De Jong's Sphere Model Test for A Social-Based Genetic Algorithm (SBGA)

Nagham Azmi AL-Madi, *Ahamad Tajudin Khader*

School of Computer Sciences, Universiti Sains Malaysia (USM), Penang, Malaysia

Summary
In this paper, we used the De Jong's first function 1, "The Sphere Model" to compare values and results concerning the averages and best fits of both, the Simple Standard Genetic Algorithm (SGA), and a new approach of Genetic Algorithms named Social-Based Genetic Algorithm (SBGA). Results from the Sphere Model test on Social-Based Genetic Algorithms (SBGA) are obtained. These results are encouraging in that the Social-Based Genetic Algorithms (SBGA) performs better in finding best fit solutions of generations in different populations than the Simple Standard Genetic Algorithm.

Key words:
Genetic Algorithms (GAs), ,Evolutionary Algorithms (EA), Simple Standard Genetic Algorithms (SGA), Social-Based Genetic Algorithm (SBGA), De Jongs' functions, the Sphere model.

1. Introduction

In the early 1960s and 1970s, new search algorithms were initially proposed by Holland, his colleagues and his students at the University of Michigan. These search algorithms which are based on nature and mimic the mechanism of natural selection were known as Genetic Algorithms (GAs) [1, 3, 5, 6, 7, 8, 9].

Holland in his book "Adaptation in Natural and Artificial Systems" [1] initiated this area of study. Theoretical foundations besides exploring applications were also presented.

As a matter of fact, "Genetic algorithms' functionality is based upon Darwin's theory of evolution through natural and sexual selection." [8]. They mimic biological organisms [5].

In GAs a solution to the problem is represented as a genome (or chromosome) [1, 3, 4, 5, 6]. The population of solutions is initialized by applying the GAs operators such as the crossover and mutation [1, 3, 4, 5, 6]. And with their natural selection they have an iterative procedure usually used to optimize and select the best chromosome (solution) in the population. This population consists of various solutions to hard complex problems and is usually generated randomly [5, 14]. Figure (1) below represents the Simple Standard GA evolution flow.

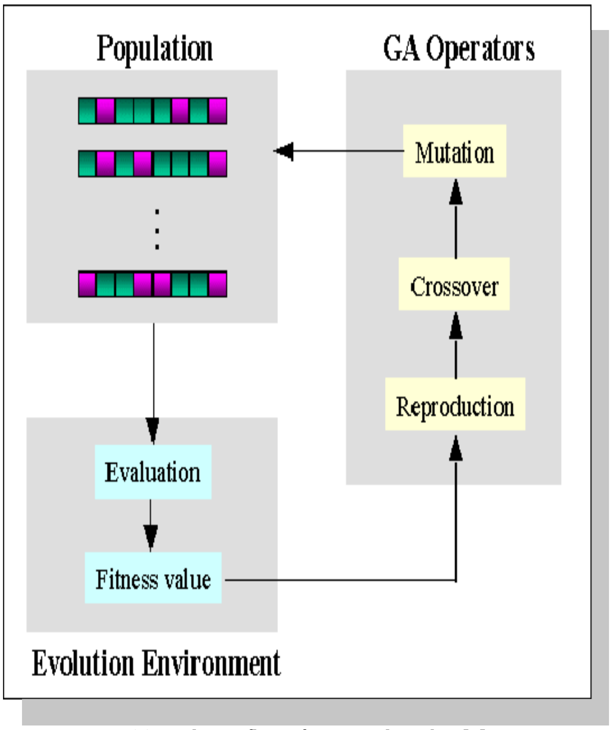

Figure (1) Evolution flow of genetic algorithm [5].

GAs attracted many researchers to search and optimize complex problems. In fact, they proved to be efficient in solving different combinatorial optimization problems. They are considered heuristic search algorithms that solve unconstrained and constrained problems [3]. Many applications use these kinds of algorithms in designing complex devices such as aircraft turbines, integrated circuits and many others, GAs play a main role [3].

Manuscript received March 5, 2008
Manuscript revised March 20, 2008

As a matter of fact, GAs have many advantages in terms of global optimization. On the other hand, from these advantages; potential disadvantages appear [3].

2. Related Work

John Holland, his colleagues and his students explained adaptive processes of natural systems. In order to retain some mechanisms of natural systems, they have designed some kind of artificial system software for this purpose [3, 6].

GA differs from other search algorithms in that it has a unique characteristic [3]. It only needs the input parameters of a certain problem and represent these inputs in a chromosome format. Thus, it is unaware of the problem itself. This is the reason why GAs can be applied to many types of complex problems [1, 3].

Usage of genetic algorithms began by solving academic problems such as the traveling salesman problem and the 8 Queens problem [3, 5, 6, 9]. Years later, GAs grew rapidly. In a way, they increased their applications to optimize complex scheduling problems, spatial layout and many other types that are hard to efficiently maximize [7].

The Simple Standard Genetic algorithm works randomly in selecting parents. In choosing two individuals to mate together there are no constraints [36]. Many studies have been done to tackle this problem trying to overcome it, and trying to design structured population with some control on how individuals interact [36].

From many researches on GAs different types and models of GAs appeared such as Cellular GA [36], Island GA [37], Patchwork GA [38, 39], Terrain-Based GA [40], and religion-Based GA [41]. Below we will discuss some of them briefly.

2.1 Cellular GAs (CGA)

By Gorges-Schleuter, 1989 [36]. It is called a diffusion model. A two-dimensional Grid world is used here to arrange the individuals where these individuals interact with each other by the direct neighborhood of each individual [42]. These individuals will be distributed on a graph which is connected together; each individual connects with its neighborhood by a genetic operator. This type of GAs is designed as a probabilistic cellular automation. A self-organizing schedule is added to reproduce an operator [43]. The individual which can interact with its immediate neighbors can only be held in the cell.

2.2 Terrain-based GA (TBGA)

TBGA showed better performance than the CGA with less parameter tuning [40]. This was discussed in a previous study [36]. At every generation each individual should be processed, and the mating will be selected from the best of four strings, located above, below, left, right.

It is a more self-tuning model compared to cellular genetic algorithm [40]. In which many combination parameter values will be located in different physical locations.

2.3 Island Models

According to the increasing complex problems which appear in evolutionary computation, more advanced models of evolutionary algorithms (EAs) appear. Island models are considered a family of such models [45]. Here the individuals are divided into sections. We call each section a subpopulation which is referred to as an island. These island models are able to solve problems in a better performance than standard models [46, 43]. There is a specific relation between islands through some exchange of some individuals between islands. This process is called migration; this is what island models are famous of, and without these migrations, each island is considered as a set of separate run. Therefore migration is very important [47, 45].

2.4 Patchwork Model

This type was introduced by Krink et al., (1999). A combination of ideas from cellular evolutionary algorithms, island models, and traditional evolutionary algorithms where used in this model [38, 39]. Here the grid is a two dimensional grid of fields, each field can have a fixed number of individuals. The patchwork model is considered a self-organized, spatial population structure [44]. In a GA population, in order to allow self-adaptation, patchwork model is used as a base. It contains a grid world and some interesting agents. In modeling biological systems the patchwork model is considered as a general approach.

2.5 Religion-Based EA Model (RBEA)

It was introduced by Rene Thomsen et al. [44]. The religion-based EA model is based on a part of religious concept which is attracting believers. It attracts new believers to a religion which puts more control than other models such as cellular EA and the patchwork models [41].

3. Social-Based Genetic Algorithm (SBGA)

AL-Madi and Khader [6] presented a new approach for structured population of GAs so-called Social-Based Genetic Algorithm (SBGA). They applied some constraints on the Simple Standard Genetic Algorithm (SGA) in order to control its randomness in selecting parents.

3.1 SBGA Chromosome Representation

According to the Social-Based Genetic Algorithm (SBGA) [6] which is based on nature and social selection, an attribute is given to each individual in the population specifying its sex whether male or female. In addition, being in the same society- as the population is divided into subgroups or islands- is a dependable constraint for recombination. The problem of age is considered also by adding an attribute for the age. The age attribute takes three values: youth, parent, and grandparent. This chromosome representation (the presence of father and mother pointers) will keep all family relations which divides the subgroups into a Directed Acyclic Graph (DAG).
All the standard operations in the GA will be changed in order to add restrictions on each operation including: Social constraints such as the Male/Female '*operator*', this will be added in the selection part which will restrict choosing two different couples. In addition the Birth *operator* which is generating a new population, and the Death *operator* which will discard the worse individuals.

In figure (2) below, the Social-Based Genetic Algorithm (SBGA) model which is a modification of the Simple Standard Genetic Algorithm (SGA) is shown. And all the standard operations in the SGA will be changed in order to add restrictions on each.

3.2 SBGA Method

Initially, the first individual is selected randomly from the population - this will be the first parent. Based on the first parent's type (whether a male or a female), the second parent will be chosen such that it is the opposite type of the first parent. This process is repeated for a number of individuals creating the initial population. Next comes the stages of selection and crossover, bringing up two new children or *offspring's*. Repeating this for a number of couples a second population will be generated.
Again, the previous process is repeated until the maximum number of generations is reached. (The next main important thing is that the *two individuals* must *not* share the same parents).

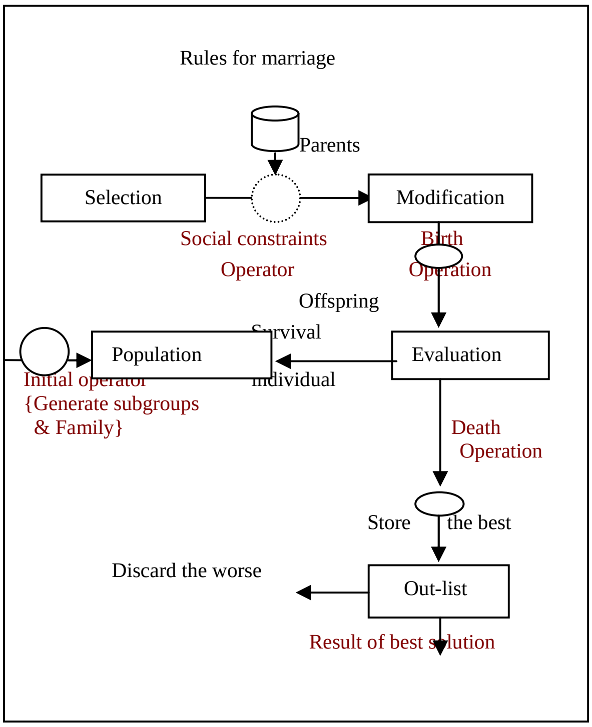

Figure (2): The Social-Based Genetic Algorithm (SBGA) model design "The Simple Standard GA (SGA) modified by new operators" [6]

4. De Jong's functions

De Jong's functions were initially introduced in his thesis entitled "An analysis of the behavior of a class of genetic adaptive systems" [8, 11]. These different functions were used as evaluation functions for the genetic algorithm structure. Many different optimization problems were explained in a novel way using these kinds of functions. This made them the most widely used functions for experimenting Genetic Algorithms functionality and allowing direct comparisons with existing available results [8, 12].

4.1 De Jong's function (1): (The Sphere Model)

De Jong's function no. (1) is considered the easiest and simplest test function among De Jong's other functions [10]. It is also called "The Sphere Model". It is a good example of a continuous, strong convex, unimodel function [9, 10].

The structure of the first functions of De Jong functions is defined as follows:

Function definition:

$$f_1(x) = \sum_{i=1}^{n} x_i^2 \qquad -5.12 \le x_i \le 5.12$$

$f_1(x)$ = sum(x(i)^2), i =1:n, 5.12<=x(i)<=5.12.

Global minimum:

f(x)=0, x(i)=0, i=1:n.

The Sphere model serves as a test case for convergence velocity and is well known and widely used in all fields of evolutionary algorithms occurring in the test sets of Schwefel, De Jong, and Fogel [9, 10]. The three-dimensional topology of the Sphere model which shows the Visualization of De Jong's function (1) is shown in figure (3) below.

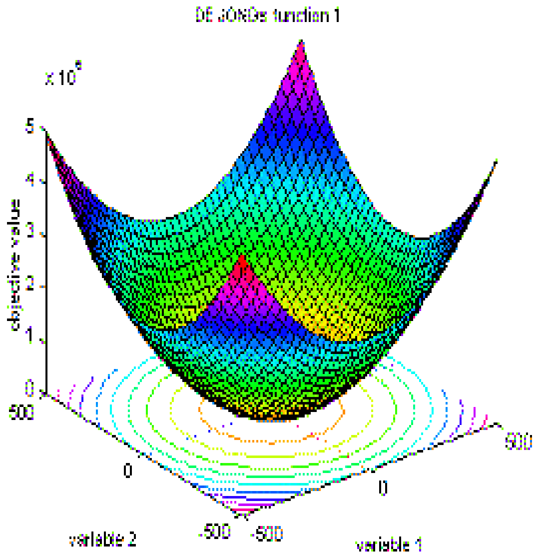

Figure 2.1 the Sphere model in a very large area from -500 to 500, [10].

5. Experimental Results

In this research we have used the first of De Jong's functions - "The Sphere model" to test the Social-Based Genetic Algorithm Model (SBGA) in [6]. We also used it as a test on the Simple Standard Genetic Algorithm (SGA) in order to compare between both algorithms.

A population size of 350 and a randomly selected one-point crossover are used in a process that is both standard and simple [34]. A random integer (crossover point) and a crossover rate of 5% are chosen according to the maximum length of the chromosome in the model. This is the place in the chromosome at which, with probability, the crossover will occur. If the crossover does occur, then the bits up to the random integer of the two chromosomes are swapped. The mutation of a solution is a random change to a gene value [34, 35]. After several experiments of different mutation rates, the most suitable mutation rate is 0.04. The selection method used is the roulette wheel. The number of generations is 100. The implementation part was programmed in C# (C Sharp) Language Version (5.0) on a Pentium 4, HP-Compaq laptop.

By applying the Sphere model on both the Simple Standard Genetic Algorithm (SGA) and on the Social-Based Genetic Algorithm (SBGA) [6] we can compare the performance of both algorithms. The comparisons in figures 3 and 4 below show that the constraints put on the new Social-Based Genetic Algorithm (SBGA) has results in better performance to SBGA than the Simple Standard Genetic Algorithm (SGA) which depends mainly on its randomness in finding the best fit solution.

It is shown that in the Social-Based Genetic Algorithm (SBGA) the average converge toward the optimal solution better than the Simple Standard Genetic Algorithm (SGA), and the best fit values in the Social-Based Genetic algorithm (SBGA) also show better findings of best fit values in comparison to the Simple Standard Genetic Algorithm (SGA).

In the following figures below we can see the comparative results of applying De Jong's function (1) "The Sphere Model" on both the Simple Standard Genetic Algorithm (SGA) and the Social-Based Genetic Algorithm (SBGA) as in [6].

In figures 3 and 4 below it shows that the average of the Social-Based Genetic Algorithm (SBGA) has better performance than the average of the Standard Genetic Algorithm (SGA). In addition, they show better finding of best fit solutions for the Social-Based Genetic

Algorithm (SBGA) than the best fit solutions of the Simple Standard Genetic Algorithm (SGA).

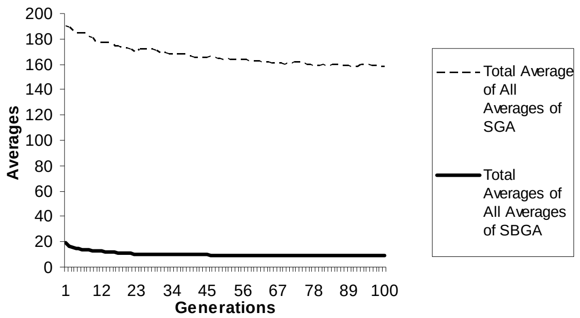

Figure (3) A Comparison between the Total Averages of all averages of 10 Runs each; between the SGA & the SBGA with a Population of size 350 and one-point cross over.

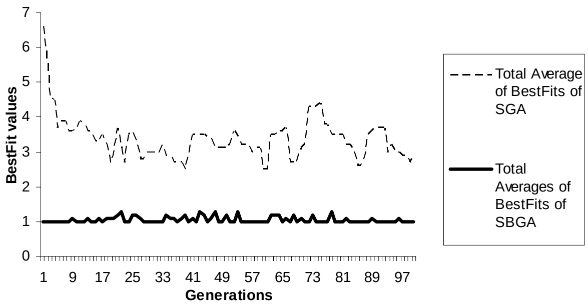

Figure (4) A Comparison between the Total Averages of the best fits of 10 Runs each; between the SGA & the SBGA with a Population of size 350 and one-point cross over.

6. Conclusion

In this paper, a test function of the De Jong's function 1 which is also called "The Sphere Model" is used to evaluate and compare results between the Simple Standard Genetic Algorithm (SGA) and a new approach for structured population of GA called the Social-Based Genetic Algorithm (SBGA) [6].

It is concluded based on the analysis results that the Social-Based Genetic Algorithm (SBGA) is better in terms of best finding as shown in our given results than the Simple Standard Genetic Algorithm (SGA).

The Average of the Social-Based Genetic Algorithm (SBGA) is trying to converge towards the minimum despite its restricted constraints to the best values. In addition, the findings of the best solutions of best fit values are better in the Social-Based Genetic Algorithm (SBGA) than in the Simple Standard Genetic Algorithm (SGA).

Acknowledgment

The authors would like to thank Universiti Sains Malaysia (USM) for its support of this study.

References

[1] J. H. Holland, "Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence" *MIT Press,* Cambridge, MA, USA, 1992.

[2] http*://ai-depot.com*

[3]Y.Zheng, S. Kiyooka,"Genetic Algorithm Applications". www.me.uvic.ca/~zdong/courses/mech620/GA_App.PDF/ . 1999.

[4]www.doc.ic.ac.uk/~nd/surprise_96/journal/vol1/hmw/article 1.html

[5] Y. Liao, and C.T. Sun, "An Educational Genetic Algorithms Learning Tool" ©*2001 IEEE*.

[6] N. A. AL-Madi, A. T. Khader, "A SOCIAL-BASED MODEL FOR GENETIC ALGORITHMS". *Proceedings of the third International Conference on Information Technology (ICIT*), AL-Zaytoonah Univ., Amman, Jordan, 2007.

[7] J. Dalton, "Genetic Algorithms" Newcastle Engineering Design Centre, http://www.edc.ncl.ac.uk. 2007.

[8] E. Bagheri, H. Deldari, 2006. "Dejong Function Optimization by means of a Parallel Approach to Fuzzified Genetic Algorithm" *Proceedings of the 11th IEEE Symposium on Computers and Communications (ISCC'06) 0-7695-2588-1/06 $20.00 © 2006 IEEE*

[9] T. Back. "Evolutionary Algorithms in theory and practice Evolution Strategies, Evolutionary Programming, Genetic Algorithms". *Accessed 2008.*

[10]*http://www.geatbx.com/docu/fcnindex.html #P74_1604.*

[11] K. Dejong, "an analysis of the behavior of a class of genetic adaptive systems" PhD thesis, University of Michigan, 1975

[12] L. Randy, Haupt S., Practical Genetic Algorithms, 2003, Wiley-IEEE Publication.

[13] T. Back. *Evolutionary Algorithms in Theory and Practice*, Oxford, New York, 1996.

[14] B. Bhanu, S. Lee, and J. Ming. Self-optimizing image segmentation system using a genetic algorithm. In R. K. Belew and L. B. Booker, editors, *Proceedings of the Fourth International Conference on Genetic Algorithms and Their Applications*, pages 362-369, San Mateo, CA, July 1991. Morgan Kaufmann.

[15] G. A. Cleveland and S. F. Smith. Using genetic algorithms

to schedule flow shop releases. In J. D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms and Their Applications*, pages 160-169, San Mateo, CA, June 1989. Morgan Kaufmann.

[16] N. L. Cramer, A representation for the adaptive generation of simple sequential programs. In J. J, Grefenstette, ed., *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*. Erlbaum, 1985.

[17] M. Dorigo and U. Schnepf. Genetic-based machine learning and behavior-based robotics: a new syndissertation. *IEEE Transactions on System, Man, and Cybernetics*, SMC-23(1):144-154, 1993

[18] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley & Sons, New York, 1966.

[19] C. Fujiki, and J. Dickinson, Using the genetic algorithm to generate Lisp source code to solve the Prisoner's dilemma. In J. J. Grefenstette, ed., *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*. Erlbaum, 1987.

[20] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Weley, Reading, MA, 1989.

[21] D. E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodal function optimization. In J. J. Grefenstette, editor, *Proceedings of the Second International Conference on Genetic Algorithms and Their Applications*, pages 41-49, Hillsdale, NJ, July 1987. Lawrence Erlbaum Associates.

[22] J. J. Grefenstette. Credit assignment in rule discovery systems based on genetic algorithms. *Machine Learning*, 3(2/3):225-245, 1988.

[23] J. J. Grefenstettte, R. Gopal, B. J. Rosmaita, and D. V. Gucht. Genetic algorithm for the traveling salesman problem. In J. J. Grefenstette, editor, *Proceeding of the First International Conference on Genetic Algorithms and Their Applications*, pages 160-168, Hillsdale, NJ, July 1985. Lawrence Erlbaum Associates.

[24] S. A. Harp, T. Samad, and A. Guha. Towards the genetic syndissertation of neural networks. In J. D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms and Their Applicatins*, pages 360-369, San Mateo, CA , June 1989. Morgan Kaufmann.

[25] K. A. De Jong. Learning with genetic algorithms: an overview. *Machine Learning*, 3(2/3:121-138, 1988.

[26] C. L. Karr. Design of an adaptive fuzzy logic controller using a genetic algorithm. In R. K. Belew and L. B. Booker, editors, *Proceedings of the Fourth International Conference on Genetic Algorithms and Their Applications*, pages 450-457, San Mateo, CA, July 1991. Morgan Kaufmann.

[27] K. Kristinsson and G. A. Dumont. System identification and control using genetic algorithms. *IEEE Transactions on System, Man, and Cybernetics*, SMC-22(5):1033-1046, 1992.

[28] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[29] J. R. Koza, *Genetic Programming II: Automatic Discovery of Reusable Programs*. MIT Press, 1994.

[30] G. G. Miller, P. M. Todd, and S. U. Hegde. Designing neural networks using genetic algorithms. In J. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms and Their Applications*, pages 379-384, San Mateo, CA, June 1989. Morgan Kaufmann.

[31] N. H. Packard, A genetic learning algorithm for the analysis of complex data *Complex Systems* 4, no. 5:543-572, 1990.

[32] G. Syswerda and J. Palmucci. The application of genetic algorithms to resource scheduling. In R. K. Belew and L. B. Booker, editors, *Proceedings of the Fourth International Conference on Genetic Algorithms and Their Applications*, pages 502-508, San Mateo, CA, July 1991. Morgan Kaufmann.

[33] M. Mitchell, *An Introduction to Genetic Algorithms*. MIT Press, 1997.

[34] E. K. Prebys,*The Genetic Algorithm in Computer Science*, MIT Undergraduate Journal of Mathematics, *ee.sharif.ir/~poshtkoohi, accessed 2007.-*

[35] *www.geocities.com*, "A Genetic Knapsack Problem Solver". *Accessed 2007.*

[36] M. Gorges-Schleuter. "ASPARAGOS an asynchronous parallel genetic optimization strategy". In J. D. Schaffer, editor, Proceedings of the Third International Conference on Genetic Algorithms, 1989, pp. 422–427.

[37] Back, T., Fogel, D. B., Michalewicz, Z., Et Al., Eds. Handbook on "Evolutionary Computation". IOP Publishing Ltd and Oxford University Press, 1997.

[38] Krink, T., Mayoh, B. H., and Michalewicz, Z. "A Patchwork model for evolutionary algorithms with structured and variable size populations". In Proceedings of the Genetic and Evolutionary Computation Conference, Vol. 2, 1999, pp. 1321-1328.

[39] Krink, T., and Ursem, R. K. "Parameter control using the agent based patchwork model". In Proceedings of the Second Congress on Evolutionary Computation (CEC-2000). 77-83.

[40] Gorden, V. S., Pirie, R., Wachter, A., and Sharp, S. "Terrain-based genetic algorithm (TBGA): Modeling parameter spaceasterrain".In Proceedings of the Genetic and Evolutionary Computation Conference, vol. 1, 1999, pp. 299- 235.

[41] Rene Thomsen, Peter Rickers and Thiemo Krink. "A Religion-Based Spatial Model For Evolutionary Algorithms". EvAlife Group, Dept. of Computer science, University of Aarhus, Denmark [Online] [Accessed 2006].

[42] Whitley, D. "Cellular genetic algorithms". In Proceedings of the Fifth International Conference on Genetic Algorithms, Forrest S. (ed.). Morgan Kaufmann 1993, pp. 658.

[43] Enrique Alba, Mario Giacobini and M. Tomassini,: "Comparing Synchronous and Asynchronous Cellular Genetic Algorithms". In J. J. Merelo et al., editor, Parallel Problem Solving from Nature – PPSN V11, Springer-Verlag, Heidelberg, 2002, 2439: 601-610.

[44] Ursem, R. K. "Multinational evolutionary algorithms". In Proceedings of the Congress of Evolutionary Computation, Vol. 3, 1999, pp. 1633-1640.

[45] Zbigniew Skolicki, and K. De Jong. "The influence of migration sizes and intervals on island models". In Proceedings of Genetic and Evolutionary Computation Conference – GECCO-2005. ACM Press, 2005, 1295-1302.
[46] Mohamed A. Belal, Iraq H. Khalifa, "A Comparative Study between Swarm Intelligence and Genetic Algorithms", Egyptian Computer Science Journal, Vol. 24, No. 1, 2002.
[47] In: 2002 Water Resources Planning & Management Conference, Roanoke, VA. American Society of Civil Engineers (ASCE) Environmental & Water Resources Institute (EWRI), 2002.

Nagham Azmi AL-Madi received the BSc. degree in Computer Science from the United Arab Emirates University, UAE, in 1991. She obtained her M.Sc. degree in Information Technology from AL-Neelain University, AL-Sudan, in 2003. During 1993-2000 she stayed at AL-Zaytoonah University, Amman, Jordan, as an instructor of Computer Sciences and Information Technology. Then she worked as a M.Sc. lecturer at AL-Zaytoonah University, Amman, Jordan from 2003-2005. Currently she is a Ph.D. student at the School of Computer Sciences, Universiti Sains Malaysia.

Ahamad Tajudin Khader is a currently an associate professor in the School of Computer Sciences, Universiti Sains Malaysia. He received the BSc. and M.Sc. in Mathematics from the University of Ohio, USA, in 1982 and 1983, respectively. He obtained his Ph.D. in Computer Science from University of Strathclyde, U.K., in 1993.