Comparing Mechanisms for Evolving Evolvability

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

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

Abstract

In evolutionary search, the term *evolvability* as defined in [Altenberg 94] refers to "the ability of a population to produce variants fitter than any yet existing". In this paper, we examine a few existing mechanisms which provide the potential for the evolvability of a population to itself evolve. One key property that we identify among such mechanisms is a many-to-one genotype-to-phenotype mapping, which permits variations in evolvability to occur independent of fitness. Another is the propensity for individuals to become increasingly conservative in parent-offspring transmission as they become more fit, a phenomenon which becomes stronger as selection pressure becomes weaker.

1 Introduction

Given a target search space, P (the set of phenotypes), a *representation* of this space maps a set of representative objects, G (the set of genotypes), to points in P. Evolutionary search employs *search operators* to act on encoded objects in a population, the parents, to produce new encoded objects, the offspring. For example, the target search space might be the set of real numbers, the representative objects could be strings of 32 bits, the mapping could be the IEEE floating-point specification, and a search operator could be probabilistic bitwise mutation.

The search operators define a transition graph, O_G, over encoded objects with directed links which specify which points in G may be reached from which others. Because evolutionary search is stochastic, each transition has a probability of occurrence that is dependent upon the specific choice of search operators.

The representation projects O_G onto the target search space. The resulting graph, O_P, indicates the probabilities of possible transitions *in the target search space* using the chosen representation and search operators, and is of major relevance to the outcome of search. For instance, evolutionary search will stagnate when the population reaches a local optimum, a point from which all other neighboring points–those reachable with some reasonable probability via the application of search operators–are of lesser fitness. Exactly which points are indeed reachable from, *i.e.* are neighbors of, any given point is determined by O_P. Thus, a joint effect of the specific choices of representation and search operators is to determine the distribution of local optima in the space and hence to largely determine the potential evolvability of populations evolving within it This joint effect is summarized via the notion of a parent-offspring *transmission function* (via [Altenberg 94]), which maps parent points to a probability distribution from which their offspring are sampled.

Because the choices of representation and search operators are often difficult to make (particularly for challenging search spaces), it is an attractive option to permit these choices, or aspects thereof, to themselves evolve (as is likely the case in nature). Such a scheme may provide for populations to evolve the transition probabilities in O_P to enhance their evolvability in the target search space, *i.e.* to more often yield offspring of higher fitness than any yet existing.

There are innumerable possible mechanisms via which traits related to evolvability (or parent-offspring transmission) may evolve in artificial evolution. We examine a couple of these, evolvable mutation rates and evolution of representation in Genetic Programming, and then in light of our observations evaluate a simple learning mechanism with respect to its ability to foster the evolution of evolvability.

2 Per-Individual Mutation Rates

Rather than specifying a fixed, global mutation rate, mutation rates may be "self-adapted" on a per-individual basis by simply encoding a mutation rate along with each individual candidate solution [Schwefel 81]. When an individual is then mutated, it is done using its own encoded rate. As evolution proceeds, those individuals with mutation rates best adapted to themselves produce the most fit offspring in the population, hence propagating the well-adapted mutation rates. In the most typical case, the trend is for mutation rates to decrease over time because as individuals become more and more fit, higher rates become increasingly likely to disrupt their fitness. Because mutation rates must continue to adapt as individuals change, the mutation rates themselves are also mutated (typically via another fixed, global rate).

Selection for mutation rates which entail the production of more offspring of higher relative fitness constitutes a clear example of a mechanism for the evolution of evolvability. We make a number of observations from this example, each of which would appear to generalize to other such mechanisms.

The first observation is that selection for mutation rates appears to be an inherently weaker force than selection for objective traits. We can see this by first noting that selection operates directly on classical, objective traits because such traits play a direct role in an individual's evaluation by the fitness function. While selection is itself often a stochastic process, selection for mutation rates is diluted by an additional stochastic step because the effect of mutation rates is only via their stochastic likelihood to yield offspring that are possessed of either more or less favorable combinations of objective traits. Even if selection is completely deterministic, due to sampling error a poorly adapted mutation rate will occasionally yield offspring of higher fitness than those of a well adapted rate, resulting in selection favoring the poor rate over the better one.

Another interesting observation is that the strength of selection has a clear influence on which mutation rates are favored (see [Glickman and Sycara 98] for a more complete treatment). For example, two individuals which are the same 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; the individual with a higher mutation rate will likely produce offspring with a lower mean fitness but with a higher chance of occasionally producing particularly high fitness individuals. 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.

The final observation we draw from this example is that practitioners of Evolution Strategies [Schwefel 81] have indicated the dangers of "elitist selection" for self-adaptation of mutation rates. Elitist selection refers to the practice of always retaining in the population from generation to generation the highest-fitness individual(s) yet found, and has been demonstrated to strongly cripple rate adaptation. In light of the above discussion of the weakness of selection for mutation rates, this result is not surprising. Retaining individuals in the population instead of forcing them to reproduce and compete via their offsprings' fitness partially bypasses the circumstances via which selection acts on mutation rates; once an individual of particularly high fitness but with a poorly adapted mutation rate appears, it may remain in the population for several generations while contributing only low fitness offspring.

3 GP and Neural Networks

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. [Altenberg 94] extensively explores the evolution of evolvability in GP.

As noted in connection with self-adapting mutation rates, as the individuals in the population become more fit, the action of the search operators becomes increasingly disruptive to their ability to transmit this fitness to their offspring; a more fit individual usually has "more to lose" in the variation process. One phenomenon addressed by Altenberg concerns how evolving programs often become over time increasingly 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 produced when subjected to 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. Such a variation that can make a program more conservative is the addition of code which plays no role in program behavior. The more functionless code there is in a program, the more likely variation will occur in this code rather than in a

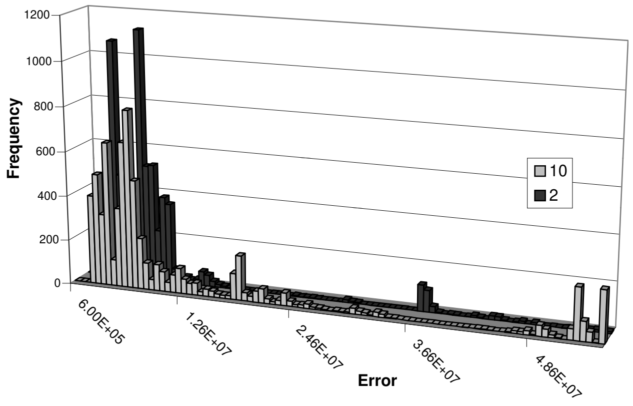

Figure 1: The fitness distributions (lower error values are higher fitness) of the offspring produced from ANN's evolved under stronger (tournament size of 10) and weaker (tournament size of 2) selection pressure.

behavior-critical section.

If, however, 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, as discussed above, selection on objective, directly fitness-related traits appears stronger than on transmission-related traits like mutation rates, this situation would likely inhibit selection for evolvability-related traits such as conservatism.

Noting the fact that per-individual mutation rates also provide a many-to-one genotype-to-phenotype mapping, we decided to further test the hypothesis that such a many-to-one representation provides for the evolution of evolvability by conducting experiments evolving connection weight vectors for artificial neural networks (ANN's) for an extended period of time at different selection strengths. We supposed that for the chosen maze-navigation task (see [Glickman and Sycara 99] for details) there could be many possible variations in ANN connection weight space that would be phenotypically-neutral but transmission-relevant. After extended evolution, we chose several individuals evolved with both selection strengths that had similar fitness scores, generated $10,000$ offspring from each, and measured their resulting fitnesses (depicted in figure 1).

Just as self-adaptive mutation rates are selected to be lower with less selection pressure, we observed that networks evolved with less selection pressure (with a tournament size of 2) indeed transmitted their fitness to their offspring more conservatively than those evolved with stronger selection, yielding more offspring clustered around their parents' high fitness (low error) values.

4 Learning as Evolvability

The final mechanism we treat is from Hinton and Nowlan's influential paper presenting a simulation of the effect of learning upon evolution [Hinton and Nowlan 87]. They present an intractable search space, a classic "needle in a haystack" problem: Each individual consists of a string of length 20 composed of the characters `0` and `1`. The evaluation function only awards fitness points to a string of all `1`'s; a string with one or more `0`'s receives none. Up until the solution point is already reached, there is no fitness upon which selection can work, rendering evolution equivalent to random search in a space of size 2^{20}.

They then provide a rudimentary learning mechanism by adding a third symbol to their strings' alphabet, `?`. Upon fitness evaluation, a `?` has an equal chance of being interpreted as either a `0` or a `1`. During each generation, each individual is evaluated 1000 times. If during an evaluation an individual is interpreted as all `1`'s, it is said to have learned the correct configuration. No further evaluations are conducted and it is awarded a fitness score of $1000 - n$, where n is the number of evaluations that the individual had already undergone. The more `1`'s there are in an individual consisting only of `1`'s and `?`'s, the sooner it is expected to hit upon the correct configuration and the higher its expected fitness. The resulting fitness scores prove sufficient to engage selection such that individuals consisting of all `1`'s are generated after far fewer generations than would be expected from evolution without learning.

Hinton and Nowlan's `?`'s appeared to us as an interesting counterpoint to variable mutation rates: Both mechanisms provide evolution the ability to attenuate exploration, but while variable mutation rates act at the operator level via probabilistic genotype transmission, the `?` mechanism acts at the representation level via probabilistic genotype-to-phenotype expression.

As learning most often entails some sense of "improvement", the true sense in which actual learning occurs in the `?` mechanism is when evaluation halts once a string is interpreted as all `1`'s. One of our findings has been that the system actually evolves faster when the learning component is removed, *i.e.* when evaluation doesn't halt at all `1`'s, but instead continues to yield a fitness score equal to the total number of times all `1`'s were interpreted (see figure 2). The enhancement of evolvability thus seems more dependent on the broader circumstance of repeatedly measuring fitness over the individual's lifetime rather than on any specific notion of learning (although this appears unimportant for Hinton and Nowlan's demonstration). Despite the fact that our implementation substitutes a mutation operator for crossover and tournament selection for their fitness-proportional scheme, this result appears general in that continued evaluation/exploration for the full lifetime simply provides more sampling trials

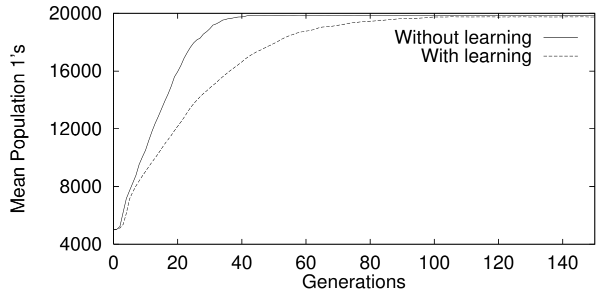

Figure 2: The mean number of ones in the population over time, with and without learning

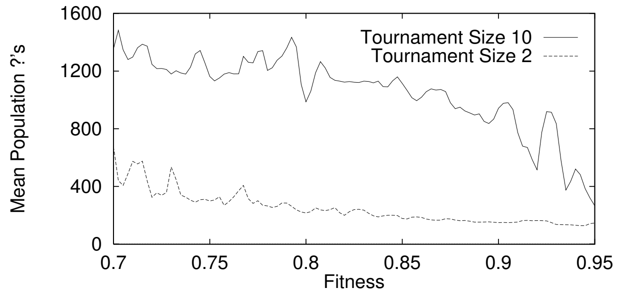

Figure 3: The mean number of question marks in the population vs. population mean fitness (for higher fitness regions) for differing selection strengths on NK landscapes (with $N = 20$, $K = 15$)

per individual as information for selection to use to better identify the individuals closest to the solution.

We also found that the Hinton and Nowlan mechanism does *not* readily exhibit the conservatism/selection-strength relationship, *i.e.* the number of ?'s selected does not appear affected by the selection strength. Two circumstances seem to explain this finding. The first is a function of the interaction between the ? mechanism and the problem domain. Once 0's are selected out, the number of ?'s in a string strongly inversely correlates with its expected fitness in the "needle" problem domain; ?'s are selected against simply because of their direct effect on fitness. This state of affairs is only compounded by the practice of repeatedly measuring fitness. The more times each individual's fitness is measured per generation, the more reliably individuals with the same number of ?'s are evaluated as equivalent phenotypes and receive the same fitness scores, yielding a one-to-one genotype-to-phenotype mapping

We eliminated these two factors by first conducting evolution in a search space replete with local optima (an NK landscape), so that having more ?'s could either raise or lower fitness, thus breaking the inverse correlation between ?'s and fitness. Then, by sampling fitness only once, many genotypes can map to the same phenotype (as well as single genotypes to multiple phenotypes). Upon making these two modifications, we did indeed see the selection-strength/conservatism effect return (see figure 3).

5 Conclusions

While the ? mechanism does indeed enhance evolvability, the enhancement does not appear directly related to learning. Moreover, evolvability does not itself appear to evolve (without substantial modifications) because of the tight coupling between the number of ?'s and fitness. However, the re-appearance of the relationship between conservatism and selection strength after modifications lends further support to the importance of a many-to-one genotype-to-phenotype mapping, as well as demonstrates that selection can act similarly upon traits affecting the probability of genotype transmission as well of genotype expression. We are currently exploring other probabilistic expression mechanisms for which the probability of expression can vary more independently.

As for the general observations above, some, such as the importance of a many-to-one genotype-to-phenotype mapping and of non-elitist selection, indicate particular properties that appear desirable to promote evolution of evolvability. Others, such as the relative weakness of selection for evolvability and the relationship between selection strength and conservatism, indicate more general dynamics to consider when accounting for the evolution of evolvability.

6 Bibliography

Altenberg, L. (1994). The evolution of evolvability in genetic programming. In: Advances in Genetic Programming, K. E. Kinnear Jr., ed. MIT Press.

Glickman, M. and Sycara, K. (1998). Evolutionary Algorithms: Exploring the Dynamics of Self-Adaptation. In J. R. Koza et al., editors, Genetic Programming 1998: Proceedings of the Third Annual Conference, Morgan Kauffmann.

Glickman, M. and Sycara, K. (1999). Evolution of goal-directed behavior from limited information in a complex environment. In W. Banzhaf et al., editors, GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference. Morgan Kaufmann.

Hinton, G. E. and Nowlan, S. J. (1987). How learning can guide evolution. In: Complex Systems, 1:495-502.

Koza, J. R. (1992). Genetic Programming: On the Programming of Computers by the Means of Natural Selection. MIT Press, 1992.

Schwefel, H.P. (1981). Numerical Optimization of Computer Models. Wiley.