
Genetic Algorithms, Efficiency Enhancement, and Deciding Well with Differing Fitness Variances

Kumara Sastry and **David E. Goldberg**
Illinois Genetic Algorithms Laboratory (IlliGAL)
Department of General Engineering
University of Illinois at Urbana-Champaign
104 S. Mathews Ave, Urbana, IL 61801
{ksastry,deg}@uiuc.edu

Abstract

This study investigates the decision making between fitness function with differing variance and computational-cost values. The objective of this decision making is to provide evaluation relaxation and thus enhance the efficiency of the genetic search. A decision-making strategy has been developed to maximize speed-up using facetwise models for the convergence time and population sizing. Results indicate that using this decision making, significant speed-up can be obtained.

1 Introduction

Significant progress has been made both in analysis and design of genetic algorithms (GAs) over the last decade. Design procedures for the development of *competent* GAs have been proposed and much progress has been made along these lines (Goldberg, 1999). A GA is called *competent* if it can solve hard problems quickly, accurately, and reliably. In essence, competent GAs take problems that were intractable with the first generation GAs and render them tractable. Competent GAs successfully solve problems with *bounded difficulty* oftentimes requiring only a subquadratic (polynomial) number of function evaluations.

However, for large-scale problems, the task of computing even a subquadratic number of function evaluations can be daunting. This is especially the case if the fitness evaluation is a complex simulation, model, or computation. This places a premium on a variety of *efficiency-enhancement techniques*. Therefore GA practitioners resort to approximate fitness functions that are less expensive to compute. Such approximations introduce error in assessing the solution quality.

Usually, one has to choose among a set of fitness func-

tions with varying degrees of error. The choice of a fitness function has a large impact on the computational resources and the solution quality. Oftentimes, practitioners choose a fitness function on an ad hoc basis which might not necessarily be the correct choice. Therefore, there is a need to investigate which fitness function should be used and under what scenarios.

However, error comes in two flavors: bias and variance. Variance and bias affect the search process in different ways and therefore have to be handled in different manner (Keijzer & Babovic, 2000). This paper considers the decision making under the presence of variance alone and decision making in the presence of bias is presented elsewhere (Sastry, 2001). This separation will not only ease the analytical burden, but also highlight the difference in the decision-making procedure.

This paper investigates the decision-making process between two fitness functions with differing variance values and computational costs. Although the fitness function with low variance requires a smaller population size and converges faster, the overall computational cost can be higher due to its higher cost. On the other hand, the low-cost fitness function is cheaper to compute, but both the population size and the convergence time increase, which in turn increases the total computational cost. Therefore, one has to choose one of the two fitness functions. The objective of this study is to develop a decision-making strategy that yields maximum speed-up. Facetwise models for convergence time and population sizing are used to predict speed-up and these models are verified with empirical results along the way.

This paper is organized as follows. Section 2 briefly discusses the past work on handling error in fitness functions. The problem addressed in this paper is defined in section 3. Then, facetwise models for convergence time, population size and total number of

function evaluations are developed in the subsequent section. The strategy that yields maximum speed-up is discussed in section 5. Finally, a summary and key conclusions of this study is presented.

2 Literature Review

Efficiency-enhancement techniques are essential for solving large-scale, complex search problems. One such technique is *evaluation relaxation*. Evaluation-relaxation schemes try to reduce the computation burden by utilizing inexpensive, but error-prone fitness assignment procedures instead of an expensive, but accurate fitness function.

Grefenstette and Fitzpatrick (1985) studied the utility of approximate evaluations in an image registration problem and obtained significant speed-up by random pixel sampling instead of complete sampling. Follow-up studies (Fitzpatrick & Grefenstette, 1988; Mandava, Fitzpatrick, & Pickens, 1989) have provided further evidence of efficiency-enhancement by using approximate fitness evaluations. Early studies of approximate function evaluations were largely empirical, and a design methodology for predicting the behavior of GAs was lacking. Miller and Goldberg (1995) provided a theoretical framework for handling noisy function evaluations. Specifically they developed convergence-time models in the presence of external noise. Miller and Goldberg (1996) extended the convergence time model for different selection methods. Miller (1997) proposed a detailed design methodology including development of population-sizing model and optimal sampling prediction for noisy environments.

Other studies exist on utilizing approximate fitness functions to speed-up the genetic search (Ratle, 1998; El-Beltagy, Nair, & Keane, 1999; Jin, Olhofer, & Sendhoff, 2000; Albert, 2001). However, an exhaustive survey is beyond the scope of this study.

3 Problem Definition

Consider two noisy fitness functions f_1 and f_2 for a search problem. Functions f_1 and f_2 consist of zero-mean Gaussian noise of variance $\sigma_{N_1}^2$ and $\sigma_{N_2}^2$ respectively. The cost of a single evaluation of f_1 is c_1 and that of f_2 is c_2 . Also, $\sigma_{N_1}^2 < \sigma_{N_2}^2$, and $c_1 > c_2$. That is, f_1 is a high-cost, low-variance function, and f_2 is a low-cost, high-variance fitness function. The objective is to correctly decide which fitness function to employ so as to obtain highest speed-up. As will be seen later, this decision has to be made spatially. To achieve this goal, we first have to develop appropriate models for the convergence time and the population size required.

4 Facetwise Models

In this section, we will develop a facetwise model for convergence time of GAs in presence of external noise. Then an existing model for population sizing is presented and these models are used to compute an expression for the total number of function evaluations. Finally, these facetwise models are verified with empirical results.

4.1 Convergence Time

Understanding run duration is one of the critical factors for analyzing GAs. Elsewhere, a motivation and the utility of understanding time has been discussed Goldberg (in press). Three main approaches have been used in understanding time: (1) Modeling of takeover time, where the dynamics of the best individual is modeled (Goldberg & Deb, 1991), (2) Selection-intensity model, where the dynamics of the average fitness of the population is modeled (Mühlenbein & Schlierkamp-Voosen, 1993; Bäck, 1995; Miller & Goldberg, 1995; Miller & Goldberg, 1996), and (3) Higher-order cumulant model, where the dynamics of average and higher-order cumulants are modeled (Blickle & Thiele, 1995; Prügel-Bennet & Shapiro, 1994).

Even though higher-order cumulant models are more accurate than selection-intensity models, they do not provide a closed-form solution for either the proportion of correct building blocks or the convergence time. Therefore, in this study we develop a selection-intensity based convergence-time model for the OneMax domain. The OneMax problem has two key properties: (1) Uniform building-block salience, and (2) Gaussian fitness distribution. Uniform building-block salience implies that the contribution of building blocks in different partition to the fitness is equal. The assumption of Gaussian fitness distribution is approximately true as recombination and other genetic operators have a normalizing effect.

Therefore the fitness distribution $F = \mathcal{N}(\mu_t, \sigma_t^2)$, and $N = \mathcal{N}(0, \sigma_N^2)$. Here, μ_t is the mean true fitness at time t . Furthermore, the noisy fitness distribution, F' can be written as $F' = F + N$, where, F is the actual fitness distribution, and N is the external noise (in this case, zero-mean Gaussian noise). Since both the actual fitness and the noise are normally distributed, the noisy fitness function is also normally distributed:

$$F' \sim \mathcal{N}(\mu_t, \sigma_t^2 + \sigma_N^2). \quad (1)$$

Under these assumptions, the expected average fitness of the population after selection, given the current av-

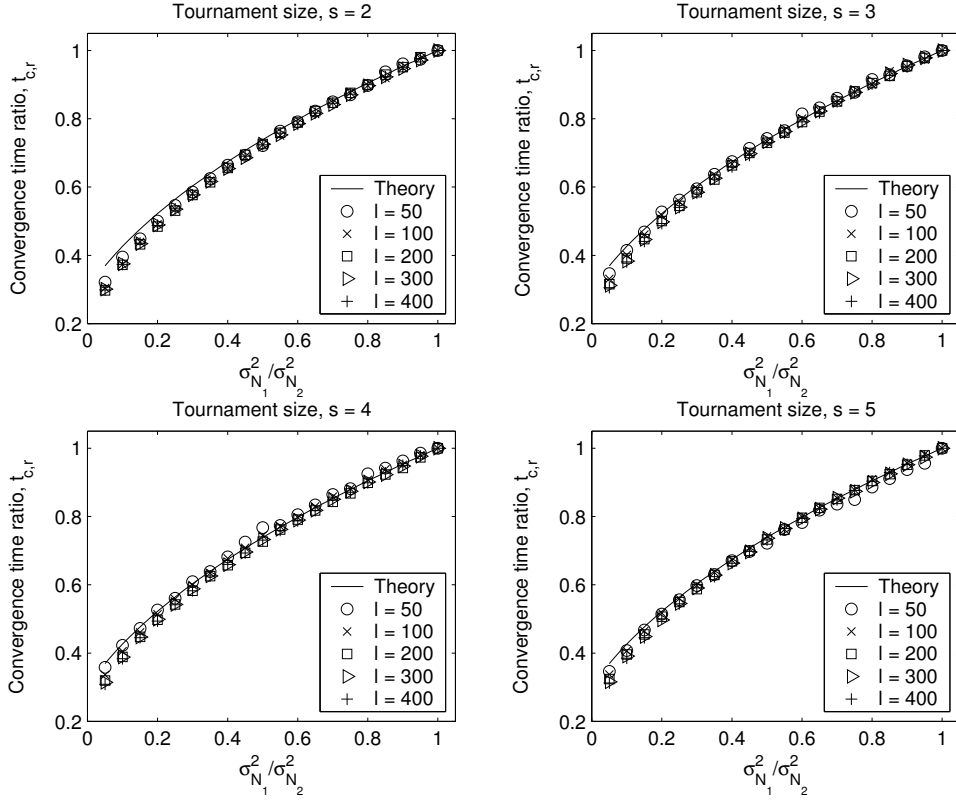


Figure 1: Empirical verification of the convergence-time-ratio model (equation 8).

erage fitness is given by (Miller & Goldberg, 1995):

$$\mu_{t+1} - \mu_t = \frac{I\sigma_t^2}{\sqrt{\sigma_t^2 + \sigma_N^2}}. \quad (2)$$

where, I is the selection intensity (Bulmer, 1985) and is defined as the expected increase in the average fitness of a population after selection is performed upon a population whose fitness is distributed according to a unit normal distribution. The selection intensity for tournament selection depends on the tournament size, s , and can be approximated by the relation (Blickle & Thiele, 1995):

$$I = \sqrt{2 \left(\ln(s) - \ln \left(\sqrt{4.14 \ln(s)} \right) \right)}. \quad (3)$$

Equation 2 can be rewritten as

$$\mu_{t+1} - \mu_t = \frac{I}{\rho_e} \sigma_t \quad (4)$$

where, $\rho_e = \sqrt{1 + (\sigma_N^2/\sigma_t^2)}$, is the *duration-elongation factor* (Goldberg, in press). Note that for a non-zero noise, $\rho_e > 1$, and the increment in the average fitness after selection would be less than that when the noise is absent. In other words, the presence of external noise,

elongates the convergence time, and this elongation is quantified by ρ_e .

Assume that ρ_e is a constant, and is equal to $\sqrt{1 + (\sigma_N^2/\sigma_f^2)}$, where σ_f^2 is the initial fitness variance. Note that for OneMax problem, $\mu_t = \ell p_t$, and $\sigma_t^2 = \ell p_t(1 - p_t)$, where p_t is the proportion of correct BBs at time t . Using these expressions, equation 4 can be written as

$$p_{t+1} - p_t = \frac{I}{\rho_e \sqrt{\ell}} \sqrt{p_t(1 - p_t)}. \quad (5)$$

Approximating the above difference equation by a differential equation, and integrating it with the initial condition, $p_0 = 0.5$ (randomly initialized population), gives us

$$p_t = \frac{1}{2} \left(1 + \sin \left(\frac{It}{\rho_e \sqrt{\ell}} \right) \right). \quad (6)$$

Equating $p_t = 1$, in the above equation we can solve for the convergence time:

$$t_{\text{conv}} = \frac{\pi \sqrt{\ell}}{2I} \sqrt{1 + \frac{\sigma_N^2}{\sigma_0^2}}. \quad (7)$$

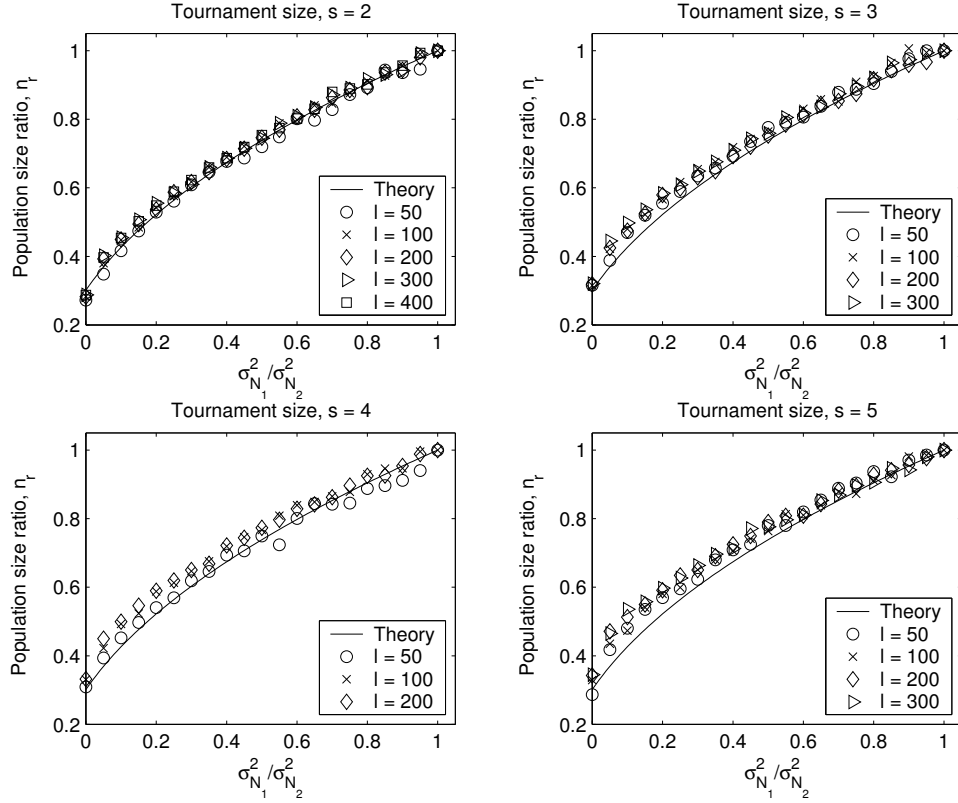


Figure 2: Empirical verification of population-size-ratio model (equation 10).

It must be noted that in deriving the above convergence-time model we assumed ρ_e to be a constant. However, ρ_e changes over time and more accurate solutions for equation 4 exist (Sastry, 2001).

In this study, we are interested in the relative value of convergence times, rather than the absolute values. Specifically, we are interested in the ratio of convergence time when fitness function f_1 is employed to that when fitness function f_2 is employed. This convergence-time ratio is given by

$$t_{c,r} = \frac{t_{\text{conv}}(\sigma_{N_1})}{t_{\text{conv}}(\sigma_{N_2})} = \left(\frac{\sigma_f^2 + \sigma_{N_1}^2}{\sigma_f^2 + \sigma_{N_2}^2} \right)^{\frac{1}{2}}. \quad (8)$$

It should be noted that using more accurate solutions for equation 4 does not improve the accuracy of theoretical model significantly

4.2 Population Size

The previous section presented a convergence-time model for tournament and other I -constant selection schemes. The other factor required to determine complexity is the population-sizing model which is presented in this section. Population size is an important

factor in determining the solution quality through a GA run. Adequate population size is required not only to ensure a good number of initial BB supply, but also a good decision-making between competing BBs.

Goldberg, Deb, and Clark (1992) proposed a practical population-sizing bounds for selectorecombinative GAs. Their model was based on deciding correctly between the best and the next best BB in a partition in the presence of noise arising from other partitions. More recently, Harik, Cantú-Paz, Goldberg, and Miller (1997) refined the population-sizing model of Goldberg et al. (1992) to compute a tighter bound on the population size. They incorporated both the initial BB supply model and the decision-making model in the population-sizing relation. Miller (1997) extended the population-sizing model of Harik et al. (1997) for noisy environments.

The following population-sizing model for noisy environments developed by Miller (1997) is used in the current study:

$$n = -\frac{\sqrt{\pi}}{2d} \chi^k \log(\alpha) \sqrt{\sigma_f^2 + \sigma_N^2}, \quad (9)$$

where, d is the signal difference and is given by the

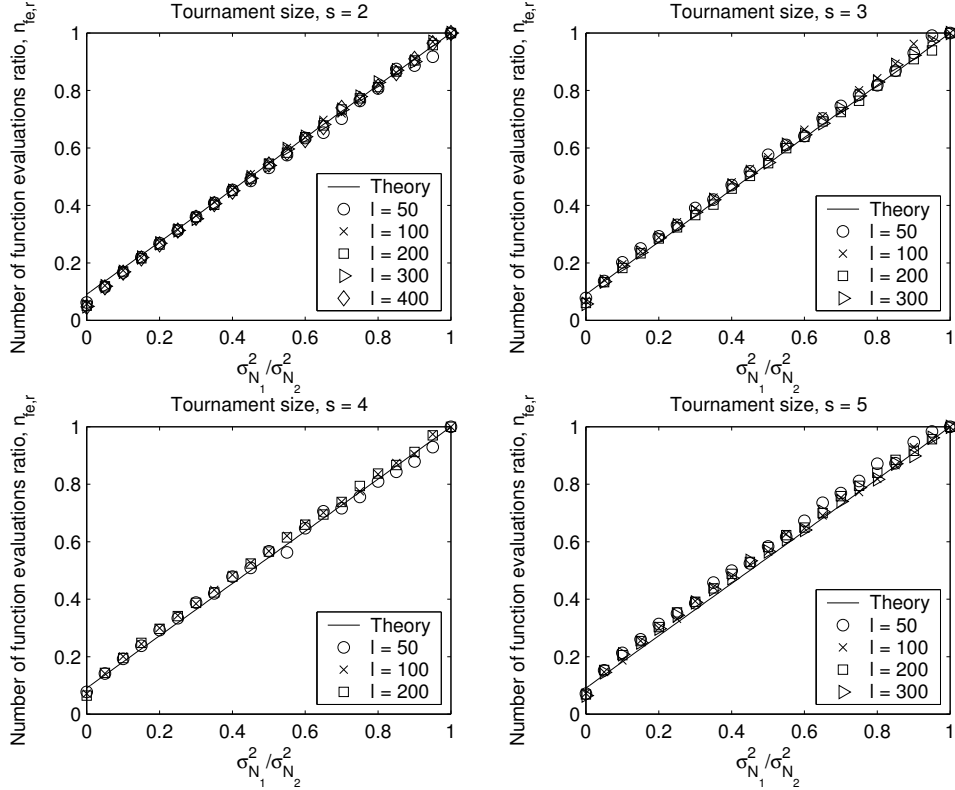


Figure 3: Comparison of empirical and theoretical results for the ratio of total number of function evaluations.

fitness difference of the best and the second best BB, χ is the alphabet cardinality, k is the BB size, and α is the failure rate.

The ratio of population size required to yield a solution of the same quality when fitness function f_1 is used to that when fitness function f_2 is used is then given by

$$n_r = \frac{n(\sigma_{N_1})}{n(\sigma_{N_2})} = \left(\frac{\sigma_f^2 + \sigma_{N_1}^2}{\sigma_f^2 + \sigma_{N_2}^2} \right)^{\frac{1}{2}}. \quad (10)$$

4.3 Number of Function Evaluations

Using equations 8 and 10, we can obtain the ratio of total number of function evaluations taken if fitness function f_1 is used to those taken if fitness function f_2 is used to obtain solution of the same quality.

$$n_{fe,r} = \frac{n_{fe}(\sigma_{N_1})}{n_{fe}(\sigma_{N_2})} = n_r t_{c,r} = \frac{\sigma_f^2 + \sigma_{N_1}^2}{\sigma_f^2 + \sigma_{N_2}^2}, \quad (11)$$

4.4 Model Validation

This section empirically verifies the models presented in the previous sections. The empirical results are obtained for the OneMax problem with string lengths ℓ

$= 50, 100, 200, 300,$ and 400 . Tournament selection without replacement with tournament sizes of $s = 2, 3, 4,$ and 5 is used. Uniform crossover with crossover probability of 1.0 is employed to ensure effective mixing of BBs. The noise variance of fitness function f_2 is taken to be $10\sigma_f^2$ and the noise variance of function f_1 is varied from 0 to $10\sigma_f^2$.

The convergence-time ratio predicted by equation 8 is verified with empirical results and is shown in figure 1. For computing the convergence time, a GA run is terminated if the proportion of correct BBs reaches a value greater than or equal to $(\ell-1)/\ell$. The population size is determined by the following relation (Goldberg, Deb, & Clark, 1992): $n = 8(\sigma_f^2 + \sigma_N^2)$. This is a conservative estimate, and is used to reduce the population-sizing effects. The empirical results are averaged over 50 independent runs. Figure 1 clearly validates the convergence-time model of equation 8. Furthermore, as the model predicts, the empirical results show that the convergence-time ratio is independent of ℓ and s values if the ratio of noise variance to the initial fitness variance is constant.

For computing n_r and $n_{fe,r}$, a GA run was terminated when all the individuals in the population converged

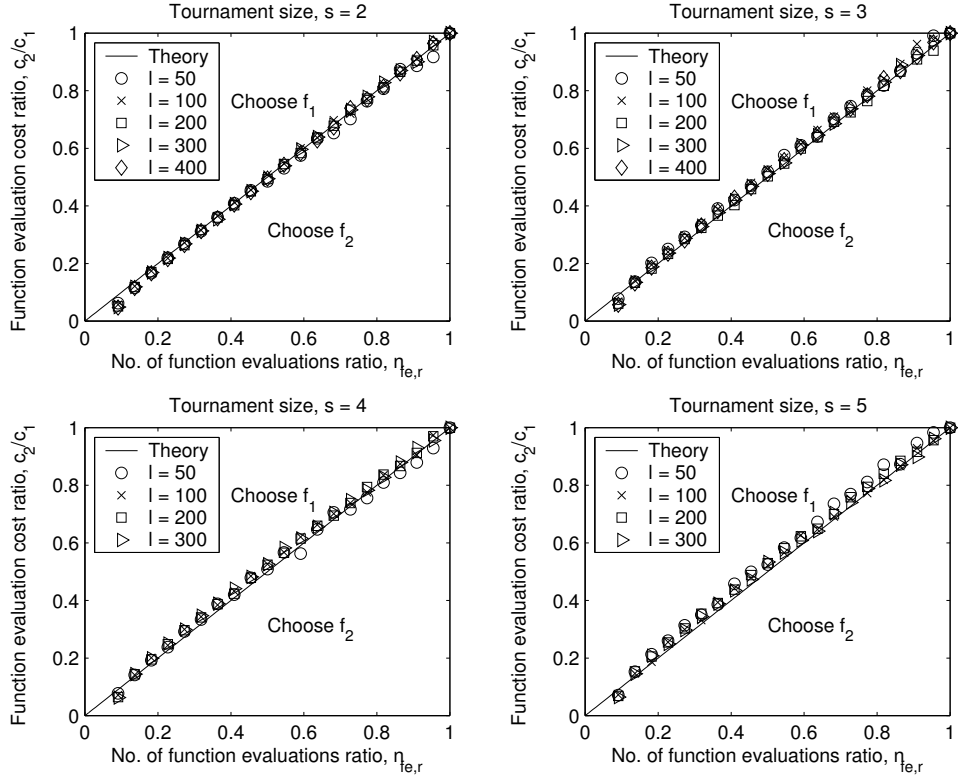


Figure 4: Verification of the optimal decision making between fitness functions with differing variance values.

to the same fitness value. The average number of correctly converged BBs are computed over 50 independent runs. The minimum population size or the total number of function evaluations required for the GA to correctly converge on an average to at least $m - 1$ BBs ($\alpha = 1/m$), is determined by the bisection method. The results are averaged over 25 bisection runs.

The population-size ratio predicted by equation 10 is verified with empirical results in figure 2. The prediction of the ratio of total number of function evaluations (equation 11) is compared to the empirical results in figure 3. The results show that the models agrees with empirical results over a broad range of parameter values (specifically, noise variance, problem-size, and tournament-size values).

5 Optimal Decision

As mentioned earlier, we have to decide between two fitness functions, one with low variance, but high cost, and the other with high noise but low cost. The ratio of total cost of employing fitness function f_1 to that of employing fitness function f_2 to obtain solution of the

same quality is given by

$$\frac{c_{\text{tot},1}}{c_{\text{tot},2}} = \frac{c_1 n_{fe,1}}{c_2 n_{fe,2}} = \frac{c_1}{c_2} \left(\frac{\sigma_f^2 + \sigma_{N_1}^2}{\sigma_f^2 + \sigma_{N_2}^2} \right), \quad (12)$$

where, $c_{\text{tot},1}$ is the total cost of employing fitness function f_1 , and $c_{\text{tot},2}$ is the total cost of employing fitness function f_2 . From the above relation, we can summarize the optimal decision as follows:

- If $c_2/c_1 > (\sigma_f^2 + \sigma_{N_1}^2)/(\sigma_f^2 + \sigma_{N_2}^2)$, then use f_1 .
- If $c_2/c_1 < (\sigma_f^2 + \sigma_{N_1}^2)/(\sigma_f^2 + \sigma_{N_2}^2)$, then use f_2 .
- If $c_2/c_1 = (\sigma_f^2 + \sigma_{N_1}^2)/(\sigma_f^2 + \sigma_{N_2}^2)$, then either f_1 or f_2 can be used.

This decision making process is shown pictorially in figure 4, where the theory is verified with empirical results. The figure plots the cost ratio of fitness functions for different values of fitness variance ratios. The empirical results shown are obtained for the OneMax problem with string lengths, $\ell = 50, 100, 200, 300$, and 400. A selectorecombinative GA with tournament selection without replacement and uniform crossover is used for this purpose.

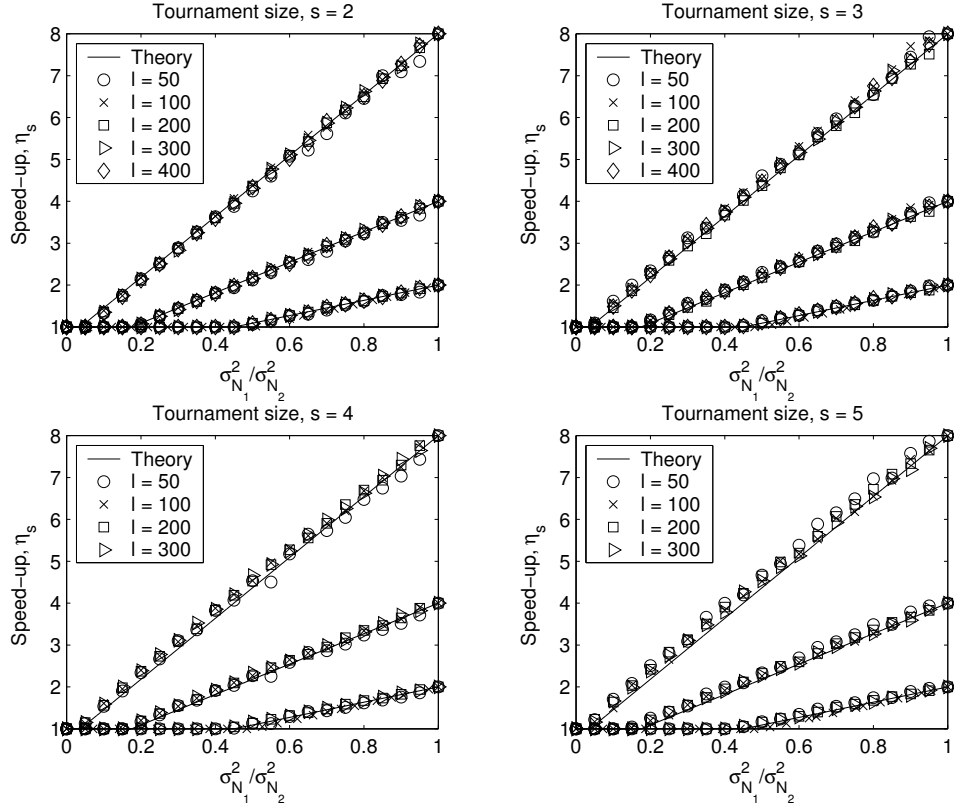


Figure 5: Empirical verification of speed-up predicted by equation 13.

Speed-up is defined as the ratio of the total cost of using a high-cost, low-variance fitness function to the total cost of using a low-cost, high-variance fitness function. Therefore, speed-up obtained by using the aforementioned optimal decision is given by

$$\eta_s = \begin{cases} \frac{c_{\text{tot},1}}{c_{\text{tot},2}} & \frac{c_1}{c_2} > \frac{\sigma_f^2 + \sigma_{N_2}^2}{\sigma_f^2 + \sigma_{N_1}^2} \\ 1.0 & \text{elsewhere} \end{cases} \quad (13)$$

This definition of speed-up assumes that one always chooses the more accurate fitness function. The above speed-up measures the improvement in efficiency when a correct decision is made instead of a naive decision. When a decision-making procedure, such as the one developed in this section is not available, the naive choice is the use the more accurate fitness function. Justification for using this definition of speed-up is given elsewhere (Sastry, 2001)

The speed-up predicted by equation 13 is verified with empirical relations in figure 5 for different cost-ratio, problem-size, and tournament-size values. The results clearly indicate the a high speed-up can be obtained if the cost-ratio of the fitness functions (c_2/c_1) is much lower than their fitness variance ratios ($\sigma_{f_1}^2/\sigma_{f_2}^2$).

The key thing is that even though we started with simplified assumptions, the decision-making is somewhat general in nature. The only control parameters in the decision making process are the relative cost and fitness variance values. Using dimensional argument, one can extrapolate the results obtained here to other problem domains. In such cases, the decision will be correct in an order-of-magnitude sense. Therefore, the core message of this section is as follows: If an optimization problem has many different fitness function with differing values of variance, and computational costs, then a fitness function with least product of cost and fitness variance should be employed.

6 Conclusions

This paper addressed the issue of deciding between fitness functions with differing variance and cost values. An approximate, but practical convergence-time model was developed and used along with a population-sizing model to develop a decision-making strategy and to predict speed-up. Although in this paper only two fitness functions were considered, the decision making can be easily extended for more than two fitness functions.

The decision-making suggests that the effect of variance can be handled spatially and the choice of the fitness function depends only on the relative cost and variance ratios of the fitness functions. Significant speed-up can be obtained by employing the decision-making strategy developed in this paper. Based on dimensional arguments, the decision-making strategy presented here, though developed for the OneMax problem, should be applicable to other fitness domains.

Acknowledgments

This work was sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant F49620-00-0163, and the National Science Foundation under grant DMI-9908252. The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright notation thereon.

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, the National Science Foundation, or the U.S. Government.

References

- Albert, L. A. (2001). *Efficient genetic algorithms using discretization scheduling*. Master's thesis, University of Illinois at Urbana-Champaign, General Engineering Department, Urbana, IL.
- Bäck, T. (1995). Generalized convergence models for tournament—and (μ, λ) —selection. *Proceedings of the Sixth International Conference on Genetic Algorithms*, 2–8.
- Blickle, T., & Thiele, L. (1995). A mathematical analysis of tournament selection. *Proceedings of the Sixth International Conference on Genetic Algorithms*, 9–16.
- Bulmer, M. G. (1985). *The mathematical theory of quantitative genetics*. Oxford: Oxford University Press.
- El-Beltagy, M., Nair, P., & Keane, A. (1999). Meta-modeling techniques for evolutionary optimization of computationally expensive problems: Promises and limitations. *Proceedings of the Genetic and Evolutionary Computation Conference*, 196–203.
- Fitzpatrick, J. M., & Grefenstette, J. J. (1988). Genetic algorithms in noisy environments. *Machine Learning*, 3, 101–120.
- Goldberg, D. E. (1999). The race, the hurdle, and the sweet spot: Lessons from genetic algorithms for the automation of design innovation and creativity. In Bentley, P. (Ed.), *Evolutionary Design by Computers* (Chapter 4, pp. 105–118). San Mateo, CA: Morgan Kaufmann.
- Goldberg, D. E. (in press). *Design of innovation: Lessons from and for competent genetic algorithms*. Boston, MA: Kluwer Academic Publishers.
- Goldberg, D. E., & Deb, K. (1991). A comparative analysis of selection schemes used in genetic algorithms. *Foundations of Genetic Algorithms*, 69–93.
- Goldberg, D. E., Deb, K., & Clark, J. H. (1992). Genetic algorithms, noise, and the sizing of populations. *Complex Systems*, 6, 333–362.
- Grefenstette, J. J., & Fitzpatrick, J. M. (1985). Genetic search with approximate function evaluations. *Proceedings of the International Conference on Genetic Algorithms and Their Applications*, 112–120.
- Harik, G., Cantú-Paz, E., Goldberg, D. E., & Miller, B. L. (1997). The gambler's ruin problem, genetic algorithms, and the sizing of populations. *Proceedings of the IEEE International Conference on Evolutionary Computation*, 7–12.
- Jin, Y., Olhofer, M., & Sendhoff, B. (2000). On evolutionary optimization with approximate fitness functions. *Proceedings of the Genetic and Evolutionary Computation Conference*, 786–793.
- Keijzer, M., & Babovic, V. (2000). Genetic programming, ensemble methods and the bias/variance tradeoff - introductory investigations. *Genetic Programming: Third European Conference*, 76–90.
- Mandava, V. R., Fitzpatrick, J. M., & Pickens, III, D. R. (1989). Adaptive search space scaling in digital image registration. *IEEE Transactions on Medical Imaging*, 8(3), 251–262.
- Miller, B. L. (1997). *Noise, sampling, and efficient genetic algorithms*. Doctoral dissertation, University of Illinois at Urbana-Champaign, General Engineering Department, Urbana, IL. (Also IlliGAL Report No. 97001).
- Miller, B. L., & Goldberg, D. E. (1995). Genetic algorithms, tournament selection, and the effects of noise. *Complex Systems*, 9(3), 193–212.
- Miller, B. L., & Goldberg, D. E. (1996). Genetic algorithms, selection schemes, and the varying effects of noise. *Evolutionary Computation*, 4(2), 113–131.
- Mühlenbein, H., & Schlierkamp-Voosen, D. (1993). Predictive models for the breeder genetic algorithm: I. continuous parameter optimization. *Evolutionary Computation*, 1(1), 25–49.
- Prügel-Bennet, A., & Shapiro, J. L. (1994). An analysis of a genetic algorithm using statistical mechanics. *Physics Review Letters*, 72(9), 1305–1309.
- Ratle, A. (1998). Accelerating the convergence of evolutionary algorithms by fitness landscape approximation. *Parallel Problem Solving from Nature*, 5, 87–96.
- Sastry, K. (2001). *Evaluation-relaxation schemes for genetic and evolutionary algorithms*. Master's thesis, University of Illinois at Urbana-Champaign, General Engineering Department, Urbana, IL. (Also IlliGAL Report No. 2002004).