Reasons for Premature Convergence of Self-Adapting Mutation Rates

Matthew R. Glickman
Computer Science Department
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213-3891
glickman@cs.cmu.edu

Katia Sycara
The Robotics Institute
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213-3891
katia@ri.cmu.edu

Abstract-

To self-adapt ([Schwefel, 1981], [Fogel *et al.*, 1991]) a search parameter, rather than fixing the parameter globally before search begins the value is encoded in each individual along with the other genes. This is done in the hope that the value will then become adapted on a per-individual basis.

While this mechanism is very powerful and in some cases essential to achieving good search performance, the dynamics of the adaptation of such traits are often complex and difficult to predict. This paper presents a case study in which self-adapting mutation rates were found to quickly drop below the threshold of effectiveness, bringing productive search to a premature halt. We identify three conditions that may in practice lead to such premature convergence of self-adapting mutation rates. The third condition is of particular interest, involving an interaction between self-adaptation and a process referred to here as "implicit self-adaptation". Our investigation ultimately underlines a key aspect of population-based search: namely, how strongly search is directed toward finding solutions that are not just of high quality, but those which also produce other high quality solutions when subjected to the chosen variation process.

1 Introduction

1.1 Self-Adaptation

There are several parameters that are normally set prior to conducting evolutionary search. These parameters specify some aspect of how the search will be conducted, *e.g.* the selection strength, the crossover probability, the mutation rate, etc.

When the best candidate solutions reproduce to form the next generation, the offspring are subjected to some stochastic variation process. Many search parameters, such as the mutation rate or the crossover probability, affect this process. Their impact is upon the distribution of offspring produced by any given parents. This distribution is also known as the parent-offspring *transmission function* ([Slatkin, 1970], [Altenberg, 1985], [Altenberg, 1994]).

These parameters are typically set before beginning search, are held constant for the whole run, and are applied invariantly in all situations. Self-adaptation (for a review see [Angeline, 1995]) instead encodes such values in each individual along with its other genes. When individuals are then subject to variation, the process draws upon each individual's own encoded parameters (which are themselves subject to variation as well).

Because the encoded parameters affect only parent-offspring transmission, they have no direct impact on the fitness-evaluation of an individual; two individuals that differ only in the values of their encoded search parameters will have the same expected fitness. For this reason, the action of selection on these parameters is necessarily indirect: Individuals of higher fitness selected for reproduction will simply *tend* to have well-adapted search parameters because those parameters are the ones that were used to generate the high fitness individual in question.

The most common class of trait to be self-adapted is that of mutation rates (see [Schwefel, 1981], [Fogel *et al.*, 1991], [Bäck, 1992b], [Bäck, 1992a]). Mutation rates are an intuitively appealing feature to adapt because they specify some measure of the breadth or scope of the parent-offspring transmission function, thus providing selection with a mechanism for adaptively tuning the degree of exploration versus exploitation in search over time. Because a more fit solution has more to lose via random mutation, the general trend is for selection to favor increasingly lower mutation rates as the fitness of the candidate solutions approaches optimal.

Self-adaptation would seem to be of most interest for complex or other search spaces whose structure is difficult to characterize. Under such conditions, advantageous parameter values may simply be hard to estimate, or some value may be appropriate for one region of the search space but not for another (this is more likely to arise during extended search [Glickman and Sycara, 1998]).

1.2 The Case of the Vanishing Mutation Rates

We were investigating the performance evolutionary search in such a complex search space. The domain involves training Artificial Neural Networks (ANN's) with recurrent connections (RANN's) to yield effective goal-oriented behavior in a challenging environment (specifics are given in section 2; see also [Glickman and Sycara, 1999]). Evolving RANN's is particularly challenging because of the extreme interdependence of the weights. It appeared likely that variable mutation rates might provide a substantial benefit in such a sensitive search

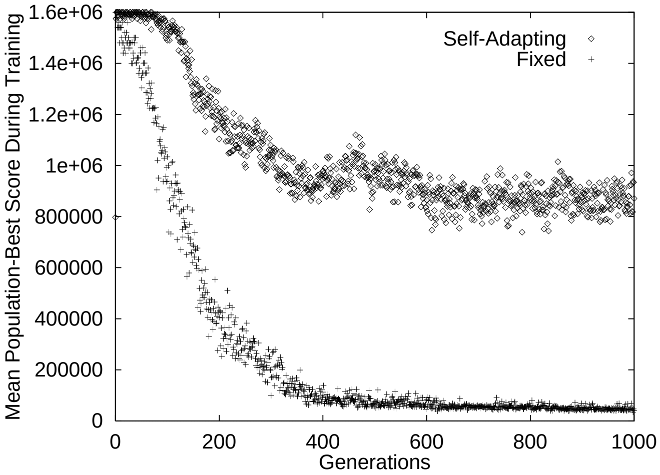

Figure 1: Mean population-best solution during training for both self-adapting and fixed mutation rates (note: A *lower score* corresponds to *superior fitness*)

space.

Instead, what we found was that allowing mutation rates to self-adapt had a very negative impact on performance (as shown in figure 1). The reason for this poor performance can been seen in figure 2: Mutation rates were selected down to near zero so rapidly that effective search quickly came to a halt.

There are a number of reasons why mutation rates may drop below the threshold of effectiveness before useful search is completed. We present three such reasons in section 3: one referred to here as "the bowl effect"(section 3.1), another involving the relationship between mutation rate and selection strength (section 3.2), and finally the one that appears most likely to be at fault in this case (section 3.4), an interaction between variable mutation rates and a phenomenon that may be called "implicit self-adaptation". We discuss our conclusions in section 4.

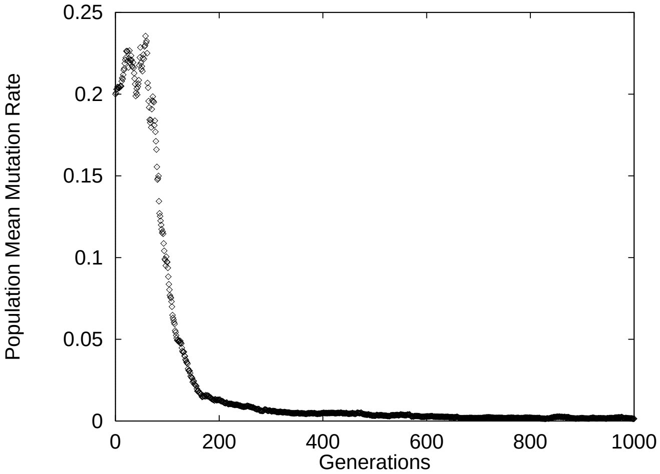

Figure 2: Avg. population-mean mutation rate during training

2 Evolving RANN's for Goal-Oriented Behavior

2.1 The Start-Anywhere Maze Task (SAM)

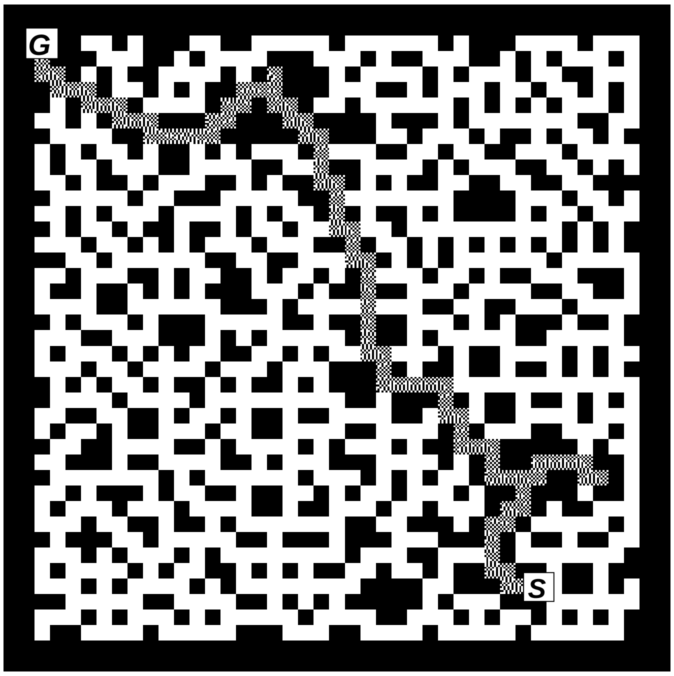

Figure 3: Navigating a typical 40x40 maze, from S, a randomly chosen start-point, to G, the fixed goal-point.

The target task for many of the results presented below involves navigation in a given maze (see figure 3 for an example). As opposed to most maze tasks an agent is rewarded according to how quickly it can find its way to the designated goal when started from *any* random point in the maze rather than only from a designated start point.

When an agent is started from a given point, it has no *a priori* knowledge of its location. Significantly compounding the difficulty of this task is the fact that at any given time-step, the agent only "sees" the eight squares immediately surrounding its position (see figure 4). Thus, an agent must begin by exploring, progressively integrating observations in order to localize itself so that it can narrow in on a strategy to quickly and reliably reach the goal. This process requires memory while in the maze, which is provided via the recurrent links in the evolved networks.

Another difficult aspect is that the agent receives a reward only if and when it reaches the goal; if, during a single trial, one agent manages to come within one step of the goal without actually reaching it while another never even gets close, the two receive equally poor scores. Such extremely *delayed reward* significantly limits the information available to the selection process for differentiating between alternate behavioral policies.

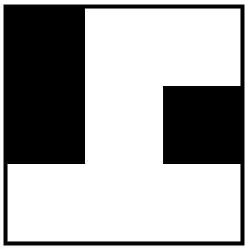

Figure 4: Field-of-view of an agent in the maze (note - orientation is significant)

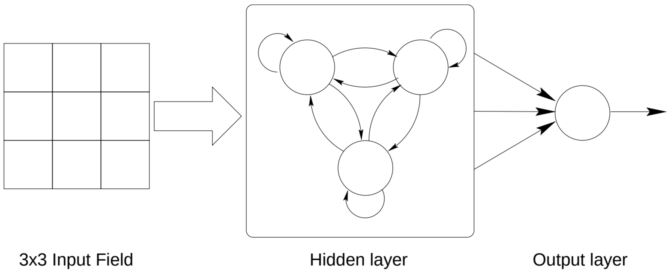

Figure 5: The network topology

2.2 The RANN's

Studies applying evolutionary search to training RANN's (*e.g.* [Angeline *et al.*, 1994]) often search the space of network topologies, possibly concurrently with the connection weight-space. For reasons of simplicity, we simply chose a single, very powerful topology and conducted search only in connection weight-space. This arrangement yields a search space of a fixed number of dimensions.

Individuals in the population are encoded as vectors of real values, each corresponding to a connection weight in the network. The unit+s (with sigmoid activation functions) are arranged according to a fixed, recurrent topology, an Elman [Elman, 1990] network. The input layer (9 inputs, 1 for each of the pixels in the 3x3 visual field) is fully connected to a single hidden layer. The hidden layer is fully connected to both the single output unit *and to itself* (see figure 5). The recurrent connections in the hidden layer are effectively time-delayed by one time-step: All of the hidden units' outputs are determined in parallel, and thus they each see each others' output values from the previous time-step when calculating their current output value. By granting each hidden unit access to the activation levels of all the hidden units in the previous time-step, recurrent connections provide the capacity to retain state over time. What information is retained and how it is encoded is left to be determined via evolution.

If the value produced by the output unit is below a particular threshold, the agent turns left and tries to advance one grid-unit, if above a different threshold, to the right and forward one unit, and if between these two thresholds, straight ahead one unit. Unless otherwise noted, results presented here are for agents with a total of 12 hidden units, which yields a complete weight vector of 277 weights in length (including bias unit connections).

2.3 The Evolutionary Algorithm

An Evolutionary Algorithm (EA) consists of two primary processes, selection and variation, that are alternately performed on a population of candidate solutions. In these experiments we employed a population of size 100.

The specific form of selection we've used is *tournament selection*. Given a population of size n and user-defined parameter, k ($n = 100$ and $k = 10$ for our purposes here), n "tournaments" are conducted whereby k individuals are randomly chosen from the current population and the one with the best score is copied into the next population.

Scoring is conducted in *trials*. In a trial, an individual is dropped into the maze and is given a fixed amount of time (400 time-steps in this case) in which to find its way to the goal. An agent's score for the trial is then the square of the amount of time it took to reach the goal (which is simply the full trial-length when the goal is not reached). For each generation, 10 random starting places in the maze are chosen, and each individual's training score is the sum of its trial scores when starting from each of the 10 specified points. One potentially confusing feature here is that lower scores represent a quicker time to the goal, and thus *lower* scores correspond to *higher* fitness.

In the manner often favored by EA's such as Evolution Strategies [Schwefel, 1981] and Evolutionary Programming [Fogel *et al.*, 1966], the variation process simply mutates each new population member via the addition to each weight of Gaussian noise (sampled from a distribution with a fixed standard deviation defined by the user, 0.2 in this case) to each weight. For simplicity, the EA used here does not employ any form of crossover operator.

3 Reasons for Premature Rate Convergence

3.1 The Bowl Effect

Figure 6 shows a classical local minimum (near x = 4.5) in a region of the function $x + 5 \sin x$. Suppose at some point that the evolving population becomes clustered close to this minimum.

There will be some range of mutation rates expressed in the population. In the case in figure 6, the rates vary from the lower rate a to the higher rate b. Individuals with the higher mutation rate b will tend to generate offspring that stray farther from the minimum than those with rate a. Because these offspring tend to end up higher in the bowl, their expected fitness is lower and selection will tend to favor those with mutation rate a. We can then see that if another mutation rate lower than a appears in the population, their mean offspring fitness will be yet higher. In this way, selection will favor smaller and smaller mutation rates.

Consider, however, the case in figure 7. Here, the rates

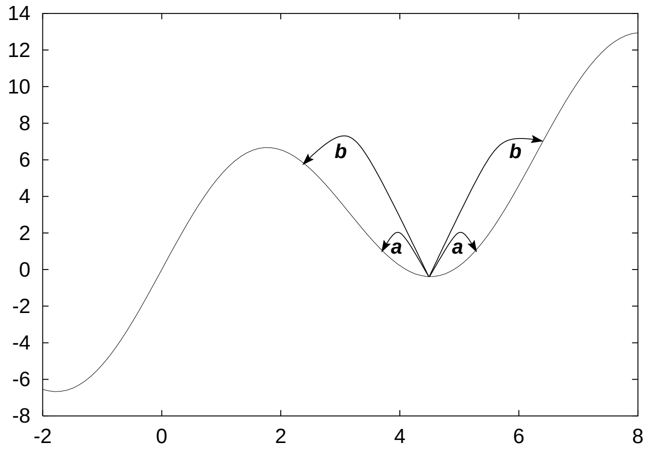

Figure 6: A local minimum of the function $x + 5\sin x$, showing the range of offspring produced via mutation rates a and b by individuals located on the minimum.

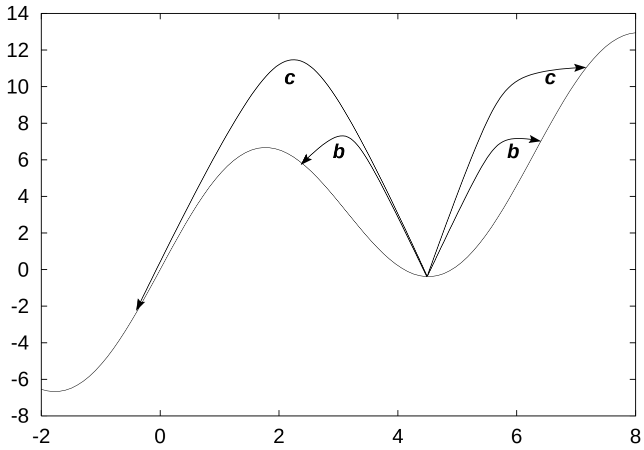

Figure 7: A local minimum of the function $x + 5\sin x$, showing the range of offspring produced via mutation rates b and c by individuals located on the minimum.

range between mutation rate b and a yet higher mutation rate, c. In this case, the rate c is high enough that it will with reasonable probability yield a few offspring that fall far enough outside the bowl that they will be of higher fitness than any other members of the population. Selection will therefore favor individuals with mutation rate c over those with rate b.

We can see then that if the range of mutation rates in the population extends above some threshold, we can expect that higher rates will be favored, and that the population will escape the local minimum. However, if the range of rates does not exceed the threshold, then rates will tend to decrease and the population will entirely converge upon the local minimum.

To observe this effect directly, we conducted a large number of runs starting with populations of identical individuals. In all runs every member of the population was placed directly at the local minimum depicted in figures 6 and 7. However, the runs differed in that each population was initialized with one of five different initial mutation rates. For all the runs starting with each mutation rate, we calculated the log-mean mutation rate over time as shown in figure 8. Some of the chosen initial rates were clearly above the critical threshold , resulting in a tendency for rates to rise over time[1], while others fell below the threshold leading to a tendency to drop toward zero.

In the case of the RANN/SAM domain, however, raising the initial mutation rates did not solve the premature convergence problem. If the rates were started above some very high threshold, mutation was so severe that even if the EA found some solution that was measurably better than random, heredity was so poor no progress could be made.

[1]Rising indefinitely in this case, because (1) the function has no minimum, and (2) once the mutation rates have exceeded the threshold they are (practically) no longer subject to becoming trapped in the local minima

3.2 Selection Strength

The strength of selection has a clear influence on which mutation rates are favored by which fitness levels (see [Glickman and Sycara 98] for a more complete treatment). For example, two individuals identical except for their encoded mutation rates will appear identical to the fitness function and hence receive the same number of reproductive opportunities. However, the range of the fitnesses of the offspring they then produce will quite likely be different.

Mutation tends to degrade solution quality, and stronger mutation will tend to trigger greater degradation. At the same time, stronger mutation also increases the overall variance in post-mutation fitness. The result is that the individual with a higher mutation rate will likely produce offspring with a lower mean fitness, but with perhaps a higher chance of occasionally producing particularly high fitness individuals than the equivalent individual with a lower mutation rate. Under relatively low selection pressure, many individuals will get an opportunity to reproduce and the offspring of the lower mutation rate parent may well receive more total reproductive opportunities because of their higher mean fitness. However, under high selection pressure, only the very most fit individuals will reproduce. Because of the higher likelihood of the parent with the higher mutation rate to yield offspring of particularly high fitness, this situation tends to in general favor higher mutation rates.

In the case of the RANN's, the selection strength effect seemed quite likely to be of at least some help in forestalling the mutation rates' convergence. Nonetheless, we found the effect in this case to be negligible at best. Measuring the effect is complicated by the fact that search is particularly effective only within a relative narrow band of selection strength (a tournament size of 10 seems about optimal) even with a fixed rate. Significantly higher or lower strength tends to re-

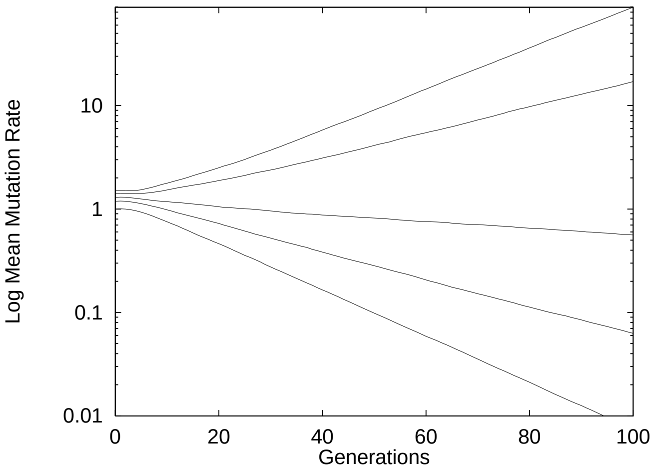

Figure 8: Mean mutation rates over time for different initial mutation rates for a population starting at the local minimum (note: The mean mutation rate of some runs that rise and some that fall will always be large. To depict the general tendency of rates to either rise or fall, we took the mean of the log mutation rates, rather than the log of the mean.)

sult in relatively poor solution quality, making it difficult to adequately assess the relationship between fitness and the selected mutation rate as selection strength varies. There exists the possibility that larger populations may provide effective search coupled with higher selection strengths, but the additional computational cost seems unlikely to justify foreseeable performance benefits.

In other attempts to use selection strength to operationalize variable mutation rates, we experimented with several dynamic pressure schemes, based on or simulating the pressure induced by antagonistic, co-evolving populations. While these schemes often resulted in performance benefits in the case of fixed rates, no significant effect was seen on variable rates.

3.3 Some Other Possibilities

Another factor which might possibly influence the viability of self-adapting rates is the parameter τ, which governs the variance of mutation rates when they are themselves mutated. However, this prospect seems unlikely for a number of reasons, including: (1) Limited experimentation in varying τ has yielded no significant results; (2) τ is found to be a rather insensitive parameter in general; and (3) The results of experiments varying the initial rates show that significantly higher rates by themselves provide no benefit.

Another possibility is that in the highly epistatic RANN domain some traits are likely to be at times extremely mutation-sensitive. Perhaps it is these extra-sensitive traits that are driving mutation rates down so quickly, a condition which might be alleviated via the powerful technique of allocating multiple mutation rates per individual, each which affect different parts of the genome. Although attractive, one problem with this approach is that there are simply too many weights to support one mutation rate each, which leaves open the problem of how to best associate n mutation rates with m weights, where $n < m$. At the same time, because a single, fixed rate can yield quite reasonable results it seems that multiple rates are not a necessity for good performance.

3.4 Explicit and Implicit Self-Adaptation

3.4.1 Implicit Self-Adaptation

Genetic Programming (GP, [Koza 92]) entails the evolution of programs, most commonly in the form of LISP S-expressions. There are typically two search operators to induce variation: crossover, which swaps sub-trees between two selected programs, and mutation, which spontaneously transforms nodes in a program tree from one type to another.

As noted above, as the population becomes more fit, the variation process becomes increasingly disruptive to individuals ability to transmit this fitness to their offspring; a more fit individual usually has "more to lose" in the variation process. A phenomenon addressed by [Altenberg, 1994] concerns how evolving programs often become over time increasingly transmission-conservative, *i.e.* more likely to conserve their fitness score when acted upon by the search operators.

A key property of GP underlying this phenomenon is that any particular program behavior can be generated by any number of syntactically distinct S-expressions, *i.e.* the representation provides a *many-to-one mapping* from the set of genotypes to the set of phenotypes. This property is of import because while syntactically distinct programs may be the same in the eyes of the fitness function, the distribution of the fitness of their offspring (after the action of the search operators) may well be different. This makes it possible for variations to occur that affect the transmission-related properties of a program independent of any immediate effect upon its fitness. In other words, GP's many-to-one representation makes possible variations whose effect is akin to that of self-adapting transmission-related search parameters. We refer to this process here as "implicit self-adaptation" because the effect is similar even in the absence of an explicit self-adaptive mechanism.

One kind of variation that can make a program more conservative is the addition of code which plays no role in program behavior. Such regions of an encoded individual that have no discernable effect on fitness are often referred to as *introns* [Levenick, 1991], a term from biology which refers to segments of DNA that don't contribute to protein formation. The more of such functionless code there is in a program, the more likely variation will occur in this code rather than in a behavior-critical section. In this way, the "implicit mutation rate" of programs may be adapted over time.

Note that if the genotype-to-phenotype mapping were one-to-one, then any variation would produce a phenotypic change and a potential fitness change. A program might thus not be able to produce more conservative offspring without a concomitant change in fitness. Because selection acts more directly upon objective traits than upon transmission-related traits like mutation rates, this situation would likely inhibit selection for transmission-related traits such as conservatism.

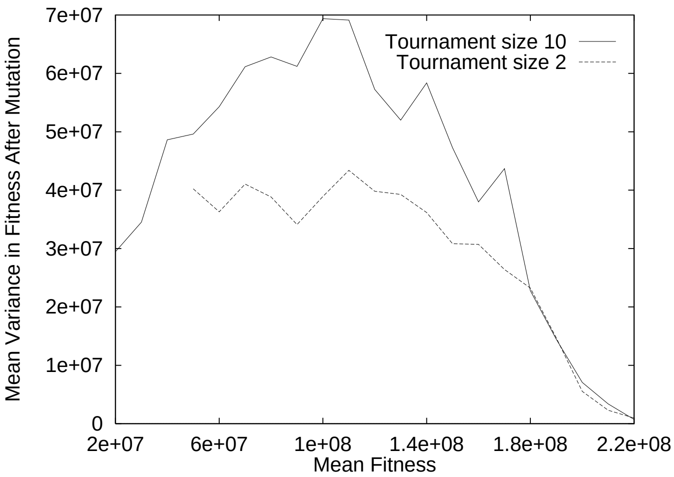

Figure 9: The relationship between mean offspring fitness and variance in offpring fitness, displayed for solutions evolved with two different selection strengths

RANN's in the SAM task would also seem to have such a many-to-one mapping: Variations in weights may occur that have no effect on the observed behavior of the network, but may well influence the distribution of behaviors observed in its offspring. We looked to see if we could observe some form of implicit self-adaptation in the RANN/SAM experiments.

We sampled the populations generated during several runs (using fixed mutation rates) with both higher (tournament size $= 10$) and lower (tournament size $= 2$) selection strengths. We then repeatedly mutated the sampled individuals to estimate their distributions of offspring fitness. Figure 9 shows the relationship between the means of the resulting fitness distributions and their corresponding variances. For the same mean offspring fitness, we observe that once the scores drop below near-random quality, for stronger selection the same mean fitness is associated with greater variation, *i.e. a higher implicit mutation rate*. The effect of selection strength upon variance in offspring fitness thus mimics that between selection strength and self-adapting mutation rates, indicating the presence of a form of mutational self-adaptation in the evolution of RANN's even in the absence of any explicit mechanism.

3.4.2 A Revealing Experiment and Another Hypothesis

Attempts to unravel our mutation rate puzzle led at one point to a very simple experiment: Evolve RANN's with self-adapting rates restricted to one of two values: 0.2 or 0.

τ (set to 0.1) in this case represented the probability of a mutation rate "flipping" to the other value when subject to variation. We found the results, depicted in figure 10, rather striking.

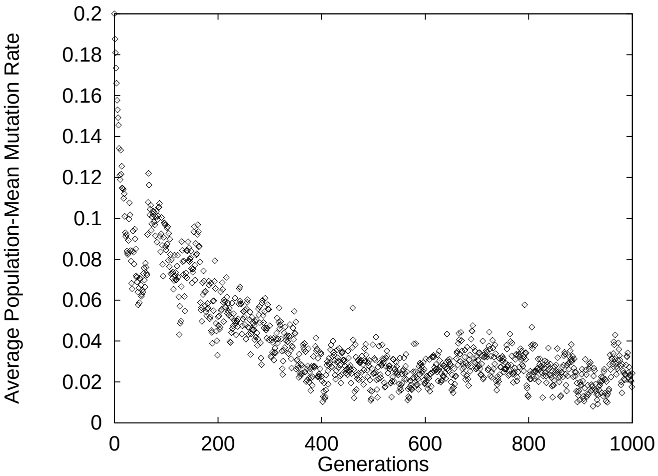

Figure 10: Average population-mean mutation rate during training for self-adapting rates restricted to values of either 0.2 or 0.0

Due to the reasonably high frequency of favorable mutations encountered using a fixed rate of 0.2, it was somewhat surprising that evolution shortly began to favor no mutation at all (the mean rate hovering not too far from the minimum rate generated via 10% probability of back-mutations). Perhaps more interesting was the observation that the 10% or more individuals that ended up with a rate of 0.2 seemed to be mostly failing to produce any significant favorable mutations, as opposed to what one might expect from observing the fixed-rate case.

Investigating further, we isolated populations from around generation 400 and repeatedly mutated the members and recorded the resulting fitnesses. What we found was that these individuals were far less likely to produce offspring of equivalent or higher fitness than those of equivalent fitness from a fixed-rate population. In other words, the ability to assume a mutation rate of zero had somehow led to the prevalence of a transmission-related mutation in the population: one which resulted in "mutation-brittle" solutions. It is our hypothesis that this process occurs in the following manner:

- Variation provides for a mix of individuals in the population: those with non-zero and zero mutation rates, and those that are more or less mutation-brittle.
- Less mutation-brittle individuals may successfully reproduce regardless of their mutation rate, while only the more mutation-brittle solutions with a mutation rate of zero tend to reproduce successfully.
- This condition then lowers the incidence of non-zero mutation rates in the population.
- The correspondingly higher frequency of zero rates now allows more mutation-brittle individuals to form.

- The process continues in this manner, with the increasing frequency of zero rates and mutation-brittle traits each promoting the reproductive dominance of the other.

It is important to note here that as stated, this process may be reasoned to run the other way as well, *i.e.* the second point above could be argued to lead not only to a higher frequency of zero mutation rates but also to an increased fraction of non mutation brittle individuals.

One factor which may bias the direction of this process is that, as previously discussed, that there exists an additional, independent force against mutation that increases in strength as solutions become more fit; the more fit a solution, the more it has to lose via random perturbation.

Another factor may be that brittle solutions are more common than non-brittle ones. Just as that, in general, increasingly fit solutions tend to be increasingly rare in the search space, those solutions with a reasonable probability of retaining or increasing in fitness when subjected to mutation may become increasingly rare as fitness increases as well. This bias toward more brittle solutions would in turn be expected to favor lower mutation rates.

3.4.3 Testing for the Effect in the General Case

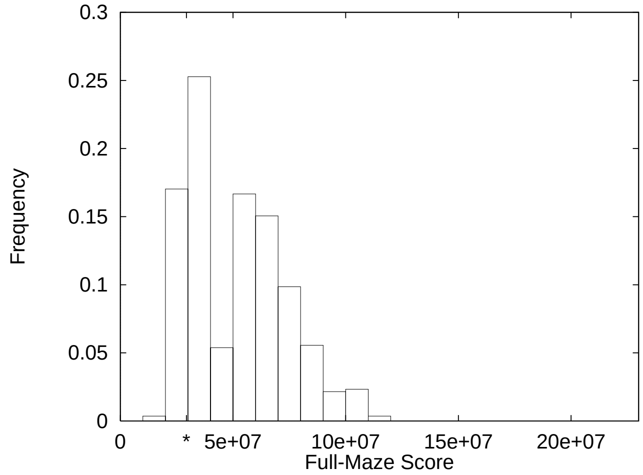

Figure 11: Fitness distribution resulting from repeatedly mutating an intermediate-quality solution evolved via fixed rates (note: the fitness before mutation is marked with a $*$).

Comparison of the fitness-distribution-after-mutation of solutions of equivalent fitness from runs with and without self-adapting mutation rates supports the generality of the "cooperative" relationship between brittle solutions and lower mutation rates. Figures 11 and 12 indicate the distribution of offspring fitness (*i.e.* the distribution of fitnesses after mutation) from a pair of solutions of equivalent fitness evolved with fixed and variable mutation rates, respectively. To enhance the comparability of this and other solution-pairs (as well as to find solutions of higher fitness while using self-adapting mutation rates), we first conducted a "seed run" with a fixed mutation rate up to generation 200. We then continued with two runs starting from this population, one with a fixed rate and another with self-adaptation.

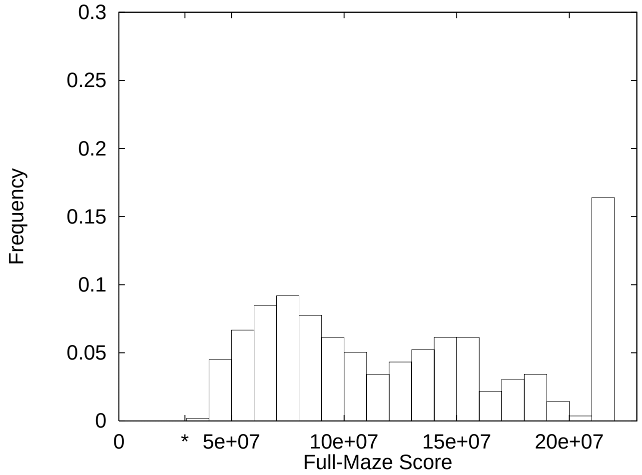

Figure 12: Fitness distribution resulting from repeatedly mutating an intermediate-quality solution evolved via variable rates (note: the fitness before mutation is marked with a $*$).

We mutated (using the fixed-rate value of 0.2) and measured the fitness of each member of each pair more than 500 times. We found the same relationship between the distributions for 9/10 solution pairs, each from different seed runs[2].

These results indicate that there is some form of co-adaptation between the response of solutions to mutation and their mutation rates. There is a clear association between solutions with low mutation rates and a poor response to mutation at at the higher rate.

It is conceivable that low-mutation-rate solutions might be selected to compensate for lower mutation rates by simply becoming more "mutation-sensitive" rather than mutation-brittle, and thus respond equivalently to the variation process regardless of their mutation rate. While it is reasonable to suppose that such solutions exist, the observed performance results indicate that they are not arising with any significant frequency.

4 Conclusions

We have seen that multiple conditions that may lead to premature convergence of self-adapting mutation rates.

We cannot yet specify the exact conditions under which the co-adaptation of mutation rates and solution brittleness may be expected to occur. However, this effect appears most likely associated with a many-to-one genotype-to-phenotype mapping which provides the possibility for transmission-related variations that are independent of changes in objective

[2]The only pair that deviated from this pattern consisted of two near-minimal performance solutions. Their resulting distributions were more or less the same.

fitness. We are exploring this effect further using a model of the system.

The ease with which brittle solutions are found in the presence of variable mutation rates suggests that brittle solutions are rather common in the search space. Thus, a key feature of population-based search would seem to be finding solutions that are not only of higher fitness, but that also have high-fitness neighbors in the search space (non-brittle solutions). This observation is further supported by the observation that simple hill-climbers do extremely poorly in this domain[3]. These hill-climbers may simply be finding brittle solutions.

Another interesting note is that the notion of "mutationally-brittle solutions" closely resembles that of "local optima"; both refer to points in the search space from which others of higher fitness are hard to reach. Consideration of this similarity suggests parallels between (a) the formation of brittle solutions with low mutation rates, and (b) becoming trapped below a threshold mutation value in the bowl effect. Further study of this parallel may yield insights into the relationships between local optima, variable mutation rate and rich, many-to-one representations.

In light of the above insights, there may still exist some mechanism for operationalizing self-adapting rates in this domain. However, the ability of individuals to implicitly self-adapt in this domain suggests that we may already be enjoying the benefits that such a mechanism would supposedly provide.

Acknowledgments

The authors wish to thank James Thomas for numerous helpful discussions. This research was partially supported by the Office of Naval Research, Contracts N00014-95-1-0591 and N00014-96-1222.

Bibliography

[Altenberg, 1985] L. Altenberg. Knowledge representation in the genome: new genes, exons, and pleiotropy. *Genetics*, 110(supplement: s41), 1985.

[Altenberg, 1994] Lee Altenberg. The evolution of evolvability in genetic programming. In K. Kinnear, editor, *Advances in Genetic Programming*. MIT Press, 1994.

[Angeline *et al.*, 1994] Peter J. Angeline, Gregory M. Saunders, and Jordan B. Pollack. An evolutionary algorithm that constructs recurrent neural networks. *IEEE Transactions on Neural Networks*, 5(1):54–65, 1994. Special Issue on Evolutionary Computation.

[3] We found similar performance in limited experiments with other, more sophisticated non-population-based hill-climbers, but more experimentation must be done.

[Angeline, 1995] P. J. Angeline. Adaptive and self-adaptive evolutionary computations. In M. Palaniswami, Y. Attikiouzel, R. Marks, D. Fogel, and T. Fukuda, editors, *Computational Intelligence: A Dynamic Systems Perspective*, pages 152–163. IEEE Press, Piscataway, NJ, 1995.

[Bäck, 1992a] T. Bäck. The interaction of mutation rate, selection, and self-adaptation within a genetic algorithm. In R. Männer and B. Manderick, editors, *Parallel Problem Solving 2*, Amsterdam, 1992. Elsevier.

[Bäck, 1992b] Thomas Bäck. Self-adaptation in genetic algorithms. In Francisco J. Varela and Paul Bourgine, editors, *Towards a Pratice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, Cambridge, MA, 1992. MIT Press.

[Elman, 1990] J.L. Elman. Finding structure in time. *Cognitive Science*, 14:179–211, 1990.

[Fogel *et al.*, 1966] L.J. Fogel, A.J. Owens, and M.J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley & Sons, New York, 1966.

[Fogel *et al.*, 1991] D. B. Fogel, L. J. Fogel, and J. W. Atmar. Meta-evolutionary programming. In R. R. Chen, editor, *Proceedings of 25th Asilomar Conference on Signals, Systems and Computers*, pages 540–545, Pacific Grove, California, 1991.

[Glickman and Sycara, 1998] Matthew Glickman and Katia Sycara. Evolutionary algorithms: Exploring the dynamics of self-adaptation. In *Genetic Programming 1998: Proceedings of the Third Annual Conference*, San Francisco, CA, 1998. Morgan Kaufmann.

[Glickman and Sycara, 1999] Matthew Glickman and Katia Sycara. Evolution of goal-directed behavior from limited information in a complex environment. In *GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference*, 1999.

[Levenick, 1991] J. R. Levenick. Inserting introns improves genetic algorithm success rate: Taking a cue from biology. In R. K. Belew and L. B. Booker, editors, *Proceedings of the Fourth International Conference on Genetic Algorithms*, San Mateo: CA, 1991. Morgan Kaufmann.

[Schwefel, 1981] H. P. Schwefel. *Numerical Optimization of Computer Models*. Wiley, Chichester, 1981.

[Slatkin, 1970] M. Slatkin. Selection and polygenic characters. *Proceedings of the National Academy of Sciences U.S.A.*, 66:87–93, 1970.