Performance of Hybrid Genetic Algorithms Incorporating Local Search

T. Elmihoub, A. A. Hopgood, L. Nolle and A. Battersby
The Nottingham Trent University, School of Computing and Technology,
Burton Street, Nottingham NG1 4BU, United Kingdom
{tarek.elmihoub, adrian.hopgood, lars.nolle, alan.battersby}@ntu.ac.uk

ABSTRACT

This paper investigates the effects of learning strategy and probability of local search on the performance of hybrid genetic algorithms. It compares the performance of two genetic–local hybrids using different learning strategies and different probabilities of local search. Two test functions are used for the comparisons. The results show that the solution quality of hybrids is not only affected by the Lamarckian or Baldwinian learning strategy, but also by the probability of local search. This probability, together with the learning strategy, has a great impact on population size requirements. These requirements are also affected by the local search method, and the fitness landscape. Reducing the population size can lead to an increase in the algorithm convergence speed.

INTRODUCTION

The ability of genetic algorithms to capture a global view of the search space, when combined carefully with the fast convergence of local search methods (Turney 1996), can often produce an algorithm that outperforms either one alone (Bobo and Goldberg 1997). Hybridizing a local search method provides the global genetic search algorithm with some local knowledge that can guide and may accelerate the search to the global optimum (Hart 1994).

The usual motivation for hybridization in optimization practice is the achievement of increased efficiency (Goldberg and Vosser 1999). The efficiency of any hybrid depends on many factors, e.g. how the hybrid decides between global and local knowledge (Bobo and Goldberg 1997), how it strikes a balance between the cost and value of local knowledge (Hart 1994), and how successfully local knowledge are utilized by the global genetic algorithm (Whitley et al. 1994). The efficiency of any hybrid can be measured by comparing its performance with that of the global genetic algorithm alone. Espinoza et al. (2001) have proposed an adaptive hybrid algorithm that can increase convergence speed to the global optimum. The same authors also show the effect of a local search method on reducing the population size of the algorithm compared with the population size of the standard genetic algorithm (Espinoza et al. 2003).

In this paper, a further step is taken in this direction by investigating the effect of the learning strategy and probability of local search on the performance on both the adaptive hybrid algorithm and the standard staged hybrid. The effect of both these factors on the population size requirements, convergence speed and solution quality has been studied.

LAMARCKIAN EVOLUTION AND BALDWIN EFFECT

One of the important issues of hybrid genetic algorithms is how the information gained during local search is used by the global algorithm. Either the Lamarckian or the Baldwinian approach can be used. In the Lamarckian approach the traits acquired during the learning process are passed from parents to their offspring. This means that both the genetic structure of an individual and its associated fitness value are modified to reflect the changes in phenotype structure as a result of performing local search (Whitley et al. 1994). The Baldwin Effect is somewhat Lamarckian in its results but using different mechanisms (Turney 1996) In the Baldwinian approach the learning process can help the individual to adapt to its environment and as a result to survive and gain more chance to pass on its traits to the next generation. In this case, only the improved fitness value is modified to reflect the effect of performing local search, thereby allowing individuals with the ability to learn to proliferate in the population.

Although Lamarckian evolution has been universally rejected as a viable theory of genetic evolution in nature, using it as learning strategy in genetic algorithms can improve their convergence speed (Whitley et al. 1994). The Lamarckian strategy can disrupt schema processing of genetic algorithms in staged hybrid algorithms and in some cases this may lead to the premature convergence problem (Whitley et al. 1994). In many real-world applications, it is not possible to use the Lamarckian approach because the inverse mapping from phenotype to genotype is computationally intractable (Turney 1996). The Baldwinian approach, in spite of being characterized by slow convergence speed compared with that of Lamarckian (Whitley et al. 1994), has a smoothing effect on the search landscape and does not disrupt the global genetic search (Gruau and Whitley 1993).

Utilizing either form of learning is more effective than the standard genetic algorithm approach without a local

Proceedings 18th European Simulation Multiconference
Graham Horton (c) SCS Europe, 2004
ISBN 3-936150-35-4 (book) / ISBN 3-936150-36-2 (CD)

improvement procedure (Whitley et al. 1994). The effectiveness of pure Lamarckian, pure Baldwinian or any mixture of them is affected by the fitness landscape, the representations, and local search method used (Whitley et al. 1994; Houck et al. 1997;Michalewicz and Nazhiyath 1995).

BALANCE BETWEEN COST AND VALUE OF LOCAL KNOWLEDGE

In any hybrid algorithm, a local search can be applied either to every individual in the population or only few individuals. Applying a local search to every individual in the population can waste resources without providing any more useful information than applying it to only a small fraction of the population. The use of a large fraction of the population can limit exploring the search space by allowing the genetic algorithm to evolve for a small number of generations. A more selective use of local search can improve the efficiency of hybrids (Hart 1994). Deciding on the optimal fraction of the population that should perform local search, and the basis on which these individuals are chosen, has been investigated by Hart (1994). The cost of local knowledge is measured by the number of function evaluations performed by a local search method to gain that knowledge and its value is measured by its effect on increasing the convergence speed and/or solution quality of the algorithm. The probability of local search can affect the minimum population size of the hybrid which, in turn, can affect the convergence speed of the algorithm. This effect should not be ignored when deciding between different local search probabilities.

POPULATION SIZE REQUIREMENTS

Efficient Population sizing is critical in genetic algorithms for getting the most out of a fixed budget of function evaluations. In (Harik et al. 1997) two factors that influence convergence quality are considered to estimate the population size of genetic algorithms. These factors are the initial supply of building blocks and the selection of the best building blocks over their competitors. The gambler ruin model is used to derive the following relation for population size of genetic algorithms

$$N = \frac{-2^{k-1}\ln(\alpha)\sigma_{bb}\sqrt{\pi(m-1)}}{d}$$

where k is the building-block order, which represents the minimum number of binary digits that have physical significance to the solution of the problem. α is the probability of failure, σ_{bb} is the standard deviation of the building blocks, d is the signal difference between the best and second-best building blocks, and m is the maximum number of building-blocks within a single string. The term $\sqrt{\pi(m-1)}$ represents the noise interference between competing building-blocks. This term can be approximated using the fitness function standard deviation, $\sigma_{fitness}$ (Reed et al 2000).

The computational complexity of a genetic algorithm is measured as the number of function evaluations that are required to attain an optimal solution. The number of function evaluations can be calculated by multiplying the population size (N) by the number of generations required for convergence (t). The number of generation required is strongly affected by the relative rates at which genes within the population converge. The lower and upper bounds for the convergence rates for genetic algorithms applications are functions of $O(\sqrt{l})$ and $O(l)$ for tournament selections, where l is the string length (Thierens et al. 1998). The building blocks of most engineering problems converge at variable rates within the population (Reed 2000). This phenomenon is known as domino convergence. The expected number of generations (t) required under domino convergence for all locations to be converged is given by the following equation.

$$t_{domino} \approx 2l$$

Another phenomenon that is closely related to domino convergence is "genetic drift" (Thierens et al. 1998). This phenomenon occurs in a population when crossover and mutation cause genes to fluctuate and converge to non-optimal values in the absence of selection pressure. Although the genes with reduced relevance to the solution experience reduced selection pressure, they may converge to non-optimal values under the crossover and mutation operations. The expected number of generations for genes to converge in the absence of selection within a randomly generated initial population is given by the following equation (Thierens et al. 1998):

$$t_{drift} \approx 1.4N$$

Domino convergence to optimal solution should occur before genetic drift can occur. The following inequality needs to be satisfied:

$$t_{domino} < t_{drift}$$

or, in terms of population size and string length,

$$N > 1.43l$$

Since carefully designed hybrid genetic algorithms often converge faster than standard genetic algorithms, their convergence to the global optimum can occur even if population size is not greater than 1.43l. The population size for hybrid algorithm should satisfy the following relation

$$2l\frac{t_{domino}}{t_{drift}} < 1.43N$$

where t_{hybrid} is the number of generations required to for a hybrid to converge.

The local search method affects the signal difference between the best individual and the second best, and this can either increase or decrease the population size. It can also decrease the standard deviation of the population and this leads to a decrease in the population size.

ALGORITHMS AND TEST FUNCTIONS

Two hybrids with different mechanisms for deciding between global and local search were used to gain some insight into the effect of learning strategy and probability of local search on the performance of hybrids. The standard staged hybrid genetic algorithm (SSH) (Mathias and Whitley 1994) and the adaptive staged hybrid genetic algorithm (ASH) (Espinoza et al. 2001) have been tested using two multimodal test functions.

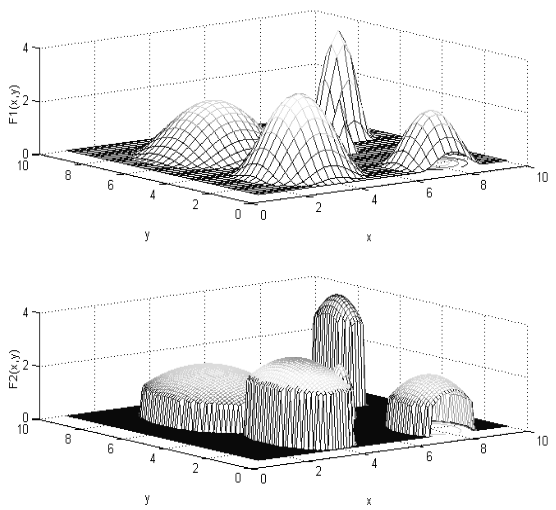

Figure 1: Fitness Landscapes for the Test Functions

In the standard staged hybrid genetic algorithm (SSH), the local search step is defined by three basic parameters: frequency of local search, probability of local search and number of local iterations. The local search frequency measures how frequently local search is performed; the probability of the local search represents the fraction of individuals in the population that undergo local search at each local search iteration; and the number of local search iterations represents the number of local search iterations performed at each local search process.

The adaptive staged hybrid genetic algorithm (ASH), uses feedback from the current state of the search process to direct the algorithm to decide between global and local methods (Espinoza et al. 2001). The algorithm works with the same operators as SSH. It performs local search only if new regions of search space are being discovered, and local knowledge can help to guide the search. The probability of the local search is controlled by a deterministic rule that keeps this probability less than a specific value. When local search no longer improves the average fitness more than the most recent global search iteration, the search goes back to the global search.

Two multimodal test functions, with multiple basins of attraction, have been used in the current work. The first function, F1, has conical basins of attraction. Its global maximum is 4 and is located at (7.0,8.5) (Goldberg and Vosser 1999; Espinoza et al. 2001). The second function, F2, has elliptical basins of attraction. This function has a global optimum of 4 located at (7.0, 8.5) (Espinoza et al. 2001). Figure 1 shows the fitness landscapes of F1 and F2.

The steepest descent method (Press et al.1993) was used as a local searcher. The steepest descent algorithm uses the derivatives of the function to estimate the best step size to climb to the local optimum from the current position in the basin of attraction.

SIMULATIONS AND DISCUSSION

In order to evaluate the effect of learning strategy and local search probability on the hybrids' performance, a set of experiments was performed. Both hybrids use the simple elitist genetic algorithm with binary tournament selection, single-point crossover, and simple mutation. For all experiments, the probability of local search was 0.4 and the probability of mutation was $1/N$ where N is the population size (Reed et al. 2000). For SSH, the frequency of local search was 3 and the number of local iterations was 3. For ASH, the maximum number of local iterations was 3, e was 0.2, and the local threshold value was 0.6. Each variable was represented by 30-bit string with a total of 60 bits for each chromosome. The stopping criterion for all experiments was that 80% of the population had converged to the solution.

Effects on Convergence Speed

In the experiment to evaluate the effect of learning strategy on convergence speed of hybrid algorithms, both the adaptive and standard staged algorithms used a probability of local search of 0.1, and population sizes of 800 and 1200 for F1 and F2 (Espinoza 2001).The stopping criterion was that 80% of the population converged within 0.000001 boundaries of the best ever found solution.

The results show, as expected, that increasing the fraction of the population that evolves according to the Lamarckian approach leads to an increase in the convergence speed. This increase is not linear. For example, when applying ASH on F2, the speed of convergence increases sharply as the learning approach changes from pure Baldwinian (100% Baldwinian) to a mixture of 80% Baldwinian and 20% Lamarckian. In this interval the number of function evaluations decreases from 85,000 to about 37,000, while it decreases to 25,000 evaluations for the pure Lamarckian approach. Figure 2 shows the effect of learning strategy on the convergence speed of the adaptive staged hybrid. The effect of learning strategy on the convergence speed of standard staged hybrid and the adaptive staged hybrid are similar for both test functions.

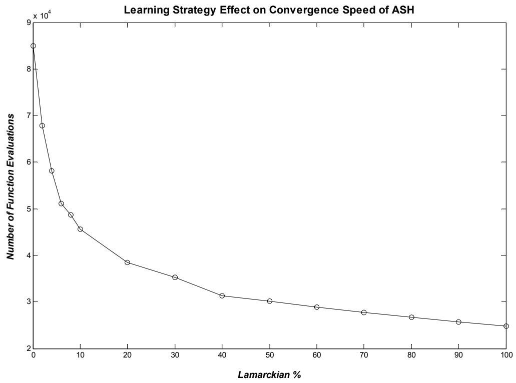

Figure 2: Effect of Learning Strategy on Convergence Speed

Effects on Solution Quality

The results of previous experiments show no clear relation between learning strategy and solution quality. This led us to consider how the local search probability interacts with the learning strategy and how this interaction affects the quality of solutions. An experiment was carried out to consider the effect of local probability on the solution quality for different population sizes (100, 400, 800, and 1200). The results of these experiments show that as probability of local search increases, the effect of learning strategy becomes apparent (figure 3). The graphs in figure 4 show that, when the probability of the local search is kept small, the quality of the solution is insignificantly affected by the learning strategy. As this probability increases, the quality of the solutions degrades with an increasing Lamarckian percentage in the learning process. This means using small local search probabilities for both algorithms, even with pure Lamarckian, can produce high quality solutions because the disruption to schema processing caused by these small probabilities is neglected and has no effect on global search process.

The results in Figure 4 show that a mixture of 20% Lamarckian and 80% Baldwinian produces the most stable solution quality for F2, regardless of the probability of the local search. A mixture of 75% Baldwinian and 25% Lamarckian produces the most stable solution quality for F1 (Figure 5) .The results from both hybrid algorithms show that a pure Baldwinian approach does not always produce the optimal solution quality and that the optimal learning strategy depends on the probability of local search.

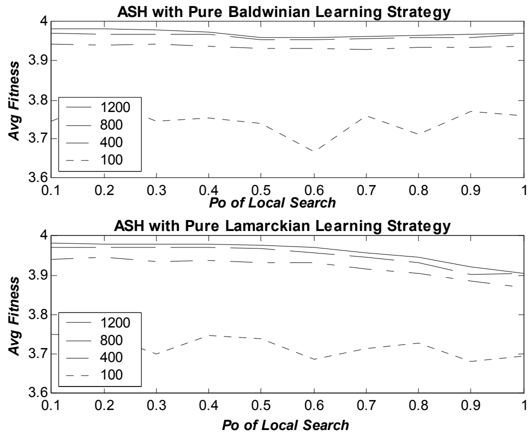

Figure 3: Effect of Learning Strategy and Search Probability on Solution Quality

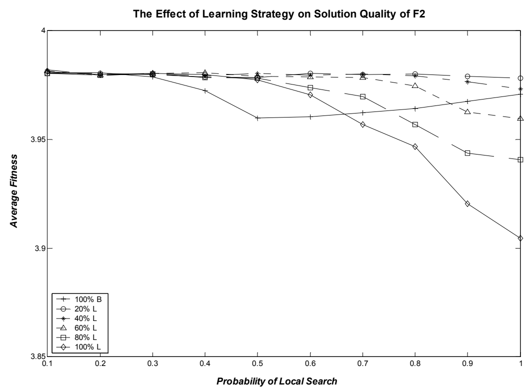

Figure 4: Solution Qualities for F2

Effect on Population Size

The aim of this experiment was to show how the probability of local search and learning strategy affect the minimum size requirements for both hybrids. The results were obtained by using bisection method. Starting with a population size of 10, the population size is doubled until the population converges to the desired solution quality. After the solution quality is attained, the population size is set midway between the current size and the last unsuccessful population size. This process is repeated until the difference between population sizes is less than or equal to 10. The stopping criterion was that 80% of the population converged within 0.000001 boundaries of the global optimum. The settings of other parameters were as in the previous experiments.

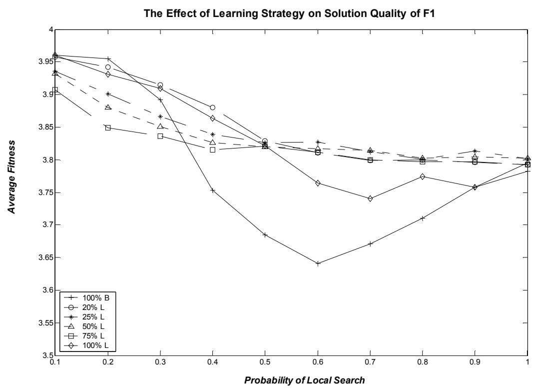

Figure 5: Solution Qualities for F1

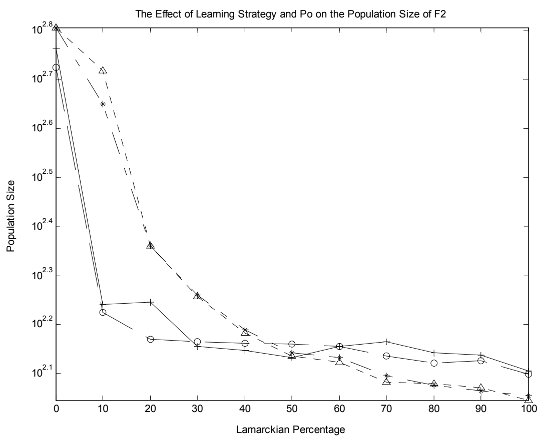

Figure 6: Effect of Learning Strategy and Search Probability on Population Size

The results of SSH and ASH on the second test function are similar. Figure 6 shows that, as the probability of local search increases, the population size decreases for a pure Lamarckian approach. On the other hand, with a pure Baldwinian strategy, the population size increases as the probability of local search increases. For a pure Baldwinian strategy with local search probability of more than 0.4, the population size exceeds that of a pure genetic algorithm (minimum population size=640). The results also show that the relationship between the local search probability and the change in the population size depends on the learning strategy used. For example, using a partial Lamarckian of 50% or more, an increase in the local search probability results in a decrease in population size. With a partial Lamarckian of less than 50%, an increase in the local search probability leads to an increase in the population size. For both hybrids, a decrease in population size leads to an increase in the convergence speed. In general, increasing the Lamarckian percentage decreases the population size and increases the convergence speed. The experiments also show that the solution quality of the pure Baldwinian approach is the optimal and the solution quality is degraded as both the Lamarckian percentage and the probability of local search increase. The solution quality for impure Baldwinian strategies, as shown in Figure 7, seems to be more dependent on the probability of local search than on the learning strategy.

The local search can decrease both the standard deviation of the population and the signal difference between the best and second-best solutions, since the population size depends directly on the standard deviation of the population and the signal difference. A decrease in the former decreases the population size and a decrease in the latter increases the population size.

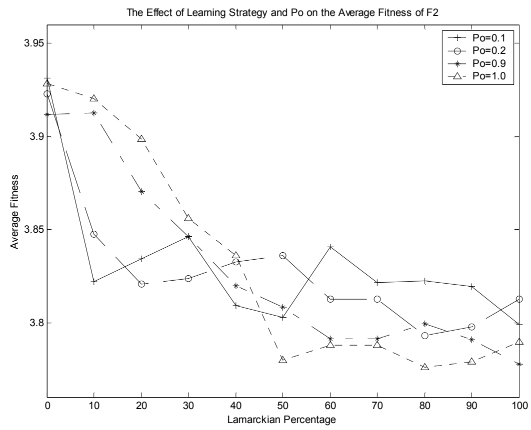

Figure 7: Effect of Lamarkian Proportion and Search Probability on Solution Quality

The increase in the population size requirements for the pure Baldwinian approach can be explained as follows. In a pure Baldwinian, the local search needs some help from evolution process to keep decreasing the ratio of standard deviation to signal difference. Pure Baldwinian can reduce this ratio at the end of the local search. However, in the next global iteration, if the value of local knowledge is insufficient to keep the global genetic algorithm reducing this ratio, the algorithm will lose some of its resources (i.e. local function evaluations) without reducing that ratio. In this case, a high probability of local search cannot lead to any reduction in the population size since it increases the probability of losing the algorithm's resources. However, a low local search probability reduces the probability of lost resources while increasing the probability of maintaining the reduction in the above-mentioned ratio by the global genetic algorithm. In addition to the probability of local search, the effectiveness of pure Baldwinian in reducing the population size depends on the value of local knowledge and this depends on the method of local search and fitness landscape.

On the other hand, the opportunity to keep the gained reduction in this ratio is improved by using a partial Lamarckian strategy. As the percentage of Lamarckian increases, the probability of keeping this reduction increases. An increase in the probability of local search increases the probability of reducing the ratio and reducing the population size.

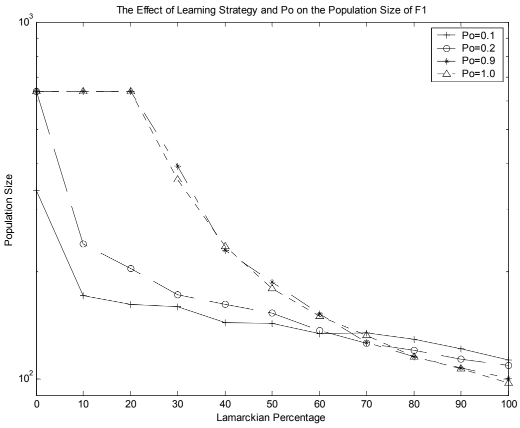

Figure 8: Effect of Learning Strategy and Search Probability on Population Size of F1

Figure 8 shows the results of running the same experiment on the first test function. For a Lamarckian percentage of 65% or more, an increase in the probability of local search results in a reduction in the population size. For other percentages, an increase in this probability leads to an increase in the population size requirements. The convergence speed depends on the population size; as the population size decreases the convergence speed increases. Comparison of Figures 6 and 8 shows that the switch point on the Lamarckian axis between increasing and reducing the population size is shifted from about 50% for F2 to about 65% for F1. This is due to the differences in the fitness landscape of both functions. While the local search can provide more significant local knowledge in F1 than in F2, an impure Lamarckian approach requires a more partial Lamarckian to accelerate the genetic assimilation process.

Additionally, the effect of the local search method on F1 is to enable any solution in a basin of attraction to climb to the exact local optimum in a single step. Consequently, increasing the probability of local search does not necessitate decreasing the signal difference between the best and second-best solutions. It also makes the selection process more difficult as the search process progresses when using a pure Baldwinian approach. In contrast, in F2 the local search method sends any point in the basin of attraction to a point near local optima and not to the local optimum itself.

The local search method can provide more significant local knowledge from the landscape of F1 than F2. This is why the reduction in the population size requirements of F1, using a pure Lamarckian approach, is greater than that of F2. This also makes the genetic assimilation process more difficult for F1 using a pure Baldwin effect compared with F2. The use of a partial Lamarckian can accelerate the genetic assimilation process. The exact value of the switch point depends on the value of the local knowledge.

CONCLUSIONS AND FUTURE WORK

The simulations show that using a low probability of local search and using a pure Lamarckian learning strategy can improve the convergence speed of the algorithm without disrupting the schema processing of the global genetic algorithms. They also show that, depending on the learning strategy used, increasing the probability of local search can decrease or increase the population size. As a result, the convergence speed is affected by the probability of local search. The results show that there is a relation between the probability of local search and the population size.

These experiments have attempted to provide an insight into how the probability of local search and learning strategy affect the population size requirements. We now plan to study how the population size can affect the optimal local search probability by developing a self-adaptive hybrid algorithm that encodes the number of local iterations within the chromosomes themselves and to study how their values propagate during the evolution process.

REFERENCES

Bobo, F. and Goldberg, D. 1997. "Decision Making in a Hybrid Genetic Algorithm", In T. Back (Ed.) Proc. the 1997 IEEE International Conference on Evolutionary Computation, pp. 121-125. IEEE Press

Espinoza, F., B; Minsker, B. and Goldberg, D. 2001. "A Self Adaptive Hybrid Genetic Algorithm". L. Spector, E. Goodman, A. Wu, W.B. Langdon, H.-M. Voigt, M. Gen, S. Sen, M. Dorigo, S. Pezeshk, M. Garzon, and E. Burke, editors. Proceedings of the Genetic and Evolutionary Computation Conference, GECCO'2001. San Francisco, Morgan Kaufmann Publishers.

Espinoza, F.; Minsker, B. and Goldberg, D. 2003. "Performance Evaluation and Population Size reduction for Self Adaptive Hybrid Genetic Algorithm (SAHGA) ", Proc. the Genetic and

Evolutionary Computation Conference, 2723, 922-933, Morgan Kaufmann
Goldberg, D. and Vosser, S. 1999. “Optimizing Global-Local Search Hybrids” Proc. the Genetic and Evolutionary Computation Conference, pp 220-228,Morgan Kaufmann.
Gruau, F. and Whitley, D. 1993.”Adding Learning to the Cellular Development of Neural network: Evolution and Baldwin Effect”, Evolutionary Computation, 1(3),213
Harik, G.; Cantu-Paz, E; Glodberg, D., and Miller, B. 1997. ” The Gambler’s Ruin Problem, Genetic algorithms, and the Sizing of Populations”, In T. Back (Ed.) Proc. the 1997 IEEE International Conference on Evolutionary Computation, pp. 121-125.
Houck, C.; Joines, J.; Kay, M; and Wilson, J. 1997. "Empirical investigation of the benefits of partial Lamarckianism", Evolutionary Computation, v.5, n.1, 31- 60.
Hart, W. E. 1994 .“Adaptive Global Optimization with Local Search”. Doctoral Dissertation, University of California, San Diego, CA
Mathias, K.; Whitley, D. ; Stork, C. and Kusuma, T. 1994. ”Staged Hybrid Genetic Search for Seismic Data Imaging” ,1994 ,International Conference on Evolutionary Computation
Michalewicz, Z. and Nazhiyath, G., 1995. “Genocop III: A Co-evolutionary Algorithm for Numerical Optimization Problems with Nonlinear Constraints” , Proc. of the 2nd IEEE International Conference on Evolutionary Computation, Vol.2, 647-651
Press, W.; Teukolsky, S.; Vetterling, W. and Flannery, B. 1993. "Numerical Recipes in C ", Cambridge University Press
Reed, P.; Minsker, B. and Glodberg D. 2000. “Designing a Competent Simple Genetic algorithm for Search and Optimization”. Water Resource Research, 36(12),3757-3761
Thierens, D., Goldberg, D. and Guimaraes P.1998.:” A. Domino Convergence, Drift, and the Temporal-Salience Structure of Problems. In The 1998 IEEE International Conference on Evolutionary Computation Proceedings, pp. 535-540, IEEE Press, New York, NY
Turney, P. 1996 ”Myths and Legends of the Baldwin Effect” ,Poc. The ICML (13th International Conference on Machine Learning), 135-142.
Whitley, D.Gordon, V. and Mathias, K. 1994. “Lamarckian Evolution, the Baldwin Effect and Function Optimization”, Parallel Problem Solving from Nature-PPSN III, 6-15.

BIOGRAPHIES

TAREK EL MIHOUB is a PhD student at the Nottingham Trent University. He obtained his MSc in Engineering Multimedia in 2002 from Nottingham Trent University. He worked as a teaching assistant in Al-Fatah University in Libya and as a manager of IT department in the Libyan Environment general authority. His current research is in the field of optimization, genetic algorithms, and artificial intelligence.

ADRIAN HOPGOOD is professor of computing and head of the School of Computing and Technology at the Nottingham Trent University, UK. He is also a visiting professor at the Open University. His main research interests are in intelligent systems and their practical applications. He graduated with a BSc (Hons) in physics from the University of Bristol in 1981 and obtained a PhD from the University of Oxford in 1984. He is a member of the British Computer Society and a committee member for its specialist group on artificial intelligence.

LARS NOLLE graduated from the University of Applied Science and Arts in Hanover in 1995 with a degree in Computer Science and Electronics. After receiving his PhD in Applied Computational Intelligence from The Open University, he worked as a System Engineer for EDS. He returned to The Open University as a Research Fellow in 2000. He joined The Nottingham Trent University as a Senior Lecturer in Computing in February 2002. His research interests include: applied computational intelligence, distributed systems, expert systems, optimisation and control of technical processes.

ALAN BATTERSBY obtained an MSc in Computer Science from Hatfield Polytechnic in 1977. Prior to joining the Computing department at Nottingham Trent University, Alan was a Computing Development Officer for Bedfordshire Education Authority. His research interests include: Fuzzy Logic applied to Robotics, wavelets, compression and the Internet.