The Effects of Selection on Noisy Fitness Optimization

Sergiu Goschin
RL^3, Computer Science
Rutgers University
Piscataway, NJ, USA
sgoschin@cs.rutgers.edu

Michael L. Littman
RL^3, Computer Science
Rutgers University
Piscataway, NJ, USA
mlittman@cs.rutgers.edu

David H. Ackley
Computer Science
University of New Mexico
Albuquerque, NM, USA
ackley@cs.unm.edu

ABSTRACT

This paper examines how the choice of the selection mechanism in an evolutionary algorithm impacts the objective function it optimizes, specifically when the fitness function is noisy. We provide formal results showing that, in an abstract infinite-population model, proportional selection optimizes expected fitness, truncation selection optimizes order statistics, and tournament selection can oscillate. The "winner" in a population depends on the choice of selection rule, especially when fitness distributions differ between individuals resulting in variable risk. These findings are further developed through empirical results on a novel stochastic optimization problem called "Die4", which, while simple, extends existing benchmark problems by admitting a variety of interpretations of optimality.

Categories and Subject Descriptors

F.2.m [**Analysis of Algorithms and Problem Complexity**]: Miscellaneous; G.1.6 [**Numerical Analysis**]: Optimization

General Terms

Theory

Keywords

Evolutionary noisy optimization, convergence analysis, selection algorithms

1. INTRODUCTION

Genetic algorithms provide one of the most powerful and versatile approaches to optimization in common use today. Even in some well-studied and difficult combinatorial problems like the traveling salesman problem, evolutionary approaches are among the most reliable [5].

For real-life problems in which fitness evaluations require physical measurement or complex simulations, the same individual can be assigned different scores [7]. Optimizing in the face of such noisy fitness functions adds several additional difficulties beyond the challenge of ordinary optimization. One well known fact is that these inconsistent evaluations can mislead an algorithm, causing it to spend too much or too little of its resources on an individual relative to its "worth". One common response is to repeat and average fitness evaluations, which provides more accurate estimations at the expense of additional evaluations [10]. Note, however, that while this multiple-sample approach shrinks the variance of fitness distributions, it cannot reduce them to a single point—the problem of noisy optimization remains.

A second difficulty in noisy optimization is that the very notion of which individual is the best can have multiple interpretations. Given that an individual can have more than one fitness value, which one should "count" in the optimization process? The maximum? The mode? Depending on the form of the noise distribution, these options might not even make sense. A natural choice is the mean, or expected value. However, the median might be more appropriate for some applications, say if the fitness distribution has extreme outliers.

Existing analyses [11] and evaluations [8] of evolutionary algorithms optimizing noisy fitness functions focus on problems in which "reasonable" objective functions align. That is, the individual with the highest mean also has the highest median, 25th percentile, etc. In the case of the evaluation, this decision was made explicitly so that results remain comparable even if participants choose to optimize different objectives.

Consider, however, applications such as finance where the "right" decision depends critically on one's risk attitude. An optimization procedure applied to such a problem without concern to what objective it is attempting to optimize is of limited utility to the user. The user would want to choose an algorithm that optimizes a desired quantity, or, even better, would like to tell the algorithm what to optimize. Note that this notion of risk is relevant even in more traditional optimization problems such as jet-engine design where cost–safety tradeoffs play an important role.

In this paper, we undertake an analysis and empirical study of exactly the issue of what objective function an evolutionary algorithm strives to optimize. We note that two evolutionary algorithms that differ only in their selection rule can optimize different objectives and provide several concrete examples. Specifically, we show that whereas proportional selection seeks to optimize the expected value of the fitness distribution, truncation selection optimizes or-

GECCO'11, July 12–16, 2011, Dublin, Ireland.

der statistics such as the median. Tournament selection, in contrast, has no well defined target of optimization.

Section 2 provides the definition of the algorithms we consider. Section 3 introduces a noisy optimization problem called "Die4" that serves as our motivating illustration. Section 4 analyzes several selection mechanisms in the infinite population limit. Section 5 presents empirical results on finite-population algorithms running on Die4 to show how the theoretical findings help explain the results of experiments.

2. SELECTION MECHANISMS

Briefly, a genetic algorithm defines the evolution of a population of individual genomes under the influence of reproduction, mutation, and selection operations. We study a class of algorithms that has the following form.

The population size is set to a fixed N, and an initial collection of N genomes is produced. For each generation, each individual genome i in the population is evaluated, resulting in a fitness score f_i. In the setting we consider, this score is a random variable and thus the fitness score is more properly thought of as being sampled from a genome-specific *fitness distribution*. A new generation is then constructed depending on the sampled fitness values and the selection mechanism in use. We next describe several possible selection mechanisms.

In *truncation selection*, the N genomes are sorted based on their fitness scores. Each of the top $\theta \times N$ scoring genomes (breaking ties randomly, but consistently) is used, round-robin-style, to populate the next generation. A fitness threshold is defined at each generation and any individuals scoring below that line are dropped from the population. The evolutionary strategies $(\mu + \lambda)$ and (μ, λ) are examples of this scheme. In our work, the two behaved similarly so we chose not to report their results separately.

The idea of *fitness proportional selection*, or sometimes "roulette" selection, is that individual genomes are selected for the next generation at random from the entire current population. However, those with higher fitness scores have a better chance of being included. The new population is built up one genome at a time, with each picked in proportion to its fitness divided by the sum of the fitness of all genomes in the population. Absent any rescaling, (1) this scheme assumes only non-negative fitness, (2) genomes with a fitness of 0 have no chance of producing offspring, and (3) if all members of the population have zero fitness, the algorithm terminates.

The *tournament selection* scheme also builds up an entire new population each generation. A separate tournament is held for each position in the new population. During each tournament, τ distinct parents are chosen uniformly at random from the entire current population, and the genome with the largest fitness score of those τ individuals produces an offspring for the next generation.

In all these schemes, whenever an offspring is created, it has a 99% chance of being identical to the parent and a 1% chance of being "mutated" (a problem-specific modification to a new, but perhaps related, genome).

Although many other selection rules have been studied [6], we focus on these because of their relative simplicity and broad appeal. We defer an analysis of a more complete set of rules to future work.

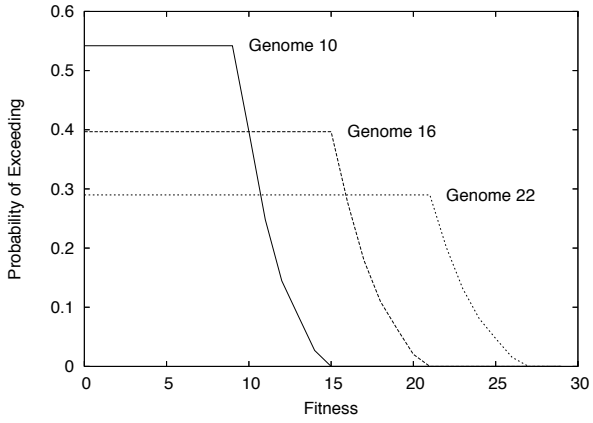

Figure 1: Plot of probability of exceeding a given value for three genomes discussed in the text.

3. DIE4: RISKY OPTIMIZATION

Die4 is a simple game we created to study optimization under risk. It is played with a single 6-sided die, which the player can roll repeatedly. Each time the player rolls the die, the resulting value is added to the player's score. At any time, the player can stop the game and declare the current total his score. However, if the die ever comes up 4, the player "dies", losing his points and ending the game. The individual's fitness is exactly the score when the game ends. Note that, although the probability of failure remains constant at 1/6 throughout the game, the number of points the player risks losing by rolling a 4 increases steadily from round to round.

We seek the best cutoff rule for playing Die4. Specifically, an individual genome consists of the value T and a fitness function evaluation consists of the playing the game using the rule "continue until a 4 is rolled *or* the sum reaches at least the target value T". Which value of T is best?

To explore the theoretical properties of this system, we performed a set of evolutionary runs on three specific genomes (those with thresholds T of 10, 16, and 22).

The fitness distributions of these genomes are depicted in Figure 1 via their complementary cumulative distribution functions (ccdfs). That is, the x-axis represents possible fitness values and the y-axis represents the probability that the corresponding genome will be assigned a fitness equal to or greater than that fitness value. If the fitness distribution for genome i is represented by the probability density function $f_i(x)$, the ccdf can be expressed as $G_i(x) = \int_x^\infty f_i(s)ds$. Note that while Genome 10 has a higher probability of a non-zero fitness, Genomes 16 and 22 have a higher probability of reaching fitness scores of 20 or more.

We examined the effect of running different selection rules on populations constrained to these three genomes. For each rule, we initialized a population of $N = 100$ genomes by selecting from this set of three genomes uniformly at random, ran for 1000 generations, and repeated this process 20 times. For each of these runs, we recorded the distribution over genomes up to generation 1000. To mutate an individual, we replaced the genome with one of the three uniformly at random.

Figure 2 reports the results of this study, where generation number appears along the x axis and the grey levels

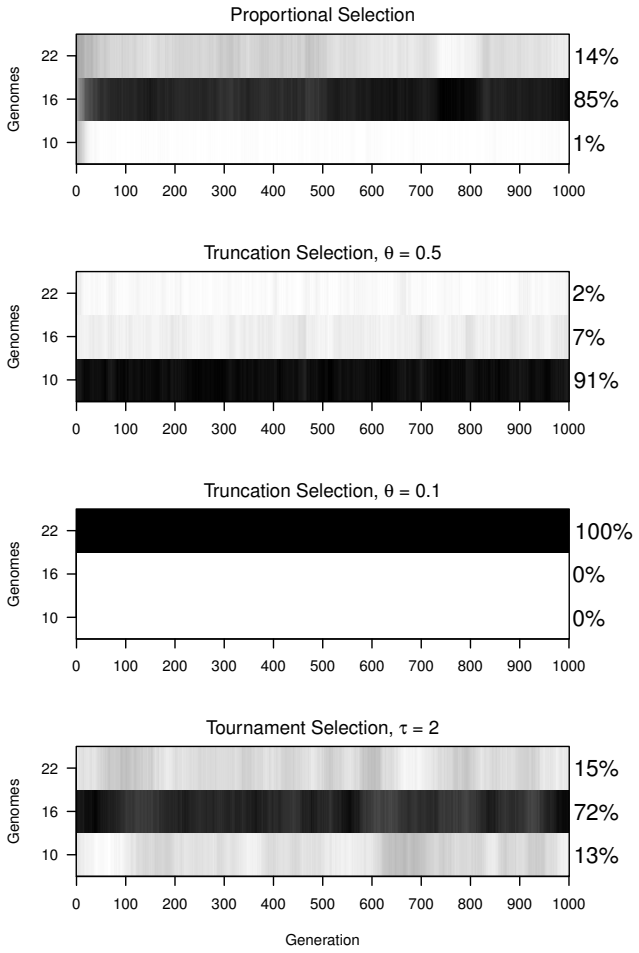

Figure 2: Die4, population 100, repetitions 20, mutation rate 0.5%. The percentages on the right side of each graph represent the fraction of the total number of individuals of a particular type through all generations.

corresponding to the three genomes emphasize the fraction of the population consisting of copies of that genome. Given that fitness evaluations are noisy, it is not unreasonable to expect that results vary from run to run. However, what we observe is a more complicated pattern. For instance, we see truncation selection with $\theta = 0.1$ *consistently* converging to Genome 22, while truncation with $\theta = 0.5$ consistently converges to Genome 10. Proportional selection leans heavily toward Genome 16, as does tournament selection $\tau = 2$, but with a much patchier time series.

In the remainder of the paper, we show that this pattern of results is well explained by analyzing the behavior of the different selection rules in the infinite population limit.

4. INFINITE POPULATION ANALYSIS

We analyze the behavior of the selection mechanisms in the classical infinite population model [14] extended to treat noisy fitness evaluations. This model provides a useful abstraction that allows for analysis without the complication of finite sampling effects. While the effects of stochasticity from a finite population are important, the goal here is to get insight into the general direction that selection mechanisms push a population in the context of noisy fitness functions. Of course, this simplification is a double-edged sword. While it reduces complications in proofs, it also hides possibly relevant details. Hence, Section 5 provides supporting evidence from computational simulations.

An infinite population is fully described by a finite set of genomes $\Omega = \{1, 2, \ldots, n\}$ and a real-valued weight vector w representing population densities, with $\sum_{i=1}^{n} w_i^t = 1$ and $w_i^t \geq 0$ for $t \geq 0$, where t is the index of the generation. We can assume without loss of generality that $w_i^0 > 0$ (all genomes are present in the initial population) and this vector is viewed as an input parameter. The weights describe the proportion of each genome in the initial population. It is worth noting that our analysis is focused on selection (with no mutation or crossover), so if $w^t = 0$, $w^{t'} = 0$ for all $t' > t$ (no new genomes are introduced).

To introduce noisy fitness to the model, for all genomes $i \in \Omega$, let $f_i(x)$ be its probability density function, or *fitness distribution*. We assume the $f_i(x)$s are non-zero and continuous and have strictly decreasing complementary cumulative distribution functions (ccdfs) $G_i(x)$ (over their support) and for simplicity they have common support over the space of fitness values $\mathcal{X}$. Note that, due to the constraints on the $G_i(x)$s, all of the probability density functions have associated quantile functions q_i that are properly defined. (For all $x \in \mathcal{X}$, there exists a unique value $y \in [0, 1]$ such that $q_i(y) = 1 - G_i^{-1}(y) = x$.)

An example meeting these constraints is if $\mathcal{X} = (0, 1)$ and $f_i(x)$ are beta density functions with different parameters. We will use a running example to help ground the intuition behind the proof; see Figure 3 for an example of three different beta densities.

Given that $w_i^t \in (0, 1)$ represents the proportion of the population at time t occupied by genome i,

$$f^t(x) = \sum_{i=1}^{n} w_i^t f_i(x)$$

is the mixture distribution that characterizes the entire population at time t. We call this combined distribution over fitness values the *population distribution*. (As an example visualization, in Figure 3, the population distribution at generation 0 is shown in red (darker solid line), while the population distribution after a few generations of using proportional selection is shown in green (lighter solid line).)

We will show that proportional selection and truncation selection push the weights towards a vertex of the unit simplex and the identity of that vertex is determined by the characteristics of the fitness distributions (and is potentially different depending on the selection mechanism). In contrast, tournament selection need not converge to any fixed weight vector under our assumptions.

4.1 Proportional Selection: Expected Value

While it appears to be a folk theorem in the literature that proportional selection finds the individual whose fitness distribution has the maximum expectation [9], we were unable to locate a proof of this claim for a general noisy setting in the literature. We provide such a proof in this section. As a corollary, note that transforming the fitness before doing selection optimizes the expectation of the transformed fitness.

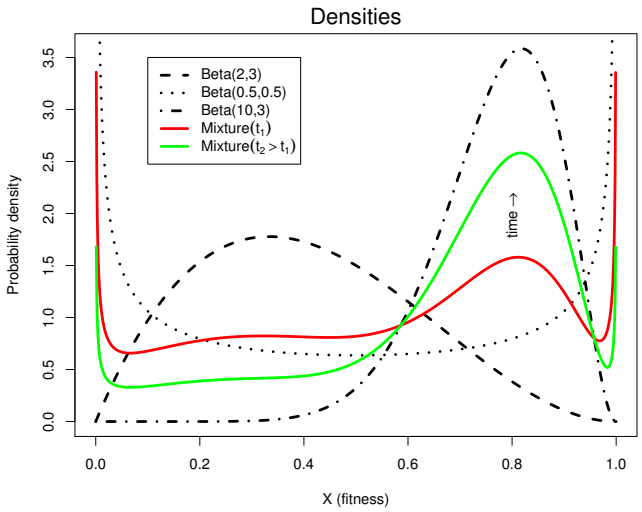

Figure 3: Three example fitness distributions and two mixtures shown as densities.

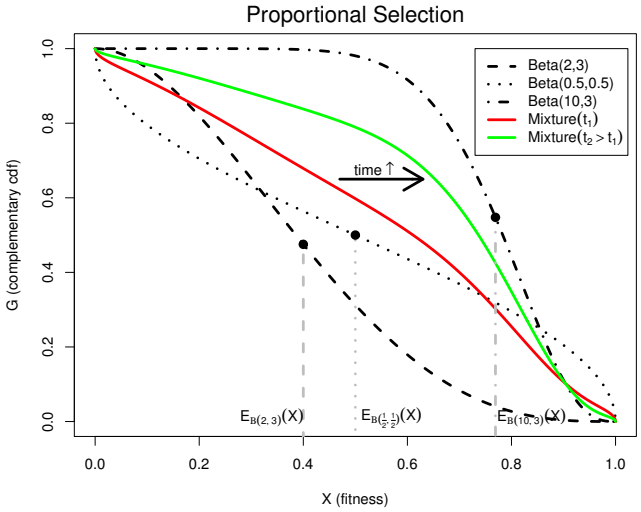

Figure 4: The same distributions as ccdfs. Expected values are marked with vertical lines.

Under the proportional selection rule, individual genomes receive a fitness value and then their probability of reproduction is proportional to this value. Ignoring the normalization factor for the moment, in the infinite population model, genome i occupies w_i^t fraction of the population at time t and those individuals have their fitness drawn from the density $f_i(x)$. Because we are imagining an infinite population, each of these fitness values actually appears and the fraction of the time fitness value x appears due to genome i is $w_i^t f_i(x)$. The total fitness weight of this genome in the resulting population is therefore $w_i^{t+1} = \int_{\mathcal{X}} w_i^t x f_i(x) dx / Z$.

As a result, if we define

$$v_i = E_{x \sim f_i}[x] = \int_{\mathcal{X}} x f_i(x) dx, \tag{1}$$

the expected value of the fitness distribution for genome i, then the evolution of the weights under proportional selection is given by:

$$w_i^{t+1} = \frac{w_i^t v_i}{\sum_{j \in 1 \ldots n} w_j^t v_j}, \forall i \in 1 \ldots n \text{ and } \forall t \geq 0. \tag{2}$$

For simplicity and without loss of generality, we assume that $v_1 < v_2 < \cdots < v_n$. That is, genomes are sorted in increasing expected fitness order and all expected fitness values are unique. (If v_i is not unique, we can consider the mixture of the distributions that have the same v_i as a single genome that has as its fitness distribution the weighted mixture.)

To demonstrate convergence, we prove the following theorem.

THEOREM 4.1. (Convergence of proportional selection)

$$\lim_{t \to \infty} w_i^t = 0, \forall i \in 1 \ldots n-1, \text{ and } \lim_{t \to \infty} w_n^t = 1. \tag{3}$$

The theorem states that the weight of the fitness distribution with the largest expected value will converge to 1, asymptotically. Before giving the formal proof, we will give a visualization of the result. In Figure 4, the genome with the fitness distribution of Beta(10, 3) has the highest expected value, marked with the vertical line. The theorem states that, over a series of generations, the weights of the other genomes decrease to 0 and thus the weight of the maximum expected value fitness distribution Beta(10, 3) will dominate the population in the limit. As a result, the population distribution will converge to the Beta(10, 3) distribution. As can be seen in the figure, the population distribution at one time step (green / light gray ccdf) is closer to the maximum expected value fitness distribution than at an earlier time step (red / dark gray ccdf).

PROOF. Using induction, we show $w_i^t = w_i^0 / (\sum_j w_j^0 (v_j / v_i)^t)$. For the base case, note that $w_i^0 = w_i^0 / (\sum_j w_j^0)$ because w^0 is normalized.

For the inductive step,

$$\begin{aligned}
w_i^{t+1} &= w_i^t v_i / (\sum_j w_j^t v_j) \\
&= w_i^t / \sum_j w_j^t (v_j / v_i) \\
&= \frac{w_i^0 / (\sum_j w_j^0 (v_j/v_i)^t)}{\sum_j [w_j^0 / (\sum_k w_k^0 (v_k/v_j)^t)](v_j/v_i)} \\
&= \frac{w_i^0 (v_i)^t / (\sum_j w_j^0 (v_j)^t)}{\sum_j [w_j^0 (v_j)^t / (\sum_k w_k^0 (v_k)^t)](v_j/v_i)} \\
&= w_i^0 (v_i)^t / \sum_j w_j^0 (v_j)^t (v_j/v_i) \\
&= w_i^0 / \sum_j w_j^0 (v_j/v_i)^{t+1}.
\end{aligned}$$

For genome n and any $i < n$, $v_n > v_i$. Thus, using the result above,

$$\begin{aligned}
\lim_{t \to \infty} w_n^t &= \lim_{t \to \infty} w_n^0 / (\sum_j w_j^0 (v_j/v_n)^t) \\
&= \lim_{t \to \infty} w_n^0 / w_n^0 = 1.
\end{aligned}$$

Since $\sum_j w_j^t = 1$, for $i < n$, $\lim_{t \to \infty} w_i^t = 0$.

□

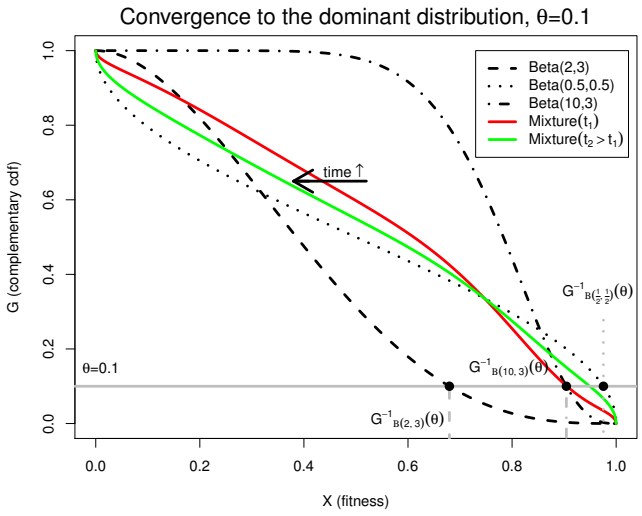

Figure 5: Visualization of population fitness approaching the fitness distribution with the highest x_i in truncation selection ($\theta = 0.1$)

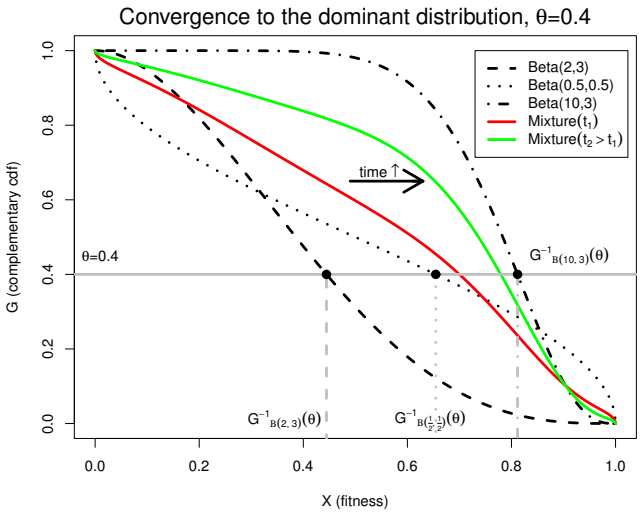

Figure 6: Visualization of population fitness approaching the fitness distribution with the highest x_i in truncation selection ($\theta = 0.4$)

Thus, in proportional selection, the genome with the highest expected fitness is preferentially reproduced at each generation and therefore comes to dominate the others.

4.2 Truncation Selection: Order Statistics

We next show that, in θ truncation selection, the dominant genome is the one with the largest θ order statistic. Let $\theta \in (0, 1)$ be the threshold parameter of the algorithm (the top θ percent of the population gets to reproduce into the next generation).

Define $x_i \in \mathcal{X}$ such that $\Pr_{x \sim f_i(x)}(x \geq x_i) = \theta = G_i(x_i)$. Thus, x_i is the value of the quantile function $q_i(1-\theta)$ of fitness distribution f_i and it is unique by our assumptions noted at the beginning of this section. It is the value x_i at which θ fraction of the fitness values are above x_i and $1-\theta$ fraction of the fitness values are below x_i. For example, if $\theta = 1/2$, x_i is the median of f_i. Figure 5 provides a visualization in which the x_is are marked with vertical lines and appear at the intersection of the G_i functions with the horizontal line at θ (labeled as $G^{-1}_{\text{Beta}()}(\theta)$ in the figure). Here, $\theta = 0.1$ and Beta$(\frac{1}{2}, \frac{1}{2})$ has the highest x_i value for this θ. Once again, we assume without loss of generality that $x_1 < x_2 < \cdots < x_n$—all of these fitness thresholds are unique and sorted.

Let $x_c^t \in \mathcal{X}$ be the *population common point* at generation t, which is the value for which

$$\sum_{i \in 1 \ldots n} w_i^t G_i(x_c^t) = \theta. \tag{4}$$

Thus, x_c^t for the population distribution is analogous to x_i for genome i's fitness distribution. In Figure 5, x_c^t is the intersection of the population distribution's ccdf (at time t_1 or t_2) with the θ line. This x_c^t threshold plays the important role of deciding the change in weights for the genomes in the next generation. When fitness evaluations are made for the w_i^t fraction of individuals with genome i, $G_i(x_c^t)$ fraction of them will survive because their fitness values will surpass x_c^t. With the normalization factor included,

$$w_i^{t+1} = \frac{w_i^t G_i(x_c^t)}{\theta}, \forall i \in 1 \ldots n \text{ and } \forall t \geq 0 \tag{5}$$

are the weights at generation $t+1$ as a function of the weights at generation t. The change in weights from generation to generation induces a change in the population distribution and thus change the next population common point x_c^{t+1}. The exact impact on x_c^{t+1} is complex and thus the analysis of truncation selection is a bit more indirect than that of proportional selection.

Convergence to genome n is proven by the following theorem.

THEOREM 4.2. (Convergence of truncation selection)

$$\lim_{t \to \infty} w_i^t = 0, \forall i \in 1 \ldots n-1 \text{ and } \lim_{t \to \infty} w_n^t = 1. \tag{6}$$

For our running example with beta distributions, it is interesting to note that, for $\theta = 0.4$, truncation selection will lead to the same distribution dominating the population as with proportional selection (compare Figures 4 and 6), but setting $\theta = 0.1$ leads to distribution Beta$(\frac{1}{2}, \frac{1}{2})$ converging to 1. (See Figure 5.) This example illustrates our main point—the genome that dominates the population in the long term is a function of the selection mechanism used.

PROOF. Our argument consists of three major steps:

1. $\forall t > 0, x_c^t < x_c^{t+1}$: The population common point is increasing.

2. $\exists t' > 0$ s.t. $x_c^{t'} > x_{n-1}$: After a finite number of generations, the population common point exceeds the second largest fitness threshold (and will not go below it again due to Step 1).

3. $x_c^t > x_{n-1} \implies \lim_{t' \to \infty} w_n^{t'} = 1$: If the population common point is above the second largest fitness threshold, then the population will converge to the genome with the largest fitness threshold.

Putting these three facts together completes the proof. We prove each in turn.

1. The population common point always increases.

At generation t, let $W = \{i \text{ s.t. } x_i > x_c^t\}$ be the "winners" (any genome i with fitness threshold above the population common point at time t) and $L = \{i \text{ s.t. } x_i \leq x_c^t\}$ be the "losers" (any genome i with fitness threshold at or below the population common point at time t). (We suppress the dependence of W and L on t to simplify notation).

Define $\delta_i = G_i(x_c^t) - \theta$ if $i \in W$ and $\delta_i = \theta - G_i(x_c^t)$ if $i \in L$. It captures the amount that genome i's G_i value deviates from θ at the current population common point. Note that $\delta_i \geq 0$ for all i and that $\delta_i > 0$ for at least one $i \in W$ (otherwise, all genomes are tied in their order statistics). Also, note that $w_i^{t+1} > w_i^t$ for $i \in W$ and $w_i^{t+1} \leq w_i^t$ for $i \in L$. These facts follow from Equation 5 and the definition of W and L.

We proceed by contradiction. Assume the fitness common point remains the same or decreases, $x_c^{t+1} \leq x_c^t$. This assumption implies $G_i(x_c^{t+1}) \geq G_i(x_c^t)$ (by the fact that G_is are strictly decreasing). Now,

$$\begin{aligned}
\theta &= \sum_{i \in 1 \ldots n} w_i^{t+1} G_i(x_c^{t+1}) \\
&\geq \sum_{i \in 1 \ldots n} w_i^{t+1} G_i(x_c^t) \\
&= \sum_{i \in W} w_i^{t+1} G_i(x_c^t) + \sum_{i \in L} w_i^{t+1} G_i(x_c^t) \\
&= \sum_{i \in W} w_i^{t+1} (\theta + \delta_i) + \sum_{i \in L} w_i^{t+1} (\theta - \delta_i) \\
&= \theta + \sum_{i \in W} w_i^{t+1} \delta_i - \sum_{i \in L} w_i^{t+1} \delta_i \\
&> \theta + \sum_{i \in W} w_i^{t} \delta_i - \sum_{i \in L} w_i^{t} \delta_i \\
&= \sum_{i \in W} w_i^{t} (\theta + \delta_i) + \sum_{i \in L} w_i^{t} (\theta - \delta_i) \\
&= \theta,
\end{aligned}$$

which is a contradiction ($\theta > \theta$). Thus, the fitness common point must increase.

2. The population common point eventually exceeds the second largest fitness threshold.

Let $\Delta = G_n(x_{n-1}) - \theta$. Note that $\Delta > 0$ because $G_n(x_n) = \theta$ (by definition of x_n) and $G_n(x_{n-1}) > G_n(x_n)$ (because the x_is are sorted and the G_is are strictly decreasing). This quantity represents how likely it is for a fitness evaluation for genome n to fall between x_{n-1} and x_n.

Now, as long as $x_c^t \leq x_{n-1}$, $G_n(x_c^t) \geq G_n(x_{n-1})$ (because the G_is are strictly decreasing). By Equation 5,

$$\begin{aligned}
w_n^{t+1} &= w_n^t G_n(x_c^t)/\theta \\
&\geq w_n^t G_n(x_{n-1})/\theta \\
&= w_n^t (\Delta + \theta)/\theta \\
&= w_n^t (1 + \Delta/\theta).
\end{aligned}$$

Thus, $w_n^{t'}$ grows without bound as long as $x_c^{t'} \leq x_{n-1}$. Therefore, there must be some time point t' when $x_c^{t'} > x_{n-1}$ (and, due to Step 1, it will never go below x_{n-1} again).

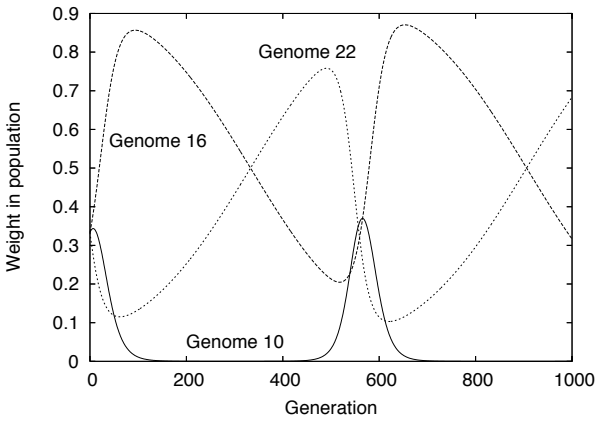

Figure 7: Oscillation of tournament selection ($\tau = 2$) in the infinite population model.

3. Once the population common point exceeds the second largest fitness threshold, convergence to genome n is guaranteed.

Let $\hat{t}$ be the first time the population common point goes over the second largest fitness threshold (and this threshold is well defined as the time is discrete and the fitness support is continuous) and define a to be the probability that the fitness of the genome with the second largest fitness threshold is larger than θ and smaller than the resulting population common point: $a = \theta - G_{n-1}(x_c^{\hat{t}})$. For all $i < n$,

$$\begin{aligned}
w_i^{t+1} &= w_i^t G_i(x_c^{\hat{t}})/\theta \\
&\leq w_i^t G_{n-1}(x_c^{\hat{t}})/\theta \\
&= w_i^t (\theta - a)/\theta \\
&= w_i^t (1 - a/\theta).
\end{aligned}$$

As a result of the fact that w_i^t is multiplied by a number bounded away from 1, as t increases, w_i^t goes to 0, as desired. □

4.3 Tournament Selection: Oscillation

We briefly look at tournament selection with $\tau = 2$. We find that the consistent drive toward a "maximum" genome, seen in truncation and proportional selection, does not appear to be a feature of tournament selection.

In the infinite-population framework, tournament selection with $\tau = 2$ consists of drawing two genomes from the population in proportion to their weights and then comparing their fitness values head to head. If h_{ij} is the probability that genome i has a higher fitness value than genome j, the change in weights can be written

$$w_i^{t+1} = 2\, w_i^t \sum_j h_{ij} w_j^t. \tag{7}$$

(Note that $h_{ij} + h_{ji} = 1$.) We expect convergence of this iteration if there is some genome i for which $h_{ij} > 0.5$ for all j. However, as in the famous example of "non-transitive dice" [13], it can be the case that there is i, j, k such that $h_{ij} > 1/2$, $h_{jk} > 1/2$, and $h_{ki} > 1/2$ leading to a rock-paper-scissors-like ring of dominance.

If we assume fitness ties are broken at random, the Die4 example of Section 3 exhibits precisely this pattern. Specifically, $h_{16,10} = 0.535$, $h_{10,22} = 0.548$, and $h_{22,16} = 0.504$.

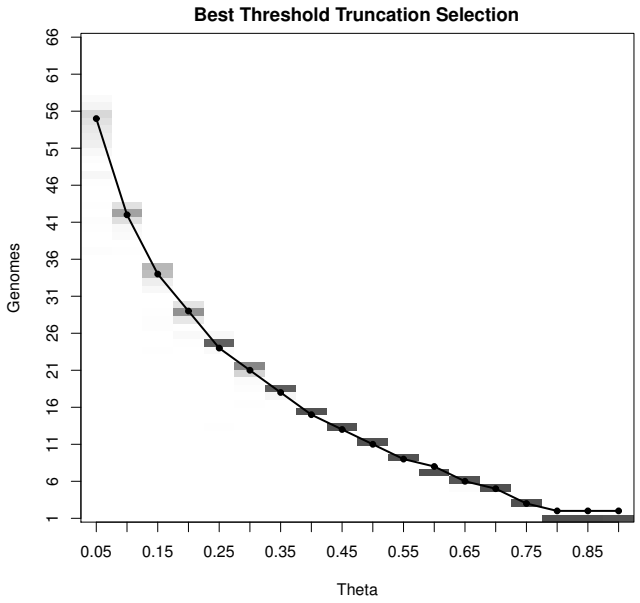

Figure 8: Different thresholds with their corresponding optimal values and empirical histograms

Genome 16 beats genome 10 because they succeed nearly as often and genome 16 gets the higher score. Genome 22 beats genome 16 for a similar reason. However, genome 10 beats genome 22 because it is significantly more likely to get a non-zero score.

Figure 7 illustrates changes in the population over generations when tournament selection ($\tau = 2$) is used in an infinite population version of Die4. Note that instead of converging to one genome, we see oscillatory behavior with each genome growing and then shrinking over time. This plot suggests an explanation for the tournament results reported in Figure 2—instead of steady state behavior or convergence, the weights for the three genomes fluctuate over time.

5. EXPERIMENTAL COMPARISONS

To validate our analysis in a more realistic setting, we evaluated Die4 in a context where a larger number of genomes have a chance to be part of the population. In contrast to Section 3, populations could include any genome from 1 to 64. (Recall that genome i continues rolling until a 4 is rolled or a total of at least i is reached.)

In all the experiments from this section, we only use selection followed by mutation (no crossover), with a mutation rate of 1%. When a genome was mutated, it transformed uniformly at random into one of the 64 possible genomes. (Similar results are obtained using mutations that change the genome to a nearby value.)

Using our knowledge of the Die4 game, we used dynamic programming to analytically compute that the genomes with maximum expected fitness are genomes 17 and 18, the maximum median is achieved by genome 11, and the maximum 10th percentile is that of genome 42.

To understand the extent to which finite population sizes are well modeled by the infinite population analysis in Section 4, we varied the population size and plotted heatmaps

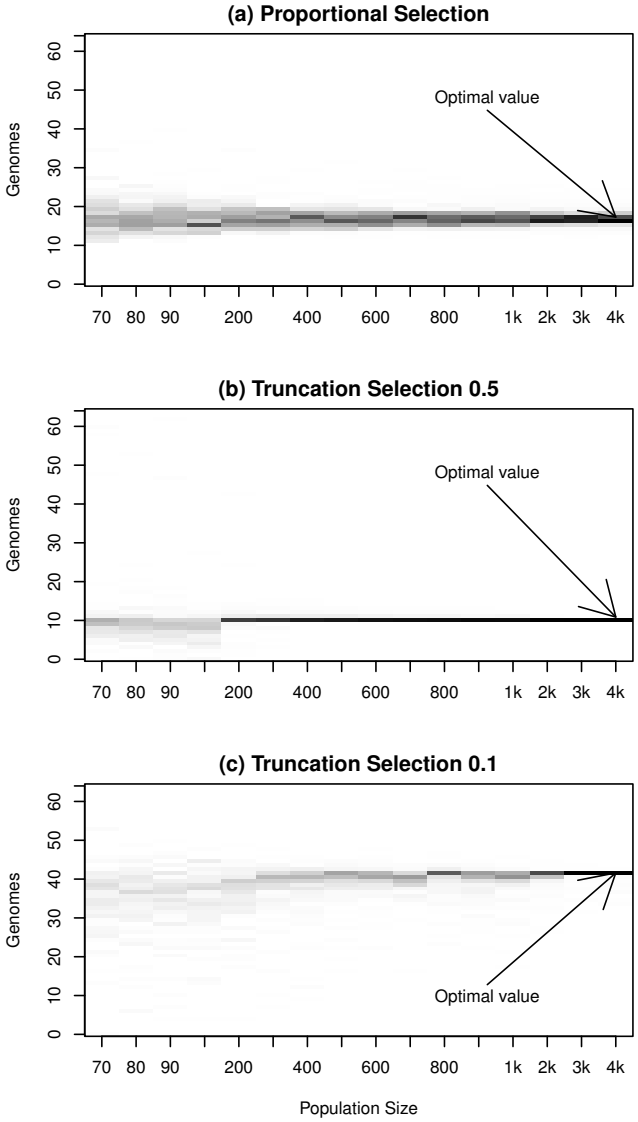

Figure 9: Variable population size, number of generations fixed at 1000, mutation 1% uniformly random. The infinite population model predicted value ("Optimal value") is also shown in the figures

of the distribution of genomes at the end of a fixed number of generations. The parameters we used were: population size 70 to 4000, number of generations 1000, number of repetitions 100.

As shown in Figure 9(a), proportional selection has a more diffuse set of "winners" after 1000 generations when the population is small, whereas, as the population size increases, the distributions of best genomes becomes peaked around the optimal values. It is interesting to note that, while our analysis does not apply directly to small populations, the final fitness values in small population runs are still grouped around the value predicted for the infinite population limit.

For truncation selection, we tested both scenarios mentioned in Section 3, $\theta = 0.5$ (Figure 9(b)) and $\theta = 0.1$ (Figure 9(c)). For truncation with $\theta = 0.5$, the convergence to the behavior predicted by the infinite population model is

fast (for a population of 200 individuals, after 1000 generations, the dominant genome is the optimal one in the vast majority of the experiments). On the other hand, it takes a population of over 1000 individuals for truncation selection with $\theta = 0.1$ to converge to optimal, with very noisy behavior for populations smaller than 400 individuals.

The close correspondence of the infinite population analysis and the finite population experiments across a wide range of θ values is illustrated in Figure 8. There, for each θ level in $(0.05, 0.9)$ (with a 0.05 increment), we plotted as a heatmap the histogram of the weights of the genomes after 1000 generations for a population size of 500 and 100 repetitions of the experiment for each θ. We also computed the theoretic maximum in the infinite population model and plotted it as a thin black line. For intermediate values of θ, the two line up well. For values of θ close to 1, the empirical behavior of this relatively small population slightly deviates from the expected behavior of the infinite population.

6. RELATED WORK

In the context of genetic algorithms, one of the first papers to empirically investigate the effects of noise [7] looked at the tradeoff between sampling a genome multiple times versus increasing the size of the population. On the theoretical side, a lot of analysis has looked at the effects of normal, additive noise [11], for which the genome with maximum expected fitness must match the one with maximum median fitness. The same holds true for work in evolution strategies, where the theoretical analysis has usually focused on the noisy sphere model [2]. Other papers, however, have looked at different types of randomness such as Cauchy noise [1].

More closely related to the present paper are those that study the effects of the noisy fitness on the selection mechanism. The effects of normal additive noise on stochastic tournament selection has been studied [4]. Another relevant paper [12] argued for changing the selection mechanisms in the case of noisy fitness optimization, observing the fact that the total order between genomes in the case of deterministic optimization becomes a partial order in the noisy case, requiring a different perspective on selection. Thorough and readable reviews of the existing literature on evolutionary noisy optimization are available [3, 9].

7. CONCLUSIONS

This paper presented a formal analysis in the infinite population model of the behavior of several widely used selection algorithms—proportional selection, truncation selection and tournament selection—when fitness functions are noisy. We showed that different selection methods optimize different objectives (proportional selection maximizes expected value, truncation selection maximizes the order statistic corresponding to the truncation threshold, and tournament selection need not maximize any concrete metric). We introduced a new stochastic optimization domain (Die4) and verified that the predicted theoretical behavior is consistent with the experiments for finite populations.

A natural next step is to provide more detailed analysis for the finite-population case. Also, a lot of valuable theoretical work has focused on analyzing the convergence rate of different methods. Understanding the behavior of these algorithms in the more general setting could be of great use when applying evolutionary algorithms to naturally occurring noisy optimization problems.

In future work, we want to drop some of the restrictive modeling assumptions we made such as the common support for the noise distribution and the constraint for the ccdf function to be strictly decreasing. Insights from such studies, in turn, should lead to a better understanding of the behavior of selection algorithms in noisy evolutionary optimization.

8. REFERENCES

[1] D. Arnold and H.-G. Beyer. On the effects of outliers on evolutionary optimization. In *Intelligent Data Engineering and Automated Learning*, volume 2690 of *Lecture Notes in Computer Science*, pages 151–160. Springer Berlin / Heidelberg, 2003.

[2] D. V. Arnold and H.-G. Beyer. *Noisy Local Optimization with Evolution Strategies*. Kluwer Academic Publishers, Norwell, MA, 2002.

[3] H.-G. Beyer. Evolutionary algorithms in noisy environments: Theoretical issues and guidelines for practice. *Computer Methods in Applied Mechanics and Engineering*, 186(2-4):239 – 267, 2000.

[4] J. Branke and C. Schmidt. Selection in the presence of noise. In *Genetic and Evolutionary Computation: GECCO 2003*, Lecture Notes in Computer Science, pages 766–777. Springer Berlin / Heidelberg, 2003.

[5] M. Dorigo and L. M. Gambardella. Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1997.

[6] S. Ficici, O. Melnik, and J. Pollack. A game-theoretic and dynamical-systems analysis of selection methods in coevolution. *Evolutionary Computation, IEEE Transactions on*, 9(6):580–602, 2005.

[7] J. M. Fitzpatrick and J. J. Grefenstette. Genetic algorithms in noisy environments. *Machine Learning*, 3:101–120, 1988.

[8] N. Hansen, S. Finck, R. Ros, and A. Auger. Real-parameter black-box optimization benchmarking 2009: Noisy functions definitions. Technical Report 6869, INRIA, 2009.

[9] Y. Jin and J. Branke. Evolutionary optimization in uncertain environments—A survey. *IEEE Transactions on Evolutionary Computation*, 9(3):303–317, 2005.

[10] V. H. Meisner and C. Igel. Hoeffding and Bernstein races for selecting policies in evolutionary direct policy search. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 401–408, 2009.

[11] B. L. Miller and D. E. Goldberg. Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, 9:193–212, 1995.

[12] G. Rudolph. A partial order approach to noisy fitness functions. In *Congress on Evolutionary Computation, Seoul, Korea*, pages 318–325, 2001.

[13] C. Rump. Strategies for rolling the Efron dice. *Mathematics Magazine*, 74(3):212–216, 2001.

[14] M. D. Vose. *The Simple Genetic Algorithm: Foundations and Theory*. MIT Press, Cambridge, MA, 1998.