Global Journal of Advanced Engineering Technologies and Sciences

PERFORMANCE ANALYSIS OF GENETIC OPERATORS ON TEST FUNCTIONS OF SINGLE OBJECTIVE OPTIMIZATION PROBLEMS

Dr.VishnuRaja Palanisamy
Department of Computer Science and Engineering, Kongu Engineering College, Erode, Tamilnadu, India.

Abstract
Multi Clustered Parallel Genetic Algorithm is a type of multi population based genetic algorithm which gives equal importance to low fit individuals. It has been applied to 0/1 knapsack problem and found to perform well compared to the Standard Genetic Algorithm. This paper explores the working principle of multi clustered parallel genetic algorithm for the standard test functions for the single objective optimization problems and compared with the standard genetic algorithm. The performance is compared with the standard genetic algorithm, the standard test functions of single objective optimization problems are used and the result shows that proposed method performs better with convergence velocity.

I. INTRODUCTION

On the notions of genetics and natural evolution, Genetic Algorithms (GA) (Holland 1992) is the stochastic search algorithms. "Survival of the fittest", the Darwin's theory is applied in the search space to direct towards the search process from the randomized initialization to a more prospective direction in the search space, which is very large. In the process of searching a solution in the search space, a number of genetic operators are applied to help the process of investigation.

This paper focus on both standard genetic algorithm and multi clustered parallel genetic algorithm. Multi clustered parallel genetic algorithm works under the principle of "Birds of the same feather flock together". Based on the type of the application the standard selection mechanisms like roulette wheel selection, tournament selection and rank based selection mechanisms are used in standard genetic algorithm. All these standard selection mechanisms aim to select high fit individuals in different proportion to perform genetic operations like crossover and mutation. The low fit individuals are given less chance to perform genetic operations thus the diversity of the population is reduced. But if the chance is given to the low fit individuals, they may produce good chromosomes in the further generations. This fact is given more importance in the multi clustered parallel genetic algorithm.

In Multi Clustered Parallel Genetic Algorithm (MCPGA), the initial population are grouped into various groups based on the fitness value. Individuals in each group mate with each other to produce good chromosomes. In a group, if any chromosome comes up with better fitness value, it leaves the group and combines with the group which has the similar fitness value. The selection mechanisms followed in MCPGA insists on recombination within the same group thus providing equal chances for mating to the group with lower fitness value. This allows the multi clustered parallel genetic algorithm to maintain the diversity of individuals in the population.

The performance of the MCPGA is proved by applying the standard test functions of single objective optimization problem for implementation. The same test functions are implemented with standard genetic algorithm and the results are compared in terms of convergence velocity and the profit obtained.

II. LITERATURE REVIEW

Due to its faster convergence, Multi population Genetic Algorithms (MGA) are popular, since each group evolves independent of each other. MGAs reduce the number of generations to find the best optimal solution or the near optimal solution. It is also more defiant to premature convergence.

In [Petty et al, 1987], after each generation migration takes place and a copy of the best individual in each group is transferred to the neighbour group. 5 sub-population was used by Grosso in 1985 and the individuals were exchanged with fixed migration rate. Chaotic migration strategy [Chen et al, 2004] was implemented in MGA which employs the asynchronous migration of individuals during its parallel evolution.

In each individual the mating tag has been added , [Booker ,1982] and [Goldberg, 1991]. The tag must match before a cross is permitted. Migration was implemented earlier to maintain the diversity of individuals in the initial population [Rebaudengo and Reordo, 1993 and Power et al, 2005]. Where as in genitor II by Whitely in 1988, the parallel GA, where the individuals migrate from one processor to another.

III. MATERIALS AND METHODS

Multi Clustered Parallel Genetic Algorithm

Multi Clustered Parallel Genetic Algorithm (MCPGA)[Vishnu Raja and Murali Bhaskaran, 2012] is proposed with an objective to reduce the selection pressure of the global search space. In order to improve the performance of the GA, the algorithm is tuned in such a way the entire population is divided into several subpopulations and executed simultaneously. For effective grouping of the population, ranking of chromosome is done.

IV. Methodology

Initially the GA is started with a set of populations generated by random as its initial population. In MCPGA, based on the fitness evaluation the entire population is divided into several sub populations called clusters. Initial clusters are formed at random. The individuals present in a cluster can mate with each other to produce new offspring. Parent selection mechanisms, the genetic operators (crossover and mutation) are applied within the group. The offspring produced from each group will have different fitness values. In the next iteration, again ranking of chromosome is done after the fitness evaluation. And the algorithm continues till the termination condition is satisfied. The individuals migrate to other clusters based on the fitness value of each individual.

The important property that makes the Multi Clustered Parallel Genetic Algorithm works better is the parallel implementation of multi population individuals. The parallel implementation of genetic operators like crossover and mutation in each group is done to improve the performance of the algorithm. In each generation, the worst individuals also gain the profit and migrate with the groups. This migration between the groups helps the worst individual to gain the fitness value after crossover and mutation process to retain the individual survival in the competition. When a chance is given, sometimes even the worst individuals can contribute towards the final solution. The pseudo code of MCPGA is given below:

The pseudo code of Multi Clustered Parallel Genetic Algorithm is given below

1. [Initialization]
   Generate the initial population by random.
2. [Fitness Evaluation]
   Calculate the fitness value of each individual in the population.
3. [Grouping]
   Sort the individuals based on the fitness function.
   Based on the fitness value arrange the population into groups.
4. [Breeding]
   Each group will have individual population. For each group perform the following steps
   - Select the parents from the population using selection mechanisms.
   - Mate the parents to produce new offsprings.
   - Mutate the new offsprings.
   - Calculate the fitness of offspring.
   - Replace the offspring to the same group.
5. [Migration]
   Combine the groups into a single population.
   Calculate the fitness value for all the individuals.
   Sort the individuals based on the fitness value.
6. [Termination]
   Repeat the process from step -3 till the termination condition is reached.
   Select the best solution from the current population.

In MCPGA, the cluster size remains constant. Hence, without disturbing the cluster size the migration can be achieved between the groups to sort individuals from all the groups and grouping them again based on the fitness value. Since the cluster size is constant, the number of individuals in each cluster remains the same after migration. If one individual enters into a cluster based on the fitness value say, cluster 1 to cluster 2, then another individual from cluster 2 has to migrate to some other cluster based on the fitness value. By migration the individuals, it has the high fitness value remains in the first cluster and the individuals with low fitness value remains in the last cluster.

V. Benefits of MCPGA

From the process of MCPGA, notable benefits are observed.

- The selection pressure is reduced. All the individuals in the population of a cluster are given chance to mate with each other in the same group.

- Since the groups are formed based on the fitness value. It is not necessary to worry about the mate between the worst individuals, if they produce best individuals they migrate with some other cluster.
- The convergence is quicker compared to the single population GA.
- It is very simple to use.

It can be applied to any problems of any domain.

Single objective Optimization Test Functions

In Computational science, the optimization is to find the best solution to a problem. The objective chosen in this paper is a set of test functions of single objective optimization problems. These test functions are generally used for evaluating the performance of the evolutionary algorithms. The important consideration for the evaluation of the algorithm is to identify the problems where the performance is better. This will help to formulate the test set for which the algorithm should be evaluated.

The adequate test set was designed by Eiben and Back. This test set has well characterized test functions that allow us to obtain and generalize as far as possible. A function F(x) is multi model if it has two or more local optima [Back, 1996]. A function of p variables is separable if it can be rewritten as a sum of functions of just one variable.

Separable functions are optimized for each variable where as the non separable functions are more difficult to optimize as the accurate search a direction depends on one or more genes. The optimization problem is more difficult if the function F(x) is an multi model function. The search process must be able to avoid the regions around local minima in order to approximate, as far as possible, the global optimum. The most complex case appears when the local optima are randomly distributed in the search space.

The goal of any optimization function is to find the best possible solutions x* from a set X according to a set of criteria F = {f1, f2, . . . fn}. These criteria are called objective functions expressed in the form of mathematical functions. An objective function is a mathematical function $f : D \subset R^n \rightarrow R$ subject to additional constraints. The set D is referred to as the set of feasible points in a search space. In the case of optimizing a single criterion f, an optimum is either its maximum or minimum. The global optimization problems are often defined as minimization problems, however, these problems can be easily converted to maximization problems by negating f. A general global optimum problem can be defined as follows:

$$\textit{minimize } f(x)$$

The true optimal solution of an optimization problem may be a set of $x^* \in D$ of all optimal points in D, rather than a single minimum or maximum value in some cases. There could be multiple, even an infinite number of optimal solutions, depending on the domain of the search space. The task of any good global optimization algorithm is to find globally optimal or at least sub-optimal solutions. The objective functions could be characterized as continuous, discontinuous, linear, non-linear, convex, non-convex, unimodal, multimodal, separable and non-separable.

Following are the test functions taken up for the experimentation.

- Ackleys Function
- Sphere Function
- Rosenbrock function
- Matyas Function
- Booths Function

Ackleys Function

Ackley is a Continuous, Differentiable, Non-Separable, Scalable, Multimodal function which was first proposed by Ackley and the generalized figure was given by Back. The function has an exponential term that covers its surface with numerous local minima. The function definition is as follows:

$$f(x) = \sum_{n=1}^{D} \left(e^{-0.2} \sqrt{(x_i^2 + x_{i+1}^2\,)} + 3(\cos(2x_i) + \sin(2x_{i+1})) \right)$$

Subject to $-35 \le x_i \le 35$. It is a highly multi model function with two global minimum close to the origin.

x=f({−1.479252,−0.739807}, 1.479252,−0.739807}),

$f(x^*) = -3.917275$.

Sphere Function

Sphere function is a Continuous, Differentiable, Separable, Scalable, Multimodal function and has been used in the development of the theory of evolutionary strategies. De jong used sphere function as a test function for the evaluation of genetic algorithms.

$$f(x) = \sum_{i=1}^{p} x_i^2$$

subject to $0 \le xi \le 10$. The global minima is

located x* = f(0, … , 0),
f(x*) = 0.

Rosenbrock function

Rosenbrock function is a Continuous, Differentiable, Non-Separable, Scalable, Unimodal function. is a non-convex function used as a performance test problem for optimization algorithms introduced by Howard H. Rosenbrock.

$$f(x) = \sum_{i=1}^{D-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$$

subject to $-30 \le xi \le 30$. The global minima is located at x* = f(1, ... , 1),

f(x*)=0.

Matyas Function

Matyas function is a Continuous, Differentiable, Non-Separable, Non-Scalable, Unimodal function and has no local minima except the global one. This function is used as a test function in order to evaluate the performance of optimization algorithms.

$$f(x) = 0.26(x_1^2 + x_2^2) - 0.48x_1x_2$$

subject to $-10 \le x_i \le 10$. The global minimum is located at x* = f(0, 0),

f(x*)=0.

Booths Function

Booths function is a Continuous, Differentiable, Non-separable, Non-Scalable, Unimodal function. This function has several numbers of global minima.

$$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$$

subject to $-10 \le x_i \le 10$. The global minimum is located at x* = f(1, 3),
f(x*) = 0.

VI. RESULTSAND DISCUSSION

Experimental Setup

The chromosomes are represented in the form of double vector. The optimal parameters [10] for MCPGA are

No of Individuals: 200
Selection Mechanism: Tournament Selection
Crossover Type: Uniform Crossover
Crossover Rate: 0.90
Mutation Type: Flip Bit Mutation
Mutation Rate: 0.20
Group Size: 4

The performance of MCPGA was compared with standard genetic algorithm in several ways. Many numbers of experiments has been carried for the performance analysis of both MCPGA and SGA with fixed problem size.

- Fixed number of generations
- Variable number of generations

The simulations of SGA are done in Matlab and the simulations of MCPGA in Java. Each Test functions were subjected to 15 continuous executions for both SGA and MCPGA and the optimal value and the number of generations taken for the convergence are noted.

Comparison of SGA and MCPGA

The experiments were carried out for continuous 25 executions. The fitness value and the corresponding generations taken to obtain the fitness value for the first 15 executions were shown in the graph.

Figure 1 and 2 shows the fitness value obtained and the number of generations taken in each execution for Ackley Function.

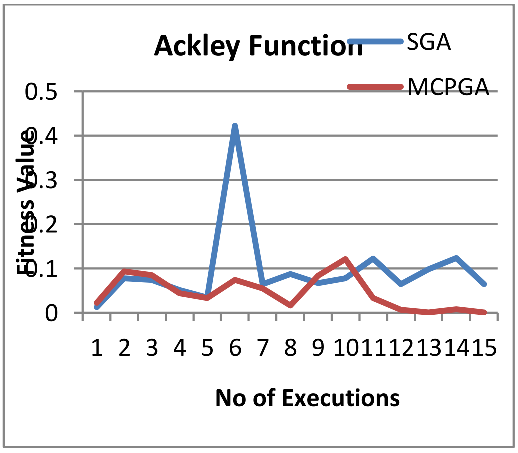

Figure 1: Comparison of Fitness value for Ackley Function.

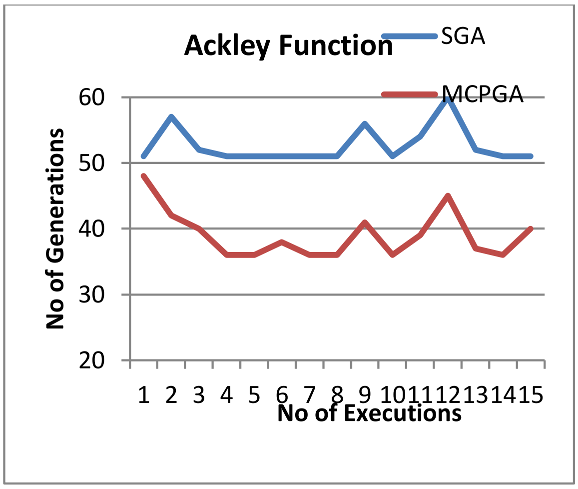

Figure 2: Comparison of Convergence velocity for Ackley Function.

Figure 3 and 4 shows the fitness value obtained and the number of generations taken in each execution for Sphere Function.

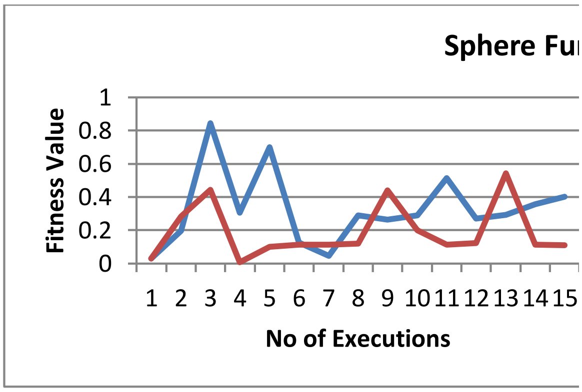

Figure 3: Comparison of Fitness value for Sphere Function.

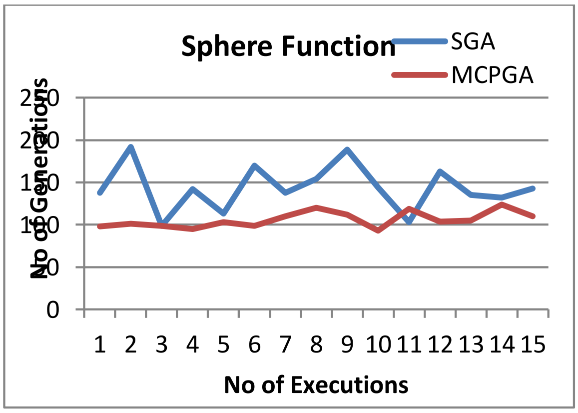

Figure 4: Comparison of Convergence velocity for Sphere Function.

Figure 5 and 6 shows the fitness value obtained and the number of generations taken in each execution for Rosenbrock Function.

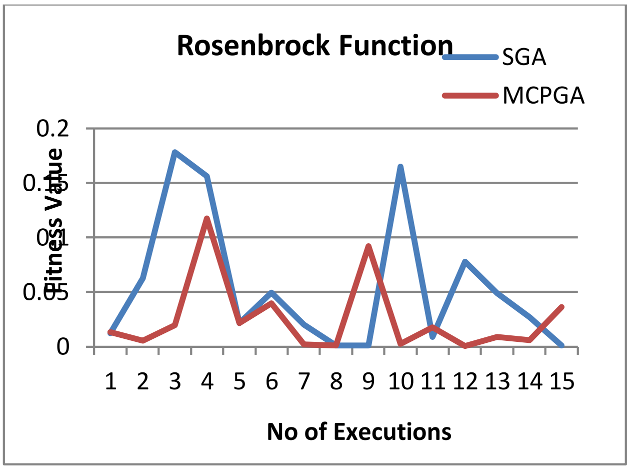

Figure 5: Comparison of Fitness value for Rosenbrock Function.

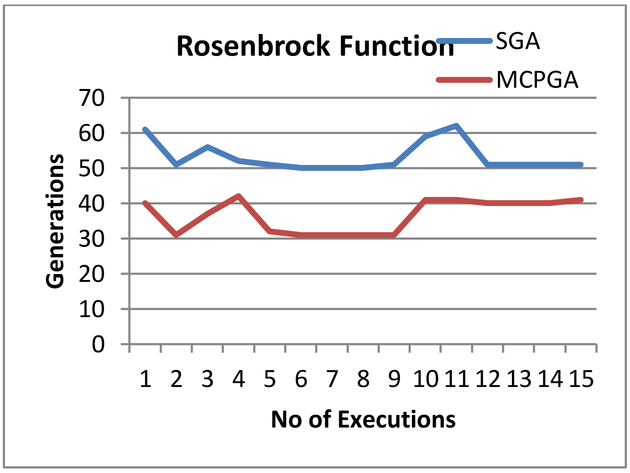

Figure 6: Comparison of Convergence velocity for Rosenbrock Function.

Figure 7 and 8 shows the fitness value obtained and the number of generations taken in each execution for Matyas Function.

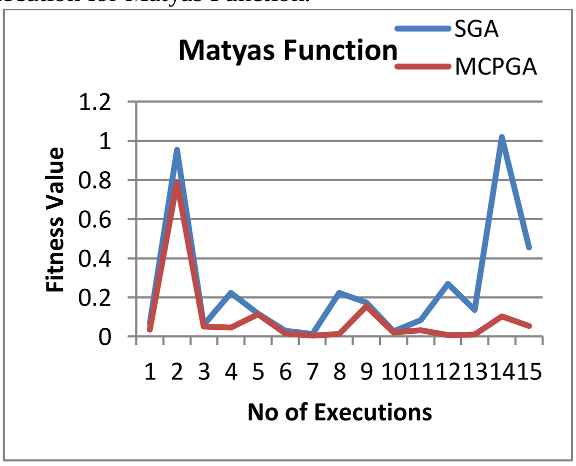

Figure 7: Comparison of Fitness value for Matyas Function.

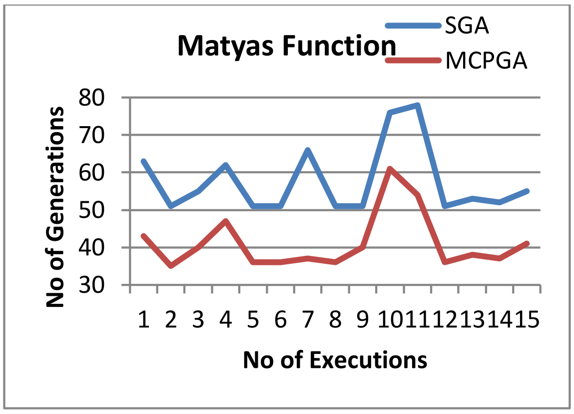

Figure 8: Comparison of Convergence velocity for Matyas Function.

Figure 9 and 10 shows the fitness value obtained and the number of generations taken in each execution for Booths Function.

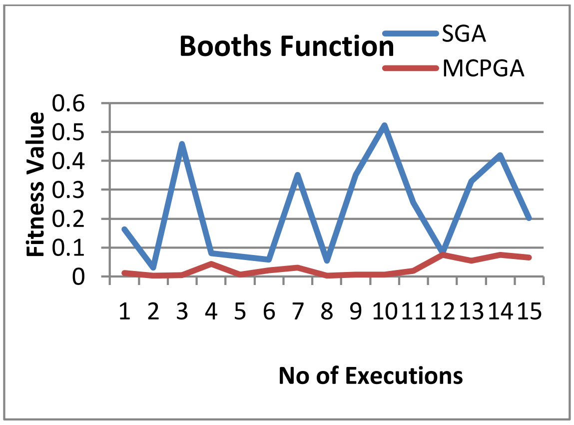

Figure 9: Comparison of Fitness value for Booths Function.

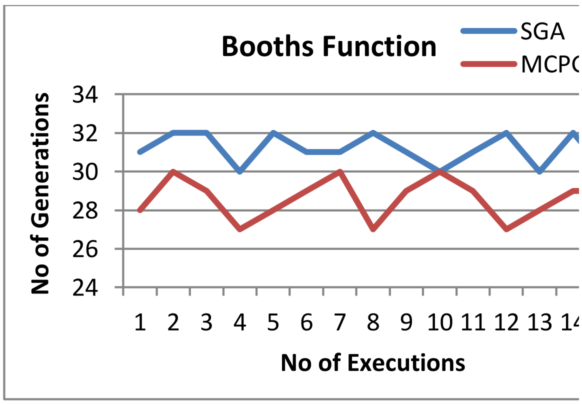

Figure 10: Comparison of Convergence velocity for Booths Function.

On implementing the above mentioned single objective test functions, it is clear that MCPGA converges faster than the standard genetic algorithm. Due to its independent nature the functions behave differently in terms of the fitness value obtained and the number of generations taken for convergence.

Ackley function almost shows a smooth regular value for MCPGA. The sphere function on implementing in MCPGA gave much more optimised fitness value than SGA. In SGA, the sphere function did not converge to the global optima beyond the limited value.

The figures 5 and 6 of Rosenbrock function show that the values are grouped nearer to the global optimal value. It is identified that there are some high peaks for SGA which is not found for MCPGA. The Matyas function which belongs to the same genere shows the similar characteristics. MCPGA was able to converge faster compared to SGA. Booths Function converged efficiently for SGA, but it is found some variations in few iterations. But MCPGA converged uniformly in a better manner.

VII. CONCLUSION

The results clearly shows that multi population based Multi Clustered Parallel Genetic Algorithm performs well when compared with Standard Genetic Algorithm in terms of producing the good chromosomes with best fitness value in less convergence. The test functions which come under Multi Objective optimization problems are currently under study. The MCPGA offers a good synchronized method for solution to any practical problems with the desirable quality of giving importance to low fit individuals.

REFERENCES

I. Petty.C.B, Leuze.M.R and Grefenstette.J.J., 1987,A Parallel genetic algorithm, in: proceedings of the second international conference on Genetic Algorithms, pp. 155-161.

II. Whitley.D and Keith.J., 1988, Genitor: A Different genetic algorithm, in Proceedings of the Rocky Mountain conference on Artificial Intelligence, pp 118-130.

III. Rebaudengo.M and Reordo.M.S., 1993, An Experimental Analysis of the effects of Migration in parallel genetic algorithms, in: Proceedings of Euromicro workshop on parallel and Distributed Processings, pp.232-238.

IV. Grosso.P.B., 1985, Computer simulations of genetic adaptation: parallel subcomponent interaction in a multilocus model. The University of Michigan.
V. Booker,L.B., 1982, Intelligent behaviour as an adaptation to the task environment. Doctoral Dissertation, University of Michigan.
VI. Back.T., 1996, Evolutional Algorithms in Theory and Practice, Oxford University Press.
VII. Chen.X, Gui.W and Chen.L., 2004, A Multi population based GA on chaotic migration strategy and its applications to inventory programming, in : Proceedings of the 5th world congress of Intelligent control and Automation, pp 2159-2162.
VIII. Power.D, Ryan.C and Azad.M.A., 2005, Promoting diversity using migration strategies in distributed genetic algorithms, in: Proceedings of the IEEE Congress on Evolutionary Computation, Vol.1, pp.1831-1838.
IX. Vishnu Raja.P and Murali Bhaskaran.V., 2012 'Impact of Population Size and Mutation Rate on Multi Clustered Parallel Genetic Algorithm', International Journal of Modern Engineering Research (IJMER), vol.2, no. 3, pp-1096-1098.
X. Vishnu Raja P. and Murali Bhaskaran .V., 2012 'Performance Analysis of Multi Clustered Parallel Genetic Algorithm with Gray Value', American Journal of Applied Sciences (AJAS), Impact Factor : 1.4, vol. 9, no. 8, pp. 1268-1272.
XI. Goldberg D.E and Deb.K., 1991, A Comparative analysis of selection schemes used in genetic algorithms, in: G.J.E Rawlins, Foundation of Genetic Algorithms, Morgan Kaufmann, Los Altos, pp. 69-93.