How Are We Doing? Predicting Evolutionary Algorithm Performance

Mark A. Renslow[1], Brenda Hinkemeyer[2], and Bryant A. Julstrom[3]

Department of Computer Science
St. Cloud State University, St. Cloud, MN 56301 USA
[1]markminn@mac.com
[2]brenda@nikosha.net
[3]julstrom@eeyore.stcloudstate.edu

Abstract. Given an evolutionary algorithm for a problem and an instance of the problem, the results of several trials of the EA on the instance constitute a sample from the distribution of all possible results of the EA on the instance. From this sample, we can estimate, non-parametrically or parametrically, the probability that another run of the EA, independent of the initial ones, will identify a better solution to the instance than any seen in the initial trials. We derive such probability estimates and test the derivations using a genetic algorithm for the traveling salesman problem. We find that while the analysis holds promise, it should probably not depend on the assumption that the distribution of an EA's results is normal.

1 Introduction

Evolutionary algorithms are applied to instances of computationally difficult optimization problems for which optimum solutions are not known. Because EAs are probabilistic, we cannot in general know how good the solution returned by any one run might be, and we often carry out multiple trials of an EA on an instance.

The results of such trials constitute a sample from the distribution of all possible results of the EA on the target instance. We can use this sample to determine whether subsequent runs might or might not be worth doing. Specifically, we can estimate, from an initial set of trials, the probability of identifying a better solution on the next run, or in the next fifty runs. Such information would be useful, for instance, when deciding whether to carry out more trials of a computationally expensive EA.

We consider, then, the following problem. Given an evolutionary algorithm for a problem, an instance of the problem, and the results of an initial set of trials of the EA on the instance, what is the probability that one additional trial, independent of those already completed, will return a solution better than any observed in the initial trials?

A non-parametric analysis identifies a probability that depends only on the number of initial trials. A parametric analysis assumes that the values the EA re-

K. Deb et al. (Eds.): GECCO 2004, LNCS 3103, pp. 82–89, 2004.

turns conform to a particular distribution—here, normal—and identifies a probability that depends on that assumption and on the mean and standard deviation of the initial results. The two analyses are tested with a simple genetic algorithm on five instances of the well-known traveling salesman problem. We find that neither analysis is as accurate as we might like, and the second is flawed by the assumption of normality, but the method itself holds promise.

The following sections of the paper define the problem precisely, present non-parametric and parametric derivations of the probability that another trial identifies a better solution, and describe the tests of both analyses using a genetic algorithm for the traveling salesman problem.

2 The Problem

Let Q be an optimization problem, like the traveling salesman problem or the 0-1 knapsack problem, and let E be an evolutionary algorithm for Q. The fitness of a genotype in E is the objective function value of the solution the genotype represents. When E is applied to an instance of Q, it reports, on its termination, the single best solution represented in its population.

Without loss of generality, assume that Q seeks to minimize its objective function, as in the TSP, and let Q_o be a particular instance of Q. Assume that Q_o is difficult enough for E that E is unlikely to return an optimum solution to Q_o. Since E is probabilistic, repeated independent trials of it on Q_o will return solutions with a variety of fitnesses.

Let the random variable X be the fitness of the solution E returns at the conclusion of one trial on Q_o. X has some (unknown) distribution. Given n probes into this distribution—that is, given the results of n independent trials of E on Q_o—what can we say about the distribution of X? In particular, how likely is it that another trial will improve on the best result observed in the initial trials?

Note that the problem just posed is a special case of a more general one: Given a sample from an unknown distribution, what can we say about that distribution and probabilities based on it?

3 A Non-parametric Analysis

Consider a sequence of n probes into the unknown distribution of X; that is, a sequence of n independent runs of the evolutionary algorithm E on the problem instance Q_o. The trials will return values $X_1, X_2, \ldots, X_n$, the fitnesses of the n best solutions to Q_o that E discovers.

Assume that the probability that two trials return identical results is negligible. Then the probability that the second trial will return a solution with smaller evaluation than the first is $P[X_2 < X_1] = 1/2$; the probability that the third trial will return a better solution than the first two is $P[X_3 < X_1, X_2] = 1/3$;

and so on. In general, the probability that the last of k trials will return the solution with the smallest evaluation is

$$P[X_k < X_1, X_2, \ldots, X_{k-1}] = \frac{1}{k}. \tag{1}$$

When n trials have been completed and the values $x_1, x_2, \ldots, x_n$ observed, it is no longer strictly accurate to say that the probability that the next trial will improve on its predecessors is $1/(n+1)$, since this probability now depends on the recorded values and on the distribution of X. However, that distribution is unknown, so we will reckon the probability that the next trial returns a better result than its predecessors as $p = 1/(n+1)$. More generally, if we carry out m additional trials, at least one of them will improve on the first n trials if it is not the case that none do. The probability that one additional trial fails to return a better result is $1 - p = n/(n+1)$, so the probability that none of m additional trials returns a better result is $(1-p)^m = (n/(n+1))^m$, and the probability that at least one of them does improve on the best of the first n trials is then

$$1 - (1-p)^m = 1 - \left(1 - \frac{1}{n+1}\right)^m. \tag{2}$$

For example, if $n = 20$, then the probability that the twenty-first trial returns a solution with smaller evaluation than any of the first twenty is $1/21 = 0.0476$. If we carry out 25 additional trials, the probability that at least one of them returns a better solution than any in the first batch is

$$1 - \left(1 - \frac{1}{21}\right)^{25} = 0.705. \tag{3}$$

4 A Parametric Analysis

Again, let $X_1, X_2, \ldots, X_n$ be the fitnesses of the solutions returned by n independent trials of the evolutionary algorithm E on the problem instance Q_o. If we can assume a particular distribution for X, more precise predictions of the evolutionary algorithm's behavior become possible.

For example, assume that X has the normal distribution $N(\mu, \sigma^2)$ with mean μ and variance σ^2. The mean $\overline{X}$ of the n returned values is an unbiased estimator of μ, and their variance $S^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \overline{X})^2$ is an unbiased estimator of σ^2; we approximate $N(\mu, \sigma^2)$ with $N(\overline{x}, s^2)$, where $\overline{x}$ and s^2 are the mean and variance of the trials' results.

The probability that another trial will return a solution with a smaller evaluation than any of the first n trials depends on the smallest evaluation among them, on $\overline{x}$, and on $s = \sqrt{s^2}$, as Figure 1 illustrates. In particular, if $x_{\min} = \min\{x_1, x_2, \ldots, x_n\}$, then

$$p = P[X < x_{\min}] = P\left[Z < \frac{x_{\min} - \overline{x}}{s}\right] = 0.5 - \Phi\left(\frac{\overline{x} - x_{\min}}{s}\right), \tag{4}$$

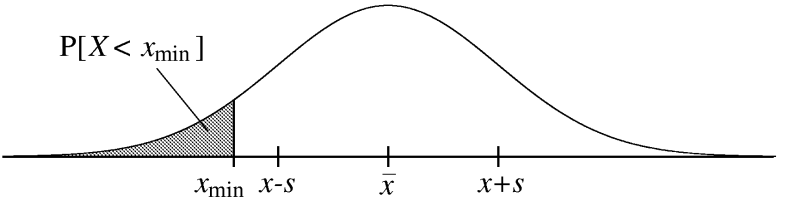

Fig. 1. Given the results of n trials with mean $\overline{x}$, standard deviation s, and smallest value $x_{\min}$, the probability $P[X < x_{\min}]$ that a new trial will return a solution with smaller evaluation than $x_{\min}$, under the assumption that X is normally distributed

where Z is the standard normal random variable with distribution $N(0,1)$ and $\Phi(z)$ is the area under Z's density function above z.

Again, the probability that at least one of m additional trials returns a solution with smaller evaluation than any among the first n is $1-(1-p)^m$. In addition, it is possible to estimate the probability that an additional trial will return a solution with evaluation smaller than an arbitrary value x_o:

$$P[X < x_o] = P\left[Z < \frac{x_o - \overline{x}}{s}\right]. \tag{5}$$

For example, if twenty trials of the evolutionary algorithm E on the problem instance Q_o have returned solutions whose evaluations have mean $\overline{x} = 455.3$, standard deviation $s = 18.6$, and minimum $x_{\min} = 426.8$, then the probability p that another trial will return a value less than $x_{\min}$ is

$$\begin{aligned} p = P[X < 426.8] &= P\left[Z < \frac{426.8 - 455.3}{18.6}\right] = P[Z < -1.53] \\ &= 0.5 - \Phi(1.53) = 0.5 - 0.437 = 0.063, \end{aligned} \tag{6}$$

and the probability that at least one of 25 additional trials will improve on $x_{\min}$ is $1-(1-0.063)^{25} = 0.803$.

Moreover, the probability that another trial will identify a solution with evaluation less than, say, 425.0 is

$$\begin{aligned} P[X < 425.0] &= P\left[Z < \frac{425.0 - 455.34}{18.6}\right] = P[Z < -1.63] \\ &= 0.5 - \Phi(1.63) = 0.5 - 0.448 = 0.052. \end{aligned} \tag{7}$$

The distribution of X, the value returned by one trial of E on Q_o, depends not only on E's coding, fitness function, and operators but also on all its features and parameter values; change any of these, and the distribution changes. Also, X's distribution is bounded below by the evaluation of an optimum solution and so is likely to be skewed rather than normal. However, an analysis like that above can be carried out with any other distribution, and a χ^2 statistic can be used to test the hypothesis that X conforms to the chosen distribution.

5 Tests

The analyses of the previous two sections were tested with a straightforward (and not particularly effective) generational genetic algorithm. The GA addressed the well-known traveling salesman problem (TSP) in which, given a collection of cities and the distances between them, we seek a tour that visits each city exactly once and has minimum total distance. The GA represents candidate tours as permutations of the cities. The order of the cities in a permutation indicates the order in which the permutation's tour visits them, and the fitness of a permutation is that tour's length, which we seek to minimize.

The GA initializes its population with random permutations, and it chooses permutations to be parents in k-tournaments. The crossover operator is edge recombination [1], and the mutation operator is subtour inversion [2, pp.219–220]. Each new permutation is generated by exactly one operator, never both. The GA is 1-elitist, preserving the best permutation from the current generation into the next, and it runs through a fixed number of generations.

In these experiments, the GA's population contained 100 permutations. The size of its selection tournaments was $k = 2$, and a tournament's winner always became a parent. The probability that it would apply crossover to generate the next new permutation was 50%, and the probability of mutation was therefore also 50%. The GA ran through 1 000 generations.

The GA was run on five TSP instances taken from TSPLIB[1] [3]. These instances contain between 52 and 280 cities. On each instance, the GA was run twenty independent times, and the mean tour length, the standard deviation of the lengths, and the shortest tour length were recorded. Based on these statistics, non-parametric and parametric (normal) probabilities that another run would identify a tour shorter than the shortest were derived according to the discussions in the previous two sections. These probabilities were tested by running the GA 2 000 more times on each instance and noting the relative frequency of the event that a trial identified a tour shorter than the initial minimum.

Table 1 summarizes the results of these trials. For each TSP instance, it lists the mean, standard deviation, and minimum of the tour lengths returned by the GA's initial twenty trials; the two estimated probabilities that another trial would identify a shorter tour; and the number of additional trials, the number of those that found a shorter tour, and the relative frequency of the shorter tours.

The non-parametric probability estimates are based only on the sizes of the initial samples. Since these sizes are all the same (20), so are the non-parametric estimates (0.0476). The parametric probability estimates assume that, as a random variable, the outcome of a run of the GA on a particular TSP instance has a normal distribution. We estimate the distribution's mean and variance with the sample mean and sample variance, so the parametric estimates depend on the mean, standard deviation, and minimum of the initial trials' results. These estimates vary from 0.0281 to 0.0582.

[1] elib.zib.de/pub/Packages/mp-testdata/tsp/tsplib/index.html

Table 1. Results of the trials of the genetic algorithm on five TSP instances. For each instance, the table lists the mean, standard deviation, and minimum of the tour lengths the GA returned in twenty initial trials; the non-parametric and parametric (normal) estimated probabilities that one more trial would identify a shorter tour; and the number of additional trials, the number that found a shorter tour, and the relative frequency of shorter tours

	Initial sample			Est. probs.		Add'l	Num	Rel.
Instance	Mean	StdDev	Min	Non-par	Normal	trials	$<$ min	freq.
berlin52	9834.8	497.80	8883	0.0476	0.0281	2000	18	0.0090
st70	1153.7	64.91	1052	0.0476	0.0582	2000	130	0.0650
eil76	890.1	38.81	828	0.0476	0.0548	2000	188	0.0940
ch130	17396.2	627.43	16216	0.0476	0.0301	2000	109	0.0545
a280	15148.1	314.99	14573	0.0476	0.0336	2000	201	0.1005

In the 2 000 additional runs of the GA on each instance, the relative frequencies of the event that a trial returns a tour shorter than the shortest among the initial runs varies from 0.0090 to 0.1005. On two instances (st70 and ch130), the non-parametric probability estimates are close to the relative frequencies, from which they differ by about 27% and 13%, respectively. On the other instances, the estimates differ from the relative frequencies by at least 47%. Similarly, on only one instance (st70) is the parametric estimated probability close to the relative frequency, differing from it by about 10%. On eil76 the difference is about 42%, and on the remaining three instances the difference is greater.

Though the non-parametric estimates are slightly more accurate than the parametric estimates, as we might expect with small sample sizes, these results make it difficult to conclude that either the non-parametric or the parametric probability analyses provide useful estimates of the probability of observing a better result on a future trial.

One explanation for the failure of the parametric analysis to accurately estimate the desired probability may lie with the size of the initial sample. Larger samples would provide more accurate estimates of the distributions' means and standard deviations and allow more accurate probability estimates.

To test this hypothesis, we used the mean and standard deviation of all 2 020 results on each instance to re-estimate the probabilities of new results smaller than the initial minima, and compared the new estimates to the relative frequencies. Table 2 shows the results of these calculations. For each instance, the table lists the mean and standard deviation of all 2 020 results, the original minimum result, the parametric probability estimate based on all the results, and the relative frequency of results less than the original minimum.

In Table 2, we see that the revised estimated probabilities are, with one exception, very close to the relative frequencies. This suggests that in general, the samples on which such estimates are based should indeed be larger, though it does not indicate how much larger might be adequate.

Table 2. The parametric estimated probabilities, based on all 2 020 trials of the GA on each instance, compared to the relative frequencies

	Entire sample		Initial	Est. par.	Rel.
Instance	Mean	StdDev	min	prob.	freq.
berlin52	9891.5	435.36	8883	0.0104	0.0090
st70	1140.2	58.58	1052	0.0655	0.0650
eil76	878.2	38.13	828	0.0934	0.0940
ch130	17362.8	734.86	15112	0.0011	0.0545
a280	15126.3	435.19	14573	0.1020	0.1005

It is also possible that the distributions of the GA's results on the instances are not normal. Certainly if the GA achieves results that are near optimal (not particularly the case here), we would expect the distribution of those results to be skewed, and if a distribution is not normal, probability estimates based on the assumption that it is are not likely to be accurate.

To examine the possibility that the underlying distributions were not normal, we performed chi-square tests of goodness-of-fit on the five sets of 2 020 results. Each tested the null hypothesis that the values conformed to a normal distribution whose mean and variance were the mean and variance of all the values. On two instances (st70 and eil76), we reject this hypothesis at the 1% significance level. On two others (berlin52 and ch130), we reject the hypothesis at the 10% level. Only for the one remaining instance (a280) can we not conclude that the 2 020 values do not conform to the specified normal distribution.

6 Conclusion

Estimates of the probability that an evolutionary algorithm will identify an improved result would clearly be useful. We have investigated two techniques for generating such estimates. One assumes nothing about the distribution of an EA's results on a particular problem instance. The other assumes that those results conform to a normal distribution. Though tests of these techniques used a simple genetic algorithm for the traveling salesman problem, nothing in them depends on a particular problem, coding, set of operators, or EA design.

In tests using the genetic algorithm and five instances of the TSP, we found that neither technique produced accurate estimates of the probability that another run of the GA would produce a better result than the best in a small initial sample. The non-parametric estimates were slightly more accurate than those based on the assumption of normality.

The parametric (normal) estimates may have failed for either of two reasons. The initial samples of the GA's performance may have been too small to accurately estimate the underlying distribution's mean and variance, and the underlying distribution may not have been normal. Still, we suggest that methods like these, based on larger initial sample sizes and assuming other underlying distributions, should be able to accurately estimate the probabilities we seek.

References

1. Whitley, D., Starkweather, T., Fuquay, D.: Scheduling problems and traveling salesmen: The genetic edge recombination operator. In Schaffer, J.D., ed.: Proceedings of the Third International Conference on Genetic Algorithms, San Mateo, CA, Morgan Kaufmann Publishers (1989) 133–140
2. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolutionary Programs. Third edn. Springer, Berlin (1996)
3. Reinelt, G.: TSPLIB - A traveling salesman problem library. ORSA Journal on Computing **3** (1991) 376–384