ICIC Express Letters
Volume 12, Number 12, December 2018

ICIC International ©2018 ISSN 1881-803X
pp. 1267–1275

A STRUCTURED-POPULATION HUMAN COMMUNITY BASED GENETIC ALGORITHM (HCBGA) IN A COMPARISON WITH BOTH THE STANDARD GENETIC ALGORITHM (SGA) AND THE CELLULAR GENETIC ALGORITHM (CGA)

Nagham Azmi AL-Madi, Khulood Abu Maria, Eman Abu Maria
and Mohammad Azmi AL-Madi

Faculty of Science and Information Technology
AL-Zaytoonah University of Jordan
P.O.Box 130, Amman 11733, Jordan
{ nagham.a; khulood; eman.maria }@zuj.edu.jo; almadi_moh@yahoo.com

Received April 2018; accepted July 2018

Abstract. *The structured populations in genetic algorithms are found to be proven to be superior in their performances in comparison with the Standard Genetic Algorithm (SGA) since they can control two opposite processes, that is, the exploration and exploitation in the search space. Previous studies do not use the genes information of the individuals. Hence, the structure of the sub-populations that are based on this information might help achieve better performances and more efficient searching strategies. The Human Community Based Genetic Algorithm (HCBGA) is considered an improved genetic algorithm, which is mainly based on social constraints. While using the knapsack as a maximizing test bed, a study on the behavior of the Standard Genetic Algorithm (SGA), the Cellular Genetic Algorithm (CGA) and the Human Community Based Genetic Algorithm (HCBGA) models is performed in terms of the quality of the solutions being found. Tests show that the HCBGA outperforms the SGA and the CGA in terms of finding the maximum optimal solution. Further, the HCBGA produces stable results after performing an approximate number of 100 generations under improved constraints. Finally, it can be inferred from the obtained results that the mean values of the HCBGA converge toward either the global maxima or the global minima.*

Keywords: Standard Genetic Algorithm (SGA), Knapsack problem, Cellular Genetic Algorithm (CGA), Structured population, Convergence

1. Introduction.

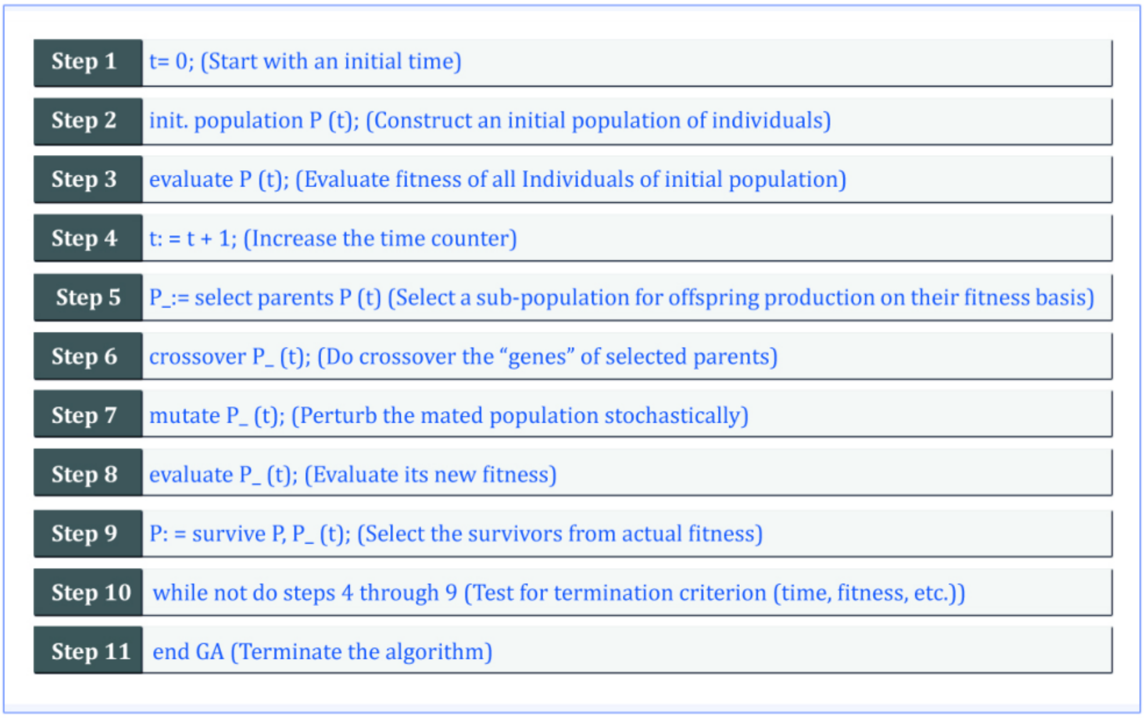
The Standard Genetic Algorithm (SGA) is considered to be the basis for the genetic algorithm. In fact, it is the standard of the entire algorithms for which they are improved based on it, and for which they are compared with it. Besides, SGAs form the best choice that can be frequently used in many difficult optimization problems such as NP-hard problems [1]. Since SGAs work on a population of solutions, these solutions consist of a sequence of genes, which are so-called individuals or chromosomes. In practice, this is performed via some operators, such as selection, crossover (recombination), and mutation. New individuals replace existing individuals with some laid out policies [2].

The general idea of the Standard Genetic Algorithm (SGA) is best elaborated by the following scheme as shown in Figure 1.

Nonetheless, the SGAs could fall in local solutions by the crossover, the mutation operations, and the undesired population diversity loss based on the selection operations, which constantly decrease the variety of its specimens [3].

The SGAs are usually used to solve many scientific and engineering areas, such as the graph coloring problem [4], the multi-objective optimization of systems, the 0/1 multiple knapsack problem [5], and the assembly line balancing problem [6].

DOI: 10.24507/icicel.12.12.1267



Step 1 t= 0; (Start with an initial time)

Step 2 init. population P (t); (Construct an initial population of individuals)

Step 3 evaluate P (t); (Evaluate fitness of all Individuals of initial population)

Step 4 t: = t + 1; (Increase the time counter)

Step 5 P_:= select parents P (t) (Select a sub-population for offspring production on their fitness basis)

Step 6 crossover P_ (t); (Do crossover the "genes" of selected parents)

Step 7 mutate P_ (t); (Perturb the mated population stochastically)

Step 8 evaluate P_ (t); (Evaluate its new fitness)

Step 9 P: = survive P, P_ (t); (Select the survivors from actual fitness)

Step 10 while not do steps 4 through 9 (Test for termination criterion (time, fitness, etc.))

Step 11 end GA (Terminate the algorithm)

Figure 1. The general idea of the Standard Genetic Algorithm (SGA) [3]

2. **The Knapsack Problem.** One of the most popular problems that are solved by the genetic algorithms is the knapsack problem [7,8]. In fact, it is a combination of the maximization discontinuous non-permutation NP-hard problem [8]. Among many algorithms, the genetic algorithm is considered to be the most successful in finding an optimal solution for the knapsack problem [7]. In addition, the genetic algorithm is considered to be the fastest algorithm, which solves this problem.

The knapsack problem is similar to a huge space full of solutions of a particular problem. Accordingly, it needs a long searching time in order to find the best possible solution. Therefore, a solution that gives an optimal solution is needed in a short time so that it can tackle this problem. Each object or solution has a weight and a value [9], and the total weight of the object should not exceed the total capacity of the knapsack [7-9]. In particular, this total capacity of the knapsack is a fixed number W, which is the maximum weight a knapsack can carry. The main aim of the knapsack problem is to fill it with different objects or solutions, and to simultaneously maximize the total profit of the included objects [7-9].

The knapsack problem is referred to as the 0-1 problem due to the fact that it accepts or rejects an object or a solution [9]. If the object is accepted, a value '1' is given to it. Otherwise, a value '0' is given.

Holland proposes the genetic algorithms, which are applied in many different areas. For instance, in [11], the researchers attempt to solve the unbounded knapsack problem based on the use of the genetic algorithms. During their research, they use the problem-specific knowledge, and incorporate a preprocessing procedure, which is however affected by the knowledge. Meanwhile, it is demonstrated that the genetic algorithms have a stumbling block that can be increased by using it with a large space, which could not give good results. Accordingly, the researchers are attempting to improve this problem based on the use of genetic algorithms [9,11]. Last, in [12], a study of the Fuzzy Logic Controller (FLC) is introduced in order to make use of the crossover as a function of the chromosomes' ages for the purpose of reducing their rate of premature convergence. A hybrid optimization algorithm that is based on a combination of the neural network, and on the genetic algorithm is presented in [13]. In fact, the researchers use the back-propagation neural network in order to improve the convergence of the genetic algorithm in large searching spaces for the global optimum solution.

As for the large population sizes and the low mutation rates, it is shown that a small population size, and a relatively high mutation rate are superior [14,15]. Studies in [15,16] use a selection probability based on the ranking of fitness values by varying the values of the crossover and mutation processes. The reason behind this usage is to improve the searching capacity.

3. **The Cellular Genetic Algorithm.** The cellular genetic algorithm has a high capacity. In fact, it is a structured population algorithm, which represents a subclass of the genetic algorithm [17]. It is an extremely efficient tool for a complex computation [18]. It is a probabilistic algorithm that is initiated by Gorges-Schleuter in 1989 and by Manderick and Spiessen [19]. This cellular genetic algorithm is so-called the diffusion model that has a spatial topology. Further, it consists of a two-dimensional grid and individuals that have direct neighbors of about four to eight individuals, which interact with each other [19,21,23]. Thus, the concept of the cellular algorithm is to arrange the population into a grid. In this context, the population is decentralized, where only the neighborhood individuals could interact with each other [19,21,23].

The researchers in [20] propose a cellular automata in order to recognize the neighborhood in the population structure, as well as the locality of it. During the genetic search, the selected individuals are controlled in order to avoid fast population loss of diversity [3]. Accordingly, it could take several generations to diffuse individuals all over the grid. Hence, the slow diffusion of the individuals throughout the grid gives the opportunity for the Cellular Genetic Algorithm (CGA) to avoid the premature convergence [24,25].

In [26,27], an algorithm for the Multi-objective Optimization Problem (MOP) is proposed. The individuals in each cell depend on its weight vector. Thus, the selection is controlled by each individual's weight vector in that cell. The selected mates are taken from the neighborhood cells [26,27].

In order to solve the multi-objective continuous problem, a new Cellular Genetic Algorithm (CGA) is presented in [27]. In particular, an external archive is used for the purpose of storing non-dominated individuals that arise from the searching process. At the end of each iteration, some of the solutions are returned back to the population from the archive randomly. These are exchanged with existing solutions from within the population. This algorithm is called the (MOCell) algorithm. Constrained and unconstrained problems are used to evaluate this algorithm. It can be found to be proven from tests that the algorithm shows a competitive result towards convergence [27].

4. Problem Statement and Preliminaries.

4.1. **The structured population genetic algorithm.** The standard genetic algorithm has no restrictions in the selection part. Hence, it selects individuals randomly. Any individual could mate with any other. They only depend on the fitness that is best to be chosen. Premature convergence in finding better solutions is noticed due to the occurrence of this randomness. Further, the loss of diversity among the individuals of the population is considered as a high chromosome flow. In short, all these are causes of this randomness [22].

Many trials are being performed in order to overcome the premature convergence problem [28,29]. Hence, a structured population that controls the individuals is introduced [28]. This structure could reduce the randomness of the selection method. The tests show that these kinds of populations give better results. Subsequently, many genetic algorithms are presented. In fact, such examples of these algorithms comprise the Island Genetic Algorithm (IGA) [31,32], the Cellular Genetic Algorithm (CGA) [24,27,30], the Terrain-Based Genetic Algorithm (TBGA) [33,34], the patchwork genetic algorithm [35], the Religion Based Genetic Algorithm (RBGA) [36,37], and the sexual reproduction model [38-40].

4.2. **The Human Community Based Genetic Algorithm (HCBGA) model.** A new approach of the genetic algorithm structured population is introduced in [41]. Additional constraints are proved to control the randomness when selecting parents to mate in the Standard Genetic Algorithm (SGA).

Referring to the extended enhancement of [41], a new enhancement of the structured population approach for genetic algorithm, based on the custom, behavior and pattern of human community is proposed in [22]. This includes gender, age, generation, marriage, birth and death. As such, this model is named the Human Community Based Genetic Algorithm (HCBGA) model. It is an evolution of the Standard Genetic Algorithm (SGA). It is based on the selection part. There are restrictions when selecting individuals to mate. This is similar to the marriage in the human community [21,29] as shown in Figure 2 below.

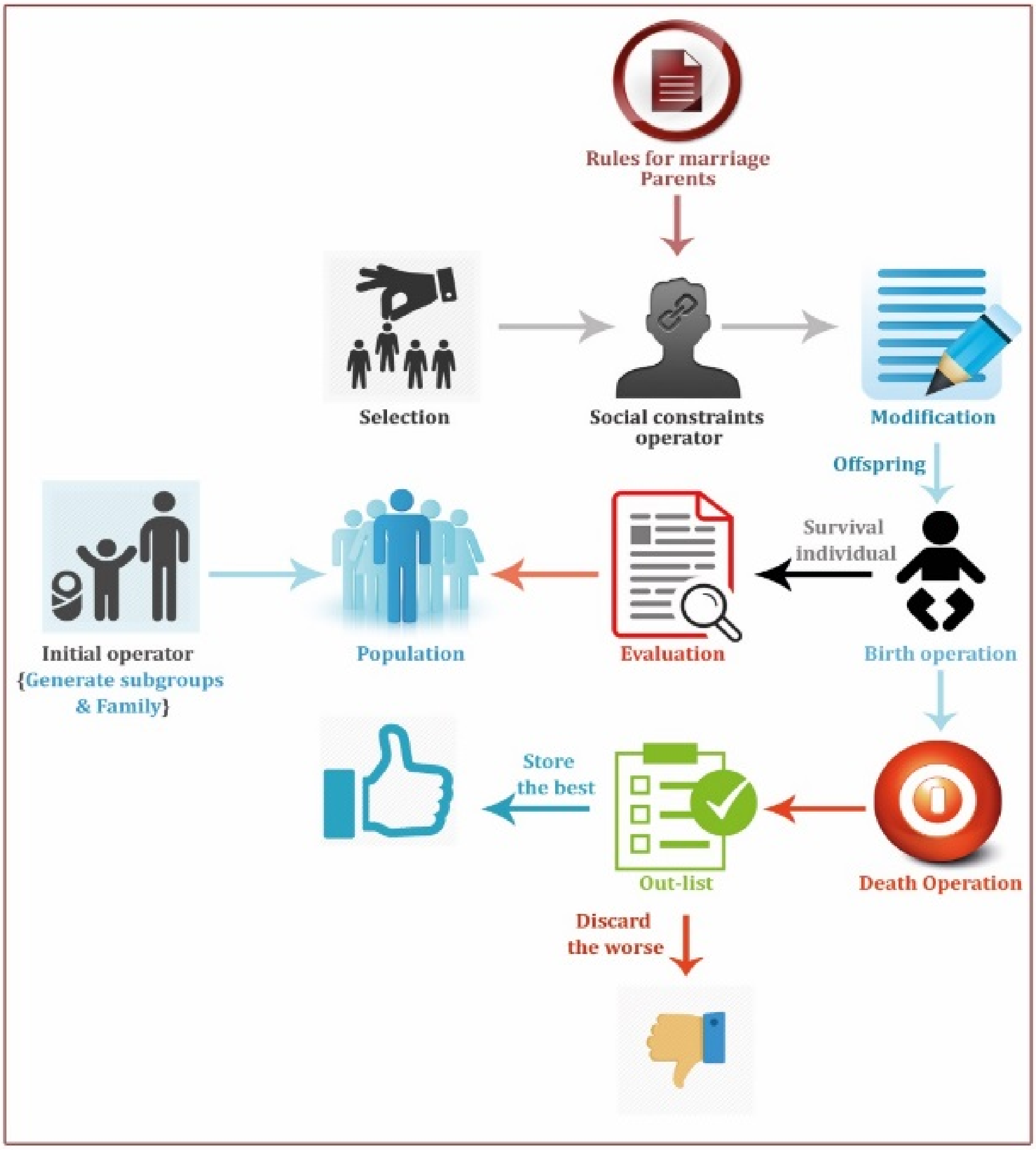


Figure 2. The Human Community Based Genetic Algorithm (HCBGA) model design modified by new operators [22]

5. Main Results.

Here are the main results in this paper.

Series of experiments were carried out on the knapsack problem. This test problem covers together discontinuous and non-permutation problem, and hence gives better results. The performance of the HCBGA is found to be far better than the SGA and an advanced model named the Cellular Genetic Algorithm (CGA) model, as this enhanced HCBGA model obtains better optimal maxima or minima, besides maintaining the diversity.

5.1. SGA and HCBGA.

Based on the analysis of the results, the HCBGA outperforms the SGA in finding the best fit values, and the mean of the entire fitness values. This could be seen in Figures 3 and 4.

The population is divided up into two different groups, in such a way they are equal in the sizes of males and females. Besides, the increase of mechanisms occurs in the HCBGA. In fact, this increase gives this model a better performance other than the SGA. Subsequently, the rest of the constraints of rules of marriage will be added to the selection and crossover parts. The rules of marriage of selecting partners in the HCBGA ensure that the mates are not close to each other. Hence, the population will have different offspring. Subsequently, this guarantees good diversity, which keeps the population heterogeneous. Figures 3 and 4 show how the HCBGA is diversified better than the SGA. These surly results require a better solution for the maximization problem.

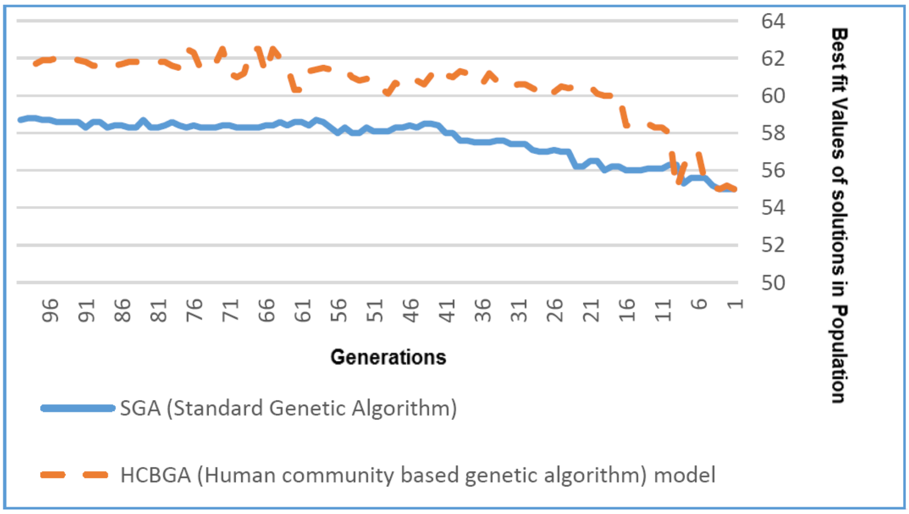


Figure 3. Best solutions of SGA and HCBGA using the knapsack problem

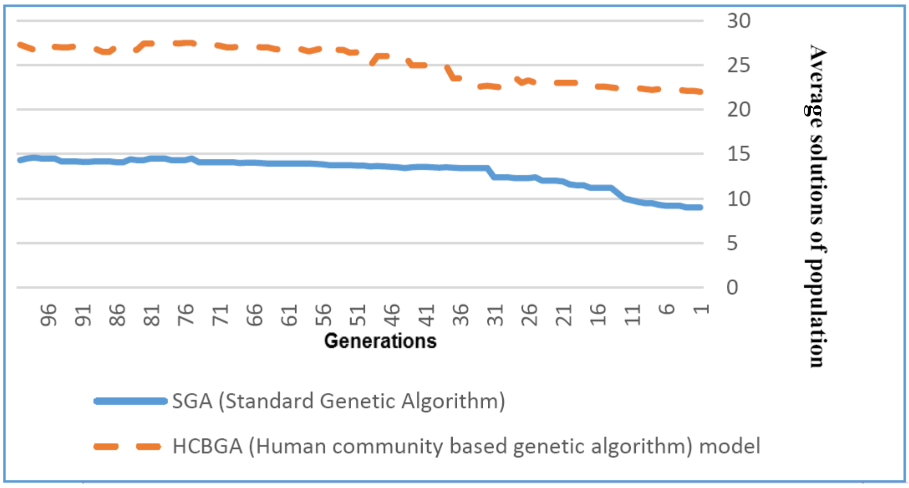


Figure 4. Average solutions of SGA and HCBGA using the knapsack problem

5.2. **CGA and HCBGA.** By delivering the comparisons between the HCBGA and the CGA, and by applying the knapsack as a test problem, the HCBGA proves to be better in finding the best fit values in comparison with the CGA as depicted in Figure 5.

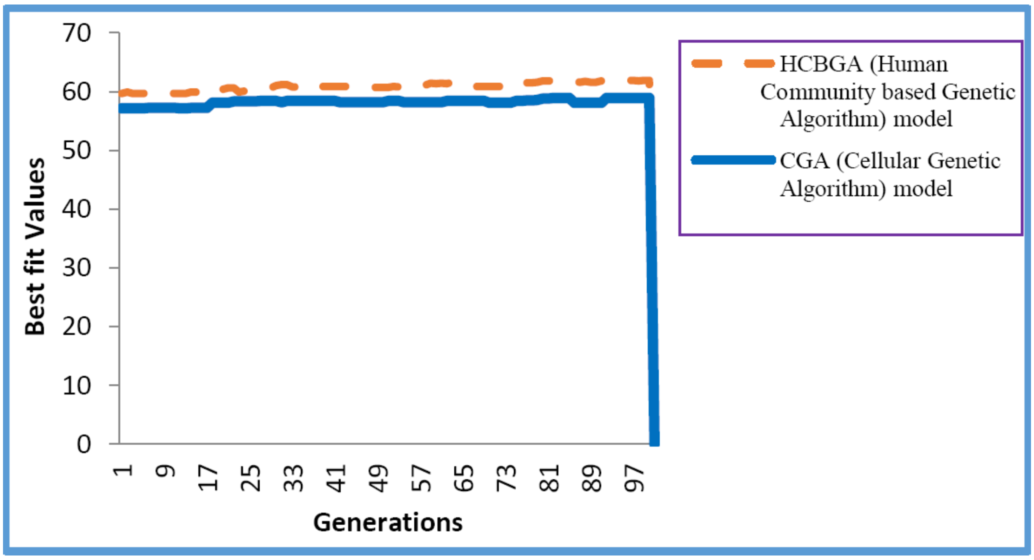


Figure 5. Average solutions of CGA and HCBGA using the knapsack problem

The HCBGA converges to a best fit value in the population with a value of 61.8, whereas the CGA converges to a best fit value of 58.9 in the same generation.

In its selection phase, the CGA is mainly based on the neighboring individuals. Accordingly, the CGA falls into a slow convergence. Nonetheless, the HCBGA individuals are being randomly distributed. In practice, these individuals are more flexible in the HCBGA. This gives the algorithm a better search for other optimal solutions. The diversity of the HCBGA, though, increases over the CGA as can be seen in Figure 5.

The statistical tests in Table 1 show that the HCBGA is the best model in terms of the highest mean, standard deviation, and variance. In any maximization problem, it is known that the higher the mean value is, the better the model is. Thus, it can be observed that the mean value of the HCBGA which is 61.084. Besides, the highest value of the standard deviation is 0.5898 as found in the HCBGA model. In fact, this explains the spread of individuals towards the maximization, and indicates that individuals in the population are spreading around the mean in a balanced distribution. In addition, the variance value of the HCBGA which is 0.354 is also higher compared to the other two models. This indicates that a variation in the data meaning is that the HCBGA model achieves more diversity between its individuals compared to the other two models. Consequently, the HCBGA model could reach a better fitness value where this indicates to achieving better performance than other models.

Table 1. Statistical analysis of the population for SGA, CGA and HCBGA models using the knapsack problem

	No. of Generations				
Models	Statistic	Mean	Std. Error	Std. Deviation	Variance
SGA	100	58.26	0.056	0.567	0.333
HCBGA	100	61.084	0.0598	0.5898	0.354
CGA	100	58.358	0.0354	0.3545	0.116

The Friedman test is a "non-parametric test (distribution-free) which is used to compare between observations repeated on the same subjects, also referred to as a non-parametric randomized block analysis of variance" (NIST, 2004). The rank is a statistical method that ranks the model according to the mean, the highest the best in a maximization problem.

Table 2 shows the mean ranks of the three models: the HCBGA, the SGA, and the CGA. In fact, it is indicated that the HCBGA model clearly outperforms both other models, as it achieves the highest rank. Since the HCBGA yields the best rank in comparison with the other two models, better fitness values are achieved by the HCBGA based on its populations along the 100 generations.

Table 2. The Friedman test shows mean ranks between SGA, CGA and HCBGA models.

Models	Mean Rank
CGA	2.01
SGA	3.06
HCBGA	5.84

Table 3. Kendall's W test shows significant differences between SGA, CGA and HCBGA models.

Parameters	Values
N	100
Kendall's W	.887
Chi-Square	430.022
Df	2
Asymp. Sig.	.000
Monte Carlo Sig.	.000

In Table 3, the Kendall's W test is conducted on the three models. N is the number of generations. The chi-square indicates a test of independence, whereas its value is very high meaning that the HCBGA model is independent from other models. The Df is the degree of freedom its value is $k - 1$ where k is the number of models tested and in this test there are 6 models so the Df value is 2. In addition, the Kendall's W value is .887 which is a high value near to 1, and this indicates a full agreement that the HCBGA model performs significantly better in exploring the search space for best solutions than other models. Finally, for a model, if the Monte Carlo significant level is less than 5% then the model is considered significantly different from other models. As such, in Table 3 it shows a Monte Carlo significant value of .000 which means the HCBGA model has a 100% effect and it has a high significant difference over the other models with a level of confidence of 99% due to .000 is less than 5%.

6. **Conclusion.** As shown in the previous section, a comparison between the best fit values of the Human Community Based Genetic Algorithm (HCBGA), the Standard Genetic Algorithm (SGA), and the Cellular Genetic Algorithm (CGA) is performed by using the knapsack problem which is conducted. As a result, it can be inferred from the obtained results that the HCBGA achieves better results in terms of finding best fit values in a comparison with the other two models, SGA, and the CGA. This implies that the HCBGA outperforms the SGA and the CGA in finding the optimal maximum solution. Further, the HCBGA produces stable results after approximately 100 generations according to improved constraints. In addition, the mean values of the HCBGA converge towards the global maxima, or the global minima.

Acknowledgment. The authors would like to thank AL-Zaytoonah University of Jordan for its support pertaining to this study. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] A. Maria, M. Pedro and P. Tania, Modified standard genetic algorithms improving convergence time for the purposes of fermentation process parameter identification, *WSEAS Trans. Systems*, vol.11, no.7, 2012.

[2] B. Abul and Z. I. Md, Advantages and limitations of genetic algorithms for clustering records, *IEEE the 11th International Conference on Industrial Electronics and Applications (ICIEA)*, 2016.

[3] H. R. Cheshmehgaz, H. Haron and M. R. Maybodi, Cellular-based population to enhance genetic algorithm for assignment problems, *American Journal of Intelligent Systems*, vol.1, no.1, pp.32-36, 2011.

[4] M. M. Hindi and R. Yampolskiy, *Genetic Algorithm Applied to the Graph Coloring Problem*, Computer Engineering and Computer Science, J.B. Speed School of Engineering, 2012.

[5] S. B. Jagtap, S. K. Pani and G. Shinde, A hybrid parallel multi-objective genetic algorithm for 0/1 knapsack problem, *Journal of Software Engineering and Applications*, vol.4, pp.316-319, 2011.

[6] B. L. Jithendrababu, RenjuKurian and T. G. Pradeepmon, Balancing labor intensive assembly line using genetic algorithm, *Proc. of International Conference on Energy and Environment-2013 (ICEE 2013)*, Kottayam, Kerala, India, 2013.

[7] F. A. O. Najadat, G. G. Kanaan, R. K. Kanaan, O. S. Aldabbas and R. F. Al-Shalabi, Genetic algorithm solution of the knapsack problem used in finding full issues in the Holy Quran based on the number (19), *Computer and Information Science*, vol.6, no.2, 2013.

[8] G. Megha, A fast and efficient genetic algorithm to solve 0-1 knapsack problem, *International Journal of Digital Application & Contemporary Research*, vol.1, no.6, 2013.

[9] T. Vikas and D. Shivali, Genetic algorithm based approach to solve non fractional (0/1) knapsack optimization problem, *International Journal of Computer Applications*, vol.100, no.15, 2014.

[10] J. H. Holland, *Adaptive in Natural and Artificial Systems*, Univ. Michigan Press, Ann Arbor, MI, 1975.

[11] A. Chehouri, R. Younes, J. Perron and A. Ilinca, A constraint-handling technique for genetic algorithms using a violation factor, *Journal of Computer Sciences*, vol.12, no.7, pp.350-362, 2016.

[12] N. Suraparaju, V. K. Surapaneni, M. Vempalli and B. M. Radhika, Clustering WSN using fuzzy logic and genetic algorithm, *IJSRD – International Journal for Scientific Research & Development*, vol.3, no.3, 2015.

[13] Z. Zukhri and I. V. Paputungan, A hybrid optimization algorithm based on genetic algorithm and ant colony optimization, *International Journal of Artificial Intelligence & Applications (IJAIA)*, vol.4, no.5, 2013.

[14] R. Olympia, F. Stefka and P. Marcin, Influence of the population size on the genetic algorithm performance in case of cultivation process modelling, *Proc. of the 2013 Federated Conference on Computer Science and Information Systems*, pp.371-376, 2013.

[15] S. L. George and B. T. Mohamed, A genetic algorithm with weighted average normally-distributed arithmetic crossover and twinkling, *Applied Mathematics*, vol.3, pp.1220-1235, 2012.

[16] A. Shukla, H. M. Pandey and D. Mehrotra, Comparative review of selection techniques in genetic algorithm, *International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE)*, 2015.

[17] M. Guzek, J. E. Pecero, B. Dorronsoro, P. Bouvry and S. U. Khan, A cellular genetic algorithm for scheduling applications and energy-aware communication optimization, *International Conference on High Performance Computing & Simulation*, 2010.

[18] K. Shirin and H. Akhtar, Sensor placement in WSN using the cellular genetic algorithm, *J. Basic. Appl. Sci. Res.*, vol.3, pp.745-750, 2013.

[19] F. Pinel, B. Dorronsoro and P. Bouvry, A new parallel asynchronous cellular genetic algorithm for scheduling in grids, *IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW)*, 2010.

[20] H. R. Cheshmehgaz, H. Haron and M. R. Maybodi, Cellular-based population to enhance genetic algorithm for assignment problems, *American Journal of Intelligent Systems*, vol.1, no.1, pp.32-36, 2011.

[21] R. K. Moghadas and S. Gholizadeh, A cellular automata based firefly algorithm for layout optimization of truss structures, *International Journal of Optimization in Civil Engineering*, vol.7, no.1, pp.13-23, 2017.

[22] N. Azmi, De Jong's sphere model test for a human community based genetic algorithm model (HCBGA), *International Journal of Advanced Computer Science and Applications*, vol.5, no.1, 2014.

[23] S. Bhargava, Note on evolutionary algorithms and its applications, *Adults Learning Mathematics: An International Journal*, vol.8, no.1, pp.31-45, 2013.

[24] A. Al-Naqi, A. Erdogan, T. Arslan and Y. Mathieu, Balancing exploration and exploitation in an adaptive three-dimensional cellular genetic algorithm via a probabilistic selection operator, *IEEE NASA/ESA Conference on Adaptive Hardware and Systems*, 2010.

[25] A. S. Aldallal, Avoiding premature convergence of genetic algorithm in informational retrieval systems, *International Journal of Intelligent Systems and Applications in Engineering*, vol.2, no.4, pp.80-85, 2014.

[26] N. Qamar, N. Akhtar and I. Younas, Comparative analysis of evolutionary algorithms for multi-objective travelling salesman problem, *International Journal of Advanced Computer Science and Applications*, vol.9, no.2, 2018.

[27] Y. Tian, R. Cheng, X. Zhang and Y. Jin, PlatEMO: A MATLAB platform for evolutionary multi-objective optimization, *IEEE Computational Intelligence Magazine*, vol.12, no.4, pp.73-87, 2017.

[28] M. A. Belal and M. H. Haggag, A structured-population genetic-algorithm based on hierarchical hypercube of genes expressions, *International Journal of Computer Applications*, vol.64, no.22, 2013.

[29] A. J. Umbarkar, M. S. Joshi and P. D. Sheth, Dual population genetic algorithm for solving constrained optimization problems, *I. J. Intelligent Systems and Applications*, pp.34-40, 2015.

[30] H. R. Cheshmehgaz, M. I. Desa and F. Kazemipour, A cellular-rearranging of population in genetic algorithms to solve assembly-line balancing problem, *Journal of Mechanical Engineering and Automation*, vol.2, no.2, pp.25-35, 2012.

[31] A. A. Gozali, J. Tirtawangsa and T. A. Basuki, Asynchronous island model genetic algorithm for university course timetabling, *The 10th International Conference of the Practice and Theory of Automated Timetabling*, York, United Kingdom, 2014.

[32] N. Mezyan and V. W. Samawi, Web information retrieval using island genetic algorithm, *Proc. of the World Congress on Engineering and Computer Science*, San Francisco, USA, 2015.

[33] S. Gatscha, *Generic Optimization of a Wind Farm Layout Using a Genetic Algorithm*, Master Thesis, University of Natural Resources and Life Science, Vienna, 2016.

[34] P. Devibala and C. Rajendran, Optimal micro-siting of wind turbines in a wind park using particle swarm optimization algorithm, *Int. J. Eng, CS&Tech*, 2012.

[35] M. Franulović and R. Basan, Objective function in genetic algorithm for material behaviour modeling, *The 14th International Research/Expert Conference "Trends in the Development of Machinery and Associated Technology"*, Mediterranean Cruise, 2010.

[36] J. D'Souza, Validation of region-based crossover for clustering problems, *International Journal of Imaging and Robotics*, vol.15, no.2, pp.53-66, 2015.

[37] O. I. Lasassmeh, A. A. Kasassbeh, A. M. Mobaedeen and E. A. Kasasbeh, Shared crossover method for solving knapsack problems, *International Journal of Academic Research in Business and Social Sciences*, vol.4, no.5, pp.227-249, 2014.

[38] A. Farasat, M. B. Menhaj, T. Mansouri and M. R. S. Moghadam, ARO: A new model-free optimization algorithm inspired from asexual reproduction, *Applied Soft Computing*, vol.10, no.4, pp.1284-1292, 2010.

[39] J. D'Souza, *Region-Based Crossover for Clustering Problems*, 2012.

[40] G. Ranjhitha and M. V. Rajkumar, Implementation of genetic algorithm based maximum power point tracking for photovoltaic system, *International Journal of Innovative Research in Science, Engineering and Technology*, vol.5, no.11, 2016.

[41] N. A. AL-Madi and A. T. Khader, A social-based model for genetic algorithms, *Proc. of the 3rd International Conference on Information Technology (ICIT)*, AL-Zaytoonah Univ., Amman, Jordan, 2007.