

Population Diversity as a Selection Factor: Improving Fitness by Increasing Diversity

James Byron
Westmont
955 La Paz Road
Santa Barbara, CA
jbyron@westmont.edu

Wayne Iba
Westmont
955 La Paz Road
Santa Barbara, CA
iba@westmont.edu

ABSTRACT

Evolutionary algorithms search for problem solutions by selecting individuals for survival and reproduction with a bias towards higher fitness. Such biases may lead to premature convergence on sub-optimal solutions. A bias toward greater diversity can help delay convergence and broaden the area searched for optimal candidate solutions. We introduce two ways to measure a population's diversity and evaluate how they interact with traditional fitness during selection. We then introduce a mechanism that includes a bias toward greater diversity in addition to traditional accuracy. Using the King-Rook-King chess endgame problem, we demonstrate that including diversity as a selection factor leads to better overall solutions.

1. INTRODUCTION

Evolutionary algorithms aim to solve complex problems by mimicking the behavior of natural biological organisms that adapt to challenges in their environment. For a genetic algorithm, the process of evolution takes place on a population of individuals, each representing a candidate problem solver for instances of a given domain. As with natural ecosystems, diversity within the population is vital for the population's ability to produce individuals that are progressively more adept at solving problems. In this paper, we explore methods of measuring diversity and explore their relationship with improving fitness. With an aim towards preserving diversity within a population, we also introduce a flexible mechanism for combining either one of our two diversity measures with traditional fitness values based on accuracy. Note, the diversity in a population reflects the *distinctiveness* of the individuals that make up the population.

In a traditional genetic algorithm, individuals in the population undergo processes of selection for survival, reproduction, and mutation. More highly fit individuals are favored over those that are less fit for reproduction and survival from one generation to the next. While this process is well known

to yield individuals that are much more fit than the initial population [3], such genetic algorithms will often fail to discover the global optimum because it prematurely converges on a sub-optimal solution. Premature convergence in a population results from the algorithm's rapid elimination of unfit, though perhaps unique, individuals in the early stages of evolution[7]. As the diversity of the population decreases, the breadth of the search for ideal problem solvers becomes too narrow to generate improvements beyond a local optimum. Many mechanisms for preserving diversity have been proposed, often reporting their results with the use of either standard benchmark or challenging problem domains.

In the next section, we describe related work on measuring and fostering diversity. We then describe two measures that we have implemented and evaluated. Next we turn to our experiments with a standard EA that we modified to employ these measures of diversity as a factor in selection, discussing the factors' respective impacts on performance. We close with discussion and our plans for future work.

2. PREVIOUS WORK

Many studies have previously addressed the challenge of maintaining diversity in a population while simultaneously allowing the population to retain its selective pressure towards greater fitness. What follows is a summary of several popular diversity preserving mechanisms.

2.1 The Island Model

The island model is well-suited to parallel implementations on networked workstations, each containing its own island population.[5]. As an added benefit, this approach tends to slow the convergence of the overall population to a local optimum because the sub-populations on each island evolve independently from one another. Periodic migration between the islands allows a small amount of genetic material to cross-fertilize the separately evolving populations. Such migration may increase the diversity within each island's sub-population by injecting novel genetic material from a separately evolving neighbor population.

2.2 DGEA

The Diversity-Guided Evolutionary Algorithm (DGEA) measures diversity within the population and alternates between modes of increasing diversity and increasing fitness at each generation, depending on the amount of diversity within the population. While calculating diversity, each individual's genetic sequence is evaluated by calculating its distance to the average point for the population[9]. The

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](http://permissions.acm.org).

GECCO'16 Companion, July 20-24, 2016, Denver, CO, USA

© 2016 ACM. ISBN 978-1-4503-4323-7/16/07...\$15.00

DOI: <http://dx.doi.org/10.1145/2908961.2931653>

population’s total diversity is then used to alternate between processes of exploration (mutation) and exploitation (survival and genetic recombination), trending toward exploration when diversity is below a predefined threshold and toward exploitation when diversity is above the threshold. While the DGEA has the extra cost of computing diversity at each generation, it allows both fitness and diversity to factor into the evolutionary process, resulting in an improvement over evolutionary algorithms that do not use diversity to guide the evolutionary process.

2.3 Fitness Uniform Selection Strategy (FUSS)

FUSS preserves diversity by selecting individuals whose fitness values are between the least and most fit individuals within the population. In the event that many individuals have similarly low fitness values, the individuals with higher fitness will be preferred for selection because they lie in a more sparsely represented part of the fitness space. The population size increases over time as FUSS adds new individuals to the population without discarding old individuals. While FUSS has the drawback of requiring large amounts of memory, there is no risk of premature convergence in the population because by not discarding individuals from the population on each successive generation, genetic material is preserved indefinitely. FUSS has proven to be successful both at preserving diversity and increasing fitness [4].

2.4 Deterministic Crowding

With Deterministic Crowding, new offspring from recombination are allowed to take the place of their parents only if they achieve higher fitness than one of their parents. This mechanism has the effect of preserving a similar genetic code together with its genetic diversity while also favoring the fittest individuals to survive [2].

2.5 Fitness Sharing

Fitness sharing adjusts the fitness of each individual in a population by a factor of how dissimilar each individual’s fitness is from every other individual in the population. Individuals with average fitness are weighted lower than those with more uncommon fitness values [2].

2.6 Ranked Space

The ranked space method preserves diversity by embedding a diversity ranking alongside the fitness ranking for each individual [6]. These two factors are combined to determine which individuals are favored for selection. Ranked space has the effect of directly enforcing a rule of diversity preservation in the population while simultaneously allowing fitness to influence selection as well.

2.7 Restricted Mating

Restricted mating influences recombination by discouraging individuals from mating whose genetic information is too similar, or who belong to a sub-group of the population [6]. This method of preserving diversity encourages a degree of separation in the population, which promotes diversity, while also allowing individuals to recombine whose genetic exchange may yield improvements.

2.8 Entropy-based Approach

Genetic diversity can be measured as a numerical total of the entropy within a population’s individuals. Squillero

and Tonda [8] proposed a new methodology for entropy calculation that incorporates the position and repetition of a genetic symbol as well as its structure. This approach recognizes the possibility that the same genetic symbol at different locations within the genetic sequence may represent novel features within the population and thus contribute to the population’s overall diversity.

3. MEASURING AND USING DIVERSITY

In this paper, we present two measures of diversity that we then proceed to employ within a standard genetic algorithm. But note that diversity is a *population-wide* measure; that is, it is a property that a given population may have to a greater or lesser degree. Diversity in a population is made up of *distinctiveness*, a measure that applies to an *individual*. When we use these measures of diversity to influence the selection of individuals for survival across generations and for breeding, we are actually using the individuals’ distinctiveness values. Throughout this paper, we will sometimes refer to the diversity of an individual but the reader should keep in mind that we actually mean distinctiveness. In this section, we describe the measures of distinctiveness and diversity and define how we compute them.

3.1 Phenotype Diversity

In a typical genetic algorithm, an individual’s genetic material governs its performance on test cases, and thereby its fitness. Phenotype diversity is based on the uniqueness of an individual’s performance rather than on the uniqueness of the individual’s genome. In other words, phenotype diversity measures *behavioral* diversity. Our phenotype measure looks at performance on each problem of the training set with respect to the collective performance of the population on that training instance. Individuals are considered more phenotypically distinctive, or unique, if they correctly answer problems of the training set that few other individuals answered correctly. To calculate phenotype diversity, we first evaluate how each individual performed on each element of the training set and store these performance values in a two-dimensional vector

$$V_{ij} = I_i \times P_j$$

where I_i is an individual in the population at index i , P_j is a specific problem in the training set at index j on which the individual is evaluated, and V is a two-dimensional vector containing each individual’s performance on each element of the training set. We then add up the i columns of the vector V so that we have a vector P with the population’s total performance on each element of the training set. We evaluate

$$P_j = \sum_{i=1}^n V_{ij}$$

where P is a vector containing the sum totals of the performance values for each element of the training set and i is the index for an individual in the vector V of each individual’s performance on the training set. The P_j value aggregates the population’s performance on each element j of the data set. Next, we compute a weighted value for each element in the training set by calculating the difference between the number of individuals in the population and the computed

value of each element in the vector of performance totals P and then divide the result by the number of individuals. We evaluate

$$W_j = \frac{n - P_j}{n}$$

where n is the number of individuals in the population, and P_j is the total performance of the population for an element of the training set. The vector W stores floating point values that represent the difficulty of a given problem and correspondingly the importance of individuals that correctly solve that problem. The population’s diversity by this measure will aggregate each individual’s importance over training set. Each value in W is then inversely proportional to how many individuals in the population correctly solve each element of the training set. Larger values in the vector of weights result from a smaller number of individuals correctly solving a particular problem. If, for any element of the training set, no individual answered it correctly, we assign a weight of 1 to that training set element and record how many elements received such a weight. Ideally, few to none of the training set elements will receive a weight of 1 because we want our population to retain sufficient diversity that at least one individual can answer each problem correctly. Next, we sum the product of the performance values of the individuals on each element of the training set stored in vector V with the corresponding weights in W for each training set element. We compute the distinctiveness, D_i , of an individual, i , as

$$D_i = \frac{1}{t} \sum_{j=1}^t V_{ij} \times W_j$$

where j ranges over the training set, V_{ij} is an individual’s performance on a specific training set element j , W_j is the diversity weight associated with that element from the training set, and t is the number of elements in the training set. The resulting vector D stores the distinctiveness values for the population. Larger values in D represent an individual that performed well on elements of the training set that few other individuals answered correctly.

3.2 Estimated Hamming Diversity

In contrast to phenotype diversity, which is based on the performance or behavior of individuals, we can define a measure of diversity as the hamming distance between all pairwise matches of individuals in the population. This approach yields a measure for how different an individual is with respect to the rest of the population. However, for large populations, the n^2 cost in the number of individuals can become excessive. Thus, we present an estimate of this measure that can be computed in linear time and suffers only a modest error (less than 3% for large populations).

Estimated hamming diversity is calculated by evaluating how different each individual’s genetic makeup is from that of the average genetic composition in the population. This approach bears some resemblance to that described in [9]; in that work, the Euclidean distance to the population average was used as a reference point for guiding the mutation of individuals. Instead here we determine a modified hamming distance between each individual and the population average as an estimate of the true average hamming distance measured between every pair of individuals. We first determine this population average by summing the bits of each

individual’s genetic code and divide the resulting values by the number of individuals in the population as

$$A_j = \frac{\sum_{i=1}^n I_{ij}}{n}$$

where n is the number of individuals in the population, i is the index of an individual in the population, A is a vector of floating point numbers for the average bit value between 1.0 and 0.0 at a given position throughout the population. We compute A_j for each bit position in the genomes. Next we compute the deviation from this ‘average individual’ by summing the amounts that each individual’s genetic code varies from the average at each bit position. The resulting number is an individual’s distinctiveness in the population. We compute the distinctiveness, D_i , of an individual, i , as

$$D_i = \frac{1}{r} \sum_{j=1}^n |I_{ij} - A_j|$$

where I_{ij} is the bit-value at position j of individual i ’s genome, A_j is the value at position j in the population average as computed above, and r is the number of bits in each individual’s genetic code. We take the absolute value of the difference of I_{ij} and A_j in order to accumulate the total distance from the population average as a positive value. As when computing phenotype diversity, we normalize the values in D by dividing them each by the number of bits in each individual’s genetic code. Once we have a measure of an individual’s distinctiveness – whether phenotype or estimated hamming – we next want to combine diversity with accuracy for the purposes of selection, to which we now turn.

3.3 Including Diversity in Fitness

Given one or more ways of measuring diversity in a population – and more importantly, of measuring the distinctiveness of individuals in that population – we would like to utilize such measures as a factor in selection. Our approach defines fitness as a weighted sum of an individual’s distinctiveness and its accuracy on the training set. As such, we think of as accuracy, or quality of behavior, on the performance task as what has traditionally been known as fitness. Where the context makes our intention clear, we may use the term ‘fitness’ to refer to accuracy.

Our approach to determining fitness combines an individual’s distinctiveness value with its performance, or accuracy, on a training set. We employ a single weight that controls the relative contribution of these factors when selecting individuals for survival and recombination. Thus, the fitness, F_i , of an individual, i , can be computed as

$$F_i = (w \times D_i) + (1 - w) \times B_i$$

Where D_i is the distinctiveness value for the individual, B_i is the behavioral accuracy reflecting how well the individual performs on the data, and w is a weight between 0.0 and 1.0. Values of w closer to 1.0 will bias selection towards distinctiveness and should lead to populations with higher diversity; similarly, when w is set close to 0.0, selection will operate more like a traditional GA. Thus, the fitness for an individual, F_i , adopts a ‘balance’ between diversity and accuracy where the nature of the balance is determined by the weight, w .

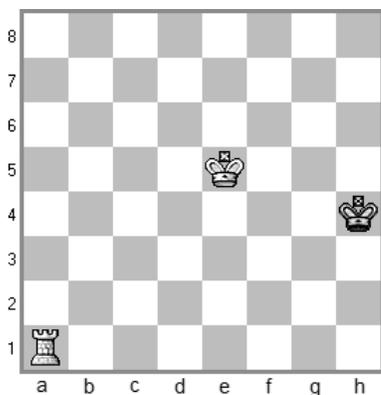


Figure 1: Sample KRK problem. In this case, White can checkmate in three moves.

4. PROBLEM DOMAIN

We use the King-Rook-King (KRK) chess endgame problem as our domain for assessing the impact of diversity as a selection factor. This domain presents a formidable challenge for a genetic algorithm, and many previous studies have addressed the challenge of solving a difficult problem while maintaining diversity among the individuals in the population. The KRK problem can be stated as follows: *given* a board configuration with rank and file values for the two Kings and a white Rook, *determine* the optimal number of moves White needs to force checkmate. As such, the domain may be thought of as a classification task. Figure 1 displays a sample problem instance from the KRK domain. In this problem, White can checkmate Black’s King in three moves.

The King-Rook-King dataset [1] consists of 28,056 labeled instances or problems. The answer, or class, for each problem is the number of moves to checkmate and has eighteen possible values: ‘zero’ (already checkmate) through ‘sixteen’, and ‘draw’ (no checkmate possible). Each problem’s board configuration is described by six numeric values between 1 and 8; each of the three pieces has both a rank and a file value.

To apply an evolutionary algorithm to this problem we need to decide how an individual’s genome is represented and how that genome determines an answer to a KRK problem. We represent an individual as a bit string consisting of eighteen segments, one segment corresponding to each possible answer. Each segment consists of six bytes, one for each feature in a problem configuration. The eight bits of each byte represent the possible values for the rank or file of one of the three pieces. Note, within a byte of an individual’s genome, none, some, or all of the bits may be set. However, problem instances get represented as six bytes, each of which has exactly one bit set, corresponding to the rank or file value of the piece in the problem. Overall, the genome of each individual is $8 \times 6 \times 18$ or 864 bits long.

When an individual classifies a problem instance from either the training or test sets, each of the eighteen segments from the individual’s genome get ‘matched’ to the representation of the problem. A single byte within a segment is masked with the corresponding byte in the problem representation. If the genome has a set bit (1) in the same posi-

tion as the problem’s single set bit, then that feature is said to have matched. For the segment, we add the number of features that match. After matching all eighteen segments, the segment with the most matches gets to make the classification. If there is a tie for the most matches and one of those tied is the correct classification, the individual gets credit for a fraction of a correct response; the fraction depends on how many segments were tied. If the correct segment did not have the most matching features, then no credit is awarded. An individual’s accuracy is computed based on how much credit it receives for correct answers over the data set.

5. EXPERIMENTS

Having defined our two measures of diversity and having described how we employ a EA in our problem domain, we now turn to evaluate whether including either of these measures as a selection factor actually improves overall performance.

5.1 Setup

Each experiment evaluates a particular selection strategy trading off accuracy and diversity. As described above, a weight controls how strongly an individual’s distinctiveness value influences selection in addition to accuracy. A weight of 1.0 selects purely on the basis of distinctiveness; a weight of 0.0 makes selections based on accuracy only. We consider weight values between 0.0 and 1.0 in increments of 0.25. With these five weight values and one or the other of the two diversity measures to be weighted, we have ten nominal experimental conditions. However, when the relevance weight is 0.0, diversity does not influence selection at all so those two conditions collapse into one.

For each of the remaining nine experimental conditions, we performed ten repetitions of the following experimental procedure. We start with a population of 400 random individuals. For each of the ten repetitions, the KRK data set is independently partitioned into 1,000 randomly selected endgame problems; these will serve as the training set for computing accuracy and phenotype diversity. The remaining 27,056 endgame problems serve as our validation set.

Within a single repetition, we run our evolutionary algorithm for either 10,000 or 30,000 generations. On each generation, 20% of the population (80 individuals) is selected to survive, and 320 individuals are created by crossover reproduction. For any given pairing of individuals for breeding, we randomly select a number between 1 and 20 (inclusive) for the number of crossover points. As described above, selection for survival and breeding is based on a fitness that is the weighted sum of one of the diversity measures and accuracy on the training set. Selection uses a tournament process with a tournament size of five individuals.

Periodically (on exponentially increasing intervals; every 10, then every 100, and so on), we measure the performance of the population on the validation set. The plots that we report in the figures reflect the measures on the validation set averaged over the ten repetitions.

In summary, selection is biased by a distinctiveness value (either phenotype or estimated hamming diversity) and by accuracy on the training set. The weight, w , controls this bias as $F_i = wD_i + (1 - w)B_i$, where D_i is the individual’s distinctiveness score and $(1 - w)B_i$ is the individual’s accuracy score. When the weight w is 1.0, then accuracy is ignored altogether; likewise, when w is 0.0, then we have a

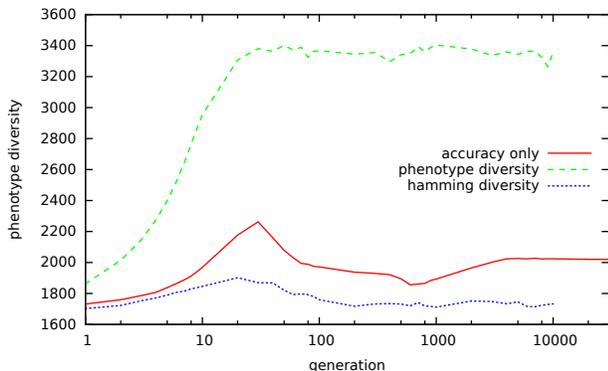


Figure 2: Phenotype diversity present in a population as a function of generation comparing selective-biases based purely on phenotype diversity (weight 1.0), estimated hamming diversity (weight 1.0), and accuracy alone.

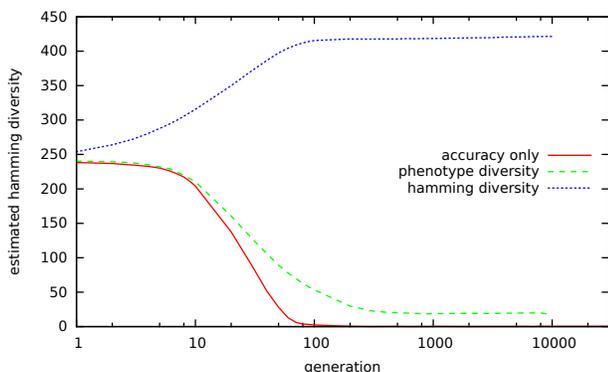


Figure 3: Estimated hamming diversity present in a population as a function of generation comparing selective-biases based purely on phenotype diversity (weight 1.0), estimated hamming diversity (weight 1.0), and accuracy alone.

traditional genetic algorithm using only accuracy to determine the winner of a tournament.

5.2 Hypotheses and Tests

Evolutionary algorithms have flourished based on the well-established phenomenon that selecting for fitness yields a corresponding increase in fitness (albeit subject to the problem of local optima and premature convergence). As our first hypothesis, we predict that selecting for diversity should lead to a corresponding increase in diversity. But conversely when selecting for accuracy, we expect diversity, as measured by phenotype and estimated hamming distance, should diminish and disappear.

To test this hypothesis, we first consider the results from our experiments in three cases: where selection is based purely on accuracy, where it is based on phenotype diversity only, and where selection is based on estimated hamming diversity only. For each of these three selective pressures, we want to see the values reported by our two diversity measures; when selecting for a given measure, that measure should increase over time and when selecting for accuracy

only, the measure should decrease. Of equal interest, when selecting for one diversity measure we want to see what happens to the other.

Figure 2 shows the phenotype diversity measure as a function of generation for the three conditions just identified. First, we see that when selecting for phenotype diversity that measure steadily increases over the first 50 generations and then levels off, remaining roughly at that level for about 10K generations. However, the other two conditions run counter to our expectations. When selecting for accuracy only, the phenotype measure does not notably decrease but rather increases over the first 50 generations (although not increasing by as much as when selecting for phenotype diversity), then decreases considerably over the next 50 generations, and finally stays somewhat level for the next 30K generations. In the third case when selecting for estimated hamming diversity, we see almost no change in the phenotype diversity measure over 10K generations.

If instead we consider the measure of estimated hamming diversity under the same three conditions, the picture is quite different. Figure 3 shows the results corresponding to Figure 2 but where the vertical axis is the hamming diversity measure. These results appear to be exactly what we predicted. When selecting for hamming diversity, the measure steadily increases and levels off. When we select for accuracy only, this measure starts to slowly decrease over the first 10 generations but then steadily decreases over the next 100 generations, at which point diversity, according to this measure, has been eliminated from the population. A similar effect is observed when we select for phenotype diversity. The drop-off is less pronounced and hamming diversity does not disappear entirely but the qualitative result is the same.

The results in these two figures partially confirm our hypothesis and yet raise further questions. On the one hand, the measure for which we are selecting indeed increases and then levels off much as we are familiar with when looking at traditional fitness measures based on accuracy alone. But we see very different effects between our two measures. Perhaps this is not so very surprising as they represent two different ways to measure diversity. However from these graphs, they appear to be extremely different measures. While this difference merits further discussion, it also raises the question of what impact selecting for these measures will have on accuracy.

Our primary goal in studying measures of diversity is the assumption that preserving or cultivating it will lead to improved performance. Thus, we hypothesize that including a distinctiveness measure in the selection process should improve the accuracy of the population on the performance task. We expect this because the population effectively explores a larger search space, thereby improving the chances that a better solution will be found. However, we also hypothesize that this improvement comes at the cost of search time. That is, a better solution should be found eventually but we may have to wait for it.

In the following results, we report *accuracy* as the performance of the ‘best’ individual in the population as a function of the number of generations. Normally, this would correspond to ‘fitness’, but when we include selective pressure for diversity we essentially change the notion of fitness from pure accuracy to something more, where that something ‘more’ is more than the sum of its parts.

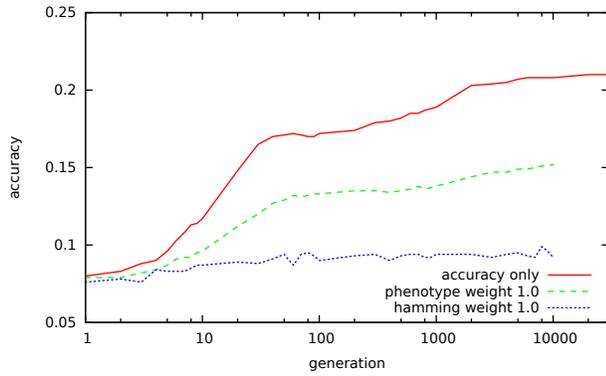


Figure 4: Comparison of the phenotype and estimated hamming diversity mechanisms to selection using accuracy alone. Selection of individuals in the case of phenotype and estimated hamming diversity is weighted by 1.0 on the diversity measure and 0 on accuracy.

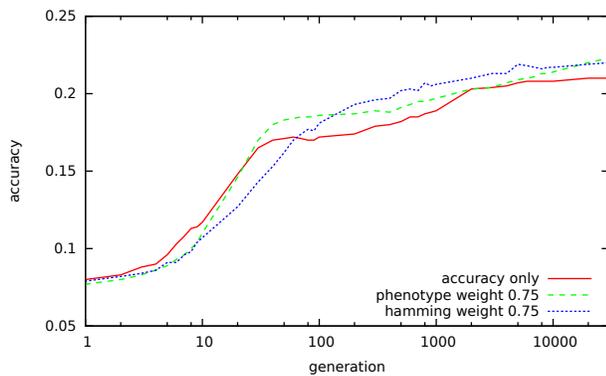


Figure 5: Comparison of the phenotype and estimated hamming diversity mechanisms to selection using accuracy alone. In the case of the diversity measures, diversity is weighted by 0.75 and accuracy by 0.25.

Figure 4 shows the performance of the best individual under three conditions: selecting for accuracy only (weight of 0.0 where diversity does not have any influence), phenotype diversity with a weight of 1.0, and estimated hamming diversity with a weight of 1.0 (where for both, accuracy has no influence). In the first case, when diversity is not considered at all, the algorithm does much better than the other two cases where the diversity measures are used exclusively. This is not really surprising as we hypothesized that *including* a selective bias for diversity would improve performance; we have no reason to expect an improvement when selecting for diversity only. That said, note that phenotype diversity does significantly better than estimated hamming diversity, and actually does improve somewhat. Because the phenotype measure is based on the performance behavior of individuals, accuracy plays an indirect role in this measure of diversity and very slowly drives the population towards improved performance.

When we change the weight w to 0.75, the picture changes significantly. Figure 5 shows the plots corresponding to Fig-

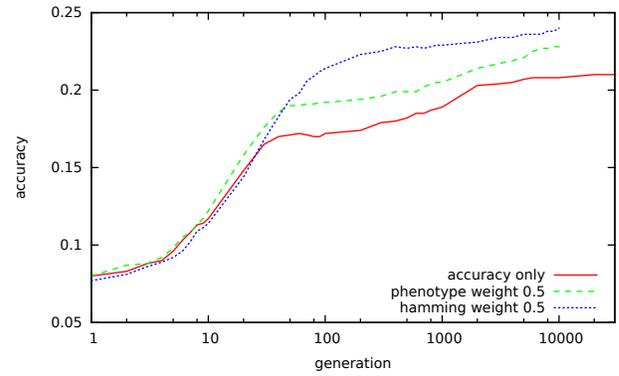


Figure 6: Comparison of the phenotype and estimated hamming diversity mechanisms to selection using accuracy alone. In the case of the diversity measures, diversity and accuracy are both weighted by 0.5.

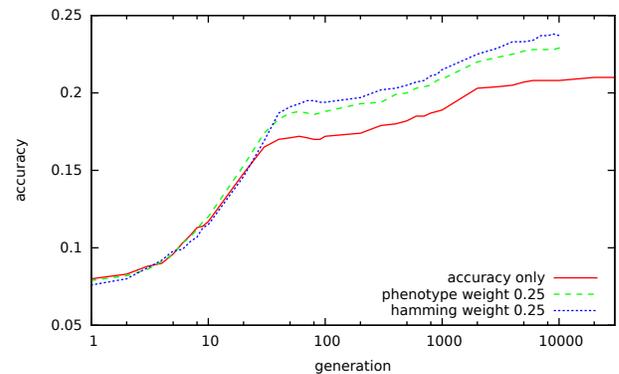


Figure 7: Comparison of the phenotype and estimated hamming diversity mechanisms to selection using accuracy alone. In the case of the diversity measures, diversity is weighted by 0.25 and accuracy by 0.75.

ure 4 but now selection is biased by the respective diversity measures and also influenced by accuracy. In this plot, we see the diversity-influenced conditions doing slightly better than selection based on accuracy only; however, the advantage is small and may not be significant. We also see a reversal of the advantage of phenotype diversity where now estimated hamming diversity appears to perform better. Note, that the phenotype diversity condition appears to continue improving and conceivably could surpass both of the other conditions; this is a topic of ongoing exploration.

Figures 6 and 7 show similar plots with weights of 0.5 and 0.25, respectively. In Figure 6 with a weight of 0.5, the estimated hamming diversity condition clearly dominates the others. In the case of weight 0.25 (Figure 7), hamming diversity still seems to have a slight advantage over phenotype diversity. Note, both of the diversity conditions appear to continue to improve in both figures.

Note that Figures 6 and 7 indicate that our original hypothesis was too conservative. We expected that the inclusion of diversity in the selection process would improve performance but that it would cost time. It appears, at least

for values of weight w between 0.25 and 0.5, that such an approach finds superior solutions at least as early as a method that selects only for accuracy. But Figure 5 shows that both conditions where selection includes diversity start off at a disadvantage with respect to a pure accuracy selection. Yet both diversity conditions surpass the accuracy-only condition between generations 20 and 100. Thus, we do find empirical support for our hypothesis: selection pressures that include diversity as a factor lead to improved performance but in some cases may require additional time to do so.

6. CONCLUSIONS

We presented two ways to measure the diversity within a population of individuals based on the distinctiveness of those individuals. Phenotype diversity reflects the behavioral distinctiveness of individuals with respect to collective performance on specific test cases. Estimated hamming diversity provides an improved method for computing the pair-wise differences between the genetic material of individuals in a population. Indeed, our comparisons suggest that estimated hamming diversity holds promise for boosting performance of genetic algorithms and merits further investigation.

Surprisingly, methods using diversity to influence selection lead to better solutions more quickly than a selection regime based purely on traditional fitness, or accuracy. We intend to explore the boundaries of this effect and hope to obtain further insight into this phenomenon.

7. ACKNOWLEDGMENTS

Thanks to Connor Riva, Echoe Jones, and Grady Goff for conversations that influenced the direction of this work. We also thank both Westmont and the James L. Stamps Foundation, Inc. for generous support that provided computer equipment on which we ran the experiments reported in this paper.

8. REFERENCES

- [1] A. Asuncion and D. Newman. UCI machine learning repository, 2007.
- [2] T. Friedrich, P. S. Oliveto, D. Sudholt, and C. Witt. Analysis of diversity-preserving mechanisms for global exploration. *Evolutionary Computation*, 17(4):455–476, Winter 2009.
- [3] D. Gupta and S. Ghafir. At overview of methods maintaining diversity in genetic algorithms. *International Journal of Emerging Technology and Advanced Engineering*, 2(5):56–60, May 2012.
- [4] M. Hutter. Fitness uniform selection to preserve genetic diversity. In *Proceedings of the 2002 Congress on Evolutionary Computation*, January 2001.
- [5] W. Iba. Searching for better performance on the king-rook-king chess endgame problem. In *Proceedings of the Twenty-fifth International FLAIRS Conference*, 2012.
- [6] C. Jassadapakorn and P. Chongstitvatana. Self-adaptation mechanism to control the diversity of the population in genetic algorithm. *International Journal of Computer Science and Information Technology*, 3(4):111–127, August 2011.
- [7] J. Jayachandran and S. Corns. A comparative study of diversity in evolutionary algorithms. In *Evolutionary Computation (CEC), 2010 IEEE Congress on*, pages 1–7, July 2010.
- [8] G. Squillero and A. P. Tonda. A novel methodology for diversity preservation in evolutionary algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 2223–2226, July 2008.
- [9] R. K. Ursem. Diversity-guided evolutionary algorithms. *Parallel Problem Solving from Nature*, 7:462–471, September 2002.