A Study on the Design Issues of Memetic Algorithm

Q. H. Nguyen, Y. S. Ong, and N. Krasnogor

Abstract— **Over the recent years, there has been increasing research activities made on improving the efficacy of Memetic Algorithm (MA) for solving complex optimization problems. Particularly, these efforts have revealed the success of MA on a wide range of real world problems. MAs not only converge to high quality solutions, but also search more efficiently than their conventional counterparts. Despite the success and surge in interests on MAs, there is still plenty of scope for furthering our understanding on how and why synergy between population-based and individual learning searchers would lead to successful Memetic Algorithms. In this paper we outline several important design issues of Memetic Algorithms and present a systematic study on each. In particular, we conduct extensive experimental studies on the impact of each individual design issue and their relative impacts on memetic search performances by means of three commonly used synthetic problems. From the empirical studies obtained, we attempt to reveal the behaviors of several MA variants to enhance our understandings on MAs.**

I. Introduction

Modern stochastic algorithms such as evolutionary algorithms (EA) draw inspiration from biological evolution. EAs, unlike conventional numerical optimization methods, produce new search points that do not use information about the local slope of the objective function and are thus not prone to stalling at local optima. Instead they involve a search from a "population" of solutions; making use of competitive selection, recombination and mutation operators to generate new solutions which are biased towards better regions of the search space. Further, they have shown considerable potentials for solving optimization problems that are characterized by non-convex, disjoint or noisy solution spaces. Modern stochastic optimizers, which have attracted a great deal of attention in recent years; include simulated annealing, tabu search, genetic algorithms, evolutionary programming, evolutionary strategies, differential evolution and many others [1], [2], [3] and [4]. These stochastic methods have been successfully applied to many real world optimization problems.

Evolutionary algorithms are capable of exploring and exploiting promising regions of the search space. They can, however, take a relatively long time to locate the exact local optimum in a region of convergence (and may sometimes not find the optimum with sufficient precision). Most recent global optimization algorithms are now designed to achieve better exploration and exploitation using a combination of dedicated population based and individual learning searches. It has the potential of exploiting the complementary advantages of EAs (generality, robustness, global search efficiency), and problem-specific local search (exploiting application-specific problem structure, rapid convergence toward local minima). Such combinations of optimizers are commonly known as hybrid methods. In diverse contexts, hybrid EAs are also commonly known as Memetic Algorithms (MAs), Baldwinian EAs, Lamarckian EAs, cultural algorithms or genetic local search. Such methods have been demonstrated to converge to high quality solutions more efficiently than their conventional counterparts, [5], [6] and [7]. Since we consider evolutionary algorithms that employ individual learning heavily during the entire lifetime of the search, the term Memetic Algorithms is most appropriately used. Besides, the name of Memetic Algorithms is more widely used now since it is believed to be more general and encompasses all the major concepts involved in the others.

Over the recent years, many dedicated MAs have been crafted to solve domain-specific problems more efficiently [8], [9], [10], [11], [12] while a distinct group of researchers has concentrated on the algorithmic aspect of MA as combinations of EAs with individual learning procedures [13], [14], [15], [16], [17]. From a survey of the field, it is now well established that potential algorithmic improvements can be achieved by considering some important issues of MA such as the choice of individual learning procedure or local improvement procedure or meme to employ [18], [19], [20], the frequency and intensity at which individual learning is used [21], [22] including the subset of solutions on which individual learning is applied.

In this paper, the aim is to provide a systematic study on the design issues of MA so as to reveal the behaviors of MAs and enhance the understandings on the search mechanism of MA. The paper is organized as follows. Section II describes several important design issues of Memetic Algorithms in details. Section III introduces the experimental procedures used in this study while Section IV presents and analyzes the experimental results obtained from studies on the mechanism of each design issue as well as their relative impacts on memetic search performances using three commonly used benchmark functions. Finally, Section V concludes this paper with a brief summary.

II. Memetic Algorithm

Memetic Algorithms are population-based meta-heuristic search methods inspired by both Darwinian principles of natural evolution and Dawkins notion of a meme as a unit of cultural evolution capable of individual learning. In a more diverse context, MA can be defined as a synergy of

Q. H. Nguyen and Y. S. Ong are with the School of Computer Engineering, Nanyang Technological University, 639798, Singapore. Krasnogor is with the School of Computer Science and IT, Jubilee Campus, University of Nottingham, Nottingham, NG8 1BB, United Kingdom (e-mail: nguy0046@ntu.edu.sg, asysong@ntu.edu.sg, Natalio.Krasnogor@nottingham.ac.uk).

evolution and individual learning. The pseudo-code of a Memetic Algorithm is outlined in Algorithm 1.

Algorithm 1 Memetic Algorithm

Initialize: Generate an initial population;
while Stopping conditions are not satisfied **do**
  Evaluate all individuals in the population.
  Evolve a new population using stochastic search operators.
  Select the subset of individuals, Ω_{il}, that should undergo the individual improvement procedure.
  for each individual in Ω_{il} **do**
    Perform individual learning using meme(s) with frequency or probability of f_{il}, for a period of t_{il}.
    Proceed with Lamarckian or Baldwinian learning.
  end for
end while

In order to locate the global optimum of a search problem accurately and efficiently under limited computational budget, a good balance between exploration and exploitation in the MA must be appropriately maintained throughout the optimization search process. In what follows, we present a brief overview on some of the core issues considered in the literature. For the sake of conciseness, we use the following definitions and notations throughout the rest of this paper:

Definition 1: Individual learning frequency, f_{il}, defines the proportion of an EA population that undergoes individual learning. For instance, if po is the EA or MA population size, the number of individuals in the population that undergoes individual improvement is then $f_{il} \times po$.

Definition 2: Individual learning intensity, t_{il}, is the amount of computational budget allocated to an iteration of individual learning, i.e., the maximum computational budget allowable for individual learning to expend on improving a single solution.

Definition 3: The subset of individuals that should undergo the individual learning procedure is denoted by Ω_{il}, where $|\Omega_{il}| = f_{il} \times po$.

The frequency and intensity of individual learning directly define the degree of evolution (exploration) against individual learning (exploitation) in the MA search, given a fixed limited computational budget. Clearly, a more intense individual learning provides greater chance of convergence to the local optima but limits the amount of evolution that may be expended without incurring excessive computational resources. Therefore, care should be taken when setting these two parameters to balance the computational budget allocated to the two objectives. If not all individuals of the population undergo individual learning, it becomes necessary to also consider which subset of individuals to improve so as to maximize the utility of MA search. Last but not least, the individual learning procedure/meme used also favors a different neighborhood structure, hence the need to decide which meme or memes to use for a given optimization problem at hand.

III. Experimental Procedure

In this section, we describe the experimental procedure used to study the various design issues of MAs. To begin, we investigate first the impacts of choice of population-based and individual learning procedures used in creating an MA. Among the candidate population-based approaches considered here are i) Simple Genetic algorithm (GA) [2], and ii) Differential Evolution (DE) [23] and iii) Evolutionary Strategy (ES) [1]. For individual learning procedures or memes, we consider the i) procedure of Davies, Swann, and Campey with Gram-Schmidt orthogonalization (DSCG) [24], ii) Broyden-Fletcher-Goldfarb-Shanno method (BFGS) [25] and iii) Lagrangian interpolation strategy [24] which are representatives of first and zeroth order exact individual learning methods commonly found in the literature. Note that we have a total of nine potential hybrid global-local search combinations or MA variants.

In addition, we conduct experiments for various configurations of t_{il} and f_{il} where $f_{il} \in [0, 1]$ and t_{il} is some integer value. Here, we limit t_{il} to the set $\{100, 200, 300, 400, 500\}$ in the present study. For each setting of t_{il}, we also experimented for different values of f_{il} at 0.1, 0.3, 0.5, 0.7, 0.9 and 1.0. Further, we examine the effects of three commonly used schemes for selecting the subset of individuals, i.e., Ω_{il}, that will undergo individual learning.

In our study, three commonly used continuous parametric benchmark test problems already extensively discussed in the literature are considered here to study the effects of the diverse MA design issues and their configurations. The benchmark problems used represent classes of Unimodal/Multimodal, Epistatic/Non-Epistatic test functions.

1) Sphere:

$$F_{Sphere} = \sum_{i=1}^{n} x_i^2 \tag{1}$$

Sphere is a smooth and unimodal function. Though it is not a great challenge for most optimization methods in finding the global optimum, Sphere poses as a useful benchmark for evaluating the convergence speed of a search algorithm.

2) Ackley:

$$F_{Ackley} = 20 + e - 20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)) \tag{2}$$

Ackley function has a search landscape containing many local optima with weak epistasis, which makes it difficult for some approaches to optimize.

TABLE I

BENCHMARK FUNCTIONS USED IN THE STUDY

Func	Range	Characteristics	
		Epistasis	Multi-modality
F_{Sphere}	$[-100, 100]^{30}$	no	no
F_{Ackley}	$[-32, 32]^{30}$	yes	yes
$F_{Weierstrass}$	$[-0.5, 0.5]^{30}$	no	yes

TABLE II

MA PARAMETERS SETTING

General parameters	
Population-based methods	GA, DE and ES
Individual learning procedures	DSCG, Lagrange and BFGS
Stopping criteria	$300,000$ evaluations or convergence to global optimum
Population size	50
Genetic Algorithm parameters	
Encoding scheme	Real number
Selection scheme	Roulette wheel
Crossover operator	Two point crossover $p_c = 0.7$
Mutation operator	Gaussian mutation $p_m = 0.03$
Differential Evolution parameters	
Crossover probability	$p_c = 0.9$
Evolutionary Strategy parameters	
Selection method	$\mu + \lambda, \mu = 50, \lambda = 100$
Mutation operator	Gaussian mutation
Individual Learning parameters	
Individual learning intensity t_{il}	300

3) Weierstrass:

$$F_{Weierstrass} = \sum_{i=1}^{n}(\sum_{k=0}^{k_{max}}[a^k \cos(2\pi b^k(x_i + 0.5))]) - D\sum_{i=1}^{n}(\sum_{k=0}^{k_{max}}[a^k \cos(2\pi b^k \cdot 0.5)]) \quad (3)$$
$$a = 0.5, b = 3, k_{max} = 20$$

The Weierstrass function also contains a large number of local optima. Hence, first or second order individual learning methods that operate the search based on gradient information generally do not work well on this function due to getting stuck at some local optimum.

For a pictorial view on the fitness landscape of the benchmark functions, the readers are referred to the appendix provided at the end of this paper. Note that here the landscapes of the search problems are also shifted so that the global optima are not at the origin. This also avoids any biased of the search algorithms on exploiting the symmetric property of the benchmark functions. Table I summarizes these functions with their notable characteristics. Note that in this study, we consider the 30-dimensional version of the benchmark functions.

IV. EMPIRICAL RESULTS

In this section, we present the experimental results of various MAs in optimizing the three benchmark functions. For the sake of brevity in our discussion, the numerical results obtained are grouped as three subsections. In the first subsection, the effects of different population based and individual learning procedures are investigated. The second subsection illustrates the effects of individual learning frequency, f_{il} and individual learning intensity t_{il}. Finally, the last subsection considers the effects of individual subset selection schemes used for selecting subset Ω_{il}.

All results presented are the average of 25 independent runs. Each run continues until the global optimum was found (i.e., the fitness function error $< 10^{-8}$) or a maximum of 300,000 function evaluations was reached. For each run, the initial population is sampled randomly within the search range. Table II summarizes the parameter settings of the evolutionary algorithms and local optimizers used in this study.

A. Impact of different Population-Based and Individual Learning procedures on MA

We begin by presenting the results of diverse MA variants with different population based and individual learning procedures. From a survey of the literature, it it worth noting that among the three population based search candidates considered, Genetic Algorithm remains to be most extensively used for forming MAs [19]. Hence it becomes common for the term hybrid-GA to be used interchangeably with Memetic Algorithm in many existing works. Recently, [26] also considered the use of Differential Evolution as one of the population based search method in MAs. Nevertheless, to the best of our knowledge, a smaller number of papers have dealt with the hybridization of evolutionary strategies and local search.

The individual learning candidates used are representatives of first and zeroth order exact individual learning methods commonly found in the literature. BFGS is a Quasi-Newton method, which determine the search directions based on the first order derivation of the fitness functions. Lagrange strategy, on the other hand, uses a second-order interpolation to generate the search points, and generally works better if the fitness surface is quadratic or closed to quadratic. Last but not least, the DSCG approach executes a line search in each dimension independently. This individual learning procedure has been shown to work extremely well on search problems having low epistasis.

Table III presents the optimization results of different multistart individual learning procedures on the benchmark functions. These are used as base-line results with which other MAs may be compared. The average number of evaluation calls incurred by the different variants of MAs to converge at the global optimum of the benchmark functions or the best solution quality obtained are summarized in Figures 1-3. We examine first the results in Figure 1 on the Sphere function. On this function, it is observed that the BFGS procedure outperforms the other two individual learning counterparts, regardless of the population based method used to form the MAs. The Sphere function is a convex, continuous and unimodal function, hence the gradient vector at any decision point would direct the search to the global minimum. Since BFGS makes use of the gradient information in guiding its search, it converges rapidly to the global optimum of the

TABLE III

OPTIMIZATION RESULTS OF MULTISTART INDIVIDUAL LEARNING ON THE BENCHMARK PROBLEMS. FOR SPHERE FUNCTION, THE AVERAGE NUMBER OF EVALUATIONS INCURRED BY THE ALGORITHMS IN LOCATING THE GLOBAL OPTIMUM IS REPORTED INSTEAD.

Func	MS-BFGS	MS-DSCG	MS-Lagrange
F_{Sphere}	(95)	(1300)	(1147)
F_{Ackley}	19.3993	1.37624	2.2336
$F_{Weierstrass}$	49.9721	13.6501	20.3064

Sphere function by simply moving in the negative gradient direction. Hence its search performance is unaffected by the choice of global method used and is capable of converging to the global optimum regardless of the starting point used. This also explains why the results obtained by a stochastic multistart individual learning is competitive to those obtained by MA.

We refer next to the search convergence performances of the MAs on the Ackley function summarized in Figure 2. Since Ackley is a multimodal function, the use of gradient information in BFGS causes it to getting stuck at some local optima hence the results obtained shows that all the MAs employing BFGS as the individual learning procedure fails to locate the global optimum successfully within the allocated computational budget. On the other hand, the functional form of the Ackley function can be easily approximated using a second-order model. As a result, it can be observed in Figure 2 that by means of quadratic approximation, the Lagrange method generally outperforms the other counterparts on Ackley, regardless of the stochastic population based methods used.

On the Weierstrass function, none of the MAs managed to converge to the global optimum within the maximum computational budget allowable. Hence, the best mean fitness values attained by each MA across 25 runs are summarized in Figure 3. The results also indicate that both the BFGS and Lagrange interpolation individual learning procedures do not fare well on this problem. We believe this is likely due to a mismatch between the neighborhood search structures of BFGS and Lagrange interpolation to the fitness landscape of Weierstrass. DSCG, on the other hand, fares best on this problem.

Further, from the results obtained, neither the population based nor individual learning procedure nor a particular synergy of MA display superiority on all the three problems considered. However, it is worth noting that the choice of individual learning procedure appears to be the key of success in the MA search. This greatly highlights the importance of selecting suitable meme for the given optimization problem at hand as discussed extensively in [18], [19]. Overall, DSCG works generally better on all three problems considered; a conclusion also obtained in [18] and [19]. The choice of population based method in MA, on the other hand, appears to have little significance on convergence speed and solution quality for the three benchmark problem considered.

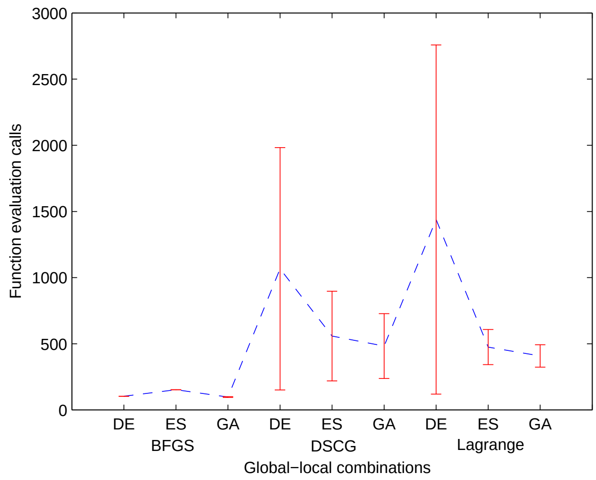

Fig. 1. Search performance of different population based-individual learning or global-local MAs on Sphere function

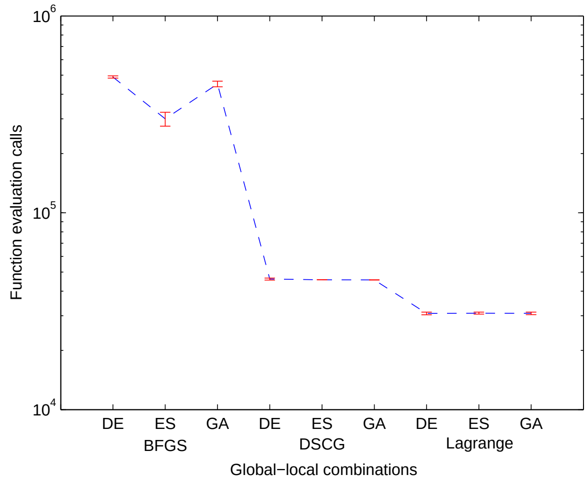

Fig. 2. Search performance of different population based-individual learning or global-local MAs on Ackley function

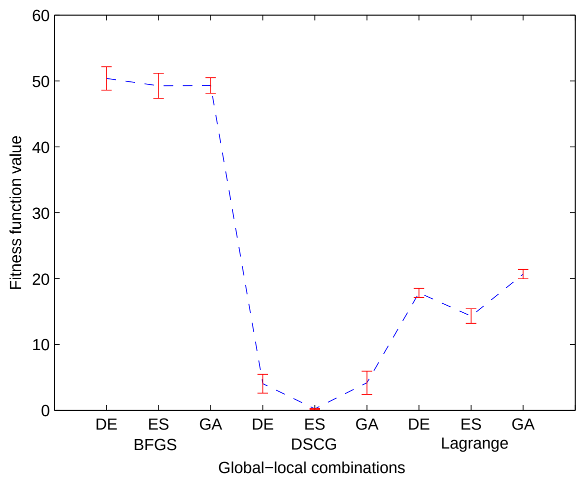

Fig. 3. Search performance of different population based-individual learning or global-local MAs on Weierstrass function

B. Impact of Individual Learning frequency and intensity on MA

Based on the results presented in the previous subsection, the MA based on a synergy of ES and DSCG or otherwise labelled in this paper as ES-DSCG is observed to obtain highest average ranking on the three benchmark functions. As a result, we will consider the ES-DSCG for studying the other design issues in the rest of this study.

In this subsection, we study the effects of individual learning intensity t_{il} and individual learning frequency f_{il} on MA search performance. t_{il}, defines the maximum computational budget allowable to each individual learning procedure. f_{il}, on the other hand, defines the proportion of individuals in each population that will undergo individual learning. Hence for a f_{il} configuration of 0.5, only half of the MA population undergoes individual learning.

A larger value of t_{il} gives more computational budget or greater emphasis on improving each individual chromosomes, thus leading to higher level of precision or accuracy in the solution quality. Similarly, with large f_{il}, i.e. $f_{il} \to 1$, more individuals in the current population will have the opportunity to undergo individual improvement, giving a higher chance of reaching the local or global optimum. In practice, however, the maximum computational budget allowable for an MA search is often limited. Hence even though more intense individual learning, i.e., a large t_{il} or f_{il}, provides greater chance of convergence to high precisions at the local or global optimum, the amount of evolution that may be expended without incurring excessive computational resources becomes limited. As a result of the scarce computational resources available in practice, the MA could fail to hit the region or basin where the global optimum lies before the potential of individual learning could start to bite.

Figures 4 - 6 present the search performance of ES-DSCG on the Sphere, Ackley and Weierstrass problems, for different combinations of t_{il} and f_{il} configurations, i.e., $t_{il} \in \{100, 200, 300, 400, 500\}$ and $f_{il} \in \{0.1, 0.3, 0.5, 0.7, 0.9, 1.0\}$. Note that in the experiments conducted, only the elite ($f_{il} \times po$) chromosomes of the current population will undergo individual learning or individual learning improvement (po is the MA population size). Figures 4 and 5 indicate that all MAs converged to the global optimum of the Sphere and Ackley problems, hence only the number of evaluations incurred have been presented. On the Weierstrass problem, none of the MA converged to the global optimum, thus the average best fitness values obtained across 25 independent runs are reported instead. From these figures, one can also observed that all the MA variants investigated performs poorly on small values of individual learning intensity, i.e., $t_{il} = 100$ or 200. Clearly, this is the result of lack of sufficient computational budget provided for individual learning to generate any positive impact on search.

On Sphere function, t_{il} at 300 evaluations looks sufficient for generating the superior search performance among the MAs. Any further increase in t_{il} actually results in detrimental effects on the MA search performances, see Figure 4. Further, it makes good sense that MA performs better with a smaller f_{il} since any starting point in the search space will lead the MA to the global optimum, especially if sufficient computational budget, t_{il}, is provided.

Next, we examine the design issue of MA on Ackley function with Figure 5 reflecting the trade-off between t_{il} and f_{il}. When f_{il} is large, i.e., approaches 1.0, MAs with $t_{il} = 300$ or 400 perform more efficiently over 500 evaluations. In contrast, MAs at $t_{il} = 500$ fares better than those counterparts with $t_{il} = 300$ or 400 and low f_{il} configurations. Clearly, this agrees with our earlier hypothesis that under some given fixed computational budget, a good balance between t_{il} and f_{il} is necessary to ensure superior search performance in the MA.

On the Weierstrass function, the results summarized in Figure 6 indicate the trends of improving solution quality with increasing individual learning intensity. On such a problem, the results indicate that one should provide a large individual learning intensity for maximum solution quality and search efficiency. It is also worth noting that the $f_{il} = 0.5$ works best over the range of different t_{il} which implies that for complex problems, it may be appropriate to undergo individual learning on half of the MA population.

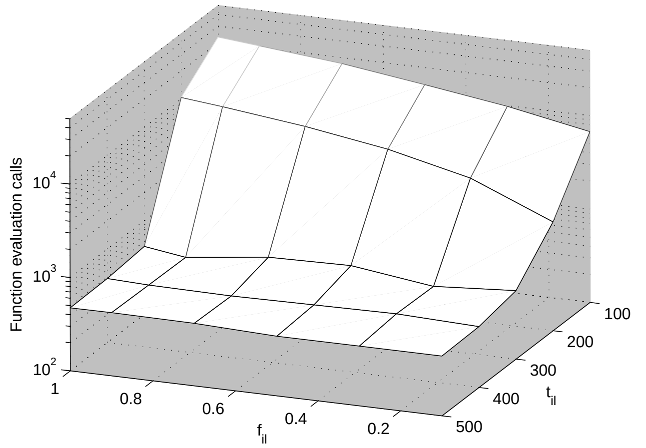

Fig. 4. Search performance of MA with different configurations of (f_{il}, t_{il}) on Sphere function

C. Impact of individual subset selection scheme, Ω_{il}, on MA

In this section, we examine three schemes for selecting the subset of chromosomes in the EA or MA that would undergo individual learning. The "Random-walk" scheme provides all chromosomes in each population equal chances of undergoing individual learning. In contrast, the "best" scheme assumes that fitter chromosomes have better chances of converging to the global optimum. Last but not least, the "stratified" scheme is a compromise between the two where individual learning is only performed on individuals that are regarded as unique. The uniqueness of individuals in a MA population may be measured by various means and may

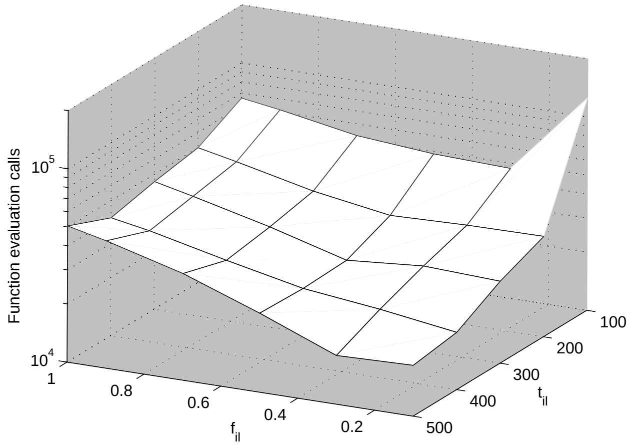

Fig. 5. Search performance of MA with different configurations of (f_{il}, t_{il}) on Ackley function

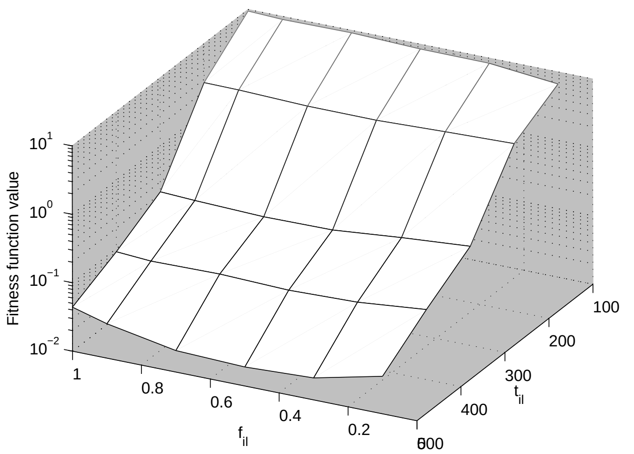

Fig. 6. Search performance of MA with different configurations of (f_{il}, t_{il}) on Weierstrass function

be categorized into those means for genotype, phenotype or fitness level. Here we considered uniqueness or the diversity based on fitness values. Table IV provides a summary of these schemes.

In studying the different selection schemes, we kept the individual learning frequency fixed at 0.5 in the experiments, i.e., half of the MA population undergoes individual learning. On the other hand, we conduct further experiments for $t_{il} =$ 200, 300 or 400 to study the relative impact of the selection scheme and t_{il} on MA search performance.

Figures 7 - 9 summarize the search performance of the MAs for various selection schemes on optimizing the benchmark problems. Figure 7 indicates that the stratified subset selection scheme works best on the Sphere function. The result appears to contradict our initial hypothesis that the "best" scheme would perform superior over the other two counterpart schemes since on the unimodal Sphere function

TABLE IV

CHROMOSOME SELECTION SCHEMES

Scheme	Description
Random-Walk(rand)	Chromosomes undergoing individual improvement are randomly chosen from the current population using uniform sampling without replacement
Stratified(strat)	The population is sorted according to their fitness values. Subsequently, the individuals are divided into n equal range where n is the number of individuals to be chosen to undergo individual learning. Next, from each range, one individual will be selected.
Best(best)	The best n individuals will undergo individual learning.

(Figure 10 in Appendix), every candidate solution should converge precisely to global optimum if sufficient individual learning intensity or computational budget is provided. Upon greater analysis, it is realized that this is due to the search structure of DSCG that does not exploit gradient information in its search. Hence a starting point that is nearer to the global optimum in a quadratic sense does not translate to faster convergence in an MA based on ES-DSCG. On the Weierstrass function, the results in Figure 9 display little difference in performances for different selection schemes. On the other hand, the "best" scheme significantly outperforms all others on the Ackley function, see Figure 8. Across the three benchmark problems considered, the choice of selection scheme in the MA appears to have greatest impact on Ackley function. The results in the figures also indicate that the selection scheme used has little impact on the relative performances of the MA for different individual learning intensities. Hence, the configuration of individual learning intensity is generally unaffected by the choice of selection scheme in the MA.

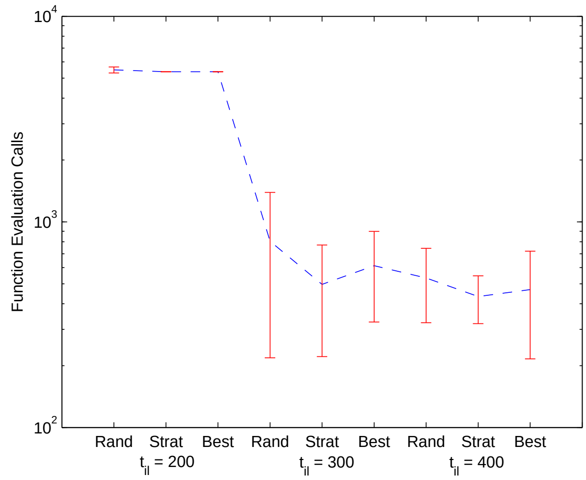

Fig. 7. Search performance of MA for diverse individual subset selection schemes and t_{il} on Sphere function

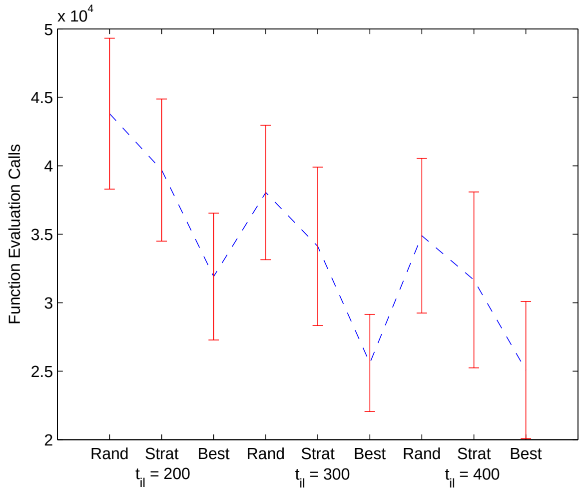

Fig. 8. Search performance of MA for diverse individual subset selection schemes and t_{il} on Ackley function

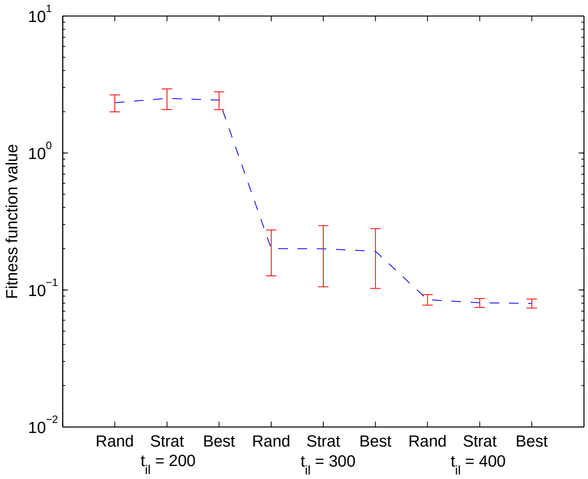

Fig. 9. Search performance of MA for diverse individual subset selection schemes and t_{il} on Weierstrass function

V. Conclusions

In this paper we have discussed several important design issues of Memetic Algorithms and presented the results obtained from a systematic study on both the impact of each individual design issue as well as their relative impact on memetic search performances using three commonly used benchmark functions. From the empirical results, we analyzed the behaviors of MAs and discussed why some synergies of stochastic population-based and individual learning optimizers led to successful Memetic Algorithms while some did not. It was shown that the choice of suitable meme for the given optimization problem at hand was more important compared to the choice of population based search when designing MA. The MA search performance is also greatly affected by the configurations of individual learning intensity and frequency. Further, a good balance between t_{il} and f_{il} must be maintained to ensure good efficiency in the MA search algorithm, under some fixed computational budget. In addition, it remains inconclusive which selection scheme works best for a problem at hand. Nevertheless, both the "stratified" and "best" scheme appeared to generate good and robust MA search performances. Last but not least, the configurations of individual learning intensity is generally unaffected by the choice of selection scheme. Finally, it is hoped that the present study would help to enhance the understandings on MA search and aid the research communities in their design of successful MAs that are appropriate for new problem domains.

Acknowledgement

This work has been funded in part under the EPSRC grants EP/D021847/1 and EP/D061571/1.

Appendix

3D plots of the two dimensional benchmark functions used in the present study

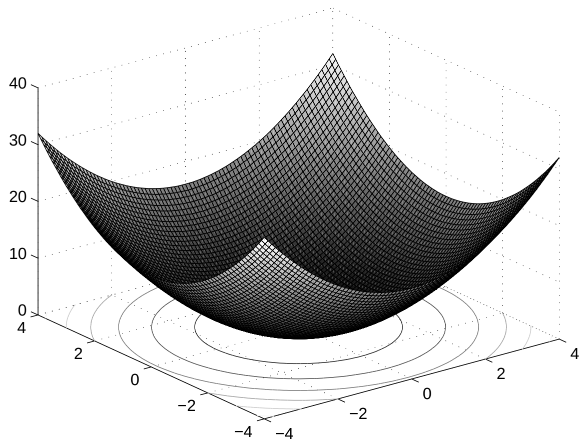

Fig. 10. 2-dimensional Sphere function

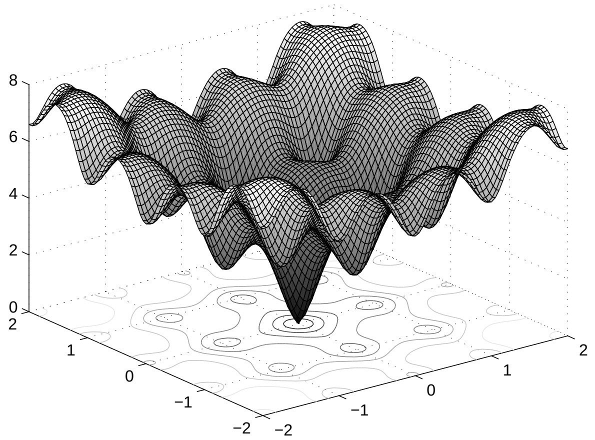

Fig. 11. 2-dimensional Ackley function

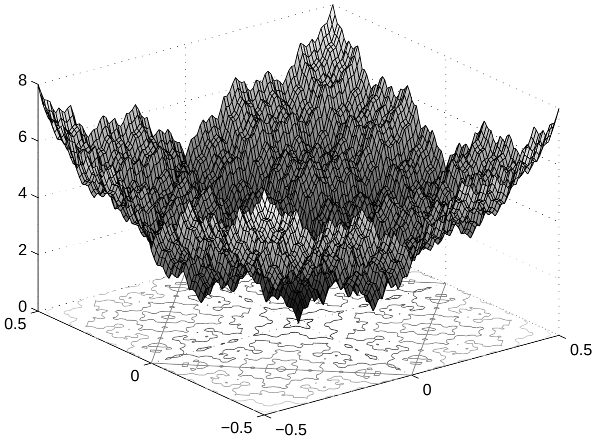

Fig. 12. 2-dimensional Weierstrass function

REFERENCES

[1] T. Back, F. Hoffmeister, and H. Schwefel. Survey of Evolution Strategies. *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 2–9, 1991.

[2] L. Davis et al. *Handbook of genetic algorithms*. Van Nostrand Reinhold New York, 1991.

[3] S. Kirkpatrick, C.D. Gelatt Jr, and M.P. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671, 1983.

[4] Z. Michalewicz. *Genetic algorithms+ data structures= evolution programs*. Springer-Verlag London, UK, 1996.

[5] P. Moscato. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. *Caltech Concurrent Computation Program, C3P Report*, 826, 1989.

[6] Y.S. Ong, P.B. Nair, and A.J. Keane. Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling. *AIAA Journal*, 41(4):687–696, 2003.

[7] Y.S. Ong, P.B. Nair, and K.Y. Lum. Max-Min Surrogate-Assisted Evolutionary Algorithm for Robust Aerodynamic Design. *IEEE Transactions on Evolutionary Computation*, 10(4):392–404, 2006.

[8] M.H. Lim, Y. Yu, and S. Omatu. Extensive testing of a hybrid genetic algorithm for solving quadratic assignment problems. *Computational Optimization and Applications*, 23:47–64, 2002.

[9] J.H. Li, M.H. Lim, and Q. Cao. A qos-tunable scheme for atm cell scheduling using evolutionary fuzzy system. *Applied Intelligence*, 23(3):207–218, 2005.

[10] B.H. Gwee and M.H. Lim. An evolution search algorithm for solving n-queen problems. *International Journal of Computer Applications in Technology*, 24(1), 2005.

[11] M.H. Lim and Y. L. Xu. Application of hybrid genetic algorithm in supply chain management. *Special issue on Multi-Objective Evolution: Theory and Applications, International Journal of Computers, Systems, and Signals*, 6(1), 2005.

[12] Y.S. Ong, P.B. Nair, A.J. Keane, and K.W. Wong. Surrogate-Assisted Evolutionary Optimization Frameworks for High-Fidelity Engineering Design Problems. *Knowledge Incorporation in Evolutionary Computation, Studies in Fuzziness and Soft Computing Series. Springer Verlag*, 2004.

[13] J. Tang, M.H. Lim, and Y.S. Ong. Parallel Memetic Algorithm with Selective Local Search for Large Scale Quadratic Assignment. *Intl Journal of Innovative Computing, Information and Control*, 2(6):1399–1416, Dec 2006.

[14] J. Tang, M.H. Lim, and Y.S. Ong. Diversity-Adaptive Parallel Memetic Algorithm for Solving Large Scale Combinatorial Optimization Problems. *Soft Computing Journal*, 11(9):873–888, July 2007.

[15] N. Krasnogor and J. Smith. A tutorial for competent memetic algorithms: model, taxonomy, and design issues. *IEEE Transactions on Evolutionary Computation*, 9(5):474–488, 2005.

[16] K.C. Tan, Y.H. Chew, and L.H. Lee. A hybrid multiobjective evolutionary algorithm for solving vehicle routing problem with time windows. *Computational Optimization and Applications*, 34(1):115–151, 2006.

[17] M.H. Lim and W. Ng. Iterative genetic algorithm for learning efficient fuzzy rule set. *Artif. Intell. Eng. Des. Anal. Manuf.*, 17(4):335–347, 2003.

[18] Y.S. Ong, M.H. Lim, N. Zhu, and K.W. Wong. Classification of Adaptive Memetic Algorithms: A Comparative Study. *IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics.*, 36(1):141, 2006.

[19] Y.S. Ong and A.J. Keane. Meta-Lamarckian learning in memetic algorithms. *IEEE Transactions on Evolutionary Computation*, 8(2):99–110, 2004.

[20] N. Krasnogor, J. Smith, et al. A memetic algorithm with self-adaptive local search: TSP as a case study. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*, pages 987–994, 2000.

[21] W.E. Hart. *Adaptive Global Optimization with Local Search*. PhD thesis, University of California, 1994.

[22] D.E. Goldberg and S. Voessner. Optimizing Global-Local Search Hybrids. *Urbana*, 51:61801.

[23] R. Storn and K. Price. Differential Evolution–A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*, 11(4):341–359, 1997.

[24] H.P. Schwefel. *Evolution and optimum seeking*. Wiley New York, 1995.

[25] C. Zhu, R.H. Byrd, P. Lu, and J. Nocedal. Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software (TOMS)*, 23(4):550–560, 1997.

[26] Y.C. Lin, K.S. Hwang, and F.S. Wang. Hybrid differential evolution with multiplier updating method for nonlinear constrained optimization problems. *Proceedings of the Congress on Evolutionary Computation (CEC)*, 2002.