

IMPROVING PERFORMANCE OF GENETIC ALGORITHMS USING DIVERSE OFFSPRING AND DYNAMIC MUTATION RATE

A dissertation submitted to the
Department of Electrical Engineering, University of Moratuwa
in partial fulfillment of the requirements for the
degree of Master of Science

by

RAJAPAKSA GAMAGE SANJAYA ANURADHA



University of Moratuwa, Sri Lanka.
PERERA
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Supervised by: Prof. Lanka Udawatta

**Department of Electrical Engineering
University of Moratuwa, Sri Lanka**

February 2011

University of Moratuwa



96810

96810

96810



621.3 "11"

621.3(043)

DECLARATION

The work submitted in this dissertation is the result of my own investigation, except where otherwise stated.

It has not already been accepted for any degree, and is also not being concurrently submitted for any other degree.

—
UOM Verified Signature


R.G.S.A. Perera

07 February 2011



University of Moratuwa, Sri Lanka.
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

~~We/I~~ endorse the declaration by the candidate.

UOM Verified Signature

Prof. Lanka Udawatta

CONTENTS

Declaration	i
Abstract	iv
Acknowledgement	v
List of Figures	vi
List of Tables	ix
1. Introduction	1
1.1 Background	1
1.2 Motivation	4
1.3 Scope of the Research	5
2. Genetic Algorithms and Performance Dependency	7
2.1 Genetic Algorithms	7
2.1.1 Features of Genetic Algorithms	7
2.1.2 Major steps of a Genetic Algorithm	8
2.1.3 Initial Population	8
2.1.4 Evaluation and Ranking	9
2.1.5 Selection	11
2.1.6 Crossover	13
2.1.7 Mutation	16
2.1.8 Termination	16
2.2 Performances Dependence	17
3. Solution and Theoretical Development	20
3.1 Proposed Solution	20
3.1.1 Solution	20
3.1.2 Methods and Techniques	21
3.2 Mathematical Derivations	24
3.2.1 Generation of Initial Population	24
3.2.2 Selection and Ranking	25
3.2.3 Crossover	25
3.2.4 Mutation	26
3.2.5 Insertion of Diverse Offspring	27
3.2.7 Fitness of the Result	28

4. Performance Testing of Genetic Algorithms	29
4.1 Performance of Genetic Algorithms	29
4.2 Implementation with Multivariate Systems	29
4.3 General Multivariate Systems	30
4.4 Performance Tests	31
5. Results and Analysis	36
5.1 Results	36
5.1.1 Sine Function	36
5.1.2 Step Function	39
5.1.3 Sphere Function	42
5.2 Comparison of Results	45
5.2.1 Step Function	45
5.2.2 Sphere Function	48
5.2.3 Rastrigin's Function	50
6. Conclusions	54
6.1 Conclusions, Remarks and Discussion	54
6.2 Recommendations for Future Research	54
References	56
Appendix - A: Programs	58
A.1 Program Output (For first generation of Step function)	58
A.2 Program Code for Genetic Algorithm	59
A.3 Program Code for Genetic Algorithm Library	63
Appendix - B: Statistical Analysis	71



Abstract

In this work a Genetic Algorithm coding and a required genetic operation library has been developed with some modifications by introducing dynamic mutation rates and fraction of diverse offspring to increase the searching probability. The improvement was done to the algorithm to automatically select the dynamic mutation rate and fraction of diverse offspring depending on the optimization problem.

The modified genetic algorithm with dynamic mutation and diverse offspring was tested with Sin, Step, Sphere and Rastrigin's benchmark functions. Same benchmark test was done with simple random search and conventional genetic algorithm to compare the performance. Also these results were compared with other researchers' results.

The results show that the genetic algorithm with Dynamic Mutation rates and diverse offspring has better searching performance than the conventional Genetic algorithm and the simple random work especially with high dimensional benchmark functions. It also shows that the risk of convergence to a false local optimum can be reduced by the introduction of diverse offspring to the population of next generation. It shows that the searching performance of a Genetic Algorithm can be significantly improved by increasing the diversity of the population using dynamic mutation rates and appropriate fraction of diverse offspring while conserving the convergence characteristics. Result shows the effectiveness of the proposed algorithm.

Acknowledgement

My heartiest thanks are due first to my supervisor, Professor Lanka Udawatta, for his great insights, perspectives, guidance and sense of humor. My sincere thanks go to the Dr. J.P Karunadasa, Head Department of Electrical engineering and the Dr. D.P. Chandima, the course coordinator of the MSc programme for their kind attention and guidance throughout the course and helping in various ways to clarify the things related to my academic works in time with excellent cooperation and guidance. Sincere gratitude is also extended to the people who serve in the Department of Electrical Engineering.

I should thank friends, colleagues and who have not been mentioned here personally in making this educational process a success.



University of Moratuwa, Sri Lanka
Electronic Theses & Dissertations
www.lib.mrt.ac.lk

Finally, I never can forget to thank to my dearly beloved parents and family members who inspired me to go ahead in my education career.

List of Figures

Figure	Page
2.1 Major steps of a genetic algorithm	9
2.2 Roulette wheel selection	11
2.3 Stochastic Universal Sampling	12
2.4 One Point Crossover	13
2.5 Multipoint Crossover	14
2.6 Uniform Crossover	14
2.7 Effect of Intermediate Recombination	15
2.8 Effect of Line Recombination	15
2.9 Binary Mutation	16
3.1 Basic structure of modified genetic algorithm	20
3.2 Illustration of dynamic mutation rate Varies with the span of search space	21
3.3 Illustration of effect of diverse offspring to convergence characteristics when the better offspring are near the global optimum	22
3.4 Illustration of effect of diverse off springs to convergence characteristics when the better offspring are near a local optimum.	23
4.1 Example of a Complex Multivariable System (Growth rate of a tree)	30

4.2 Illustration of the approach when the objective function is not defined	30
4.3 Graphical 3D representation of the Goldstein-Price Function for the range; $-10 \leq x_1 \leq 10$ and $-10 \leq x_2 \leq 10$	31
4.4 Graph of best fitness vs. number of generations for a tested genetic algorithm.	32
4.5 Graphical representations of the benchmark function with many local peaks with a global maximum.	32
4.6 Graph of Best fitness vs. number of generations for a tested genetic algorithm	33
5.1 Graph of best fitness vs. number of generations for different mutation rated of diverse offspring for Sine function.	37
5.2 Graph of best fitness vs. number of generations for different mutation rated of diverse offspring for Step function.	40
5.3 Graph of best fitness vs. number of generations for different mutation rated of diverse offspring for Sphere function.	44
5.4 Graph of best fitness vs. number of generations (Averaged over 10 runs) with diverse offspring and dynamic mutation rates for Step function.	45
5.5 Graph of best fitness vs. number of function evaluation calls (averaged over 10 runs) for 5D Step function with GA-Seek, Simple Random Walk and GA-Bci	46

5.6 Graph of best fitness vs. number of generations (averaged over 10 runs) with diverse offspring and dynamic mutation rates for Step function with 30 decision variables.	47
5.7 Graph of best fitness vs. number of generations (averaged over 10 runs) with diverse offspring and dynamic mutation rates for Sphere function with 3 decision variables.	48
5.8 Graph of best fitness vs. number of generations for Sphere function with three decision variables with Hybrid genetic algorithm.	49
5.9 Graph of best fitness vs. number of generations (Averaged over 10 runs) with diverse offspring and dynamic mutation rates for Sphere function with 30 decision variables.	50
5.10 Graph of best fitness vs. number of generations for Restraint's function with modified genetic algorithm	51
5.11 Graph of best fitness vs. number of generations for Restraint's function with Hybrid Genetic Algorithm.	51
5.12 Graph of best fitness vs. number of generations for 20D Restraint's function. With diverse offspring and dynamic mutation genetic algorithm	52
5.13 Graph of best fitness vs. number of generations for 20D Restraint's function	53

List of Tables

Table	Page
2.1 Effect of parameter setting on performance of genetic algorithm	17
4.1 Table of bench mark functions used for performance testing of genetic algorithms	34
5.1 Parameters settings used in genetic algorithm for Sin function	36
5.2 Fitness and standard deviation of the results after 0 th , 5 th , 10 th , 15 th , 19 th generations for different settings of genetic algorithm for Sin function	38
5.3 Fitness and standard deviation of the results after 0 th , 10 th , 20 th , 35 th generations for different settings of genetic algorithm for Step function	41
5.4 Fitness and standard deviation of the results after 0 th , 50 th , 100 th , 150 th , 200 th generations for different settings of genetic algorithm for Sphere function	43

