A Self Adaptive Hybrid Genetic Algorithm

Felipe P. Espinoza
Department of Civil &
Environmental Engineering
University of Illinois
4129 Newmark Lab, MC-250
205 N. Mathews Ave.
Urbana, IL 61801
fespinoz@uiuc.edu
(217)-333-6979

Barbara S. Minsker
Department of Civil &
Environmental Engineering
University of Illinois
3230 Newmark Lab, MC-250
205 N. Mathews Ave.
Urbana, IL 61801
minsker@uiuc.edu
(217)-333-9017

David E. Goldberg
Department of General Engineering
University of Illinois
Urbana, IL 61801
117 Transportation, MC 238
104 S. Mathews Ave.
Urbana, IL 61801
deg@uiuc.edu
(217)-333-0897

Abstract

This paper presents a self-adaptive hybrid genetic algorithm (SAHGA) and compares its performance to a non-adaptive hybrid genetic algorithm (NAHGA) and the simple genetic algorithm (SGA) on two multi-modal test functions with complex geometry. The SAHGA is shown to be far more robust than the NAHGA, providing fast and reliable convergence across a broad range of parameter settings. For the most complex test function, the SAHGA required over 75% fewer function evaluations than the SGA to identify the optimal solution at a 99% reliability level.

1 INTRODUCTION

A hybrid genetic algorithm (HGA) is the coupling of two processes: the simple GA and a local search algorithm. HGAs have been applied to a variety of problems in different fields, such as optical network design [Sinclair, 2000], signal analysis [Sabatini, 2000], and graph problems [Magyar et al, 2000], among others. In these previous applications, the local search part of the algorithm was problem specific and was designed using trial-and-error experimentation without generalization or analysis of the characteristics of the algorithm with respect to convergence and reliability. The purpose of this study is to develop a self-adaptive HGA (SAHGA) that can be used to reliably solve different applications without extensive trial-and-error experimentation. This paper presents the SAHGA approach and compares its performance with the simple GA (SGA) and a non-adaptive HGA (NAHGA) for several test functions.

The results show considerable promise for the SAHGA. Using two multimodal test functions, the SAHGA required less than 25% of the number of function evaluations required for the SGA at a 99% reliability level. The SAHGA algorithm was also more robust than the NAHGA, performing optimally across a broad range of parameter values.

2 HYBRID GENETIC ALGORITHM METHODOLOGY

2.1 BASIC ELEMENTS

2.1.1 Genetic Algorithm

The simple Genetic Algorithm (SGA) used in this work is defined by three basic operators: binary tournament selection, single point crossover, elitism, and simple mutation. Through the successive application of these three operators, an initial population of solutioms is evolved into a highly fit population.

2.1.2 Local Search

The local search operator looks for the best solution starting at a previously selected point, in this case a solution in the SGA population. For this application, the steepest descent method was chosen as the local search operator. This method moves along the direction of the steepest gradient until an improved point is found, from which a new local search starts. The algorithm ends when no new point can be found (this is equivalent to a gradient equal to zero). For functions with multiple local optimum, the method finds one local optima but is not guaranteed to find the global minimum. For geometries with conical shape, for example, the method finds the local optimum in one local search starting from any point inside the basin of attraction. For other geometries, the local search operator requires more than one iteration to achieve the solution.

2.1.3 Evolution: Lamarckian v/s Baldwinian

To combine the SGA and local search methods, HGAs typically use one of two approaches: Lamarckian or Baldwinian evolution [Hinton and Nolan, 1987], [Whitley et al., 1994]. Lamarck presented his theory of learned evolution in 1802 [Lamarck, 1802], in which direct learning passes the best characteristics of each individual from generation to generation. This means that both the change in genotypic information and fitness are passed to the individual as genotypic information at the end of local search (i.e., the chromosome of the individual is changed). Baldwinian evolution, also known as the Baldwin effect [Baldwin, 1896], is survival of the fittest following the direction of learning. In this case, only the improved fitness function value is changed after local search and not the genotypic information. Lamarckian evolution has been shown to cause faster convergence than Baldwinian evolution, but sometimes causes premature convergence problems [Whitley et al., 1994].

2.2 NON-ADAPTIVE HYBRID GENETIC ALGORITHM (NAHGA)

The NAHGA algorithm is a standard, non-adaptive hybrid genetic algorithm that combines an SGA with local search. The local search step is defined by three basic parameters: frequency of local search, probability of local search, and number of local search iterations. The first element for the definition of the algorithm is the frequency of local search, which is the switch between global and local search. In the NAHGA algorithm, this switch is performed every ?G global search generations, where ?G is a constant number called the local search frequency. For example, if ?G =3, local search would be performed every 3 generations during the SGA. The second element of the algorithm is the probability of local search P, which is the probability that local search will be performed on each member of the SGA population in each generation where local search is invoked. This probability is constant and is defined before the application of the algorithm. Finally, each time local search is performed, it is performed a constant number of local search iterations before local search is halted.

2.3 SELF-ADAPTIVE HYBRID GENETIC ALGORITHM (SAHGA)

The SAHGA algorithm works with the same operators as the NAHGA algorithm: frequency of local search, probability of local search and number of local search iterations. The major difference in the approaches is that the SAHGA adapts in response to recent performance as the algorithm converges to the solution. The details of the adaptations are given below.

2.3.1 Local Search versus Global Search

Instead of a constant local search frequency, local search is invoked only when the SGA's performance, as reflected by chasnges in the relative coefficient of variation of the fitness function between generations, indicates that this is needed. The coefficient of variation is defined as the ratio of the mean and the standard deviation of the population fitness. Figure 1 shows the change in the coefficient of variation and the coefficient of variation itself for a particular experiment using the SGA alone. The trend for the coefficient of variation is decreasing and approaching to zero as the population converges to the optimal solution. Using the CV, we define a new parameter CV ratio (CVR) given in equation 1 and shown in Figure 1:

$$\mathrm{CVR} = \frac{\mathrm{CV}(i)}{\mathrm{CV}(i-1)} \qquad (1)$$

where CV(i) is the coefficient of variation at generation "i." CVR represents the change in the coefficient of variation from one generation to the next. When CVR >1, the solution at generation "i" is worse than the solution at generation "i-1", which implies that local search may provide more information to improve performance. In Figure 1, a threshold of one is shown on the CVR curve to illustrate when the SAHGA would invoke local search.

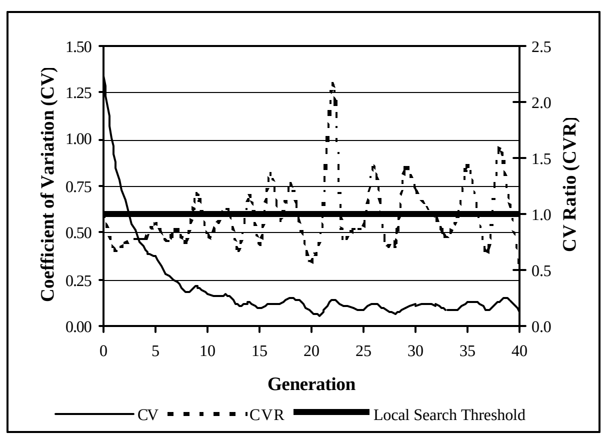


Figure 1: Global Search-Local Search Threshold Effect

2.3.2 Probability of Local Search Selection

In an HGA, local search typically operates over a small portion of the total population because the additional function evaluations required for local search can be very expensive. Therefore, when local search is achieving greater performance than the most recent global search iteration (using the criterion shown in the next section), the SAHGA algorithm is adapted to search a smaller portion of the population using the relationship shown in equation 2.

$$P = P_0\,(1-\varepsilon)^{n-1} \qquad (2)$$

In this equation, the local search probability P decreases in a constant way from the initial value. P_0 is the user-specified initial value of the local search probability, "n" is the local iteration number in the local search step, and "e" is a user-specified parameter governing the rate of decrease in the local search probability. The probability P is reset to P_0 at the beginning of every local search step in order to start with the same sampling size at the beginning of every local search.

2.3.2 Number of Local Search Iterations

One important issue for the application of the algorithm is how long the local search lasts before switching back to the global GA search. In order to make this decision, we compare the most recent fitness improvement by local search with the latest fitness improvement by global search. This criterion is presented in equation 3:

$$\text{do local search if } \frac{\Delta\text{Global}}{\text{pop}} < \frac{\Delta\text{Local}}{\text{fev}} \qquad (3)$$

where ΔGlobal is the improvement achieved between the two previous global search generations, ΔLocal is the current improvement in the local search step, pop is the population size (which is the number of function evaluations required for global search), and fev is the number of function evaluations required for the local search step. This criterion scales the fitness improvements by the computational effort required (pop and fev) so that the ratios are comparable.

When equation (3) is no longer true, or when the number of iterations exceeds a user-specified maximum value, the algorithm switches back to global search.

3 EXPERIMENTS

3.1 TEST FUNCTION

The test functions given in equation 4 are multi-modal functions with multiple basins of attraction. The coordinates $(x_{o,i}, y_{o,i})$ are the coordinates of the basin of attraction "i", which has radius r_i and depth d_i.

For this analysis, we worked with two different functions with random geometry (radius and depth). The basins of attraction for both functions are randomly distributed. Function 1 (f1) (Goldberg and Voessner, 1999) has conical basins of attraction and Function 2 (f2) has elliptical basins of attraction. Function 1 represents the best case for local search, in which only one local search is required to find the local minimum, and Function 2 represents a more realistic case in which multiple local searches are required to find the local minimum.

$$f_1(x_1,y_1) = \begin{cases} \dfrac{d_i}{r_i^2}\left(\overline{x}_1^2+\overline{y}_1^2\right)\left(2-\dfrac{\overline{x}_1^2+\overline{y}_1^2}{r_i^2}\right)-d_i & \overline{x}_1^2+\overline{y}_1^2 \le r_i^2 \\ 0 & \overline{x}_1^2+\overline{y}_1^2 > r_i^2 \end{cases} \qquad (4)$$

$$f_2(x_1,y_1) = \begin{cases} \dfrac{\overline{x}_1^2+d_i\,\overline{y}_1^2}{r_i^2}-d_i & \overline{x}_1^2+d_i\,\overline{y}_1^2 \le r_i^2 \\ 0 & \overline{x}_1^2+d_i\,\overline{y}_1^2 > r_i^2 \end{cases}$$

$$\overline{x}_1 = x_1 - x_{0,i}$$
$$\overline{y}_1 = y_1 - y_{0,i}$$

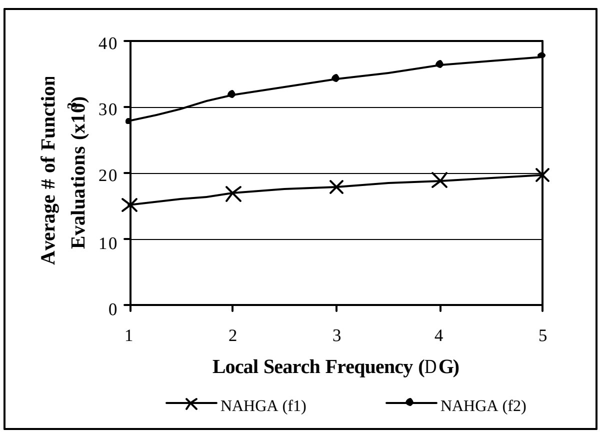
3.2 EXPERIMENTS

In order to evaluate the behavior of the SAHGA with respect to the NAHGA and SGA, we performed several experiments to test the capabilities of the method. The settings for the parameters controlling the SGA for all of the experiments (population size of 800 and 1,200 individuals, respectively, for f1 and f2; probability of crossover of 0.4; and probability of mutation of 0.0013 and 0.0008, respectively, for f1 and f2) were identified using the 3-step methodology developed by Reed et al. (2000). For local search, we used a mixture of Baldwinian and Lamarckian evolution: 25% of the local searches worked with the Baldwinian effect and 75% with Lamarckian evolution. Our initial experiments found that this mixture represented the optimal choice, giving the speed of Lamarckian evolution without causing diversity problems. The stopping criterion for the algorithm was that at least 80% of the population had converged to the solution. In order to evaluate the reliability of the method for different conditions, we averaged the results of 1,000 different initial populations from 1,000 random seeds. The results are presented in terms of "average number of function evaluations" because generations take different amounts of time for the hybrid genetic algorithm approach, depending on how many local searches are done.

3.2.1 Frequency of Local Search

The first experiment was designed to evaluate the effect of local search frequency on the solution of the problem. For the NAHGA algorithm, local search was perfomed at a pre-defined interval ?G; for the SAHGA algorithm, local search followed the threshold requirements previously explained. Figures 2 a) and b) show the results for the NAHGA and for the SAHGA algorithms, respectively. For both algorithms, the maximum number of local search iterations was three and the initial probability of local search was 0.1. For the NAHGA, it is clear that the optimal results are achieved only for one value of the variable in study, a local search frequency of 1 (meaning that local search is performed in every generation). On the other hand, for the SAHGA the optimal results are achieved for a set of different values of the "local search threshold", so the algorithm is more robust. The performance of each of the algorithms for different local search frequencies and threshold parameter values was similar for both functions.

a)



b)

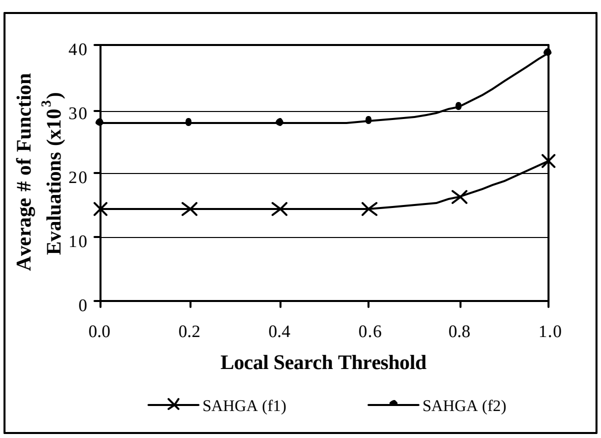
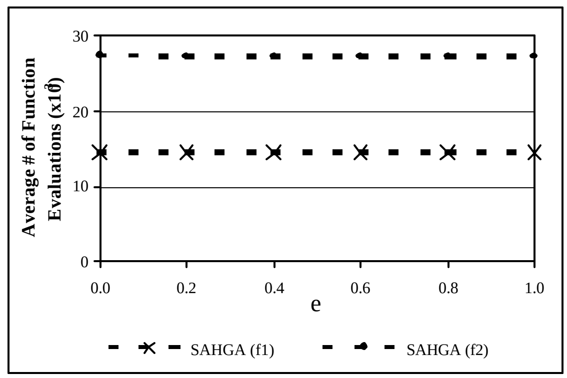


Figure 2: Local Search Effect for NAHGA Algorithm (a) and SAHGA Algorithm (b)

3.2.2 Probability of Local Search

The second experiment tested the effect of the probability of local search parameter on performance. In the SAHGA, the probability of local search is adapted using the parameter e in equation 2. Figure 3 a) shows the effect of the adaptive parameter for a specific probability of local search (P_0=0.1). For the other parameters in the algorithm, the threshold was set to 0.6 and a maximum of 3 local search iterations were performed. This experiment shows that there is only a slight improvement in performance for different values of the adaptive parameter for the two different functions. Again, this is another indication of the robustness of the SAHGA algorithm.

a)



b)

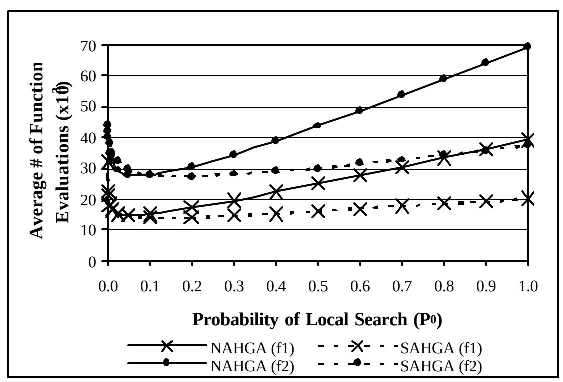


Figure 3: Adaptive Parameter Effect on Probability of Local Search (a) and Probability of Local Search Effect (b)

The next step is to evaluate the behavior of both algorithms for different probabilities of local search. For the NAHGA, the generation gap selected was 1, which gave the best performance in the first experiment. Figure 3 b) shows the results of this experiment, which indicates that the minimum number of function evaluations occurs at almost the same probability for both the NAHGA and the SAHGA algorithm. The major difference is that the NAHGA achieves the minimum for only one probability of local search and the SAHGA achieves optimal or very near optimal performance for a broader range of initial probabilities of local search due to its adaptation of P_0 during the run. This effect is achieved for both functions.

3.2.3 Maximum Number of Local Search Iterations

The final experiment analyzes the number of iterations in the local search step. For this analysis, we worked with a probability of local search equal to 0.1 for both algorithms, a ΔG equal to 1 for the NAHGA, and a threshold of 0.6 and an adaptive parameter (e) equal to 0.2 for the SAHGA. Figure 4 shows the results of this experiment. These results indicate that the number of function evaluations for the NAHGA algorithm increases with the number of local search iterations allowed, but for the SAHGA algorithm the number of function evaluations remains constant because of the adaptive stopping criterion in the SAHGA local search algorithm

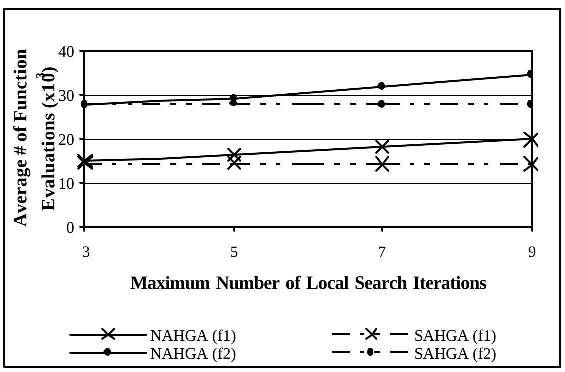
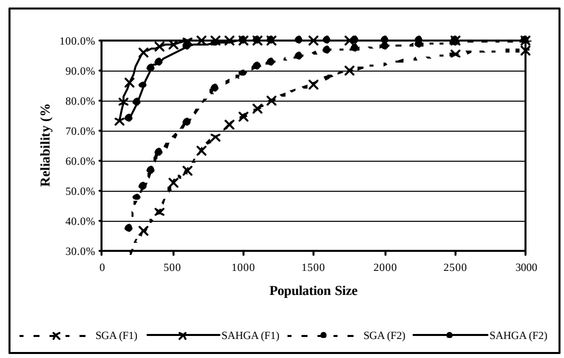


Figure 4: Maximum Number of Local Search Iterations Effect for NAHGA and SAHGA Algorithm

3.2.4 Reliability

To complete the analysis, we performed a final experiment to investigate the reliability of the SAHGA relative to the SGA for different population sizes. The analysis was performed only for the SAHGA algorithm because, as shown in the previous experiments, this algorithm worked for a broader range of parameters than the NAHGA. Figure 5 a) shows the reliability of each algorithm for different population sizes, where reliability is defined as the percentage of the 1,000 different initial populations that found the optimal solution. It is clear that the SAHGA achieves much higher levels of reliability at smaller population sizes than the SGA. Figure 5 b) shows reliability versus number of function evaluations. From this plot, it is clear that the SAHGA is able to achieve much higher reliability with far fewer function evaluations. For each level of reliability, Figure 6 shows the average number of function evaluations required for the SAHGA as a percentage of the number required for the SGA. For a reliability of 99%, the number of function evaluations required in the SAHGA for function f1 is 95% less than the number of function evaluations for the SGA algorithm. For function f2, the SAHGA requires 75% fewer function evaluations required for the SGA. More function evaluations are required for function f2 because f2 is much more complex than f1 and requires more local search iterations. These results were achieved for a population size equal to 15% and 35% of the optimal population size for the SGA, for f1 and f2, respectively. The improved performance shown in Figure 5 is a combined effect of smaller population sizes and faster convergence of the SGA.

a)



b)

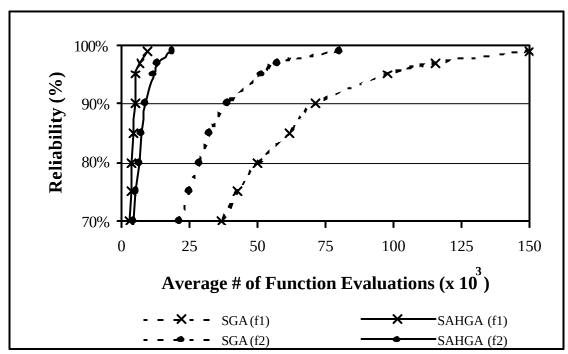


Figure 5: Reliability v/s Population Size a) and Reliability versus Number of Function Evaluations b)

4 Conclusions

The results presented in this paper clearly indicate that the adaptive capabilities of the SAHGA algorithm enabled robust solution of complex, multi-modal problems for a much greater range of parameter settings than the NAHGA. Compared with the SGA, the SAHGA was able to solve complex problems much faster because of the combined effect of smaller population sizes and increased information from local search. For the same level of reliability, the SAHGA required as much as 95% fewer function evaluations than the SGA for function f1 and as much as 75% fewer function evaluations for function f2. Further research is needed to assess the performance of the algorithm on other types of functions.

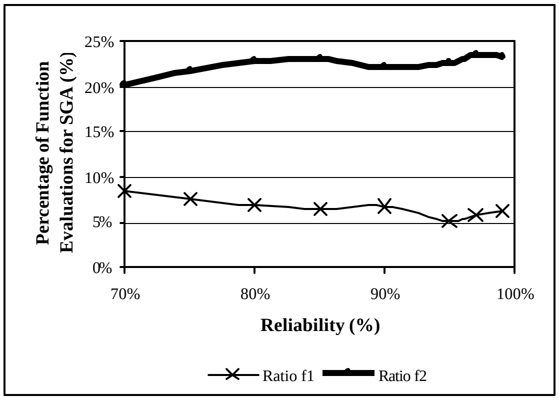


Figure 6: Average Number of function Evaluations for the SAHGA as a Percentage of the Number for the SGA at each Level of Reliability

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. BES 97-34076 CAR.

References

J. M. Baldwin, A new factor in evolution. *American Naturalist*, 30:441-451, 1896.

R. French and A. Messinger, Genes, phenes and the Baldwin effect: learning and evolution in a simulated population. *Artificial Life IV,* 277-282, 1994.

G. E. Hinton and S. J. Nolan, How learning can guide evolution. *Complex Systems*, 1:495-502, 1987.

D. E. Goldberg and S. Voessner, Optimizing global-local search hybrids, Illigal Report No. 99001, January 1999.

J. Lamarck, Zoological Philosophy, 1809.

G. Magyar, M. Johnsson, and O. Nevalainen, An adaptive hybrid genetic algorithm for the three-matching problem, *IEEE Transactions on Evolutionary Computation*, 4(2), 135-146, 2000.

P. Reed, B. S. Minsker, and D. E. Goldberg, Designing a competent simple genetic algorithm for search and optimization. *Water Resources Research*, 36(12), 3757-3761.

A. Sabatini, A hybrid genetic algorithm for estimating the optimal time scale of linear systems approximations using Laguerre models, *IEEE Transactions on Automatic Control*, 45(5), 1007-1011, 2000.

M. Sinclair, Minimum cost wavelength-path routing and wavelength allocation using a genetic-algorithm/heuristic hybrid approach, *IEEE Proceedings Comunications*, 146(1), 1999.

D. Whitley, V. S. Gordon, and K. Mathias, Lamarckian evolution, the Baldwin effect and function optimization, *Parallel Problem Solving from Nature- PPSN III,* 6-15, 1994.