Time-Dependent Performance Comparison of Stochastic Optimization Algorithms

David Shilane[*] Jarno Martikainen[†]

Seppo Ovaska[‡]

[*]Division of Biostatistics, School of Public Health, University of California, Berkeley, dshilane@stanford.edu

[†]Helsinki University of Technology, martikainen@iki.fi

[‡]Helsinki University of Technology, sovaska@cc.hut.fi

Time-Dependent Performance Comparison of Stochastic Optimization Algorithms

David Shilane, Jarno Martikainen, and Seppo Ovaska

Abstract

This paper proposes a statistical methodology for comparing the performance of stochastic optimization algorithms that iteratively generate candidate optima. The fundamental data structure of the results of these algorithms is a time series. Algorithmic differences may be assessed through a procedure of statistical sampling and multiple hypothesis testing of time series data. Shilane et al. propose a general framework for performance comparison of stochastic optimization algorithms that result in a single candidate optimum. This project seeks to extend this framework to assess performance in time series data structures. The proposed methodology analyzes empirical data to determine the generation intervals in which algorithmic performance differences exist and may be used to guide the selection and design of optimization procedures for the task at hand. Such comparisons may be drawn for general performance metrics of any iterative stochastic optimization algorithm under any (typically unknown) data generating distribution. Additionally, this paper proposes a data reduction procedure to estimate performance differences in a more computationally feasible manner. In doing so, we provide a statistical framework to assess the performance of stochastic optimization algorithms and to design improved procedures for the task at hand.

1 Introduction

Many optimization procedures iteratively estimate a function's global optimum over the course of many generations. When an *elitist selection* mechanism is employed [Bäck, 1996, Fogel, 2005], results in subsequent generations are refined estimates of those previously obtained, and these generational results may be encompassed in a *time series* data structure. We are interested in studying both the convergence of the algorithm's final result and the rate at which its estimates improve as a function of generation. For stochastic algorithms, probabilistic analysis is necessary to ascertain the quality and reliability of a procedure's estimates of the global optimum. This assessment often involves a statistical performance comparison of the competing algorithms. Shilane et al. [2006] establish a general procedure for the statistical performance comparison of competing algorithms that each result in a single candidate optimum. This paper seeks to extend this methodology to encompass the time series data structure so that performance differences may be assessed as a function of generation in iterative stochastic optimization algorithms. The proposed methodology establishes an experimental framework that collects performance data through statistical sampling and analyzes these data using multiple hypothesis testing. In doing so, we seek to identify the generational intervals in which two candidate algorithms significantly differ in performance.

Because Wolpert and Macready [1997] have shown that no single optimization algorithm can best solve all problems, we typically select among a number of candidate algorithms in particular settings based upon the available data. Because a candidate procedure's performance curve is typically unknown, we may estimate it through statistical sampling and assess performance differences among candidate algorithms using a statistical hypothesis test. Shilane et al. [2006] establish a general procedure for performance comparison of stochastic optimization algorithms seeking to solve a particular problem when run for the same number of generations. Within this framework, statistical sampling is used to collect performance data for each algorithm, and a multiple hypothesis testing procedure [Dudoit and van der Laan, 2006, Dudoit et al., 2004] based on bootstrap resampling [Efron and Tibshirani, 1994] of the data is used to identify significant performance differences. This approach allows the user to compare a single result from two algorithms for general data generating distributions [Pollard et al., 2005a, Pollard and van der Laan, 2004] and performance measures. This paper seeks to adapt the procedure of Shilane et al. [2006] to the time series data structure in order to compare algorithms in terms of their estimated performance curves.

In addition, we present a data reduction technique that estimates the test results in a more computationally feasible manner. The proposed methodology offers a convenient approach to evaluate competing stochastic optimization algorithms based upon empirical data. The procedure is applicable to general performance curves and data generating distributions under minimal assumptions and may be applied to arbitrary sets of stochastic algorithms in any optimization setting. We also provide a case study that seeks to compare the mean performance of four candidate evolutionary algorithms seeking to solve an example of Ackley's function.

2 Time Series Data

An iterative optimization algorithm, which we index by a, produces at each generation an estimate of a function's global optimum. Suppose we wish to study a function $f : \mathbb{R}^D \to \mathbb{R}$ with $D \in \mathbb{Z}^+$. If we allow algorithm a to run for $G \in \mathbb{Z}^+$ generations, then at each generation $g \in \{1, \ldots, G\}$, the algorithm produces a *point estimate* $X_{ag} = (X_{ag1}, \ldots, X_{agD})$ of the global optimum and a corresponding *fitness* value $f(X_{ag})$. For algorithms that employ an elitist selection mechanism, the algorithm's iterative estimate of the function f's global optimum at generation $g + 1$ fundamentally depends upon that obtained at generation g; indeed, if the optimization procedure cannot improve upon the previous estimate, both quantities are the same. Because each generational result depends upon the previous generation's estimate, the fitness values may be viewed as a *time series* data structure $Y_a = (Y_{a1}, \ldots, Y_{aG}) = (f(X_{a1}), \ldots, f(X_{aG}))$.

The general framework of Shilane et al. [2006] establishes a methodology for performance comparison that may consider either an algorithm's relative improvement of the fitness function given an initial candidate solution or the absolute fitness obtained from sampling an initial value on each trial. In the latter setting, a performance comparison of two algorithms at a single generation is determined in the test of a single hypothesis, and therefore we may simultaneously test hypotheses at each of G generations using a multiple hypothesis testing procedure. We will adopt this convention for the remainder of this study; however, the following procedure may be easily adapted to compare relative improvement by testing a hypothesis for each choice of the candidate initial value at each generation.

Suppose we wish to study algorithms in an index set A. Because randomized optimization algorithms follow a stochastic process, we seek to estimate the performance curve for each algorithm $a \in A$ based on sampled data. In order to do so, the researcher must establish a *performance curve* $\mu(Y_a)$ as a measure of the algorithm's quality at each generation. In practice, the user selects a G-dimensional parameter of the algorithm's data generating distribution. The resulting performance curve may be considered the algorithm's *parameter of interest.* One such choice for the performance curve is the G-dimensional vector-wise expected (mean) value of the algorithm's estimate of the global optimum as a function of generation:

$$\mu_a \equiv \mu(Y_a) = E[Y_a]; \qquad a \in A. \tag{1}$$

Collecting data from $n_a \in \mathbb{Z}^+$ independent, identically distributed trials of algorithm a results in n_a time series observations $Y_{ia} = (Y_{ia1}, \ldots, Y_{iaG})$, $i \in \{1, \ldots, n_a\}$, which may be stored in an $n_a \times G$ data matrix. Using the data collected, we can estimate the performance of algorithm a according to a statistic $\hat{\mu}(Y_a)$. An appropriate estimate of the parameter of interest applies the same function to the data collected that the parameter of interest applies to the data's distribution. For the parameter (1), the sample (empirical) mean is used:

$$\hat{\mu}_a \equiv \hat{\mu}(Y_a) = (\hat{\mu}(Y_{a1}), \ldots, \hat{\mu}(Y_{aG})) = \left(\frac{1}{n_a}\sum_{i=1}^{n_a} Y_{ia1}, \ldots, \frac{1}{n_a}\sum_{i=1}^{n_a} Y_{iaG}\right); \qquad a \in A. \tag{2}$$

3 Performance Comparison with Multiple Hypothesis Testing

For a given performance curve μ_a, algorithmic performance can be estimated based on the data collected according to (2), and competing algorithms may be ranked at each generation. However, it is not clear how generational results should be combined to compare algorithms in a generation-dependent manner. Depending upon the researcher's preferences, asymptotic performance may be the most meaningful measure; alternatively, one may consider one algorithm superior to another if its performance exceeds another's for at least some proportion of all generations (i.e. algorithm a_1 produces a larger estimated maximum of function f than algorithm a_2 for at least 75 percent of all generations). However, in randomized algorithms, any data-based estimate of performance $\hat{\mu}_a$ follows a stochastic process, and as such, estimated performance differences between algorithms may not reflect true differences simply as a result of random chance. To account for this possibility, we wish to perform a pairwise comparison of algorithms in a hypothesis testing framework to determine the generations at which significant performance differences exist.

As in Shilane et al. [2006], a multiple hypothesis testing procedure [Dudoit and van der Laan, 2006] is appropriate for performance comparison. For each pair of algorithms $a, b \in A$, the researcher must establish *null hypotheses* that define a difference in performance in terms of the algorithms' performance curves at each generation. Though others may be employed, a typical set of hypotheses is the equality of means (1) at each generation:

$$H : (\mu(Y_{a1}) - \mu(Y_{b1}) = 0; \ldots, \mu(Y_{aG}) - \mu(Y_{bG}) = 0) \qquad a, b \in A; \qquad a \neq b. \tag{3}$$

In order to test these hypotheses, we need to estimate the standard error of the observed performance difference at each generation, which relies upon the following estimate of the variance vector $\sigma^2(Y_a)$ for each algorithm a:

$$\hat{\sigma}^2\left(Y_a\right) = \left(\hat{\sigma}^2(Y_{a1}),\ldots,\hat{\sigma}^2(Y_{aG})\right) = \left(\frac{1}{n_a}\sum_{i=1}^{n_a}\left[Y_{ia1}-\hat{\mu}(Ya1)\right]^2,\ldots,\frac{1}{n_a}\sum_{i=1}^{n_a}\left[Y_{iaG}-\hat{\mu}(YaG)\right]^2\right); a\in A. \quad (4)$$

It should be noted that the bootstrap estimate of variance in (4) divides by the sample size n_a, whereas the traditional estimate of this quantity instead divides by $n_a - 1$ to create an unbiased estimate [Efron and Tibshirani, 1994]. The latter estimate may be substituted at the user's discretion, but these quantities differ by only a small amount for large sample sizes. Using the statistics (2) and the estimated variance (4), the hypothesis (3) may be tested using two sample t-statistics:

$$t = t(1,\ldots,G) = \left(\frac{\hat{\mu}\left(Y_{a1}\right)-\hat{\mu}\left(Y_{b1}\right)}{\sqrt{\frac{\hat{\sigma}^2(Y_{a1})}{n_a}+\frac{\hat{\sigma}^2(Y_{b1})}{n_b}}},\ldots,\frac{\hat{\mu}\left(Y_{aG}\right)-\hat{\mu}\left(Y_{bG}\right)}{\sqrt{\frac{\hat{\sigma}^2(Y_{aG})}{n_a}+\frac{\hat{\sigma}^2(Y_{bG})}{n_b}}}\right); \qquad a,b\in A, a\neq b. \quad (5)$$

Once test statistics are computed, the process of adapting the time series data structure to the general performance comparison framework is complete. The remainder of the hypothesis testing procedure is otherwise identical to that proposed in Shilane et al. [2006]. A bootstrap procedure is used to estimate the joint distribution of the test statistics (5), and a multiple testing procedure (MTP) must be selected to control a desired Type I Error Rate at level $\alpha \in (0, 1)$. It should be noted that time series data structures produce a highly dependent null hypothesis structure – indeed, for any optimization algorithm that stores its cumulatively optimal observed result, the performance metric at generation $g + 1$ must be at least as fit as that produced at generation g. As a result of the dependence structure in time series data, a joint MTP is necessary. A marginal MTP is not appropriate for performance curve comparisons because these tests assume the independence of the G hypotheses [Dudoit and van der Laan, 2006].

Although the choice of Type I Error Rate is left to the researcher, using the False Discovery Rate (FDR) may provide results that can be easily interpreted within a scientific context. The FDR Type I Error Rate is defined as the mean proportion $E[V/R]$ of false positives among the rejection set, where $V \in \mathbb{Z}^+$ is the number of *false positive* rejections and $R \in \mathbb{Z}^+$ is the number of total rejected hypotheses. By controlling an MTP at FDR level α, we can assure with probability $1-\alpha$ that the average proportion of false positives is α, which provides the user with a measure of reliability for the results. Within the context of the optimization problem, an MTP that controls the FDR at level α ensures that an average proportion of $1 - \alpha$ of the rejected hypotheses reflect true performance differences. Additionally, the FDR procedure ensures that if results were collected for a larger number of generations G^*, then we could also expect a proportion of $1-\alpha$ of the rejected hypotheses in the range $[G + 1, \ldots, G^*]$ to be reliable.

MTPs may be summarized in terms of adjusted p-values and confidence region plots. For each hypothesis, the adjusted p-value is the minimum value of α necessary to reject the hypothesis. Confidence regions depict a range of estimates for the difference in performance. At each generation, we reject the null hypothesis (3) if and only if the confidence region does not contain zero at that generation. Because confidence regions are a function of the data, they either contain or do not contain the true performance difference at each generation; however, if the comparison experiment is repeated a large number of times, a proportion of $1-\alpha$ of all confidence regions produced would contain the true performance difference curve for the two algorithms compared.

Estimated confidence regions are currently available for bootstrap-based MTPs controlling the Family-Wise Error Rate (FWER), which is defined as $P(V > 0)$, and the generalized Family-Wise Error Rate (gFWER) $P(V > k), k \in \mathbb{Z}^+$ [van der Laan et al., 2004]. However, deriving an estimate of FDR confidence regions or mapping from gFWER confidence regions to the FDR Type I Error Rate is currently an active area of research.

Although the bootstrap is an effective tool that allows one to estimate the distribution of effectively any parameter without relying upon assumptions, this technique is computationally intensive and may require many resamplings (i.e. $B \geq 10000$) to produce accurate results in hypothesis testing applications. Analytical techniques may be employed when one is able to validate distributional assumptions about the data (such as the Normal distribution). However, as a result of the inherent dependence of time series data structures, we propose limiting the testing methodology to data collected at regular generational intervals. Although less comprehensive than a test encompassing data collected at every generation, the loss of accuracy may be small in comparison with the computational savings. In this setting, reduced data confidence regions would be constructed only at the intervals under study, and the confidence region for the remainder of the generations may be interpolated using a smoother.

In the case of a performance curve comparison based on interval sampling, the main question of interest is how large to set the interval size h. This selection may be performed qualitatively based upon a graphical analysis, as presented in Section 4, chosen according to the Nyquist-Shannon sampling theorem [Nyquist, 1928], or selected to satisfy heuristic criteria. In practice, the researcher may choose among candidate values of h in terms of the relative improvement. For instance, if testing were conducted at every 50th generation and also exclusively at every 100th generation, then we could evaluate the efficacy of the larger interval size in terms of its ability to interpolate the confidence regions at every 50th generation according to a distance metric such as the mean squared error.

4 Example: A Performance Curve Comparison of Competing Optimization Procedures

In order to illustrate the proposed methodology, we will analyze performance curve data from several competing *evolutionary algorithms* (EAs) [Fogel, 2005] that seek to minimize the following variant of Ackley's function [Bäck, 1996]:

$$Y = -c_1 * exp\left(-c2\sqrt{\frac{1}{D}\sum_{d=1}^{D} X_d^2}\right) - exp\left(\frac{1}{D}\sum_{d=1}^{D} cos\left(c_3 * X_d\right)\right) + c_1 + exp(1). \tag{6}$$

The following parameters supplied for this example:

$$c_1 = 20;\ c_2 = 0.2;\ c_3 = 2\pi;\ D = 10;\ X_d \in (-20, 20),\ d \in \{1, \ldots, D\}.$$

The candidate EA described in Shilane et al. [2006] was applied to this problem in a study to select among four candidate mutation rates. The four corresponding algorithms will be indexed by the set $A = \{2, 4, 6, 8\}$, which respectively denote the gene-wise mutation rate of each EA expressed as a percentage. These EAs were identical in all other aspects. We will employ the expected value $\mu(Y_a)$ given by (1) as our performance curve.

A total of $n_a = 100$ trials of each algorithm $a \in A$ were conducted to collect time series data and estimate the performance curve $\mu(Y_a)$ (1) by the sample mean $\hat{\mu}(Y_a)$ given by (2). Each trial was conducted for a total of $G = 10000$ generations with data recorded at each generation. The sample mean performance curve for each algorithm is plotted in Figure 1. On average, it appears that EA 4 best minimizes the Ackley function (6) for approximately the first 2000 generations, and it is thereafter eclipsed by EA 6, which appears to outperform all other procedures for the duration of the trials. In order to substantiate the validity of these claims based upon both the empirical mean and sample variance of the performance of each algorithm, we will conduct pairwise comparisons of the algorithms via the multiple testing procedure of Section 3.

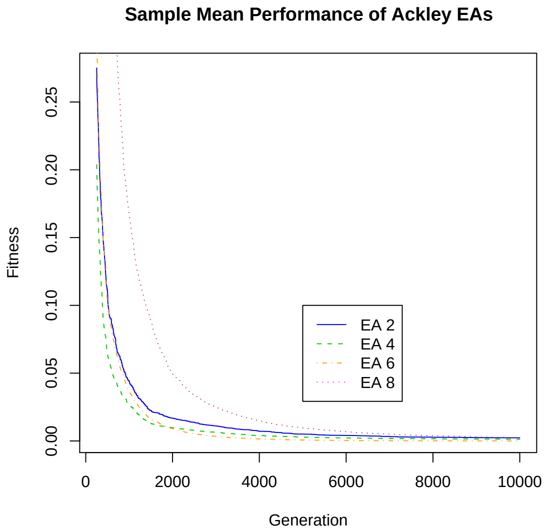

Figure 1: Sample mean fitness as a function of generation in 4 EAs seeking to optimize the Ackley function (6). Each plot is an estimate of the EA's expected value performance curve at each generation based upon $n_a = 100$ trials.

We tested each pair of algorithms $a, b \in A$; $a \neq b$ for a difference in mean performance at each generation. The null hypothesis (3) states that there exists no difference in expected performance between EAs a and b at each generation. We tested this null hypothesis using the boostrap based SSMaxT [Dudoit and van der Laan, 2006] FWER-controlling MTP and also using the FDR Conservative MTP, both of which controlled their respective Type I Error Rates at level $\alpha = 0.05$ with $B = 10000$ bootstrap re-samplings. The results of these tests are contained in Table 1:

The **Rejections** column of Table 1 shows the number of rejected hypotheses in the pairwise test. Because the null hypothesis (3) has a two-sided alternative, a rejection may correspond to a significant performance difference in either direction and may be determined by examining the sample mean performance curve plot of Figure 1. The pairwise FDR Conservative tests of $\mu_2 - \mu_4$, $\mu_4 - \mu_8$, and $\mu_6 - \mu_8$ all result in significant performance differences at all $G = 10000$ generations. As shown in Figure 1, EA 4 appears to better optimize (6) compared with EAs 2 and 8, and because all G hypotheses were rejected, we can conclude that EA 4 performs significantly better than EAs 2 and 8 for the duration of this experiment. Likewise, we can also conclude based on the data collected that EA 6 significantly outperforms EA 8 across all generations studied. However, one cannot extrapolate from these limited results the asymptotic properties of the algorithms in question; the best algorithm over the first G generations may yield to one of the other three if the experiment is extended to a larger number of generations G^*.

For each pairwise comparison, the maximum insignificant generation (**Max Insig. Gen.**) column of Table 1 indicates that last generation at which the null hypothesis is not rejected. Using these values and the mean performance plot in Figure 1, we can draw conclusions about the range at which particular algorithms outperform others. For instance, EA 2 produces a smaller empirical mean performance than that of EA 8 over much of the observed spectrum, which results in 8159 rejections for the FWER SSMaxT test. However, the null hypothesis is not rejected at the 10000th generation, meaning that EA 8's performance improves sufficiently to conclude that no significant performance difference exists between EAs 2 and 8 at the final generation. Moreover, we see that EA 6 creates significant separation in performance from EA 2 for all gen-

Test	**TI Error**	**MTP**	**Rejections**	**Max Insig. Gen.**	**Max Adjp**	**Preferred EA**
$\mu_2 - \mu_4$	FWER	SSMaxT	9952	261	-	4
$\mu_2 - \mu_4$	FDR	Conservative	10000	0	0.0256	4
$\mu_2 - \mu_6$	FWER	SSMaxT	8707	1394	-	6
$\mu_2 - \mu_6$	FDR	Conservative	8824	1243	-	6
$\mu_2 - \mu_8$	FWER	SSMaxT	8159	10000	-	2
$\mu_2 - \mu_8$	FDR	Conservative	8272	10000	-	2
$\mu_4 - \mu_6$	FWER	SSMaxT	8960	2285	-	6
$\mu_4 - \mu_6$	FDR	Conservative	9004	2285	-	6
$\mu_4 - \mu_8$	FWER	SSMaxT	9984	29	-	4
$\mu_4 - \mu_8$	FDR	Conservative	10000	0	0.005	4
$\mu_6 - \mu_8$	FWER	SSMaxT	9989	22	-	6
$\mu_6 - \mu_8$	FDR	Conservative	10000	0	0.005	6

Table 1: Summary results for pairwise performance curve comparisons of four EAs seeking to optimize the Ackley function (6) based upon multiple hypothesis testing of the null hypothesis (3) at each generation $g \in \{1, \ldots, G\}$.

erations after the 1394th generation in the FWER SSMaxT test, and likewise EA 6 significantly outperforms EA 4 at all generations after 2285.

For the three pairwise performance comparisons that resulted in rejections of all G null hypotheses, we have displayed the maximum adjusted p-value (**Max Adjp**) in Table 1 of the G simultaneous tests. This value provides the minimum possible level of α at which all hypotheses would be rejected. In the case of the FDR Conservative tests of $\mu_4 - \mu_8$ and $\mu_6 - \mu_8$, this minimum value of α is 0.005, meaning we could reduce α by a factor of 10 without changing the conclusion drawn.

Finally, the **Preferred EA** column of Table 1 displays a qualitative overall judgment of the preferred algorithm. As a heuristic standard, we choose to prefer an algorithm if it performs significantly better than another for at least 75 percent of the generations sampled. For all comparisons except that of EAs 2 and 8, this includes the final generation considered. Although EAs 2 and 8 do not differ significantly at the 10000th generation, EA 2 does significantly outperform EA 8 for more than 8000 generations. The comparisons of EAs 2 and 4 to EA 6 are also of interest. In both tests, EA 6 performs significantly worse than the others at early generations but later overtakes both algorithms. In each of these comparisons, EA 6 significantly outperforms the competing algorithm for the duration of the final 7500 generations.

These observations are further substantiated in Figures 2–13, which display FWER $1 - \alpha$ confidence regions for each of the pairwise SSMaxT tests at level $\alpha = 0.05$. In each of the comparisons, the null hypothesis is rejected at generation g if and only if the confidence region does not contain the value zero at that generation. When significant performance differences exist, the confidence region will lie below zero if the first algorithm better minimizes (6), and this region will lie above zero if the second algorithm performs significantly better. (For a maximization problem, the situation is reversed.) Each test result is depicted in two plots: the first depicts the confidence region at all generations, and the second provides a magnified view that restricts attention to generations greater than 1000.

Figures 2–13 also contain estimated confidence regions produced by restricting comparison to a subset of the data collected using the data reduction technique suggested at the end of Section 3. A total of 100 hypotheses were tested using the performance data gathered from every 100th generation. In this case, the choice of $h = 100$ for the interval size was selected. The entirety of the confidence region was estimated using a linear interpolation of the upper and lower limits at the missing generational values. Qualitatively, the

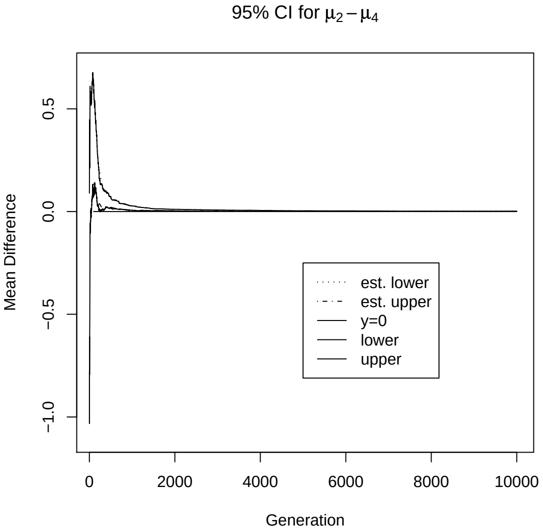

Figure 2: 0.95 confidence region for the test of $\mu_2 - \mu_4$ over the full 10000 generation interval. For greater magnification, please refer to Figure 3.

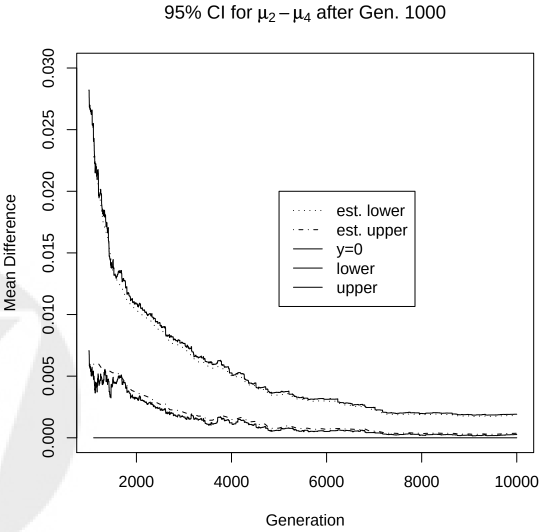

Figure 3: 0.95 confidence region for the test of $\mu_2 - \mu_4$ after generation 1000. For the confidence region over the full generational interval, please see Figure 2.

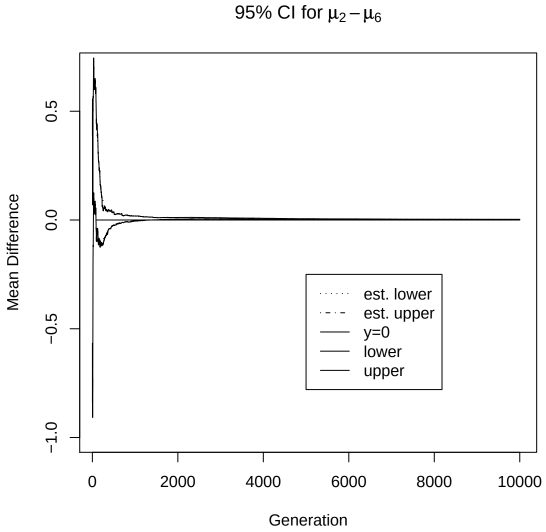

Figure 4: 0.95 confidence region for the test of $\mu_2 - \mu_6$ over the full 10000 generation interval. For greater magnification, please refer to Figure 5.

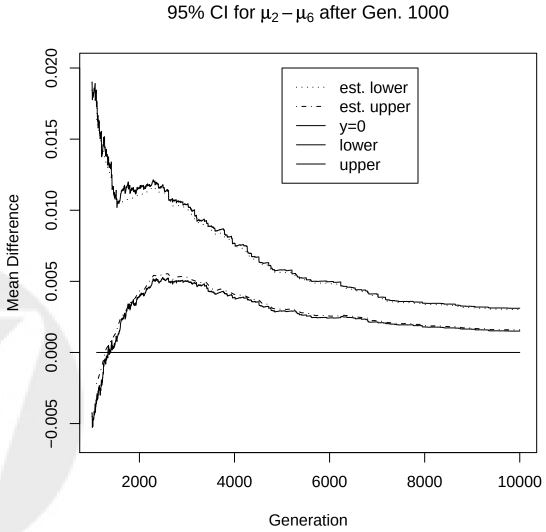

Figure 5: 0.95 confidence region for the test of $\mu_2 - \mu_6$ after generation 1000. For the confidence region over the full generational interval, please see Figure 4.

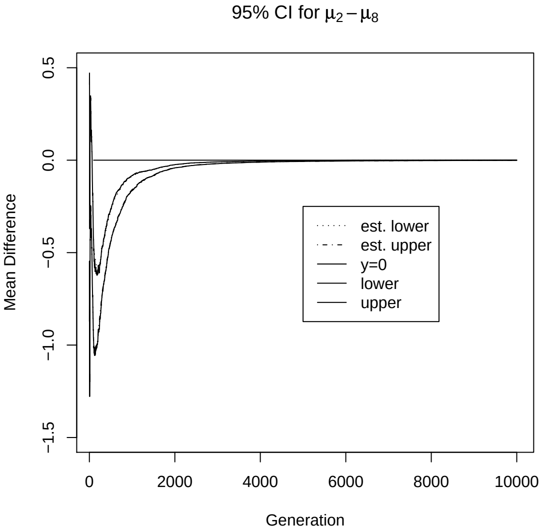

Figure 6: 0.95 confidence region for the test of $\mu_2 - \mu_8$ over the full 10000 generation interval. For greater magnification, please refer to Figure 7.

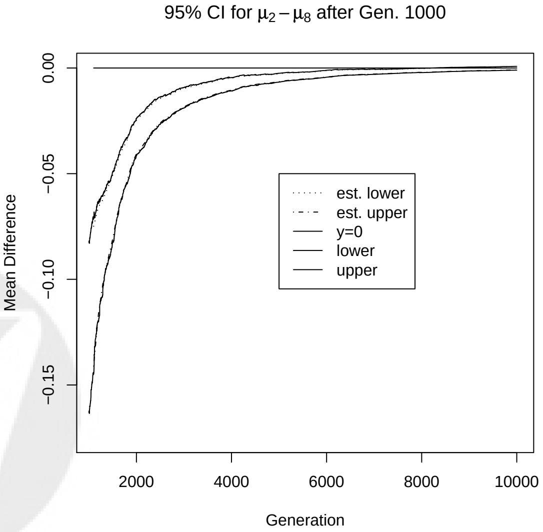

Figure 7: 0.95 confidence region for the test of $\mu_2 - \mu_8$ after generation 1000. For the confidence region over the full generational interval, please see Figure 6.

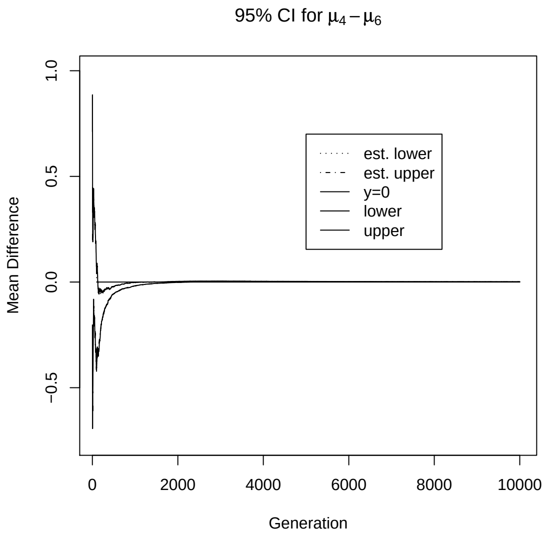

Figure 8: 0.95 confidence region for the test of $\mu_4 - \mu_6$ over the full 10000 generation interval. For greater magnification, please refer to Figure 9.

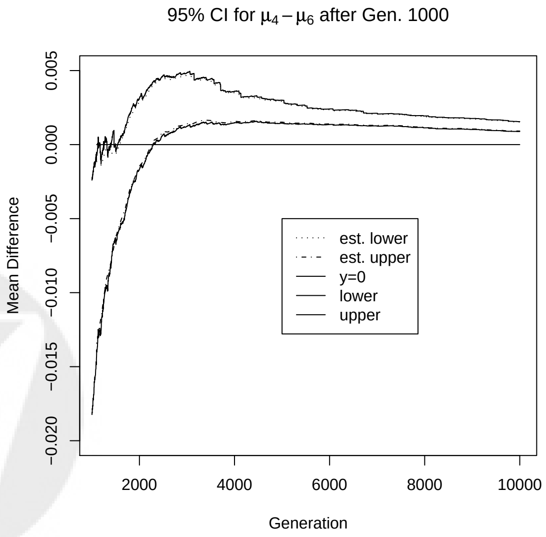

Figure 9: 0.95 confidence region for the test of $\mu_4 - \mu_6$ after generation 1000. For the confidence region over the full generational interval, please see Figure 8.

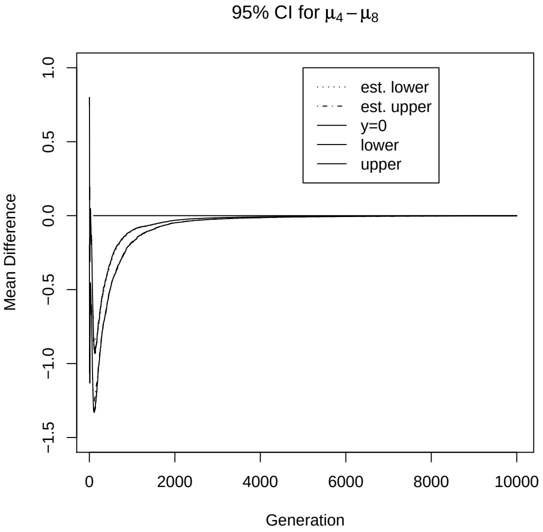

Figure 10: 0.95 confidence region for the test of $\mu_4 - \mu_8$ over the full 10000 generation interval. For greater magnification, please refer to Figure 11.

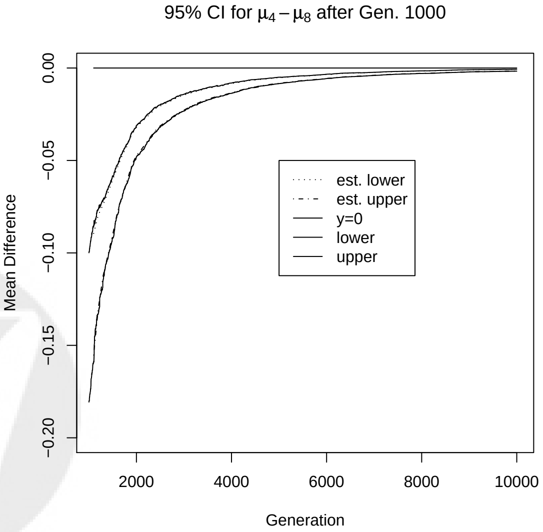

Figure 11: 0.95 confidence region for the test of $\mu_4 - \mu_8$ after generation 1000. For the confidence region over the full generational interval, please see Figure 10.

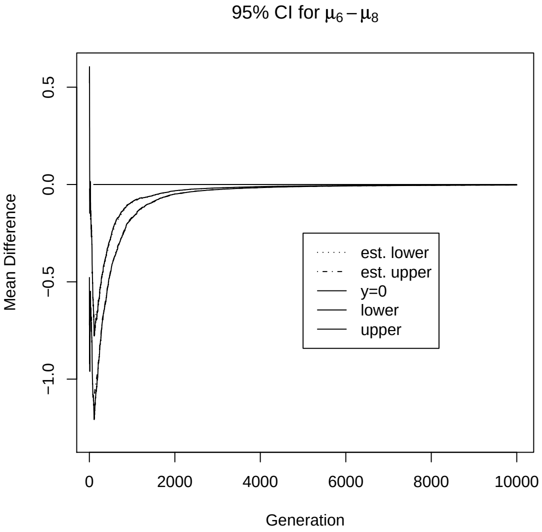

Figure 12: 0.95 confidence region for the test of $\mu_6 - \mu_8$ over the full 10000 generation interval. For greater magnification, please refer to Figure 13.

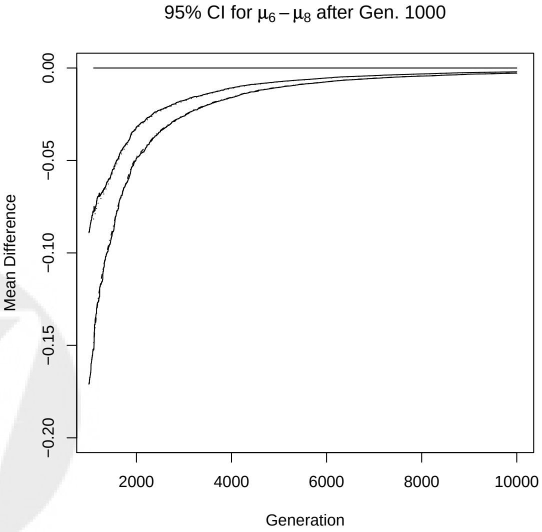

Figure 13: 0.95 confidence region for the test of $\mu_6 - \mu_8$ after generation 1000. For the confidence region over the full generational interval, please see Figure 12.

estimated confidence regions appear to approximate the full data regions reasonably well. Furthermore, the computational time required to perform all six pairwise tests of the 4 algorithms studied was reduced from approximately two days to 15 minutes using **multtest** package [Pollard et al., 2005b] of the R statistical programming language. In this particular application, it is reasonable to conclude that the computational savings justifies the relatively small loss in accuracy of the confidence region plots. For comparison purposes, a numeric summary analogous to Table 1 of the reduced data test results is contained in Table 2.

Test	**TI Error**	**MTP**	**Rejections**	**Max Insig. Gen.**	**Max Adjp**	**Preferred EA**
$\mu_2 - \mu_4$	FWER	SSMaxT	100	0	0.0022	4
$\mu_2 - \mu_4$	FDR	Conservative	100	0	0.0044	4
$\mu_2 - \mu_6$	FWER	SSMaxT	88	1200	-	6
$\mu_2 - \mu_6$	FDR	Conservative	89	1100	-	6
$\mu_2 - \mu_8$	FWER	SSMaxT	83	10000	-	2
$\mu_2 - \mu_8$	FDR	Conservative	84	10000	-	2
$\mu_4 - \mu_6$	FWER	SSMaxT	92	2200	-	6
$\mu_4 - \mu_6$	FDR	Conservative	93	2200	-	6
$\mu_4 - \mu_8$	FWER	SSMaxT	100	0	0	4
$\mu_4 - \mu_8$	FDR	Conservative	100	0	0	4
$\mu_6 - \mu_8$	FWER	SSMaxT	100	0	0	6
$\mu_6 - \mu_8$	FDR	Conservative	100	0	0	6

Table 2: Summary results for pairwise performance comparison tests of reduced data collected at every 100th generation. A total of 100 hypotheses were tested.

The results of the pairwise comparisons suggest clear performance differences between the EAs over many of the $G = 10000$ generations studied. EA 4 was shown to significantly outperform EAs 2 and 8 at all or nearly all generations, depending upon the choice of Type I error rate. EA 4 initially outperforms EA 6 by a significant margin but is eventually overtaken. Table 3 displays conclusions drawn from a closer inspection of the FDR comparison between EAs 4 and 6 at a variety of generational intervals. After frequent lead changes in the first 68 generations, EA 4 outperformed EA 6 through generation 1862. However, this performance difference became insignificant at generation 1548. Likewise, EA 6 insignificantly outperformed EA 4 from generation 1863 to generation 2285, and the results were significant for the remainder of the experiment. We may therefore conclude on the whole that EA 4 is the preferred algorithm for most of the first 1547 generations and was weakly preferred until generation 1862, whereas EA 6 was weakly preferred from generations 1863 to 2285 and significantly outperforms all other candidates in generations 2286 to 10000.

5 Discussion

The rate of convergence is a primary consideration in the design of effective optimization algorithms. In studying stochastic procedures, a probabilistic analysis is necessary to ascertain the underlying properties of these algorithms. However, even relatively simple stochastic procedures may involve complex data generating distributions when iterated over a large number of generations. By contrast, a statistical analysis of empirical data produced from the algorithms in question is relatively straightforward. The proposed methodology allows the researcher to choose how an algorithm's performance is measured, this performance curve may be studied in terms of its rate of convergence to the global optimum as a function of the generation. The time series data collected through statistical sampling provides an intuitive estimate of the algorithm's performance curve, and competing procedures may be compared at each generation using multiple hypothesis testing. The resulting confidence regions and adjusted p-values may be studied to determine whether observed performance differences are significant in specific generational intervals. This information may be

Generation Interval	Preferred EA	Difference
1 − 68	Mixed	Insignificant
69 − 127	4	Insignificant
128 − 1128	4	Significant
1129 − 1466	4	Mixed
1467 − 1547	4	Significant
1548 − 1862	4	Insignificant
1863 − 2285	6	Insignificant
2286 − 10000	6	Significant

Table 3: A comparison of EAs 4 and 6 at a variety of generational intervals in the Ackley function case study. The preferred EA is selected by mean performance at each generation, and this performance is classified as either significant or insignificant based upon the results of the FDR multiple hypothesis test of equality in means. Both EAs were competitive in the first 68 generations with frequent lead changes. In the interval from generation 1, 129 to generation 1, 466, a total of 130 test results were insignificant. All other generation ranges are homogeneous in their respective conclusions.

used to study existing optimization strategies and explore the effects of varying their components (e.g. the mutation frequency) in controlled scientific experiments.

The proposed methodology offers a convenient and flexible framework for evaluating algorithmic performance and designing improved strategies for the problem at hand. The researcher may choose any desired performance curve and does not need to rely upon distributional assumptions for the data collected. These techniques may be applied to compare arbitrary sets of stochastic algorithms in any optimization setting. Random algorithms may be compared to deterministic functions with only small changes to the hypothesis structure 3 and test statistics 5. Furthermore, the proposed data reduction technique provides an avenue for researchers to estimate the results of a full generational analysis in a more computationally tractable manner. The use of a tournament selection technique may also lead to additional computational savings in examples such as the case study of Section 4: because EA 4 significantly outperformed EAs 2 and 8 at all generations for the FDR test, the comparison of these inferior procedures to EA 6 could have been foregone completely.

Performance comparison is largely a retrospective procedure for validating experimental results. As such, it is not designed to seek out the best candidate optimum for the problem at hand; indeed, running any one of the four candidates of Section 4 for all the computational time allotted to our comparison would certainly have produced a better result than any obtained in our study. However, statistical performance comparison may be especially helpful in optimization applications that are sufficiently similar to well-studied examples. In this case, the lessons learned from performance comparison may be used to design general procedures that may be useful in a variety of settings.

As with any analysis that makes use of hypothesis testing, the proposed methodology assumes that a sufficiently large sample of data is collected to discern the performance differences between the algorithms studied. However, the selection of this sample size to ensure a minimum threshold of statistical power is currently an open problem in the multiple testing literature for general settings with unknown data generating distributions. Therefore, the researcher should collect as much data as time constraints allow.

Finally, some care should be taken to ensure that the definition of a generation is relatively consistent in all algorithms compared. One possible approach is to collect performance data at regular time intervals instead of at each generation, which may be of greater interest in comparing algorithms of differing population sizes or computational complexity within a single generation cycle.

Acknowledgment
The authors wish to thank Dr. Sandrine Dudoit for her helpful suggestions.

References

T. Bäck. *Evolutionary Algorithms in Theory and Practice.* Oxford University Press, 1996.

S. Dudoit and M. J. van der Laan. *Multiple Testing Procedures and Applications to Genomics.* Springer, 2006. (In preparation).

S. Dudoit, M. J. van der Laan, and K. S. Pollard. Multiple testing. Part I. Single-step procedures for control of general Type I error rates. *Statistical Applications in Genetics and Molecular Biology*, 3(1):Article 13, 2004. URL `www.bepress.com/sagmb/vol3/iss1/art13`.

B. Efron and R. Tibshirani. *An Introduction to the Bootstrap.* Chapman and Hall, 1994.

D. B. Fogel. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence.* IEEE Press, 2005.

H. Nyquist. Certain topics in telegraph transmission theory. *Trans. AIEE*, 47(1):617–644, 1928.

K. S. Pollard and M. J. van der Laan. Choice of a null distribution in resampling-based multiple testing. *Journal of Statistical Planning and Inference*, 125(1–2):85–100, 2004.

K. S. Pollard, M. D. Birkner, M. J. van der Laan, and S. Dudoit. Test statistics null distributions in multiple testing: Simulation studies and applications to genomics. *Journal de la Société Française de Statistique*, 146(1–2):77–115, 2005a. URL `www.stat.berkeley.edu/ sandrine/Docs/Papers/SFdS05/SFdS.html`. Numéro double spécial *Statistique et Biopuces.*

K. S. Pollard, S. Dudoit, and M. J. van der Laan. *Bioinformatics and Computational Biology Solutions Using R and Bioconductor*, chapter 15: Multiple Testing Procedures: the multtest Package and Applications to Genomics, pages 249–271. Statistics for Biology and Health. Springer-Verlag, New York, 2005b. URL `www.bepress.com/ucbbiostat/paper164`.

D. Shilane, J. Martikainen, S. Dudoit, and S. Ovaska. A general framework for statistical performance comparison of evolutionary computation algorithms. 2006. URL `www.bepress.com/ucbbiostat/paper204`.

M. J. van der Laan, S. Dudoit, and K. S. Pollard. Augmentation procedures for control of the generalized family-wise error rate and tail probabilities for the proportion of false positives. *Statistical Applications in Genetics and Molecular Biology*, 3(1):Article 15, 2004. URL `www.bepress.com/sagmb/vol3/iss1/art15`.

D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.