

On the Strength of Size Limits in Linear Genetic Programming

Nicholas Freitag McPhee, Alex Jarvis, and Ellery Fussell Crane

University of Minnesota, Morris, Morris MN 56267, USA

{mcphee, jarv0063, cran0117}@mrs.umn.edu,

<http://www.mrs.umn.edu/~mcphee>

Abstract. Bloat is a common and well studied problem in genetic programming. Size and depth limits are often used to combat bloat, but to date there has been little detailed exploration of the effects and biases of such limits. In this paper we present empirical and theoretical analyses of the effect of size limits on variable length linear structures. Specifically, we examine the relationship between size limits and the average size of individuals in a population and define the notion of size limit strength. When a size limit is strong, the average size of a population converges to a relatively stable value. When a size limit is weak, no such convergence occurs. The average size instead appears to perform a random walk within a bounded region. We use schema theory to show this is likely a result of uneven sampling of the search space.

1 Introduction

The causes and effects of code growth in genetic programming (GP) have been extensively researched [19,5,4,6]. In order to avoid the negative repercussions of bloat, a variety of corrective measures are commonly employed to keep program sizes in check [13,18,7,3]. One frequently used method is to employ a fixed limit on program size by restricting either the depth or the size of syntax trees.

While these limits have the desired effect of keeping the sizes down, little is known about what other impacts such limits might have on the dynamics of GP. Previous research has shown that decisions such as these can have significant effects on the behavior of runs [2] and on important structural features such as the size and shape distributions of populations [8,10,9]. It would therefore be useful to better understand what structural effects size limits might have, especially given their widespread use.

To shed some light on this issue we collected a large body of data on the impact of size limits on a test problem with variable length linear structures; this is described in Section 2. We present and analyze the empirical results in Section 3. The data shows a definite difference in behavior among size limits, leading us to define the notions of *strong* and *weak* size limits. Strong size limits cause the average size of a population to converge to a relatively stable value. Weak size limits, however, achieve no such convergence. Instead weak limits cause the population's average size to perform what we believe to be a bounded

random walk over a large area. In Section 4 we perform a theoretical analysis of our results using the theory of holes [8] based on exact schema theory for GP [16]. This analysis shows that there is a balance between the set of individuals exceeding the size limit and the set of short, unfit individuals. Section 4 also addresses how these results on variable length linear structures might generalize to N-ary trees. We discuss future work in Section 5 and provide conclusions in Section 6.

2 Terms and Concepts

2.1 Average Population Size

In this work the number of individuals in a population remains constant across the entirety of the runs. To simplify the discussion we will use phrases like “the average size of a population” to describe the average size over all the individuals in the population.

2.2 Convergent Average Size

When there is bloat pressure, the average size of individuals in any population increases rapidly during the early generations of a run. In runs using a strong size limit the average size will eventually reach a stable value from which it will seldom deviate significantly (see Fig. 2). In such cases, we define the population’s *convergent average size* to be the mean of the average size over time. A key goal of Section 3 is to better understand how size limits affect convergent average size.

2.3 Size Limit Strength

One of the important observations of Section 3 is that run dynamics are governed in a significant way by the choice of size limit. We say that a size limit is *strong* for a run if that run has a convergent average size. Otherwise we say the size limit is *weak*.

2.4 Crossover on Variable Length Linear Structures

The work reported here is all on GP with variable length linear structures (similar to those used in systems like [11]). We used linear structures because the theoretical analysis is more manageable and the computations are more tractable. This has yielded a number of important results for the linear case, and preliminary results suggest that many of the key ideas here are also applicable (at least in broad terms) to the non-linear tree structures typically used in GP (see Section 4.2).

Because our primary interest is the effect of size limits on code growth, we will focus exclusively on the standard subtree-swapping GP crossover operator

which is known to induce bloat. This operator acts by removing a non-empty suffix of an individual and replacing it with a new suffix taken from another individual.

More formally, in linear structure GP where \mathcal{F} is the set of non-terminal nodes and \mathcal{T} is the set of terminal nodes, individuals can be seen as sequences of symbols $c_0c_1 \dots c_{N-1}$ where $c_i \in \mathcal{F}$ for $i < N - 1$ and $c_{N-1} \in \mathcal{T}$.

Crossover, then, acts on two parent individuals by removing a non-empty suffix from one parent and replacing it with a non-empty suffix from the other parent. The removed suffix is $c_jc_{j+1} \dots c_{N-1}$ where j is chosen uniformly such that $0 \leq j < N$. The inserted suffix is $d_{j'}d_{j'+1} \dots d_{N'-1}$ where j' (which could differ from j) is chosen uniformly such that $0 \leq j' < N'$.

2.5 The One-Then-Zeros Problem

We chose to study the impact of size limits using the well-studied *one-then-zeros* problem. In this problem we have $\mathcal{F} = \{0, 1\}$ and $\mathcal{T} = \{0\}$, where both 0 and 1 are unary “operators”. This gives us a problem that is essentially equivalent to studying variable length strings of 0’s and 1’s, with the constraint that the strings always end in a 0. Fitness in this problem will be 1 if the string starts with a 1 and has zeros elsewhere, i.e., the string has the form $1(0)^a$ where $a > 0$; fitness will be 0 otherwise.

One of the reasons for studying this problem is that under selection and crossover this problem induces bloat [8], which ensures that size limits will have an impact. Another advantage is that this problem is amenable to schema theory analysis [15,8,10,17,9].

3 Empirical Results

3.1 Experimental Setup

All the runs presented in this paper use the same parameters except where noted.

Number of generations. All runs in Fig. 1 were for 1,000 generations, and all runs in Fig. 2 were for 2,000 generations.

Control strategy. We use a non-elitist generational control strategy.

Selection mechanism. Since all individuals have either fitness 0 or 1, we used uniform random selection from the set of individuals with fitness 1.

Initialization. The populations were initialized entirely with fit individuals having size 10, i.e., strings of the form $1(0)^9$.

Size limits. These were implemented so that an otherwise fit individual received a fitness of 0 if it’s size was *strictly greater* than the size limit. A variety of size limits were used; see below for details.

Operators. We exclusively use crossover in these experiments, so every individual is constructed by choosing two fit parents and performing linear structure crossover as described above. There is no mutation or copying of individuals from one generation to the next.

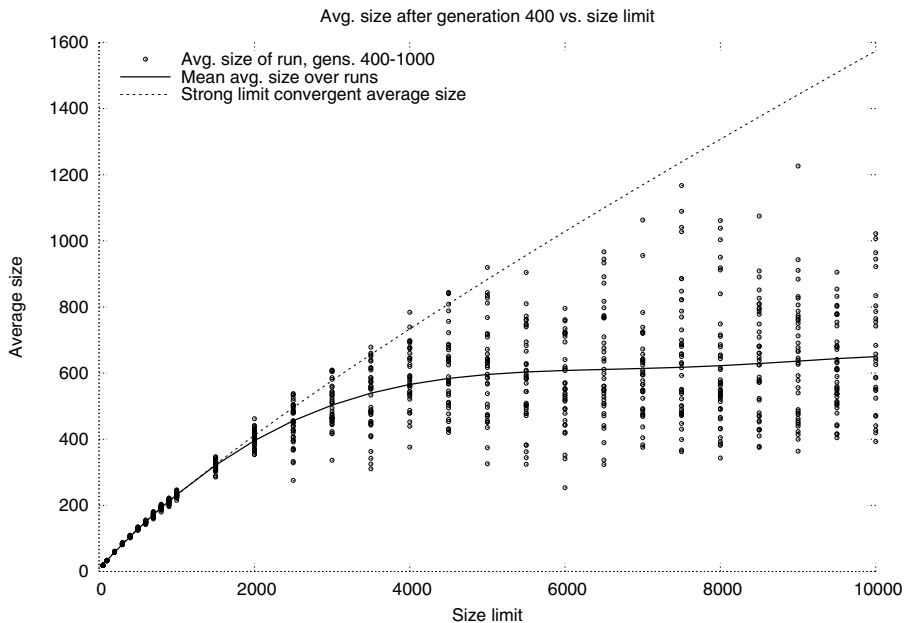


Fig. 1. The average sizes of runs from generation 400 to generation 1,000 versus the size limits used in the runs. The solid line connects the mean values of the runs at each size limit. The dotted line shows the results of a regression over the data for size limits 50 to 1,000. All data was gathered from populations of size 1,000.

In each run the average size of the population was calculated by taking the mean value of the population averages in the range of generations 400 to 1000. This region was selected because in the case of strong size limits the population had converged by generation 400.

3.2 Convergent Average Sizes and Size Limits

The graph in Fig. 1 displays the distribution of run average sizes for a variety of size limits. The average sizes remain very clustered in the lower size limits and gradually become more variable as the size limits grow. It also appears that after a certain point this variance does not increase, instead remaining fairly stable. It was this qualitative change in behavior that led us to propose the notion of size limit strength (see Section 2.3).

Fig. 1 also shows a clear change in the magnitude of the average sizes as we move from strong to weak size limits. The solid line in the figure connects the mean value of the average sizes for all runs at each size. After climbing steadily over the strong size limits, the mean of the average sizes clearly flattens for the

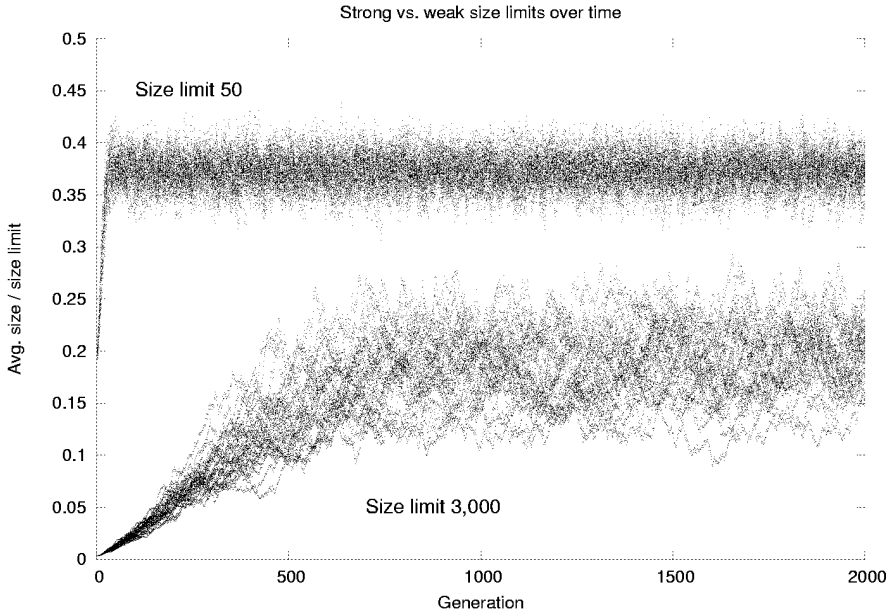


Fig. 2. Average size divided by size limit over time of multiple runs with size limits 50 and 3,000. The average sizes were divided by their respective size limits to provide a uniform scale. In all cases the population size was 1,000. The limit of 50 is clearly strong, as the average sizes of all of the runs converge onto a stable value with limited variance. The limit of 3,000 is much weaker, and appears to perform a random walk within a bounded area.

weaker limits. To better illustrate this we performed a regression on the average sizes for limits in the range 50 to 1,000 and graphed that as the dashed curve in Fig. 1. This clearly shows divergence that starts around size limit 1,500 and increases over time.

It is also interesting to note that the average sizes were consistently less than one third of the size limit in cases where the size limit was strong. The average sizes were proportionally even smaller for weak limits.

3.3 Impact of Size Limit Strength

Another method of examining the differences in size limit strength is presented in Fig. 2. This figure displays the average sizes of multiple runs where each dot is the average size across the population for a single run at a particular generation. The upper set of data comes from runs using a strong size limit of 50, whereas the lower set used the weak size limit of 3,000. In the graph the average sizes of each run were divided by the size limit to provide a uniform scale for the two data sets. The first and third quartiles for the unscaled limit 50 data are 18.11 and 19.17 respectively, so the average size typically stays within 0.5 of the mean

of 18.64. The first and third quartiles for the unscaled limit 3,000 data are 496.1 and 639.4, so the average size varies quite substantially from the mean of 571.4.

The average sizes for runs in which the size limit is 50 clearly increase rapidly to a convergent size and stay within a narrow range from then on. The upper bound is presumably generated by the size limit culling overly large individuals, and the lower bound is presumably a result of bloat pressure.

The average size of the runs using size limit 3,000, however, does not converge. The average sizes increase fairly steadily in the early generations, but seem to perform a random walk in later generations. This early behavior is an example of traditional bloat, but it's less clear what's happening later on. It appears, however, that there is a both an upper and lower bound to the random walk. The apparent lower bound is likely tied to the fact that if the average size is too small, it becomes increasingly probable that crossover will generate individuals that don't match the one-then-zeros pattern. The upper bound is presumably a function of the size limit where if the average size is too large, crossover is likely to generate individuals whose size exceeds the limit. This is discussed further in the theoretical analysis in the next section.

4 Theoretical Analysis and Discussion

4.1 Explaining the Data: The Theory of Holes

The data presented in Section 3 clearly indicates that there is a continuum of strengths for size limits in the one-then-zeros problem. The question, then, is why strong size limits tightly constrain the average size to such a narrow range, and weak limits allow the average size to drift over a substantial range of values.

In this section we'll present a theoretical analysis of these empirical results using findings from exact schema theory for genetic programming ([12,16]). This analysis suggests that the strength of a size limit is a function of how well the crossover operator samples the set of unfit individuals. In particular, we find that there is an overall balance between the sampling of the set of individuals that fail to match the one-then-zeros target pattern and the sampling of the set of individuals that exceed the size limit.

An important result from [15] uses schema theory to derive a predictive relationship between the average size of a population, the average size of its unfit individuals, and the change in the average size over time. (This was subsequently used as the basis of the Tarpeian bloat control mechanism in [13].) In this section we will present a simplified summary of those results, which can be used to better understand the empirical data presented earlier.

First assume, as is the case in the one-then-zeros problem, that the problem's fitness function has only two values: 0 and 1. We will refer to the region of the search space where the fitness is 0 as the "hole". In the standard one-then-zeros problem the hole is the set of individuals that don't match the one-then-zeros pattern. If we add size limits to the one-then-zeros problem, then the hole also contains all strings that exceed the size limit.

The key result from the theory of holes in [15] is that the average size of the population will move away from the average size of the hole. To formalize this let us first present some definitions:

- $A(i)$ to be the size of an individual i
- $|S|$ to be the size of a set (of individuals)
- $\mu(t)$ to be the average size of the population at time t
- $E[\mu(t + 1)]$ to be the expected average size of the population at time $t + 1$ given what is known about the population at time t
- $U(t)$ to be the set of unfit individuals at time t
- $\mu(U(t))$ to be the average size of the set $U(t)$, i.e., $\mu(U(t)) = \frac{\sum_{u \in U(t)} A(u)}{|U(t)|}$

Given these definitions we can express the result from [15] as:

$$\begin{aligned} \mu(U(t)) > \mu(t) &\iff E[\mu(t + 1)] < \mu(t) \\ \mu(U(t)) < \mu(t) &\iff E[\mu(t + 1)] > \mu(t) \end{aligned}$$

This tells us that if the average size of the unfit individuals is greater than the average size of the population, we would expect the population to shrink (move away from the hole). Similarly, if the average size of the unfit individuals is less than the population’s average size, we would expect the population to grow (again moving away from the hole).

Implicit in this result, but not discussed in [15], is the fact that if the average population size is relatively stable, i.e., $E[\mu(t + 1)] \approx \mu(t)$, then the average size of the hole must be approximately the same as the average size of the population, i.e., $\mu(U(t)) \approx \mu(t)$. Thus if we observe experimentally that the average size is roughly constant over a number of generations, we can infer that the average size of the unfit individuals must also be roughly the same as the average size of the population over that period of time. This result, in fact, still holds (at least in the aggregate) even if the average size varies considerably over time as long as the mean over time remains fairly constant. Thus while $\mu(U(t))$ might differ significantly from $\mu(t)$ for a particular generation t , if we average over a number of consecutive generations we would expect the mean of $\mu(U(t))$ to be close to the mean of $\mu(t)$.

The previous result holds for any GP system (linear or not) where the fitness function has just two values. We can further extend this result in the case of the one-then-zeros problem with size limits by noting that the set U of unfit individuals can be split into two disjoint sets $U = Z \cup L$, where

- $Z(t)$ is the set of individuals (at time t) that have fitness 0 because they don’t match the one-then-zeros pattern, but that are legal in the sense that they are not too large.
- $L(t)$ is the set of individuals (at time t) that are larger than the size limit, regardless of whether they match the one-then-zeros pattern.

In this case we can rewrite as follows:

$$\mu(U(t)) = \mu(t)$$

$$\begin{aligned}
 &\equiv \langle \text{Avg. length is the sum of lengths over number of strings} \rangle \\
 &\frac{\sum_{u \in U(t)} \Lambda(u)}{|U(t)|} = \frac{|U(t)|\mu(t)}{|U(t)|} \\
 &\equiv \langle \text{Splitting } U \text{ into } Z \text{ and } L \rangle \\
 &\frac{(\sum_{z \in Z(t)} \Lambda(z)) + (\sum_{l \in L(t)} \Lambda(l))}{|Z(t)| + |L(t)|} = \frac{|Z(t)|\mu(t) + |L(t)|\mu(t)}{|Z(t)| + |L(t)|} \\
 &\equiv \langle \text{Moving } \mu(t) \text{ inside the summations} \rangle \\
 &\frac{(\sum_{z \in Z(t)} (\Lambda(z) - \mu(t))) + (\sum_{l \in L(t)} (\Lambda(l) - \mu(t)))}{|Z(t)| + |L(t)|} = 0
 \end{aligned}$$

Assuming that $|Z(t)| + |L(t)| \neq 0$ (i.e., there is at least one unfit individual) this is equivalent to

$$\sum_{z \in Z(t)} (\mu(t) - \Lambda(z)) = \sum_{l \in L(t)} (\Lambda(l) - \mu(t)) \tag{1}$$

This means that if the average size is (roughly) constant we would expect the sum of the distances between $\mu(t)$ and the length of the elements of $Z(t)$ to be (roughly) the same as the sum of the distances between $\mu(t)$ and the length of the elements of $L(t)$. Since we know that all the elements of $L(t)$ are larger than the size limit, they are also presumably larger than $\mu(t)$ in almost all typical circumstances. Thus the right hand side of Eq. (1) will be positive except for pathological circumstances (e.g., having all the individuals in the initial population being larger than the size limit). Consequently the left hand side must also be positive, indicating that the bulk of the legal individuals that don't match the one-then-zeros pattern are smaller than $\mu(t)$.

Given this result, several important observations about Eq. (1) can be made. First, Eq. (1) says only that the sums are (roughly) equal, and nothing about the relative number of individuals in $|Z(t)|$ or $|L(t)|$, or the relative magnitudes of $\mu(t) - \Lambda(z)$ or $\Lambda(l) - \mu(t)$. While these can be close, it is in fact more likely that they will be quite different. The natural distribution of lengths for linear structures when using crossover alone is a discrete gamma distribution (see [15, 9]) where the majority of the individuals have below average size and are then balanced out by a smaller numbers of fairly long individuals. The same sort of distribution appears to hold even with size limits. A strong limit leads to an average size that is roughly one third of the size limit, and a weak limit leads to an average size that is even smaller. Thus what one sees in practice is a small number of individuals in $L(t)$ whose sizes are significantly greater than $\mu(t)$. They are then balanced by a larger number of individuals in $Z(t)$ whose average distance from $\mu(t)$ is considerably smaller.

The second observation is that this result is actually quite general, only depending on the ability to split U up into two disjoint groups, which will be

possible whenever size limits are employed. There are no assumptions here about linear structures so this would apply just as well to N-ary trees. There are also no reliance on the details of the one-then-zeros problem (except for the fact that it only has two fitness values), and would apply to any problem with just two fitness levels. While this may seem like a serious restriction, in practice it is not uncommon for a population to have a very limited range of fitness values, especially in the later stages of a run. This can (depending on the details of the selection mechanism) lead to an effective division of the population into two groups. One group has such low fitness that the chances of a parent coming from this group is nearly 0, and fitnesses in the other group are sufficiently similar that their probability of being selected are effectively the same. In such a situation we would expect Eq. (1) to hold – at least approximately.

The final observation is that these results and those in [15] make it clear that changes in the average size of a population are driven largely by how well (or poorly) the crossover operator samples $Z(t)$ and $L(t)$. Thus we see marked code growth in the early stages of a run when crossover doesn't sample $L(t)$ at all (i.e., no individuals are constructed whose size exceeds the limit) and the average size of individuals not matching the pattern is less than $\mu(t)$. As $\mu(t)$ grows, however, the probability of sampling $L(t)$ increases and the probability of sampling $Z(t)$ decreases (because there are fewer destructive crossovers).

Let, then, $P(S_{Z(t)})$ be the probability of (the crossover operator) sampling $Z(t)$, and $P(S_{L(t)})$ be the probability of sampling $L(t)$. Then the strength of the size limit is then determined by whether $P(S_{Z(t)})$ drops to nearly 0 before $P(S_{L(t)})$ rises appreciably. If $P(S_{L(t)})$ rises quickly enough that both $Z(t)$ and $L(t)$ are both being consistently sampled, then the balance point defined by Eq. (1) is reached and $\mu(t)$ is constrained fairly tightly over time (see the size limit 50 data in Fig. 2). If, on the other hand, $P(S_{L(t)})$ rises slowly, the population reaches a state where there is little chance of consistently sampling either $Z(t)$ or $L(t)$. In this case $\mu(t)$ essentially performs a bounded random walk over time. $\mu(t)$ is then free to drift up and down, with the constraints that if it drifts too high there will start to be consistent sampling of $L(t)$, which will push $\mu(t)$ down, and if drifts too low it will start to sample $Z(t)$, pushing $\mu(t)$ back up. This is illustrated by the size limit 3,000 data in Fig. 2.

4.2 Generalization to N-ary Trees

An important question is how these results will change when applied to more traditional (non-linear) GP trees. While depth limits and size limits are equivalent for variable length linear structures (unary trees), they can behave quite differently than bushier N-ary trees. A key difference is that size is essentially conserved in the sense that if nodes are added to one part of a tree an equivalent number of nodes need to be removed elsewhere to maintain the same size. On the other hand, nodes can be added to a tree in a way that changes the depth of parts of the tree without changing the depth of the overall tree. Thus when using depth limits there can still be considerable growth (measured as increasing

tree size) even among trees that are at or near the depth limit. Alternatively, trees near a size limit cannot grow but can only redistribute their nodes.

It then seems likely that these results will more readily generalize to size limits for N-ary trees. For the unary trees both size and depth are conserved, and there is in fact considerably less flexibility than in the N-ary case. Since there is only one possible shape for a given size, there is no longer the option to redistribute nodes while preserving size.

In the case of N-ary trees near a size limit, the tendency of the crossover operator to generate offspring that are too large (i.e., sample the space of trees that are larger than the size limit) is likely to be similar to the sampling behavior presented here. We conjecture, however, that the sampling is weaker in the case of depth limits, especially for fairly balanced trees, as there are likely to be many crossovers that don't increase the depth.

While these sampling differences between unary and N-ary structures would seem to be a major concern, it's unclear how important the difference is in practice. First, there is considerable evidence that N-ary GP often generates very deep narrow trees (see [1] for excellent visualizations of this phenomena). In such cases the distribution of sizes and lengths of random subtrees will be much more uniform, and therefore closer to the results presented in this paper.

5 Future Work

Because we were primarily focused on code growth, we used only the subtree crossover operator in this work as it was known to induce bloat. Prior results on different mutation operators [10,14] and combinations of operators [9], however, suggests that this work could be extended to include mutation operators. Since mutation operators tend to act as size limiting factors (see [9]) it would be interesting to see what effects they would have on convergent average sizes and the transition from strong to weak size limits.

We conjectured in Section 4.2 that many of these results will generalize in broad terms from linear (unary) structures to more traditional N-ary trees. While some of the theoretical arguments from Section 4.1 suggest that we should see some of the same dynamics, the details are likely to be quite different, and this obviously deserves further exploration.

We have collected preliminary data that suggests that population size plays a significant role in determining the value of the convergent average size for a given size limit. The specifics of this relationship are not yet known, however, and warrant further investigation.

6 Conclusion

The empirical results from Section 3 clearly show a qualitative and quantitative difference between strong and weak size limits. Strong size limits cause the average size of a population to converge to a relatively stable value. When a size

limit is weak, no such convergence occurs. The average size instead appears to perform a random walk within a bounded region.

In Section 4.1 we use the schema theory of holes to better understand the impact of size limit strength. The key result of that analysis is that if the average size is approximately stable then there is a balance between the set of individuals not matching the one-then-zeros pattern and the set of individuals exceeding the size limit.

This work again highlights the challenges of inferring broad structural trends based on empirical data, especially in the face of unavoidably limited resources. The inferences one would make from Fig. 1, for example, would be quite different if one only had data for size limits below 1,000. Similarly, the behavior shown in Fig. 2 changes quite dramatically over time. It seems plausible, for example, that the distribution of mean sizes for limit 3,000 has (in the aggregate) leveled off, but the data given is not sufficient to warrant any strong conclusion. Theory, then, becomes especially important as at least some theoretical results (e.g., Eq. (1) from Section 4.1) are sufficiently general that they can be used to predict behavior even in distant, unseen regions of the problem domain.

Acknowledgement. We would like to thank Engin Sungur for his generous assistance and comments.

References

1. J. M. Daida, A. M. Hilss, D. J. Ward, and S. L. Long. Visualizing tree structures in genetic programming. In E. Cantú-Paz and *et al*, editors, *Genetic and Evolutionary Computation – GECCO-2003*, volume 2724 of *LNCS*, pages 1652–1664, Chicago, 12-16 July 2003. Springer-Verlag.
2. C. Gathercole and P. Ross. An adverse interaction between crossover and restricted tree depth in genetic programming. In J. R. Koza and *et al*, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 291–296, Stanford University, CA, USA, 28–31 July 1996. MIT Press.
3. J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
4. W. B. Langdon and R. Poli. *Foundations of Genetic Programming*. Springer, 2001.
5. W. B. Langdon, T. Soule, R. Poli, and J. A. Foster. The evolution of size and shape. In L. Spector, W. B. Langdon, U.-M. O’Reilly, and P. J. Angeline, editors, *Advances in Genetic Programming 3*, chapter 8, pages 163–190. MIT Press, Cambridge, MA, USA, June 1999.
6. S. Luke. Modification point depth and genome growth in genetic programming. *Evolutionary Computation*, 11(1):67–106, Spring 2003.
7. S. Luke and L. Panait. Lexicographic parsimony pressure. In W. B. Langdon and *et al*, editors, *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 829–836, New York, 9-13 July 2002. Morgan Kaufmann Publishers.
8. N. F. McPhee and R. Poli. A schema theory analysis of the evolution of size in genetic programming with linear representations. In J. F. Miller and *et al*, editors, *Genetic Programming, Proceedings of EuroGP’2001*, volume 2038 of *LNCS*, pages 108–125, Lake Como, Italy, 18-20 April 2001. Springer-Verlag.

9. N. F. McPhee and R. Poli. Using schema theory to explore interactions of multiple operators. In W. B. Langdon and *et al*, editors, *GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 853–860, New York, 9-13 July 2002. Morgan Kaufmann Publishers.
10. N. F. McPhee, R. Poli, and J. E. Rowe. A schema theory analysis of mutation size biases in genetic programming with linear representations. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001*, pages 1078–1085, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea, 27-30 May 2001. IEEE Press.
11. M. O’Neill and C. Ryan. Grammatical evolution. *IEEE Transaction on Evolutionary Computation*, 5(4), 2001.
12. R. Poli. Exact schema theory for genetic programming and variable-length genetic algorithms with one-point crossover. *Genetic Programming and Evolvable Machines*, 2(2):123–163, June 2001.
13. R. Poli. A simple but theoretically-motivated method to control bloat in genetic programming. In C. Ryan and *et al*, editors, *Proceedings of the Sixth European Conference on Genetic Programming (EuroGP-2003)*, volume 2610 of *LNCS*, pages 204–217, Essex, UK, 2003. Springer Verlag.
14. R. Poli and N. F. McPhee. Exact gp schema theory for headless chicken crossover and subtree mutation. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001*, pages 1062–1069, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea, 27-30 May 2001. IEEE Press.
15. R. Poli and N. F. McPhee. Exact schema theorems for GP with one-point and standard crossover operating on linear structures and their application to the study of the evolution of size. In *Genetic Programming, Proceedings of EuroGP 2001*, *LNCS*, Milan, 18-20 Apr. 2001. Springer-Verlag.
16. R. Poli and N. F. McPhee. General schema theory for genetic programming with subtree-swapping crossover: Part 1. *Evolutionary Computation*, 11(1):53–66, Spring 2003.
17. J. E. Rowe and N. F. McPhee. The effects of crossover and mutation operators on variable length linear structures. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, San Francisco, California, USA, 7-11 July 2001. Morgan Kaufmann.
18. S. Silva and J. Almeida. Dynamic maximum tree depth. In E. Cantú-Paz and *et al*, editors, *Genetic and Evolutionary Computation – GECCO-2003*, volume 2724 of *LNCS*, pages 1776–1787, Chicago, 12-16 July 2003. Springer-Verlag.
19. T. Soule and J. A. Foster. Effects of code growth and parsimony pressure on populations in genetic programming. *Evolutionary Computation*, 6(4):293–309, Winter 1998.