Towards Identifying Populations that Increase the Likelihood of Success in Genetic Programming

Jason M. Daida
The University of Michigan
Center for the Study of Complex Systems and the Department of Atmospheric, Oceanic and Space Science
2455 Hayward Avenue, Ann Arbor, 48109-2143 USA
1 (734) 647-4581
daida@umich.edu

ABSTRACT
This paper presents a comprehensive, multivariate account of how initial population material is used over the course of a genetic programming run as while various factors influencing problem difficulty are changed. The results corroborate both theoretical and empirical studies on factors that influence population dynamics. The results also indicate a clue for a possible empirical measurement that could be used in tuning initial populations for increasing the likelihood of success.

Categories and Subject Descriptors
I.2.2 [**Artificial Intelligence**]: Automatic Programming – *program synthesis.*

General Terms
Algorithms, Performance, Experimentation, Theory.

Keywords
Population dynamics, initial populations, *binomial*-3, GP problem difficulty, building blocks, selection methods.

1. INTRODUCTION
There are many factors that affect populations and that influence the ability of GP to solve problems. Koza has identified many of the fundamental ones [15]. The short list would include function and terminal sets (which are the constituents that build individuals), initialization methods (which influence how these constituents are initially assembled), and population size. Over the years, this list has become more nuanced as Koza and others have argued for large (e.g., [16]), small (e.g., [14]), and dynamically sized populations (e.g., [12, 19]). Factors concerning structure and topology of multiple interacting populations have also become considerations as GP has moved from single to parallel machines (e.g., [13, 25]). Various metrics have been proposed that measure dynamics within a population (e.g., diversity [2, 3, 21]), so that mechanisms can be devised to influence them. Some of those mechanisms that could influence content in a population bring in their own set of associated factors, such as those involved with alternative means for doing population initialization (e.g., [5, 18]).

Yet in spite of this work, the flow of population material from start to finish, as it is being used and discarded by GP, is neither fully known nor well characterized. Populations in GP are highly dynamic, where most new individuals are formed from stochastic recombination of pre-existing material. There are a wide variety of factors in play as well, some of which are operator-specific, while others are problem-specific. Although theory and empirical data have shown that GP uses building blocks [17], there is an absence of work that shows either how GP adapts the usage of building blocks to adjust to variations on problems or how populations change as a result of those adjustments.

To know how GP adapts the use of population material would be of particular interest to a practitioner, as well as a theoretician in GP. A practitioner would want to know how various factors—like population size, function set and terminal set composition—can be adjusted to increase the likelihood of a successful outcome. A theoretician would want to know more about the causes that underlie the dynamics of GP populations. Much of the potential work in GP that could be used to illuminate the flow of population material has involved investigations in population diversity [2-4, 21]. Although many of those works have been done with the intent of either maintaining or enhancing population diversity, the methods used are pertinent to studies of population dynamics. In particular, auditing is a fairly straightforward, brute-force empirical method that can track the flow of population material from start to finish. Auditing, in GP, means that the history of components or actions is recorded in enough detail to permit a reconstruction of what has happened. Not only are the contents of a population recorded throughout the course of a run, but there is also a presumption of indexing of the elements within an individual in that population to allow for a reconstruction of a prior state.

GECCO'05, June 25-29, 2005, Washington, DC, USA.

1.1 Challenges

Although the general approach to audit GP runs is straightforward, the task of actually doing so when investigating the efficacy of initial populations involves several challenges:

- *Model problems that can control the likelihood of GP success by varying population attributes are not commonly used in this field.* Many of the model problems that are used for experimental studies are those that require modification of a fitness function. Model problems that can examine the consequences of problem difficulty just by changing an attribute of a population (e.g., tuning the difficulty of a problem by varying just the constituents represented by function and terminal sets) are not common in this field.
- *The amount of data to examine can be substantial.* An audit for a single GP run can mean a population dump (along with its indexing data that allows for reconstruction) to disk for every generation of a run. Furthermore, since GP is a stochastic method, multiple runs are needed to ensure any kind of statistical validity. The total amount of data for what would otherwise be considered a modestly sized experiment can easily approach a terabyte.
- *Techniques for multivariate data exploration in GP are not well developed.* Much of previous work—including those for population diversity—has relied on analysis methods for two and three variables, including time *t*. It is not unusual, since GP is a nonlinear method, to be concerned with four or more variables simultaneously (e.g., time, parameters that control problem difficulty, variables that measure population content).

1.2 About this paper

The purpose of this paper is to provide a first, detailed account of material flows within populations while various factors influencing problem difficulty are changed. The main motivation for doing this is to identify a potential metric of an initial population that could be used to predict for success.

It is organized in the following manner. Section 2 discusses the experimental procedure used to generate the data for study, as well as the procedure involved in the audit of that data. Section 3 defines the terms used in describing the results from the audit. Section 4 discusses the results, as well as comments on the techniques required to extract those results from the data. Section 5 discusses the implications of the audit's findings in the context of current work. Section 6 concludes.

2. EXPERIMENTAL PROCEDURE

A particular, well documented, tunably-difficult test problem was used (i.e., *binomial-3*). The problem has been designed as a probe for understanding GP dynamics and is representative of the kinds of problems found in data modeling. The problem can also be seen as one in which one can study the effect of tuning attributes of a population to increase the likelihood of GP success.

The *binomial-3* is discussed in [6, 9]. In brief, the problem is an instance taken from symbolic regression and involves solving for the function $f(x) = 1 + 3x + 3x^2 + x^3$. Fitness cases are 50 equidistant points generated from $f(x)$ over the interval [-1, 0). The function set is {+, –, ×, ÷}, which corresponds to arithmetic operators of addition, subtraction, multiplication, and protected division. The terminal set is {`X`, **R**}, where `X` is the symbolic variable and **R** is the set of random constants that are distributed uniformly over the interval $[-\alpha, \alpha]$. The tuning parameter is α, which is a real number that controls problem difficulty. The *binomial-3* can be tuned from a relatively easy problem to a difficult one by adjusting the range over which these random constants occur. In general, values of α that are farther from unity result in settings that increase the difficulty for GP to solve this problem.

Table 1. Parameter settings

Parameter	Setting
Selection	Tournament q=7 or Proportionate
Population Size M	500
Initialization Method	Ramped Half-and-Half
Initialization Depths	2–6 Levels
Max Generations G	200
Maximum Depth	26
Internal Node Bias	90% internal, 10% terminals
Termination Criteria	Run reaches G
Binomial-3 α	1 or 1000
Number of Runs	200

A modified version of lilgp [28] similar to that used in [8] was used for this investigation. Most of the modifications were for bug fixes and for the replacement of the random number generator with the Mersenne Twister [20].

Other significant modifications included changes that facilitated audits of how material is used during the course of a GP run. In particular, every data structure that is associated with a node in a GP individual was altered to include an integer `ID,` which subsequently serves as that node's serial number. Each `ID` is unique to a node and is generated during population initialization. This ID labeling scheme was a result of [6], but is similar to that described in [21]. McPhee and Hopper's scheme called for tagging each node in the initial population with integer label pairs (`ID`:`memID`). The `ID` part of their label is assigned at population initialization and consists of an integer that is unique to a node relative to the set of nodes that make up the initial population. `ID` is used as a serial number that can be used to track individual nodes. `memID` is used for providing an audit trail for subtree memberships. For our work, we implemented what amounts to just the `ID` portion of their integer pair.

Table 1 lists the parameter settings considered in this paper. Most of the GP parameters were similar to those mentioned in [15], Chapter 7.

We used four different experimental configurations, given that we considered two different selection methods and two different difficulty settings. These particular settings were chosen primarily because they bracket the conditions under which GP finds this problem either "hard" or "easy." Although the difference in settings seems fairly innocuous, the difference in the likelihood that GP would identify a successful solution was chosen to be unambiguous (i.e., GP would likely be able to identify successful solutions twice as many times under "easy" conditions than under "hard" ones). There were 200 runs taken per configuration for a total of 800 runs.

Although the number of runs is modest, the amount of data generated was not. Each run recorded every instance of every node that was used for all individuals in a population for all generations of a GP run. This was repeated for all 800 runs. The total amount of data that was generated by these four configurations was about 0.5 TB.

Also because the focus of this investigation was on an audit, each run was executed through to the maximum number of generations, which for all cases was 200. This value is about four times longer than what is used in practice and was chosen with the presumption that much of whatever was not used or needed in the derivation of a solution has been filtered out.

GP runs and data reduction were run on Linux workstations. Visualizations were done on a Power Mac.

3. DEFINITIONS

Of all the potential means of summarizing the results of this investigation, the most promising were those that focused on the usage of material from the initial generation at the level of an individual. As it turned out, GP did not assemble its solutions from fragments scattered across a broad cross-section of individuals. Rather, most solutions were assembled using parts from a small subset of individuals in the initial population.

Given that every node was issued a unique serial number at population initialization and since there were no new nodes that were introduced during the course of a run (by, for example, mutation), every individual in the initial population could be identified by a unique set of serial numbers. It then became possible to track which members of an initial population had any kind of presence—either as wholes or fragments—in later generations.

The following definitions are used to describe the results of this experiment:

- V_0. Given an initial population P_0, let every node (i.e., vertex) in this population be uniquely identified and labeled with an integer. V_0 is then identified as the set of integers that are associated with nodes from P_0. For example, an initial population P_0 of 500 individuals could consist of well over 14,000 nodes, depending on the type of population initialization scheme that was used. The membership of V_0 presumes that each of these 14,000+ nodes is considered as uniquely labeled. Let V_0 denote the material present in a population.
- $\mathcal{A}_i$. Let $\mathcal{A}_i$ specifically denote an individual from the initial population P_0. Since all nodes in the initial population are unique, it follows that an $\mathcal{A}_i$ consists of a unique set of nodes $A_i \subseteq V_0$ that is mutually exclusive from the set of nodes A_j, which characterizes an $\mathcal{A}_j$, where $i \neq j$. For example, a three-node tree in P_0 is presumed to be uniquely identified by a set $A = \{a_1, a_2, a_3\}$, such that $A \subseteq V_0$ and the elements a_1, a_2, and a_3 are not a part of any other tree in P_0,
- n_s. Let n_s for a given population denote the number of $\mathcal{A}_i$ that contribute at least one node to that population. For example, in the initial population P_0 that is of population size M, $n_s = M$.

4. RESULTS

As a consequence of focusing on individuals $\mathcal{A}_i$ from the initial population P_0, the analysis of the data that is summarized in this section was done in two stages. The first stage reorganized and reduced the data relative to the individuals that were present in an initial population P_0.

The second stage visualized the reduced data. Even with a reduction in the amount of data to be analyzed, the amount of data and the number of variables were still large and still exceeded the typical treatment of data commonly presented in a GP paper. Simply visualizing this data set has merited its own investigation. Details of the design for these visualizations were based on [11].

Both first and second stages were custom-coded: the first stage was coded in PERL; the second stage, in *Mathematica*. There were 80,400,000 trees that were parsed and analyzed in this manner (i.e., 4 configurations, 200 runs per configuration, 201 generations per run, 500 individual trees per generation). Post-processing time was about one CPU-month per configuration.

Of particular interest to this paper are the summaries of results that highlight two aspects of initial population individuals $\mathcal{A}_i$. In particular, Section 4.1 summarizes the number of individuals from an initial population that are eventually used to derive a GP solution. Section 4.2 summarizes the rank of those individuals in its initial population as a predictor of its use in a GP solution.

4.1 Number of Individuals Used From an Initial Population

One would expect that building blocks for a solution could come from any number of individuals in a population. The results suggest, however, that just a fraction of an initial population is ultimately used for deriving an outcome.

Figure 1 shows the results as a six-variable visualization for all 800 runs. This visualization depicts approximately 80 million trees that were traced back to 400,000 initial population individuals $\mathcal{A}_i$. The variables are as follows: time t (in generations), number of initial population individuals n_s that have any representation in a particular generation, difficulty setting α, measured success rate (percentage of runs that result in an individual that fully meets a problem's criteria for a successful solution), selection method, and cumulative distribution. Note that these visualizations omit the number of initial population individuals for the first several generations because of their magnitude. (The maximum number of initial population individuals is population size M, which occurs at generation 0. Consequently, the complete range for each density plot is [0, M], where $M = 500$. What is shown for clarity instead is the range [0, 150].)

Each density plot in Figure 1 shows four variables: t, n_s, cumulative distribution, and measured problem difficulty. The x-axis corresponds to time t (in generations), while the y-axis corresponds to the number of initial population individuals n_s that have any kind of representation at time t. Tone in the density plot is correlated to cumulative distribution: the darker the tone, the greater the number of runs that have had that many initial population individuals at that particular time. Measured success rate is represented as a thermometer graphic: higher values on the thermometer mean that GP solved the problem more frequently. For example, a thermometer value of 100% means that GP found a successful solution in all of its runs.

The remaining two variables—selection method and difficulty setting α—were portrayed by arranging the four density plots as elements of a two dimensional matrix. Each density plot subsequently corresponds to a variation of one of these two remaining variables.

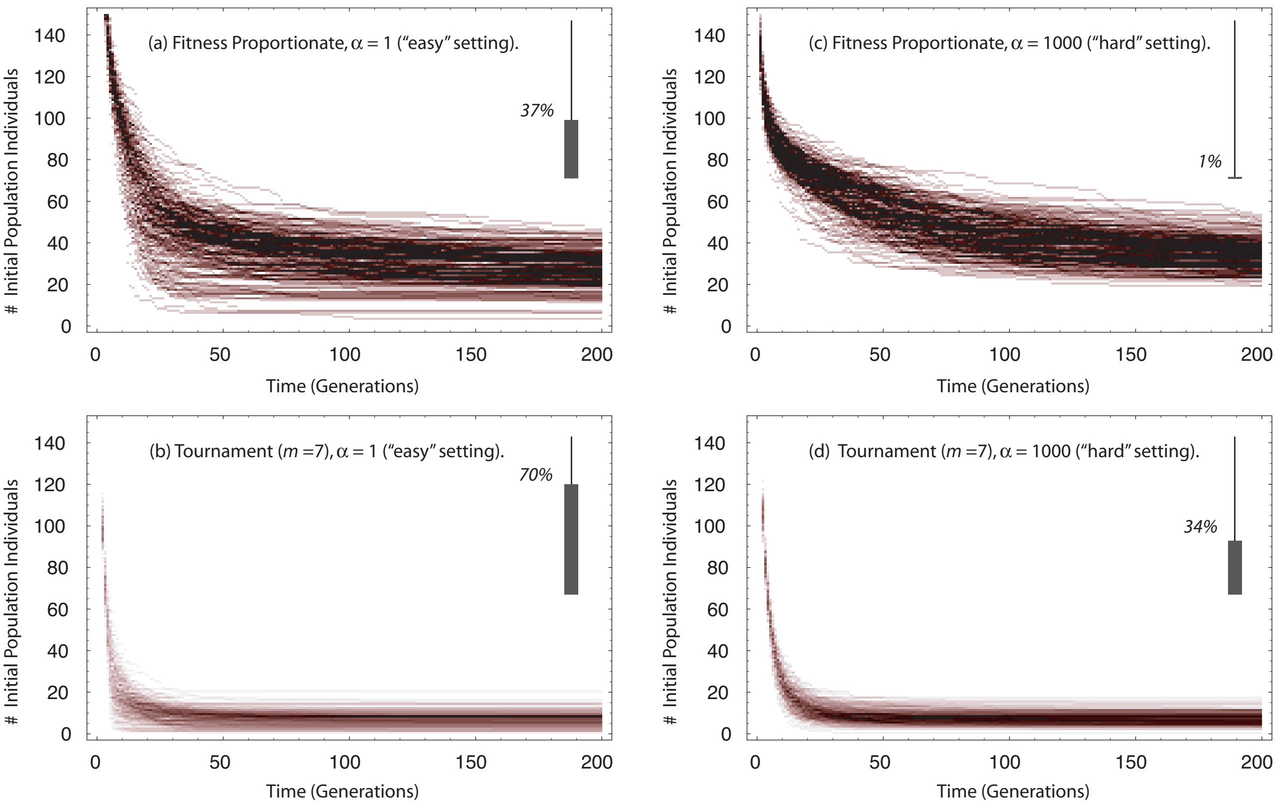

Figure 1. Number of initial population individuals that are represented in a population as a function of time. Each plot summarizes 200 trials and corresponds to a particular selection method and difficulty setting. Thermometer graphs indicate measured success rates (likelihood of deriving a successful solution at the end of a GP run).

Given that the maximum number of initial population individuals is 500 for this population, the results show that no more than 12% of the initial population remains by generation 200 under fitness proportionate selection. Under tournament selection, the number is substantially smaller and shows that no more than 4% of the population remains by generation 200. It is presumed that for all cases, the population at generation 200 consists mostly of initial population individuals that contributed to the best solution GP can identify in any given run.

Furthermore, the results show that tournament selection was more likely to yield a successful solution, even though about three times fewer individuals from the initial population were used. Unlike fitness proportionate selection, tournament selection did not require (on average) any more individuals from the initial population under the "hard" problem setting than under the "easy" one. A question that arises from the results shown in Figure 1 is this: If only a small fraction of the initial population was used by GP to derive a solution, from which initial population individuals was GP using for building blocks? The next section addresses this question.

4.2 Initial Population Rank as a Predictor of Use

Identifying which individuals in the initial population are most likely to provide building blocks that are used in a final solution bears some similarities to Markov analyses. A typical Markov analysis of population dynamics in GP would focus on the transition of probabilities from one time step to the next (e.g., [24]). One goal of Markov analysis is to focus on system properties that are abstracted away from a search space. However, in an analysis where problem difficulty is an issue, this level of abstraction could remove nuances that might indicate behaviors that are dependent upon a search space. For that reason, this analysis of population dynamics focuses on the transition probabilities relative to the initial population. If one knew which initial population individuals would be used to formulate a solution, perhaps one could somehow leverage those individuals to enhance GP's performance.

Similar to what one would have in a Markov analysis, Figure 2 shows the empirical probabilities that at least a subset of nodes from an initial population individual of a given rank would appear in generation $t = 1$. The layout of this visualization is similar to that shown in Figure 1, except that scatter plots of computed likelihoods as a function of rank are shown instead. The x-axis corresponds to the rank of an initial population individual, where *increasing* rank corresponds to *increasing* fitness. The y-axis corresponds to the probability that a subset of nodes from individual $\mathcal{A}_i$ with a given rank would appear in the population at $t = 1$.

Each scatter plot depicts 500 points, since there were 500 individuals in an initial population. Likelihoods were normalized to 200, which is the number of runs associated with each plot. For example, a point that has a probability of existence equal to unity for rank 500 means that the fittest individual in every initial population had a subset of its nodes that could be found in the first generation for all 200 runs.

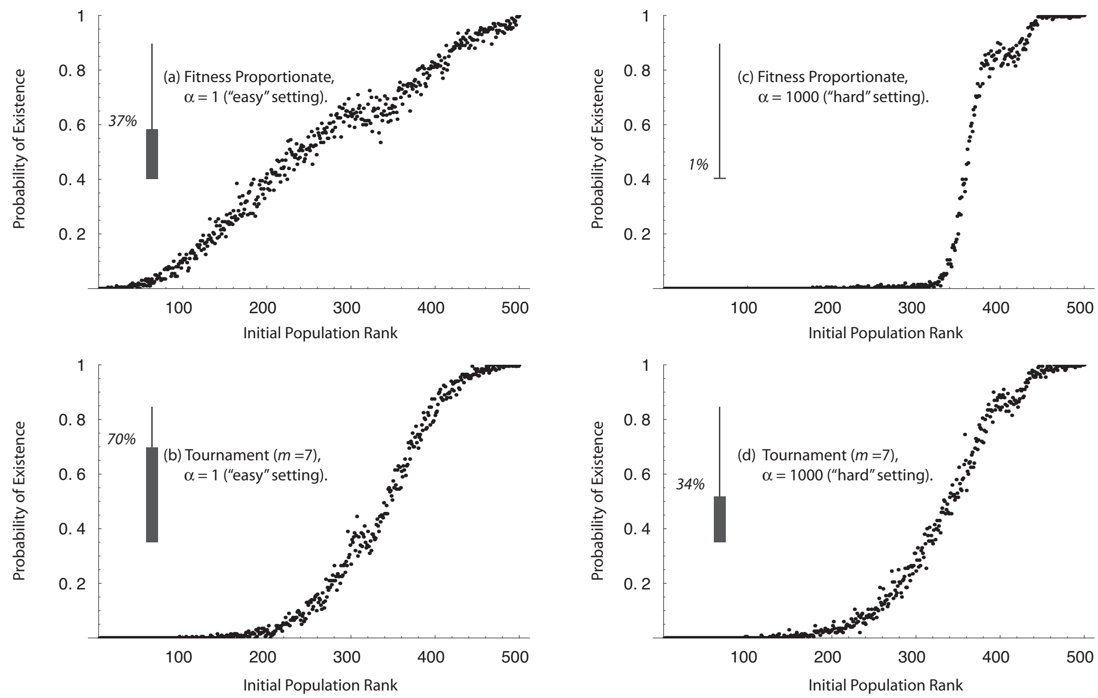

Figure 2. Computed likelihood of a particular initial population individual appearing in the first generation (i.e., $t = 1$). Higher ranks correspond to higher fitness. Each plot summarizes 200 trials and corresponds to a particular selection method and difficulty setting. Thermometer graphs indicate actual success rates (likelihood of deriving a successful solution at the end of a GP run).

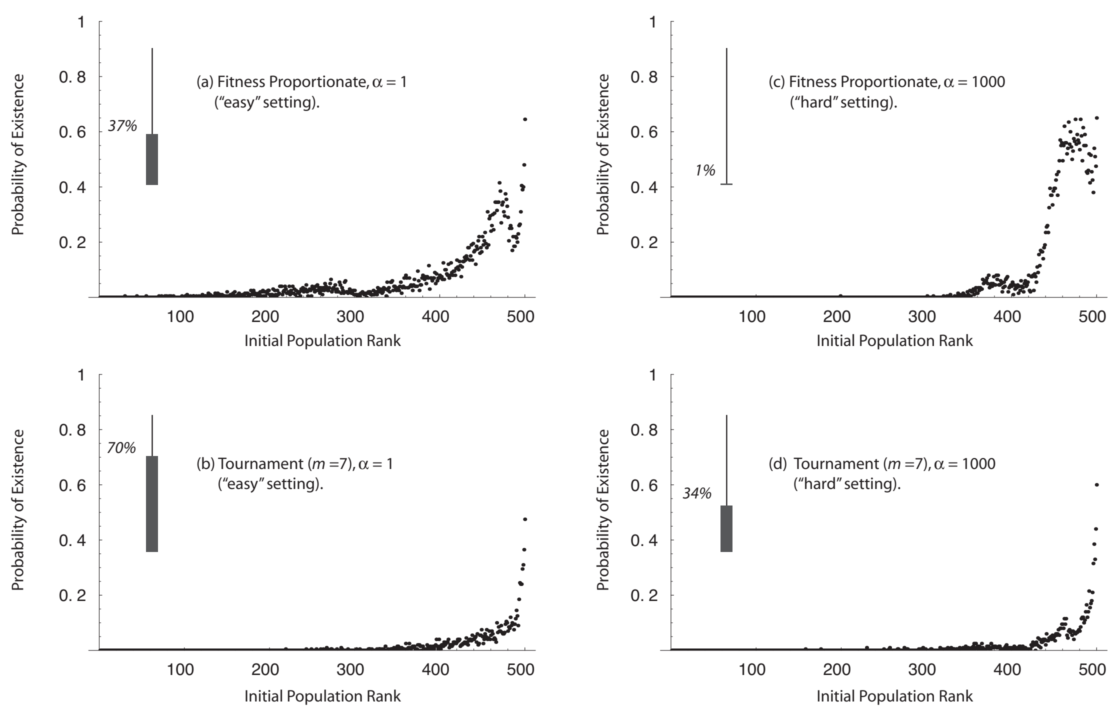

Figure 3. Computed likelihood of a particular initial population individual appearing in the last generation (i.e., $t = 200$).

Figure 3 is similar to that of Figure 2, except that it shows the empirical probabilities that at least a subset of nodes from an initial population individual would appear in the last generation (i.e., $t = 200$). As in the previous figure, each scatter plot in Figure 3 depicts 500 points. Each point represents the likelihood that a subset of nodes from an initial population individual with a given rank did survive in the population through $t = 200$. Whereas the population material V_0 is random, population material V_{200} is presumed to be strongly correlated with whatever solution GP has found. Consequently, nodes from initial population individuals

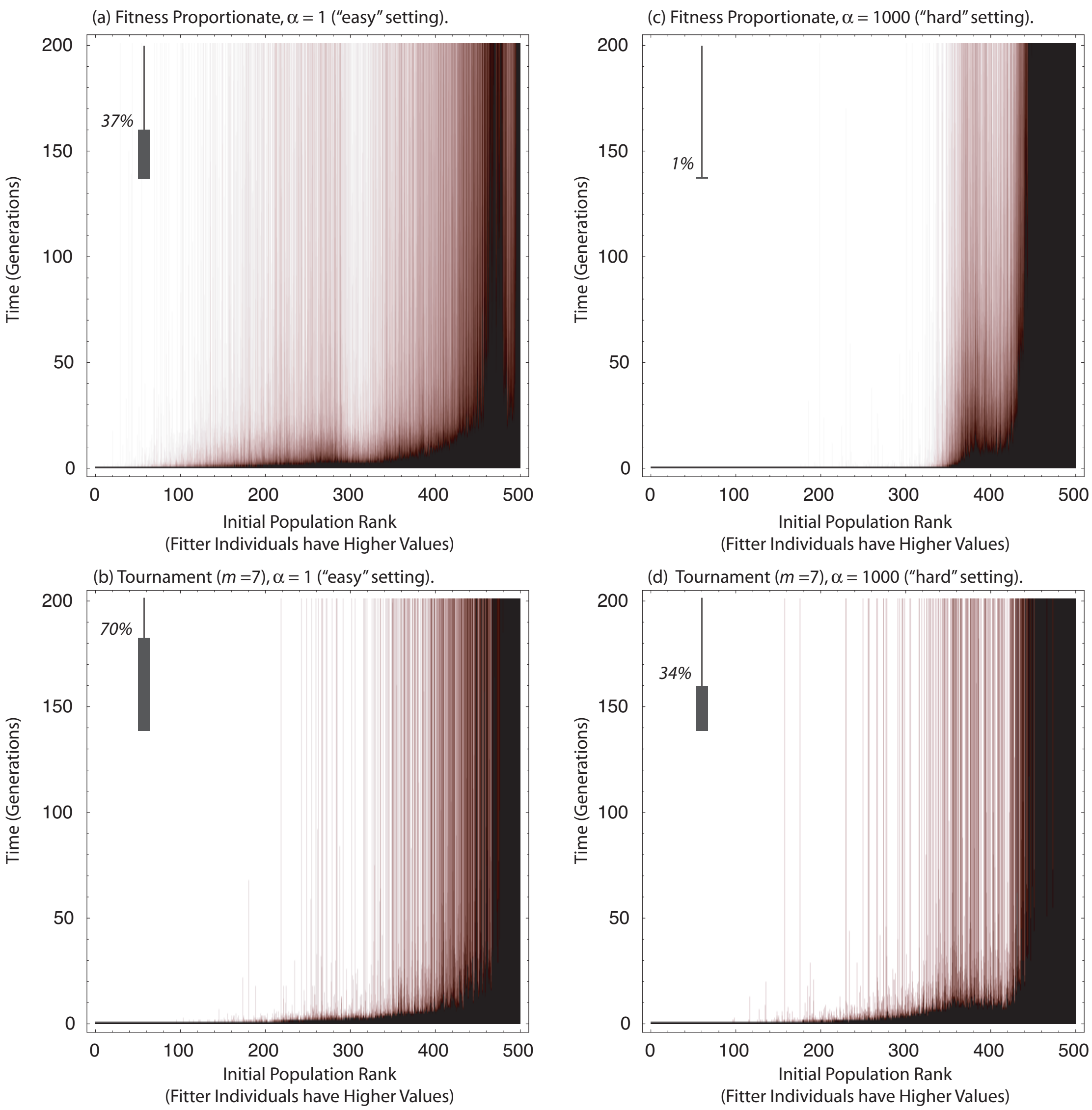

Figure 4. Computed likelihood of a particular initial population individual appearing in the last generation as a function of time and rank. Each plot summarizes 200 trials and corresponds to a particular selection method and difficulty setting. Thermometer graphs indicate actual success rates (likelihood of deriving a successful solution at the end of a GP run).

that persist through to generation 200 likely belong to those individuals that have contributed in some way to the solution that GP has derived.

Figures 2 and 3 can be thought of as slices in time that show the probability of extinction as a function of initial population rank that occurs at the beginning and end of a GP run. Consequently, Figure 4 depicts the full transition in probabilities from $t = 1$ to $t = 200$. Figure 4 has a layout that is similar to the previous figures, except that each scatter plot has been replaced by a density plot. Each horizontal slice in a density plot represents the probability of existence as a function of population rank for a particular time t. Darker tones represent higher probabilities. Each density plot is built up of 201 slices, which corresponds to the slice at $t = 0$ (i.e., the initial population) and the subsequent number of generations in a GP run (i.e., $t = 1$ to $t = 200$, inclusively).

5. DISCUSSION

The most basic result of this work is by itself not a surprise: *The composition of an initial population at the level of which functions and terminals are used can play a significant role in determining how tractable a problem is to GP*. Although this observation has been made since near the inception of the field [15], it forms the basis under which tunable difficulty is made for this particular problem [6]. As shown here, the effect was not small and resulted in order of magnitude differences in performance, particularly under fitness proportionate selection (i.e., from 1% to 37% success, *cf.* [9]).

However, the results go beyond this basic finding and provide a glimpse into the population dynamics of GP. Population dynamics inform one how initial populations change during GP: if one knew how populations transform over time, one could use this information to identify successful initial populations. One difficulty has been to synthesize an integrated understanding of population dynamics from various studies that focus on just a part of those dynamics. This would include the dynamics of schema (e.g., [17, 23]), selection methods (e.g., [1, 22]), and problem difficulty (e.g., [7, 27]).

Although a variety of problems is desired for studying the dynamics of GP populations, only one problem, albeit a rigorously characterized one, was considered. Section 5.1 discusses the interrelationship between selection method, material flows, and building blocks. Section 5.2 describes additional linkages for population size, while Section 5.3 describes a metric that could benefit practitioners for increasing the likelihood of success in using GP.

5.1 Selection Method, Material Flows, and Building Blocks

A major finding of the audit was in observing just how much of an effect selection methods have on population dynamics. (See Figures 1a and 1c in comparison to 1b and 1d, as well as Figures 4a and 4c in comparison to 4b and 4d.) In particular, flows of material from the initial population under tournament selection appeared largely independent of difficulty setting.

On one hand, it has been known for some time that tournament selection is "lossy"—referring to the finding that a certain fraction of individuals are expected not to propagate in one generation to the next methods (e.g., [1, 22]). Although [1] showed that loss occurs regardless of the underlying fitness distribution, the audit data in Figures 1 and 4 seem to show the implications of this over time.

On the other hand, even though tournament selection "lossy," it significantly *improved* performance. This finding is somewhat counterintuitive to the idea that preserving diversity or introducing more diversity is conducive to better problem solving in GP [e.g., [2, 21]]. [For the purposes of this paper, diversity would be measured as the number of distinct elements from V_0 that remain in a population. This definition is similar to one used in [21].] In particular, systemic and complete losses of initial population individuals $\mathcal{A}_i$ (up to 80%, as shown in Figures 4b and 4d) would not seem to be a rational response to improving performance *if* maintaining diversity was also a desired outcome. [Contrast to [26], which uses a different, non-audit metric for determining diversity.]

Why would "lossy-ness" contribute to an improvement in performance?

In [6], we posed a minor hypothesis about building block formation: not only does GP require methods for assembling building blocks out of the nodes that it needs, but it also does require methods for contending with nodes it does not. We argued that tunable difficulty for the *binomial*-3 occurs partly because GP has a finite capacity for removing or absorbing the effects of deleterious material, such as those posed by errant random constant values. A loose analogy is that deleterious material "parasitizes" GP solutions, in part because they are "trapped" in trees and branches.

At the time, we neither had the means to test this hypothesis with an audit nor did we consider tournament selection.

This work's audit result and the use of tournament selection do apparently support this hypothesis. The significant dissimilarities between flows of material under tournament selection and underlying fitness proportionate do suggest the possibility that the underlying dynamics are also dissimilar. I suggest that the dynamics have occurred in this way because tournament selection affords another means for eliminating deleterious material, whereas fitness proportionate selection does not. A detailed analysis of this is left to future work, especially since the data does *not* support the notion that more "lossy-ness" in and of itself results in better performance.

If anything, the results suggest a nuanced view and use of diversity. Studies in diversity generally presume that more diversity is conducive to problem solving in GP (e.g., [2, 21]). To some degree, the results support that presumption. Figures 2a and 2c indicate that when most of the initial population can be used to form the first generation under proportionate selection, the problem corresponds to one that is dramatically easier to solve. However, Figures 2b and 2d, as well as Figures 4a – 4d, indicate that diversity is symptomatic and not causal: although proportionate selection maintained population diversity for a longer time than did tournament selection (Figures 1a and 1c), tournament selection greatly increased the likelihood of success over its fitness proportionate counterparts.

5.2 Initial Population and Population Size

Another major finding of the audit was in indicating a strong correlation between individual rank within an initial population, the material that is ultimately used in a solution, and initial population tuning.

The results suggest that not only do useful building blocks need to be in sufficient supply within a population, but that these building blocks also need to be part of the fittest individuals of an initial population. In particular, both Figures 3 and 4 indicate that such building blocks would need to be in the individuals that rank in the upper 20% of an initial population. Although material V_0 that are from individuals that are ranked lower than 20% can persist in populations for several generations, the amount dwindles to zero as more and more of a population consists of progeny from what is likely a GP solution. This means one could predict in advance of a completed GP run which individuals would be used by GP to derive a solution. This prediction (at least for this problem) would hold, regardless of whether it was "easy" or "hard" for GP to solve.

Because a fraction of an initial population is indicated, population size would likely be a factor since it would influence the total number of initial population individuals that contribute to a GP solution.

This finding, if true for other problems, would support the use of seemingly contradictory methods for population sizing in GP. Koza et al. has advocated the use of very large populations to solve difficult real-world problems [16], while Luke et al. has explored the use of shrinking populations to small sizes [19]. If it turns out that GP recombines material ultimately from just a small fraction of a population for most problems (similar to what is shown in Figure 4), an exceptionally large population not only increases the likelihood of having the appropriate building blocks in a population, but also that those building blocks are concentrated in initial population individuals that make up that fraction. Dynamic shrinking also makes sense in this context, since the determination of which fraction would be useful to a solution could be done immediately after the determination of rankings of the initial population.

Note that the results from this paper do not by themselves have anything to say about the effect of population sizes, since only one population size was considered. However, in another paper [10], we show corroborative evidence for this linkage with another problem (i.e., *Highlander*) in which population size was varied.

5.3 What Identifies Potential for Success?

A *tentative* answer to this question is this: What identifies the potential of an initial population to yield a successful outcome is the probability distribution of whether individuals from that initial population are propagated into the first few generations under fitness proportionate selection (*cf.* Figures 2a and 2c). Initial populations with broader propagation distributions are more likely to produce successful outcomes in contrast to initial populations with narrower distributions. However, to enhance the possibility of a successful outcome with an initial population, tournament selection should be used, instead.

There are a number of ways that an initial population can be tuned without needing to know the specifics of what kinds of building blocks are eventually needed to solve a particular problem. One way is to adjust the composition of function and terminal sets, as was done for this investigation. Another is to use alternative algorithms for initializing a population (e.g., [18]).

For whichever method is chosen to initialize a GP population, this probability distribution does provide a clue as to whether one is on the right track towards synthesizing successful initial populations. This probability distribution can be determined empirically by running GP for just one generation.

Once the conditions for creating potentially successful populations has been identified using this probability distribution, subsequent GP runs using these conditions can be run under tournament selection.

A definitive answer to this question would require further tests to be conducted with other problems.

6. CONCLUSIONS

This paper presented a detailed, multivariate account of how initial population material is used over the course of a GP run as various factors that influence the difficulty of a problem were changed. Instead of focusing on just archetypes of GP runs that illustrate various behaviors, new multivariate visualization techniques were developed to analyze the dynamics of 800 populations and 80M individuals.

The paper described four key points that could be drawn from the results. First, the empirical results corroborated and linked various theoretical works that investigate building blocks, selection method, diversity, population size, and problem difficulty. Second, the results indicated the effect initial population rank, population size, and building blocks. It, furthermore, supported the use of apparently contradictory methods for population sizing in GP. Third, the results suggested a nuanced view and use of diversity. Fourth, the results identified a potential metric of an initial population that could be used to predict for success.

7. ACKNOWLEDGMENTS

I thank the following individuals for their help: I. Kristo, S. Daida, S. Long, A. Hilss, M. Hodges, D. Ward, M. Samples, and M. Byom, C. Kurecka, F. Tsa, and M. Rio.

8. REFERENCES

[1] Blickle, T. and Thiele, L. A Mathematical Analysis of Tournament Selection. in Eshelman, L.J. ed. *ICGA*, Morgan Kaufmann, San Francisco, 1995, 9–16.

[2] Burke, E., et al. Diversity in GP: An Analysis of Measure and Correlation with Fitness. *IEEE TEC*, *8* (1). 47–62.

[3] Burke, E., et al. A Survey and Analysis of Diversity Measures in GP. in Langdon, W.B., et al. eds. *GECCO,* Morgan Kaufmann, San Francisco, 2002, 716–723.

[4] Burke, E., et al. Advanced Population Diversity Measures in GP. in Merelo Guervós, J.J., et al. eds. *PPSN VII*, Springer-Verlag, Berlin, 2002, 341–350.

[5] Böhm, W. and Geyer-Schulz, A. Exact Uniform Initialization for GP. in Belew, R.K. and Vose, M.D. eds. *FOGA 4*, Morgan Kaufmann, San Francisco, 1997, 379–407.

[6] Daida, J.M., et al. Analysis of Single-Node (Building) Blocks in GP. in Spector, L., et al. eds. *Advances in GP 3*, MIT Press, Cambridge, 1999, 217–241.

[7] Daida, J.M., et al.. What Makes a Problem GP-Hard? Validating a Hypothesis of Structural Causes. in Cantú-Paz, E., et al. eds. *GECCO*, Springer-Verlag, Berlin, 2003, 1665–1677.

[8] Daida, J.M., et al. What Makes a Problem GP-Hard? Analysis of a Tunably Difficult Problem in GP. in Banzhaf, W., et al. eds. *GECCO*, Morgan Kaufmann, San Francisco, 1999, 982 – 989.

[9] Daida, J.M., et al. What Makes a Problem GP-Hard? Analysis of a Tunably Difficult Problem in GP. *GPEM*, *2* (2). 165–191.

[10] Daida, J.M., et al. Probing for Limits to Building Block Mixing with a Tunably Difficult Problem for GP. in *GECCO*, 2005.

[11] Daida, J.M., et al. Visualizing the Loss of Diversity in GP. in *CEC*, IEEE Press, Piscataway, 2004, 1225–1232.

[12] Fernandez, T. Virtual Ramping of GP Populations. in Deb, K. ed. *GECCO*, Springer-Verlag, Berlin, 2004, 471–482.

[13] Fernández, F., et al. An Empirical Study of Multipopulation GP. *GPEM 4* (1). 21–51.

[14] Gathercole, C. and Ross, P. Small Populations Over Many Generations Can Beat Large Populations Over Few Generations in GP. in Koza, J.R., et al. eds. *GP*, Morgan Kaufmann, San Francisco, 1997, 111–118.

[15] Koza, J.R. *GP*. MIT Press, Cambridge, 1992.

[16] Koza, J.R., et al. *GP IV*. Kluwer Academic, Norwell, 2003.

[17] Langdon, W.B. and Poli, R. *Foundations of GP*. Springer-Verlag, Berlin, 2002.

[18] Luke, S. Two Fast Tree-Creation Algorithms for GP. *IEEE TEC*, *4* (3). 274–283.

[19] Luke, S., et al. Population Implosion in GP. in Cantú-Paz, E., et al. eds. *GECCO*, Springer-Verlag, Berlin, 2003, 1729–1739.

[20] Matsumoto, M. and Nishimura, T. Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudorandom Number Generator. *ACM Trans Modeling and Comp Sim*, *8* (1). 3–30.

[21] McPhee, N.F. and Hopper, N.J. Analysis of Genetic Diversity through Population History. in Banzhaf, W., et al. eds. *GECCO*, Morgan Kaufmann, San Francisco, 1999, 1112 – 1120.

[22] Motoki, T. Calculating the Expected Loss of Diversity of Selection Schemes. *EC*, *10* (4). 397–422.

[23] Poli, R., et al. Analysis of Schema Variance and Short Term Extinction Likelihoods. in Koza, J.R., et al. eds. *GP,* Morgan Kaufmann, San Francisco, 1998, 284–292.

[24] Poli, R., et al. Markov Chain Models for GP and Variable-Length GAs with Homologous Crossover. in Spector, L., et al. eds. *GECCO*, Morgan Kaufmann, San Francisco, 2001, 112–119.

[25] Punch, W. How Effective are Multiple Pop in GP. in Koza, J.R., et al. eds. *GP*, Morgan Kaufmann, San Francisco, 1998, 308 – 313.

[26] Silva, S. and Almeida, J. Dynamic Maximum Tree Depth: A Simple Technique for Avoiding Bloat in Tree-Based GP. in Cantú-Paz, et al. eds. *GECCO*, Springer-Verlag, Berlin, 2003, 1776–1787.

[27] Vanneschi, L., et al. Fitness Clouds and Problem Hardness in GP. in Deb, K. ed. *GECCO*, Springer-Verlag, Berlin, 2004, 690–701.

[28] Zongker, D. and Punch, W. lilgp, Michigan State University Genetic Algorithms Research and Applications Group, Lansing, 1995.