Ed.), *Proceedings of the Fifth International Conference on Genetic Algorithms*, 523–530, San Mateo, CA. Morgan Kaufmann.

Deb, K. (1989). *Genetic algorithms in multimodal function optimization* (TCGA Report No. 89002). Tuscaloosa: The University of Alabama, The Clearinghouse for Genetic Algorithms.

Deb, K., & Goldberg, D. E. (1989). An investigation of niche and species formation in genetic function optimization. *Proceedings of the Third International Conference on Genetic Algorithms*, 42–50.

Dike, B. A., & Smith, R. E. (1993). Application of genetic algorithms to air combat maneuvering. In W. Karplus, & M. L. Padgett (Eds.), *Proceedings of the Fifth Workshop on Neural Networks: Academic/Industrial/NASA/Defense*, Vol. 2204, 84–94, San Diego. SPIE.

Farmer, J. D., Packard, N. H., & Perelson, A. S. (1986). The immune system, adaptation, and machine learning. In D. Farmer, A. Lapedes, N. Packard, & B. Wendroff (Eds.), *Evolution, games and learning* (187–204). Amsterdam: North–Holland. (Reprinted from *Physica*, 22D, 187–204)

Forrest, S., Javornik, B., Smith, R. E., & Perelson, A. (1993). Using genetic algorithms to explore pattern recognition in the immune system. *Evolutionary Computation*, *1*(3), 191–212.

Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning*. Reading, MA: Addison–Wesley.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor: The University of Michigan Press.

Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. R. (1986). *Induction: Processes of inference, learning, and discovery*. Cambridge: MIT Press.

Holland, J. H., & Reitman, J. S. (1978). Cognitive systems based on adaptive algorithms. In D. A. Waterman, & F. Hayes-Roth (Eds.), *Pattern directed inference systems* (313–329). New York: Academic Press.

Pettit, E., & Swigger, K. M. (1983). An analysis of genetic-based pattern tracking and cognitive-based component tracking models of adaptation. *Proceedings of the National Conference on Artificial Intelligence*, 327–332.

Richards, R. (1981). The revised USOA rating system. *Othello Quarterly*, *3*(1), 18–23.

Rosenbloom, P. S. (1982). A world-championship-level othello program. *Artificial Intelligence*, (19), 279–320.

Smith, R. E., Forrest, S., & Perelson, A. S. (1993). Searching for diverse, cooperative populations with genetic algorithms. *Evolutionary Computation*, *1*(2), 127–149.

Smith, R. E., & Goldberg, D. E. (1992). Diploidy and dominance in artificial genetic search. *Complex Systems*, *6*(3), 251–285.

Winston, P. H. (1984). *Artificial intelligence* (2 ed.). Reading, MA: Addison-Wesley.