Elitism, Fitness, and Growth

Gerardo Gonzalez
Robotics, Evolution, Adaptation, and Learning Laboratory (REAL Lab)
School of Computer Science
University of Oklahoma
Norman, OK USA
ggonzalez@ou.edu

Dean F. Hougen
Robotics, Evolution, Adaptation, and Learning Laboratory (REAL Lab)
School of Computer Science
University of Oklahoma
Norman, OK USA
hougen@ou.edu

ABSTRACT

Bloat is a well known phenomenon that may occur when evolutionary mechanisms allow for chromosome growth. Recently it has been shown that elitism can inhibit this bloat by constraining the growth of chromosomes. In this paper we study more closely the interaction effects between the fitness landscape, growth, and elitism when the fitness search stagnates. Our results show that in some cases elitism does not constrain the growth. Our results also show that in some cases elitism can stall the search completely, and that elitism does not cause a significant improvement in performance. We also look at elitism on fitness landscapes with different fitness slopes. All these results are informative, although whether elitism is a beneficial way to constrain growth in certain circumstances remains an open question.

Categories and Subject Descriptors

I.2.6 [**Artificial Intelligence**]: Learning—*parameter learning*

General Terms

Algorithms

Keywords

Genetic Algorithms, Elitism, Fitness, Bloat, Resilience

1. INTRODUCTION

In many evolutionary methods, variable-length chromosome representations are used. In these methods, solutions to a problem (chromosomes) usually get larger to accommodate more information. Two common evolutionary methods under the growing-chromosome paradigm are Genetic Programming (GP) and Neural Evolution (NE). In GP, evolution finds a program to solve a problem. In certain types of NE like SANE, ESP [6] and NEAT [7], neural networks are evolved from smaller to larger networks.

When growth is allowed in an evolutionary method, we hope to find sizes for the solutions that allow for the problem to be solved correctly, yet small enough that the solution is not wasteful. For example, if a NE problem can be solved with only 20 neural units, yet the evolutionary process creates a network with 50 units, then, when the network is deployed, it will require far more computations per input/output combination than are necessary, wasting CPU cycles and potentially slowing the deployed system. Unfortunately, in many cases we do end up evolving solutions that are much larger than what we need. In GP this problem is commonly known as bloat [1].

The difference between bloat and growth is that in bloat we keep on adding non-functional parts to our solution. In GP this might look something like `if(false)<foo>`. This is non-functional, because whatever is represented here by `<foo>` (calculate a value, execute a procedure, etc.) will never be executed, so this code makes the solution bigger without affecting the fitness. In NE with growing networks, bloat could keep on adding new nodes with all input (or output) connections having 0 weights. These would not affect the output of the network, yet will make it bigger, and waste computation cycles. In both GP and NE, more complex non-functional structures are possible as well. In a GP calculating a value, for instance, we might have a node that multiplies the value by a constant and another node that divides the value by the same constant, resulting in the original value after both operations have been completed. Similarly, in NE we could have two neural nodes with offsetting weights, resulting in a network that outputs the same values as an otherwise identical network with both nodes removed. Theoretically, such structures could exist for an arbitrary number of nodes for both GP and NE.

Bloat is also detrimental to the search process because solutions become unnecessarily computationally expensive, and therefore the amount of computation that is needed to evaluate solutions as they are being evolved is so great that the search slows down to unpractical levels. This brings a dilemma: if growth is constrained, a suitable solution to a problem might not be found, but if the growth is unbounded we might evolve inefficient solutions. Many different solutions to the bloat problem have been proposed [3], but lately it has been shown that elitism can constrain the growth of solutions in certain GP problems [4]. Elitism seems to be a natural and simple way to constrain growth. However, a greater understanding of this phenomenon can help us to make better decisions as to whether and when elitism can be used and for what purposes. In this paper we will show, through a simple demonstration problem, some interactions between elitism, chromosome growth, and the shape of the fitness landscape when fitness *stagnates*, that is when changes in fitness values slow greatly or cease entirely during some portion of an evolutionary search. In particular, we are interested in stagnation caused by relatively flat plateaus in the fitness landscape bounding the upward climb in fitness while selection prevents large drops in average population fitness. (New chromosomes may, of course, have reduced fitness as a result of the destructive potential of crossover or mutation. However, as these chromosomes are selected against, the average population fitness will not be greatly reduced.) This period of fitness stagnation in an evolutionary search was specifically chosen because in previous work [4] it is claimed that stagnation seems to be the time when elites constrain the growth of chromosomes.

A GP problem in which stagnation would likely be seen is the artificial ant problem with a trail containing a very large gap. Assume that there is no time limit. The GP will likely evolve solutions to find most or all of the food prior to the gap. As soon as all of that food is accounted for, the evolutionary process will go into stagnation, not being able to readily find solutions with greater fitness (a fitness plateau).

Our results show four main points. First, when fitness becomes stagnant but chromosomes do not lose any fitness through crossover, elitism will not constrain the growth of the chromosomes. Second, when fitness becomes stagnant but solutions generally lose small amounts of fitness during crossover, growth will be halted. Also, populations will tend to converge to a lower average chromosome size, and the size at which convergence happens will reflect the number of elites in the population. Third, we will show that elitism produces no significant effect on the average fitness of the population. Finally, we show that if there is another possible increase of fitness after the period of stagnation, even very small amounts of elitism could cause the search to fail to find the other section of the landscape. These results will help us make better judgments when using different parameters in our experiments, and they create a basis to search for some of these effects in more standard GP or NE problems.

2. METHODS

For our experiments we used a genetic algorithm (GA) very similar to the one used by Soule [5]. The GA proceeds in the following manner.

```
Initialize:
  population = CreatePopulation(population_size)
  generation = 1
Repeat:
  CalculateFitness(population)
  new_population[1..num_elites]
      = CloneElites(num_elites,population)
  For (counter = num_elites + 1,
      counter <= population_size,
      counter++){
    individual1
        = Tournament(tournament_size,population)
    if (rand(0..1) < mutation_rate)
        Mutate(individual1)
    if (rand(0..1) < crossover_rate){
      individual2
          = Tournament(tournament_size, population)
      if (rand(0..1) < mutation_rate)
          Mutate(individual2)
      child1, child2
          = Crossover(individual1, individual2)
      new_population[counter] = child1
      counter++
      new_population[counter] = child2
    }
    else
      new_population[counter] = individual1
  }
  population = new_population
  generation++
Until (generation == max_generations)
```

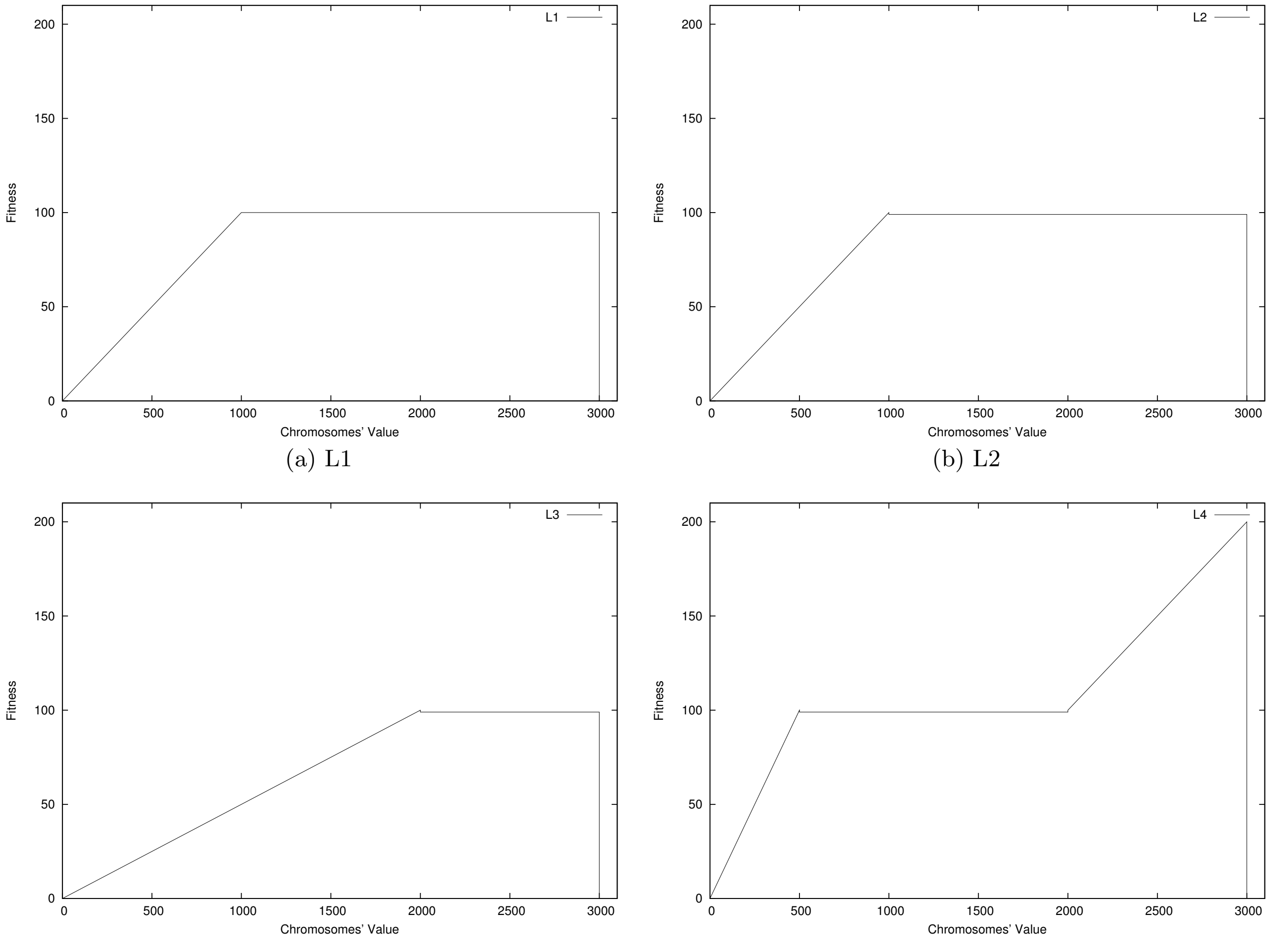
The chromosomes are arrays of integers with three possible allele values: 0, 1, or 4. When the chromosomes are created for the initial population, we choose a random size between 10 and 15 uniformly, then with equal probability we assign one of the alleles to each of the genes in the chromosome. To evaluate a chromosome, the values of all of the genes in the chromosome are added. Then, the summed value is put through a fitness function that tells us how fit that chromosome is. In our experiments we used four different fitness functions: L1–L4. In Figure 1 we can see the different "landscapes" of these functions. (Note that these are not fitness landscapes in the traditional sense [8], because we are plotting summed chromosome values versus fitness, rather than looking at fitness as it relates to similarities with respect to chromosomes and operators on them.)

The landscapes are simple, but there is a subtle difference that should be noticed. The plateau in L1 is at y=100, while in the rest of the landscapes the plateau is at y=99. L1 simulates the situation in which the fitness becomes stagnant and the chromosomes do not lose fitness at all through crossover. L2 and L3 simulate fitness stagnation but with a small loss of fitness when chromosomes cross over (a more common scenario), but with different fitness slopes. L4 has the steepest initial slope, and it has another slope after the period of stagnation.

What allows the growth of the chromosomes is a special type of crossover called *constant crossover* [5]. The idea is to take from each of the parents for exchange a segment that is largely independent of the size of the parent chromosomes. The pseudo-code of the algorithm for segment size determination is:

```
starting_point = rand(0..Size(chromosome))
segment_size = 2
while ((rand(0..1) > 0.5)
  segment_size = segment_size * 2
segment_size = Min(segment_size,
    Size(chromosome) - starting_point)
```

What this algorithm means is that with 50% probability a parent will donate a segment of size 2, with 25% probability a parent will donate a segment of size 4, and so on, given that the segment selected does not run beyond the end of the chromosome. After the segments have been chosen for each of the parents, they are exchanged between parents to produce two new offspring. Since the segments donated by each parent may be of different sizes, one of the new offspring



(a) L1 (b) L2 (c) L3 (d) L4

Figure 1: Fitness landscapes used in the different experiments.

Table 1: GA Parameters

Population Size	500
Initial Chromosome Size	random in interval [10–15]
Tournament Selection Size	3
Mutation %	0
Crossover %	90

may become larger while the other becomes smaller. It is important to note that the elites are never crossed over (they are copied exactly from one generation to the next), and that the rest of the population will come 90% from crossover, and 10% from copies of potentially non-elite individuals. We do not use mutation at all in these experiments.

The rest of the parameters for the experiments are described in Table 1.

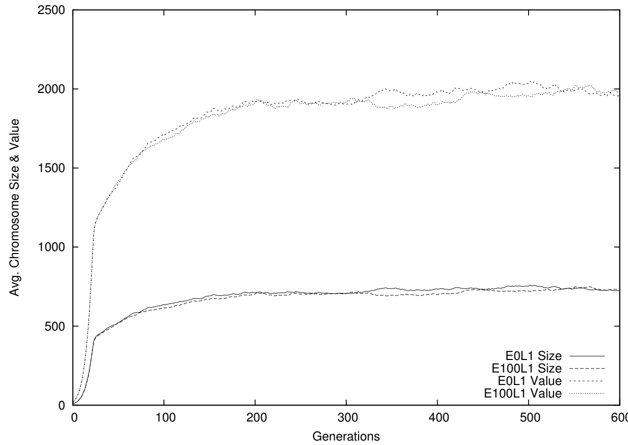
3. RESULTS

Each experiment was run 30 times, and our experimental data shows the averages of the 30 trials.

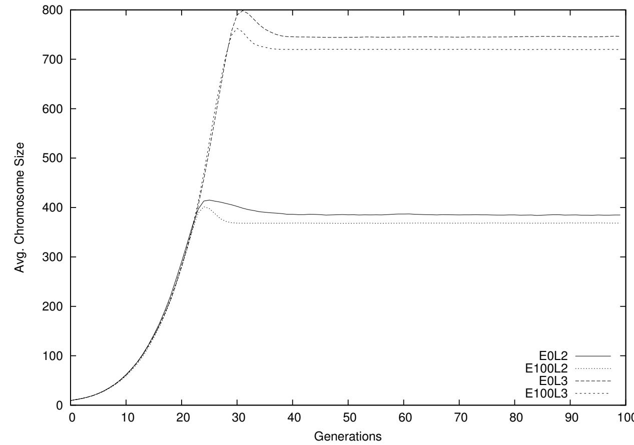
In Figure 2(a), for both elitism cases, we can see two stages in the results. The first stage shows a rapid growth pattern. Afterward, in stage two, the chromosomes continue to grow but less rapidly and less consistently. It appears that the population is slowly drifting toward larger chromosomes. (By "drift," we are referring to the slow increase in the average population size when there is no fitness increase due to growth. This is distinct from, though similar to, genetic drift, in which gene frequencies change over time without being driven by selection pressures.)

In Figure 2(b), we can see a two stage pattern similar to the one seen in Figure 2(a). The main difference is that in stage two of Figure 2(b), in which growth stalls, chromosome size actually decreases slightly, then finally stays relatively constant. It is important to note the gap between the convergences for the 0 elite experiments and the 100 elite experiments. Not only is there a gap between the elite and non-elite cases for each landscape used, but there is a gap between the two landscapes.

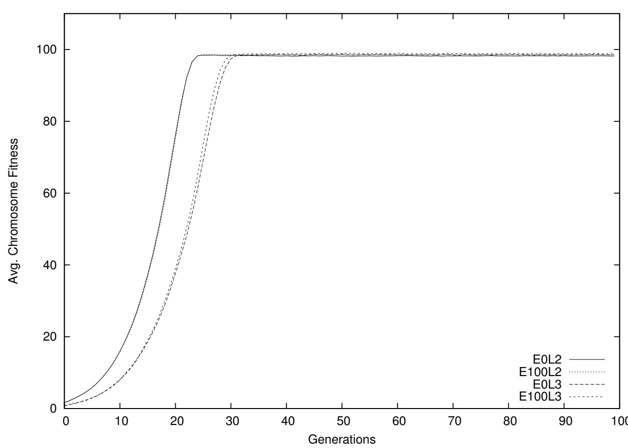
The most important thing to note about Figure 2(c) is that the use of elitism (in this case elitism of 20% which is a very high level of elitism) does not produce a very substantial effect on the performance of the algorithm (performance is being measured by how fit the population is at a given generation). Because the difference appears small, we tested the performance curves using a randomized ANOVA method as the test [2]. As we would expect for L2, there



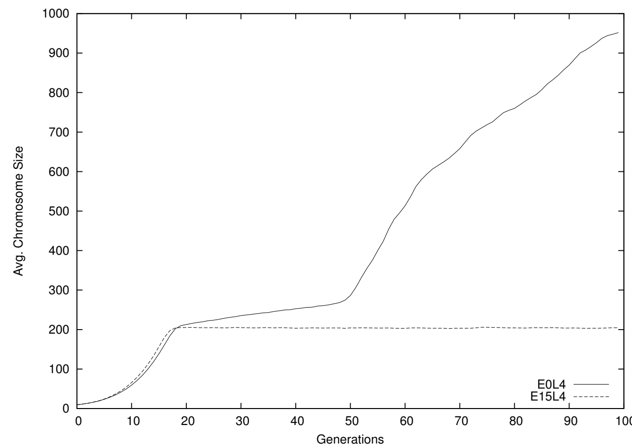
(a) Average chromosome size for landscape L1 using 0 and 100 elites (E0 and E100, respectively).



(b) Average chromosome size for landscapes L2 and L3 using 0 and 100 elites.



(c) Average fitness for landscapes L2 and L3 using 0 and 100 elites.



(d) Average chromosome size for landscape L4 using 0 and 15 elites (E0 and E15, respectively).

Figure 2: Average population values. Note different vertical and horizontal axes labels and ranges between subfigures.

is no statistically significant difference (algorithm p-value < 0.15; interaction p-value < 0.30). For L3, even though the difference is small, it is statistically significant (algorithm p-value < 0.0001; interaction p-value < 0.02).

In Figure 2(d), we can see a drift pattern for the case with zero elites. Then, after a few generations of drifting toward greater sizes, the population goes back to a rapid increase. The case with 15 elites does not drift substantially, and it never goes into the rapid increase seen with zero elites.

4. DISCUSSION

The experiments were designed to show particular effects in the interaction between growth and elitism in several cases of fitness stagnation. In the first set of experiments, shown in Figure 2(a), we wanted to show that if the population does not lose any fitness after fitness stagnates (the fitness at the end of the slope is equal to the fitness through the plateau), that elitism would not eliminate a drift in the size of the chromosomes when the plateau was reached in the landscape.

This drift is an interesting effect because once the plateau is reached, there is no fitness difference between larger and smaller chromosomes. What is important to note is that the figure shows the average population size. As the plateau is reached by the bulk of the population, there will still be some chromosomes on the slope, and some of them will be on the plateau. Chromosomes still on the slope will tend to be selected against until all chromosomes are on the plateau. Moreover, during crossover, chromosomes near the slope may lose valuable alleles (those with values of 1 or 4) and their offspring may therefore lie on the slope. When this happens, the offspring of some of the chromosomes on the plateau but near to the slope will be selected against, in favor of the offspring of longer chromosomes that are further from the slope, until no chromosomes in the population are likely to have offspring on the slope. This may be due to a kind of resilience in which an increased value of the chromosome does not increase its own fitness but does mean that its offspring are less likely to have reduced fitness. Alternately or in addition, this may be due to a one-sided version of the type of resilience seen by Soule [5]. In this type of resilience, the greater value of the chromosome does not provide the resilience; it is provided by the greater length of the chromosomes. As a chromosome grows longer, even if its value remains unchanged, the constant crossover operation will, on average, remove a smaller portion of the chromosome. For a longer chromosome to have the same value as a shorter chromosome, the longer one must contain more 0 alleles or have fewer 4 alleles and more 1 alleles. In either case, the crossover is then more likely to remove fewer valuable alleles, which increases the chance that the resulting offspring is on the plateau, rather than on the slope.

This is why we see a drifting effect as longer and higher value chromosomes come to dominate the population. Once all of the shorter/lower value chromosomes have been replaced, we see a convergence to a certain size, since no offspring chromosome has fitness less than that of the plateau. Also, because there is no fitness incentive for the population to be in a particular place, as the population drifts, the variance of the population's size gets larger and larger. This is another factor in favor of the drift, since as variance increases, those at the bottom end of the range will again approach the slope and be selected against.

The phenomenon seen in Figure 2(a) is interesting, but perhaps is not very common. In many problems, when fitness stagnation sets in, small changes in the chromosomes can cause loss of fitness. This is why in L2 and L3 we set the plateau at 99 (one unit lower than the end of the slope). We choose 99 instead of a lower value because we wanted to show how great the effect of this small change can be. The only difference between L1 and L2 is this small change in the plateau, yet in Figure 2(b) we can see that even without any elites, the drifting effect is not there anymore.

The difference between L2 and L3 is that the slope in L3 is half as steep as the slope in L2. To better understand what is going on in Figure 2(b) we will say that L2 is a *faster slope* and that L3 is a *slower slope*. L2 is faster because it is shorter, so that the population can climb it faster. L3 is slower because it is longer and to reach the plateau the chromosomes will have to be longer, taking more generations to reach the end of the slope. The reason why this is important is that it will help explain some of the small details in the graphs. For example, the rate at which the population decreases in size after stalling is greater in the slower slope than in the faster slope for the non-elite case. In the elite case there does not seem to be much of a difference.

Examining fitness variance shows that the increased effect in the slower slope is caused by the population being more tightly grouped together (having less variance) as it climbs, causing more members of the population to find the peak at the same time. Because of the slower speed of the slope, fewer chromosomes overshoot the end of the slope (becoming larger than necessary), so that there are more small chromosomes causing the decrease in size, and convergence to a smaller size.

In the faster slope, more chromosomes overshoot the end of the slope, so that quite a few chromosomes have a fitness of 99 instead of 100. This is eventually fixed as some members of the population find the peak but, as we can see, the decrease in size back to convergence takes much longer than it did for the slower slope. Just as the slow slope causes the population to have lower fitness variance (making the population tighter), the experiments in the fast slope showed greater fitness variance causing the slower return to the smaller size at which the population converged.

The last thing to discuss is the gap between the no elites case and 100 elites case for both landscapes. We believe that elitism ends up causing the population to converge to smaller chromosome sizes because elitism allows for the population to use outliers. What this means is that if the average of the population fitness is fairly low on the slope, but there is one outlier at the end of the slope, with elitism the population will pass this outlier on to the next generation from then on so that it can have a greater chance to reproduce and cause more offspring to be closer to 100. Without elitism these outliers could be easily lost if there is not enough selection pressure. In the slow slope this effect can be easily seen since the experiment with elites finds the end of the slope faster. This is not so distinguishable in the fast slope, but at the same time the gap between elites and no elites in the fast slope is not as large as the one in the slow slope.

Figure 2(c) shows that there is not a large difference in performance between the cases with and without elites. The importance of this result is one of judgment. We have seen that elitism can decrease the size of the chromosomes but in Figure 2(d) we have a case in which even a small amount of elitism caused the population to not be able to find the second slope in the landscape (L4) so that the fitness of the population could continue to increase. So, if elitism can have negative effects (in some circumstances), and if elitism does not offer substantial benefits in performance, is the small decrease in chromosome size worth the risk of halting the search?

We believe that similar effects will be seen in more authentic problems and the understanding developed in the present work regarding the interaction between growth, elitism, and fitness landscapes will help us to identify salient characteristics of landscapes and make decisions about the parameters to be used given the landscape.

5. CONCLUSIONS

In conclusion, we have seen how elitism can affect the size of the chromosomes in the population and under which conditions. We have also examined the effects of elitism, under different fitness landscape conditions, and we have analyzed the possible benefits and risks of these interactions. More specifically, we looked at four effects caused by different conditions. First, the lack of influence of elitism when fitness becomes stagnant, but solutions do not lose any fitness through crossover. Second, the halt of the growth and the convergence to a certain size in the case when fitness becomes stagnant and solutions lose a small amount of the fitness during crossover. Third, the small effect that elites have on fitness. And last, how elitism can interrupt the search for higher fitness after stagnation. In the end we are not arguing that growth or elitism are necessarily beneficial or detrimental, we are simply presenting possible scenarios that we believe will be commonly seen in authentic evolutionary computation problems (like certain GP problems and certain types of NE) and will help identify some of the effects and patterns that might be seen.

6. FUTURE WORK

As has already been mentioned, we would like to look at more conventional GP or NE problems to see if through this knowledge we can identify specific characteristics of the fitness landscape, and ways to improve the performance of the algorithm.

7. REFERENCES

[1] T. Blickle and L. Thiele. Genetic programming and redundancy. In *Genetic Algorithms Within the Framework of Evolutionary Computation (Workshop at the 18th Annual German Conference on Artificial Intelligence)*, pages 33–38, 1994.

[2] P. Cohen, J. Piater, X. Zhang, and M. Atighetchi. A randomized ANOVA procedure for comparing performance curves. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 430–438, 1999.

[3] S. Luke and L. Panait. A comparison of bloat control methods for genetic programming. *Evolutionary Computation*, 14(3):309–344, Aug. 2006.

[4] R. Poli, N. F. McPhee, and L. Vanneschi. Elitism reduces bloat in genetic programming. In *Genetic and Evolutionary Computation Conference*, pages 1343–1344, 2008.

[5] T. Soule. Resilient individuals improve evolutionary search. *Artificial Life*, 12(1):17–34, 2006.

[6] K. O. Stanley and R. Miikkulainen. Efficient reinforcement learning through evolving neural network topologies. In *Genetic and Evolutionary Computation Conference*, pages 569–577, 2002.

[7] K. O. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. *Evolutionary Computation*, 10(2):99–127, Mar. 2006.

[8] S. Wright. The roles of mutation, inbreeding, crossbreeding, and selection in evolution. In *Proceedings of the Sixth International Congress on Genetics*, pages 355–366, 1932.