# A SOCIAL-BASED MODEL FOR GENETIC ALGORITHMS

Nagham Azmi AL-Madi
University Sains Malaysia, Penang, Malaysia
Nagham@cs.usm.my

Ahamad Tajudin Khader
University Sains Malaysia , Penang, Malaysia
tajudin@cs.usm.my

## ABSTRACT

Genetic algorithms (GAs), as a general search model, have proved its success in several applications, however recently, several researchers have argued that they have slow convergence; this slowness is due to the randomness in all their operations. Therefore, recent researches have employed structured populations, in order to eliminate randomness, such as island models, cellular model, multinational evolutionary algorithms, etc. In this proposal, a social based GA is introduced; this model is trying to mimic the actual social behavior and the actual death and birth process. We will restrict the recombination for males to the only permitted females; we also divide the population into nearly separated subgroups (similar to the island model). Our motivation to such an approach is that we expect the nature to be more robust and optimal; hence the objectives of this work are to study the effects of these social rules and customs on the standard GA, and to investigate its effects on the speed of convergence of GA. The results will be analyzed according to parameters that depend on the social behavior and the natural birth and death models.

Keywords: Genetic Algorithms (GAs), Evolutionary Algorithms (EAs).

## 1.Introduction

John Holland and his colleagues at the University of Michigan were the earlier to introduce Genetic Algorithms (GAs) [21]. Until the early 1980s, the research in genetic algorithms was mainly theoretical, with few real applications; this period is marked by sample work with fixed length binary representation in the domain of function optimization by, among others [21]. GAs is defined by researchers in many different ways, as a matter of fact, they could be described as search algorithms, but differ than other search methods in a way that they mimic natural, and are inspired by evolutionary biology such as inheritance, mutation, selection and crossover (recombination) [17]. In fact they are computing algorithms, their technique in programming is to mimic biological organisms[14,16,17,18], based on natural selection using it as an iterative procedure to optimize and select the best among several solutions of hard and complex problems [14,16,17,18].

They are considered heuristic search methods which find approximate solutions that are the best among the others, you won’t know if the solution is exact, most real life problems are like that, you estimate a solution, but you don’t calculate it exactly [12,21,17]. So, GAs work on a population of possible solutions. GAs are considered quick, reliable and accurate and known as competent GAs in finding best solutions [13]. GAs could be used in many different applications such as Computer Science, Engineering, Economics, Physics, Mathematics and others. They are simple algorithms but very powerful in searching for improvement using a random choice as a tool to guide a high search through the population space.

### 1.1KnapsackProblem

The knapsack problem is used to test our new GA model. It is chosen due to its simplicity and it is widely used as test problem. The knapsack problem is how to pack known size objects into a space in a way that the value of these items will be maximized [19]. The knapsack problem is an optimization problem, it derives its name from the maximization problem of choosing possible essentials that can fit into one bag (of maximum weight) to be carried on a trip. A similar problem very often appears in business, combinatory, complexity theory, cryptography and applied mathematics. Given a set of items, each with a cost and a value, then determine the number of each item to include in a collection so that the total cost is less than some given cost and the total value is as large as possible. The decision problem form of the knapsack problem is the question "can a value of at least *V* be achieved without exceeding the cost *C*?" [19].

#### 1.1.1 Definition

In the following, we have *n* types of items, $x_1$ through $x_n$. Each item $x_j$ has a value $p_j$ and a weight $w_j$. The maximum weight that we can carry in the bag is *C*. The 0-1 knapsack problem restricts the number of each kind of item to zero or one. Mathematically the 0-1-knapsack problem can be formulated as:

maximize $\sum_{j=1}^{n} p_j x_j.$

subject to

$$\sum_{j=1}^{n} w_j x_j \le c, \qquad x_j = 0 \text{ or } 1, \quad j = 1, \ldots, n.$$

## 2. Problem statement

The standard GA has slow convergence due to the randomness in selection, recombination, and mutation. This randomness generates similar or new identical individuals, therefore the new generations don’t span the whole search space effectively, previous works that were intended to eliminate this randomness have come out with better performance in GA, this motivates us to investigate a social-based GA, which employs natural social customs and behavior for the production of new generations. Very few works have been done in using social rules comprehensively. This will be our main objective of this work as explained in the next section.

## 3. Related works

John Holland and colleagues developed GAs at Michigan University, their goals were to explain adaptive processes of natural systems, and to design artificial system software that retains important mechanisms of natural systems. There has been many important discoveries in natural and artificial systems science since this approach. As we have realized previously the EAs works randomly, and that there is no constraints when choosing two individuals to mate together [2]. These days, many researches have been tackling this problem trying to overcome it, and trying to design structured population with some control on how individuals interact [2]. Examples of work done on GAs are Cellular GA [2], Island GA [3], Patchwork GA [5,7], Terrain-Based GA [11], and religion-Based GA [20]. Many researches have been done on GAs which came up with different types and models of GAs. Below we will discuss some of them.

### 3.1 Cellular GAs (CGA)

By Gorges-Schleuter [1]. It is also called diffusion model. Here the individuals are arranged in a two-dimensional Grid world. Individuals interact with each other by the direct neighborhood of each individual [2]. There is a spatial structure that GA population has in this approach. It seems that in this kind of GAs, they are designed as a probabilistic cellular automation. The individuals will be distributed on a graph which is connected together, and each individual will have a neighborhood of some genetic operator to work with. In order to reproduce an operator there is a self-organizing schedule added to this reproduction. The algorithm converges to the global optimum. The individual which can interact with its immediate neighbors can only be held in the cell [9].

### 3.2 Patchwork Model

Krink et al. was the first to introduce this type of model. It has mixed ideas from the cellular EA, island models, and traditional evolutionary algorithms [7]. 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. Here the grid is a two dimensional grid of fields, each field can have a fixed number of individuals. According to the autonomous measure of the motivation of the individuals, they can move around the world. It is considered a self-organized, spatial population structure [4].

### 3.3 Terrain-based GA (TBGA)

It is a more self-tuning model compared to cellular genetic algorithm [11]. In which many combination parameter values will be located in different physical locations. A sort of terrain where solution will be formed. In a previous study [1], the TBGA showed better performance than CGA with less parameter tuning [11]. At every generation each individual should be processed, the mating will be selected from the best of four strings, located above, below, left, right.

### 3.4 Island Models

In evolutionary computation, when more and more complex problems appear, this requires more advanced models of evolutionary algorithms (EAs). The island models are considered a family of such models. [10]. Here the individuals are divided into sections. We call each section a subpopulation which is referred to as an island. These islands are able to solve problems better than standard models [6, 9]. 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 [8, 10].

### 3.5 Religion-Based Model EA (RBEA)

To attract believers is an important part of religious concepts; the religion-based EA (RBEA) is based on this concept. This algorithm was introduced by Rene Thomsen et al. [4]. It attracts new believers to a religion which puts more control than other models such as cellular EA and the patchwork models [20].

## 4. Methodology

Since our model is social-based, this model can be viewed as an island model in which we will divide the world into subgroups, each of which represents a community. The recombination operation is based on the natural and social selection in human societies; this requires that we add an attribute for each individual to specify his sex. Naturally recombination (marriage) depends on being in the same society (with high probability) and similarly in age and social level. This is why we need an island model in which pairs of individuals are most likely to be recombined from the same island. We will also consider the problem of age by adding an attribute for the age that can take three values: youth, parent, and grandparent. This chromosome representation (the presence of father and mother pointers) keeps all required family relations, and let the population of subgroup be divided into a Directed Acyclic Graph (DAG). In figure (1), the standard GA will be modified to include our operations, all the standard operations in the GA will be changed in order to add restrictions on each operation including: Social constraints '*operator*', Birth *operator*, and Death *operator*.

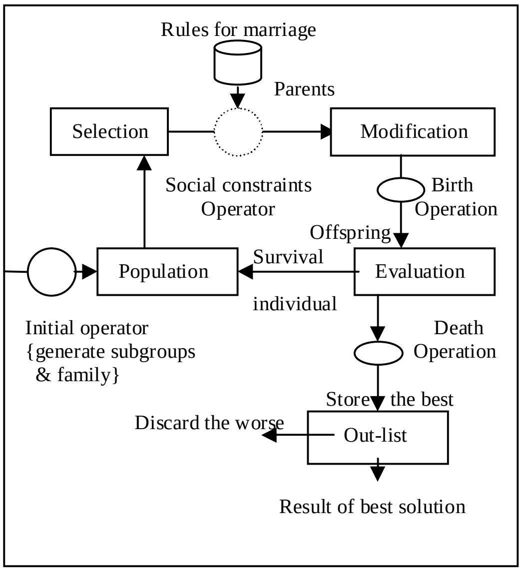


Figure (1): The proposed model design
"The standard GA modified by new operators"

## 5. Testing

Initial experiments were carried out on the knapsack problem using standard GA. This GA has a population of size 10, used one point crossover and mutation is a random change to a gene value. The selection method used is roulette wheel. Figure (2) displays the average fitness for several different runs for 20 generations. Figure (3) shows the overall average fitness for all the runs. The result shows a typical normal graph obtained using GA.

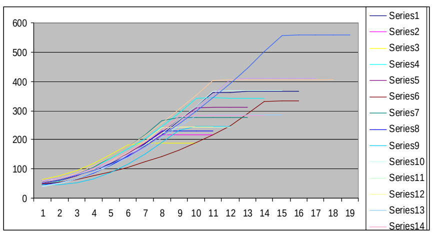


Figure (2)
Different runs using basic GA

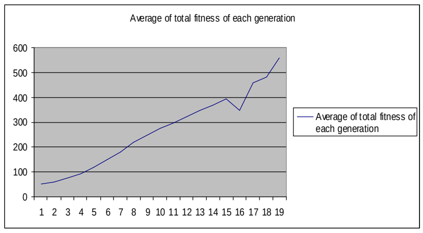


Figure (3)
Total average for all the runs of basic GA

Now, we put some constraints on the basic GA forcing the randomness of selection to be controlled by these constraints. Initially, we select the first individual randomly as our first parent. For the next parent, we choose the opposite type; for example if the first was a male then the second must be a female. We repeat this for a number of individuals creating the initial population. After generating the first population, we repeat the above phases of selection, choosing two individuals, taking under consideration the type of the parents to be opposite, then going through the same steps as above bringing up two new children whom we call *offspring*. Repeating this for a number of individuals we generate the second population. Now, we choose again a new individual from the second population repeating the previous steps, choosing the first parent, then choosing the second; and be sure to choose opposite types for the both parents. The next main important thing is; the *two individuals* must *not* share the same parents. Using these two constraints, we have run some experiments which gave the following results, as can be seen in figure (4) and figure (5) below.

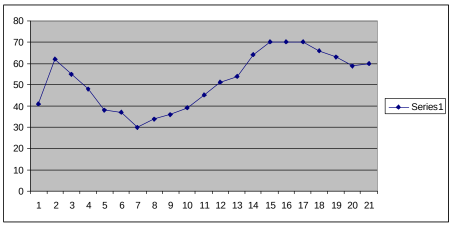


Figure (4)
Total average of 21 generations for the basic GA controlled by Social constraints

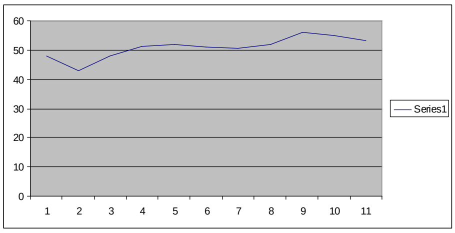


Figure (5)
Total average of 11 generations for the basic GA controlled by Social constraints

## 6. Conclusion

In this proposal, a new approach for structured population for GA is provided, this approach is social based, in which almost all the human society customs and behavior are emulated. The main motivation of this approach is that restricting randomness in the process of recombination or *marriage, and selection* which will enhance the genetic information along the generations. The experiments are still in the early stages. More experiments and fine tunings are needed before credible conclusions can be drawn.

## *References*

[1] 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.

[2] Whitley, D. "Cellular genetic algorithms". In Proceedings of the Fifth International Conference on Genetic Algorithms, 1993, pp. 658.

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

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

[5] 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.

[6] Mohamed A. Belal, Iraq H. Khalifa, "A Comparative Study between Swarm Intelligence and Genetic Algorithms", Egyptian Computer Science Journal, Vol. 24, No. 1, 2002.

[7] Krink, T., and Ursem, R. K. "Parameter control using

the agent based patchwork model". In Proceedings of the Second Congress on Evolutionary Computation.

[8] Meghna Babbar1, Barbara Minsker2, and David E. Goldberg. " A Multiscale Island Injection Genetic Algorithm for Optimal Groundwater Remediation Design".

[9] Enrique Alba1, Mario Giacobini2: "Comparing Synchronous and Asynchronous Cellular Genetic Algorithms".

[10] Zbigniew Skolicki PhD Dissertation Rasmus. "The influence of migration sizes and intervals on island models".

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

[12] Jianjun Hu. "Sustainable Evolutionary algorithms and ScalableEvolutionary Synthesis of Dynamic Systems". 2004.

[13] Claudio F. Lima, Kumara Sastry, David E. Goldberg and Fernando G. Lobo. "Combining Competent Crossover and Mutation Operators: A Probabilistic Model Building Approach", 2005.

[14] Moshe Sipper. "A brief introduction to Genetic Algorithms". The Coming Age of Bio-Inspired Computing, McGraw-Hill, New York, 2002.

[15] Mariusz Nowostawski and Riccardo Poli. "Parallel Genetic Algorithm Taxonomy". 1999.

[16] Adam Marczyk. "Genetic Algorithms and Evolutionary Computation". 2004.

[17] T.D. Collins. "The Visualization of Genetic Algorithms". Master's thesis, De Montfort University, Department of Computer Science, Leicester, UK, 1993 .

[18] Franco Busetti. "Genetic Algorithms overview". 2001.

[19] David Pisinger. "Algorithms for knapsack Problems". Ph.D. thesis, 1995.

[20] 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.

[21] David E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning,* Addison-wesley Longman, Inc. 2004.

[22] Journal Holland,J., Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, 1975.