2011 International Conference on Information and Intelligent Computing
IPCSIT vol.18 (2011) © (2011) IACSIT Press, Singapore

Learning Bees Algorithm For optimization

Fahimeh Aghazadeh[1] and Mohammad Reza Meybodi[2]

[1]Computer Engineering and Information Technology Islamic Azad University, Qazvin, Iran

[2]Computer Engineering and Information Technology, Amirkabir University of Technology, Qazvin, Iran

agazadeh.f@gmail.com and mmeybodi@aut.ac.ir

Abstract. In this paper, a new algorithm is obtained by combining Bees Algorithm and Learning Automata. In the proposed model the hive includes a Learning Automaton. The bees are use learning abality for selecting a bee and moving toward it for doing local search, try to optimize their position in the environment. In order to show the effectiveness of the proposed algorithm, the algorithm is tested through series of typical optimization problems. The experiments indicate that the proposed algorithm improves the efficiency of searching and veracity of result.

Keywords: Bees Algorithm, Learning Automata, Optimization.

1. Introduction

In the last two decades, the computational researchers have been increasingly interested to the natural sciences, and especially biology, as source of modeling paradigms. Many research areas are massively influenced by the behavior of various biological entities and phenomena. It gave birth to most of population-based Metaheuristics such as Evolutionary Algorithms (EAs), Particle Swarm Optimization (PSO), Bee Colony (BC) etc. Honey bees are one of the most well studied social insects[1]. In this work, a new hybrid model of Cellular Automata and the Bees Algorithm has been proposed.

The paper is organized as follows: Sections 2 and 3 give a brief description about basic Bees Algorithm and Learning Automata. Section 4 describes the proposed Algorithm. Section 5 gives out the experimental results of the proposed model along with its comparison to best results gained in previous works. Finally in section 6 we conclude our paper.

2. Behaviour of Bees in Nature

Behavior of bees in nature Social insect colonies can be considered as dynamical system gathering information from the environment and adjusting their behavior in accordance to it. While gathering information and adjustment processes, individual insects do not perform all the tasks because of their specializations. Generally, all social insect colonies behave according to their own division of labors related to their morphology[15]. Bee system consists of two essential components: Food sources and foragers including:

– Unemployed foragers: If it is assumed that a bee have no knowledge about the food sources, initializes its search as an unemployed forager. There are two possibilities for an unemployed forager: (1) Scout Bee (S in "Fig.1"): if the bee starts searching spontaneously without any knowledge, it will be a scout bee. The percentage of scout bees varies from 5% to 30% according to the information into the nest. The mean number of scouts averaged over conditions is about 10% [13][15]. (2) Recruit (R in "Fig.1"): if the unemployed forager attends to a waggle dance done by some other bee, the bee will start searching by using the knowledge from waggle dance.

– Employed foragers: (EF in “Fig.1”): when the recruit bee finds and exploits the food source, it will raise to be an employed forager who memorizes the location of the food source. After the employed foraging bee loads a portion of nectar from the food source, it returns to the hive and unloads the nectar to the food area in the hive. There are three possible options related to residual amount of nectar for the foraging bee. (1) If the nectar amount decreased to a low level or exhausted, foraging bee abandons the food source and become an unemployed bee. (2) it can continue to forage without sharing the food source information with the nest mates. (3) Or it can go to the dance area to perform waggle dance for informing the nest mates about the same food source.

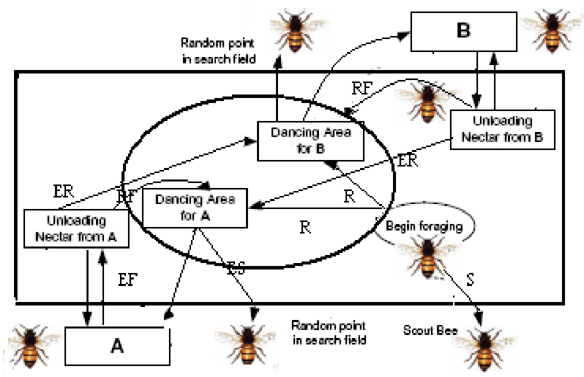

Figure 1. Typical behavior of honey bee foraging[13], [15].

– Experienced foragers: These types of forager use their historical memories for the location and quality of food sources. It can be an inspector which controls the recent status of food source already discovered. It can be a reactivated forager by using the information from waggle dance. It tries to explore the same food source discovered by itself if there are some other bees confirm the quality of same food source (RF in “Fig.1”). It can be scout bee to search new patches if the whole food source is exhausted (ES in “Fig.1”). It can be a recruit bee which is searching a new food source declared in dancing area by another employed bee (ER in “Fig.1”).

3. Bees Algorithm

The bees algorithm is a population-based search algorithm inspired by the natural foraging behaviour of honey bees first developed in 200[2]. In its basic version, the algorithm starts by scout bees being placed randomly in the search space. Then the fitnesses of the sites visited by the scout bees are evaluated and Bees that have the highest fitnesses are chosen as “selected bees” and sites visited by them are chosen for neighbourhood search. Then, the algorithm conducts searches in the neighbourhood of the selected sites, assigning more bees to search near to the best e sites. Searches in the neighbourhood of the best e sites are made more detailed by recruiting more bees to follow them than the other selected bees. Together with scouting, this differential recruitment is a key operation of the Bees Algorithm. The remaining bees in the population are assigned randomly around the search space scouting for new potential solutions. These steps are repeated until a stopping criterion is met. At the end of each iteration, the colony will have two parts, those that were the fittest representatives from a patch and those that have been sent out randomly. the algorithm performs a kind of neighbourhood search combined with random search and can be used for both combinatorial and functional optimisation[3]. “Fig.3” shows the pseudo code of basic bees algorithm[3].

Pseudo code of basic Bees Algorithm

1. *Initialise population with random solutions.*
2. *Evaluate fitness of the population.*
3. *While (stopping criterion not met)*
   //Forming new population.

4. *Select sites for neighbourhood search.*
5. *Recruit bees for selected sites (more bees for best e sites) and evaluate fitnesses.*
6. *Select the fittest bee from each patch.*
7. *Assign remaining bees to search randomly and evaluate their fitnesses.*
8. *End While.*

Figure 2. Pseudo code of the basic bees algorithm

4. Learning Automata

Learning Automaton (LA) is a machine that can perform finite number of actions. Each selected action is evaluated by a random environment. The result is presented to the LA as a positive or negative signal and then the LA will use this response to choose its next action. The final goal of LA is to learn to select the best action among all its actions. The best action is an action that maximizes the probability of receiving reward from the environment. LA and its interaction with the environment are shown in "Fig.3"[4],[5]. The environment is represented by triple $E \equiv \{\alpha, \beta, c\}$. Here, $\alpha \equiv \{\alpha_1, \alpha_2,..., \alpha_r\}$ is the set of inputs, $\beta \equiv \{\beta_1, \beta_2,..., \beta_m\}$ is the set of outputs and $c \equiv \{c_1, c_2, ..., c_r\}$ is the set of punishment probabilities. If β is a two member set, the environment is of type P. In such an environment, $\beta_1 = 1$ and $\beta_2 = 0$ are considered as punishment and reward respectively. In a Q environment, β can take a discrete value from finite values in [0,1]. In an environment of type S, $\beta(n)$ is a random variable in [0,1]. C_i is the probability that action α i has an undesirable result. In a static environment, c_i values are fixed, but in a dynamic environment, they might change as time passes.

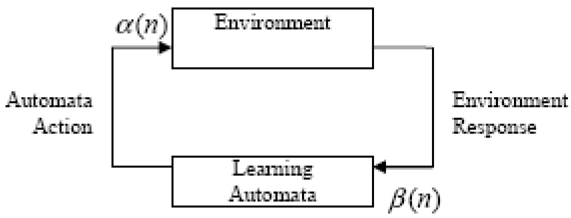

Figure 3. Relation between Learning Automata and Environment [5]

Fixed structure LA is represented by quintuple $\{\alpha, \beta, F, G, \varphi\}$ in which $\alpha \equiv \{\alpha_1, \alpha_2,..., \alpha_r\}$ is the set of actions, $\beta \equiv \{\beta_1, \beta_2,..., \beta_m\}$ is the set of inputs, $\varphi(n) \equiv \{\varphi_1, \varphi_2, ..., \varphi_k\}$ is the set of internal states at time n, $F: \varphi \times \beta \rightarrow \varphi$ is next state transition function and $G: \varphi \rightarrow \alpha$ is the output function that maps the current state of LA to next output (action). Variable structure LA can be defined as $\{\alpha, \beta, p, T\}$. Here, $\alpha \equiv \{\alpha_1, \alpha_2,..., \alpha_r\}$ is the set of actions $\beta \equiv \{\beta_1, \beta_2,..., \beta_m\}$ is the set of inputs, $p = \{p_1, ..., p_r\}$ is the probability vector of actions and $p(n+1) = T[\alpha(n), \beta(n), p(n)]$ is the learning algorithm. The following algorithm is a sample of linear learning algorithms. We assume that action α_i is selected at timestep n. In case of desirable response from the environment:

$$
\begin{aligned}
P_i(n+1) &= P_i(n) + a\left[1 - P_i(n)\right] \\
P_i(n+1) &= (1-a) P_i(n) \quad \forall j \quad j \neq i
\end{aligned}
\tag{1}
$$

In case of undesirable response from the environment:

$$
\begin{aligned}
P_i(n+1) &= (1-b) P_i(n) \\
P_j(n+1) &= (b/r - 1) + (1-b) p_j(n) \forall j \quad j \neq i
\end{aligned}
\tag{2}
$$

In equation (1) and (2), a and b are reward and punishment parameters respectively. When a and b are equal, the algorithm is called L_{RP} , when b is much smaller than a, the algorithm is $L_{R\&P}$ and when b is zero, the it is called L_{RI} .

5. Proposed Model

In this part the bees algorithm based on Learning Automata which we called Learning Bees Algorithm (LBA) has been presented. Like the bees algorithm, in this algorithm also exists a colony of the bees and every bee have a primary position that are valued randomly. The difference between proposed algorithm and standard bees algorithm is that the proposed algorithm uses a Learning Automata for controlling bee's behavior. In the proposed model while performing local searches, selection of a bee in order to move toward it, is done by Learning Automata. So the actions of Learning Automata equal the number of worker bees which exist in the colony. Since each bee is selected from all the bees by Learning Automata. In the proposed algorithm at first, the possibility vector of Learning Automata functions are initializes randomly. Then stages below repeated: Learning Automata selects one of its functions based on possibility vector of functions. By considering the selected function, the way for updating the position of bees determined and then the bees update their position. According to the results of updating ingredients position, Learning Automata function assessed and possibility vector of selecting functions are corrected. In the proposed model the population equipped with a Learning Automata which functions as the brain and controller of bee's movement. Learning Automata by choosing a function in each step is determined the way of updating bees' position in that step.

6. Exprimental Results

In [8] on the multimodal functions in low dimensions, the ABC algorithm has been introduced superior in to DE, PSO, and EA. In [9] the algorithm DE/best/1/bin, DE/best/1/exp, ABC and BA on unimodal and multimodal functions in high dimension is compared and BA algorithm and then ABC have been introduced as the best algorithms for efficient finding in multimodal functions and DE/best/1/exp as the best algorithm for efficient finding in high-dimensional functions on a unimodal ones; So we compare the proposed algorithm in low dimensions with ABC and BA, and in high dimensions with E/best/1/exp, BA and the ABC. Researches on unimodal standard test functions of Rosenbrock and Sphere and multimodal functions of Restringing and Ackley is performed on the basis of the equations (3) to (6).

$$f_2(x) = \sum_{i=1}^{D} x_i^2 \quad ,[-100,100] \tag{3}$$

$$f_2(x) = \sum_{i=1}^{D} (100(x_{i+1} - x_i^2)^2) + (x_i - 1)^2) \; ,[-5.12,5.12] \tag{4}$$

$$f_3(x) = 20 + e - 20e() - e^{1/D\sum_{i=1}^{D}\cos(2\Pi o_i)} \quad ,[-10,10] \tag{5}$$

$$f_4(x) = \sum_{i=1}^{D} (x_i^2 - 10\cos(2\Pi 0_i) + 10) \quad ,[-5.12,5.12] \tag{6}$$

The optimum point of all these functions is zero. Bee Algorithm values in accordance with the [8] have been introduced as: m=10%n ؛e=30%m ؛nep=7 ؛nsp=4؛ Cells rearrengment mechanism at the end ofeach iteration is randomly; the number of asleep bees in each cell equals to the number of research problem dimensions; the amount of fines of cells for each time is 0.1 and the top has been supposed 1. The intention of Zero in results is in lower dimensions when the result is less than 1e-50 and in high-dimensional case the result is less than 1e-300. Test mean results of 30 separate algorithms executions are available in tables 1 and 2 and a comparison chart of the average values of their logarithms are shown in “Fig.4” and “Fig.5”. The Population sizes are set as 500, 250 and 100 with 6000,4000 and 500 iterations respectively for 100, 50 and under 30 dimensional spaces.

As can be seen from “Fig.4” and “Fig.5”, the algorithm is able to achieve better results in single-sided functions and the multifaceted functions and also in all dimensions. Learning bee algorithm similar to bee algorithm, to exploit more from the rich sources more worker bees are assigned, But the main reason for the superiority of the proposed learning bees' algorithm is the feature of learning because of which the worker bees have been appointed to conduct random search at the superior gardens smartly. “Fig.6” shows the Effect of different values of parameters a and b on the results of the proposed algorithm and as can be seen, they don't have much impact on the convergence of the algorithm and this indicates the superiority of the learning bees' algorithm, because of its worker bee's ability. Although this clever selection will prevent accidental solutions but this problem is compensated with excavator bees.

TABLE I. MEAN RESULTS OF PROPOSED LBA ALGORITHM FUNCTIONS WITH DIFFERENT PARAMETERS OF A AND B

Algorithm	*Dimension*	*LBA(LRI)*	*LBA(LR&P)*	*LBA(LRP)*
Sphere	10	4.23E-48	1.78E-47	8.13E-47
	20	7.93E-22	2.31E-22	7.74E-22
	30	2.24E-13	2.86E-12	1.01E-12
Rosenbrock	10	1.13E-01	1.43E-01	1.12E-01
	20	1.35E+00	8.07E-01	2.41E-01
	30	6.77E+00	1.35E+01	4.67E+00
Rastrigin	10	0.00E+0	0.00E+0	0.00E+0
	20	3.32E-01	8.02E-09	5.24E-09
	30	4.27E-01	6.07E-01	3.55E-01
Ackley	10	1.07E-14	7.82E-15	9.47E-15
	20	3.46E-11	2.17E-11	7.73E-11
	30	3.21E-07	3.83E-07	2.31E-07

TABLE II. MEAN RESULTS OF UNIMODAL FUNCTIONS IN LOW DIMENSIONS

Algorithm	*Dimension*		*Best*	*Mean*	*Stdev*
BA	10	***Sphere***	1.70E-06	7.16E-06	9.71E-06
	20		6.67E-03	6.67E-03	3.45E-03
	30		1.62E-01	2.32E-01	1.21E-01
ABC	10		5.15E-14	4.44E-05	2.24E-07
	20		3.60E-07	6.89E+01	1.00E+05
	30		1.07E-05	8.36E+02	1.84E+06
LBA	10		0.00E+00	1.78E-47	2.64E-47
	20		7.13E-24	2.31E-22	3.05E-22
	30		4.31E-14	2.86E-12	6.17E-12
BA	10	***Rastrigin***	5.36E-06	9.72E-06	5.36E-06
	20		4.45E-03	1.96E-01	4.45E-03
	30		3.50E-01	1.81E+00	3.50E-01
ABC	10		2.36E-15	6.50E-01	2.36E-15
	20		2.67E-06	6.58E+00	2.67E-06
	30		5.04E+00	1.90E+01	5.04E+00
LBA	10		0.00E+00	0.00E+00	0.00E+00
	20		8.35E-14	8.02E-09	1.60E-08
	30		6.48E-05	6.07E-01	8.87E-01

TABLE III. MEAN RESULTS OF UNIMODAL FUNCTIONS IN HIGH DIMENSIONS

Algorithm	*Dimension*		*Best*	*Mean*	*Stdev*
BA	50	***Sphere***	8.26E-112	1.05E-97	5.67E-97
	100		1.97E-80	2.37E-74	8.62E-74
ABC	50		3.88E-37	2.08E-36	1.18E-36

	100		9.56E-26	3.37E-25	1.73E-25
EXP	50		1.44E-121	1.00E-119	2.26E-119
	100		8.05E-100	2.94E-97	5.50E-97
LBA	50		8.74E-222	2.07E-198	5.67E-97
BA	50	Rastriging	0.00E+00	0.00E+00	0.00E+00
	100		0.00E+00	0.00E+00	0.00E+00
ABC	50		0.00E+00	0.00E+00	0.00E+00
	100		1.78E-15	2.55E-11	1.78E-15
EXP	50		1.49E+01	2.67E+01	6.44E+0
	100		1.47E-14	8.57E-01	1.47E-14
LBA	50		2.77E-142	4.01E-231	2.26E-23
	100		0.00E+00	0.00E+00	0.00E+00

TABLE IV. MEAN RESULTS OF MULTIMODAL FUNCTIONS IN LOW DIMENSIONS

Algorithm	*Dimension*		*Best*	*Mean*	*Stdev*
BA	10	Roenbrock	3.89E-01	5.64E-01	9.71E-06
	20		3.31E+00	5.01E+00	1.18E+00
	30		9.70E+00	3.28E+01	9.71E+02
ABC	10		1.00E-04	4.04E+00	3.26E+01
	20		6.20E-03	9.46E+00	1.59E+02
	30		2.05E-02	2.74E+02	1.45E+06
LBA	10		3.27E-03	1.43E-01	1.27E-01
	20		8.67E-02	8.07E-01	1.41E+00
	30		5.33E-01	1.35E+01	3.11E+01
BA	10	Ackley	9.72E-06	2.21E-04	9.72E-06
	20		1.96E-01	8.21E-03	1.96E-01
	30		1.81E+00	4.13E-02	1.81E+00
ABC	10		6.50E-01	1.11E-07	6.50E-01
	20		6.58E+00	3.20E-01	6.58E+00
	30		1.90E+01	1.88E+00	1.90E+01
LBA	10		7.11E-15	7.82E-15	1.59E-15
	20		2.75E-12	2.17E-11	1.12E-03
	30		1.05E-07	3.83E-07	1.95E-07

TABLE V. MEAN RESULTS OF MULTIMODAL FUNCTIONS IN HIGH DIMENSIONS

Algorithm	*Dimension*		*Best*	*Mean*	*Stdev*
BA	50	Roenbrock	2.85E-4	8.47E-3	1.61E-2
	100		3.48E-03	1.29E-01	2.71E-01
ABC	50		3.98E-03	3.18E-02	3.31E-02
	100		1.51E-01	5.33E-01	3.17E-01
EXP	50		0.00E+00	2.66E-01	9.94E-01
	100		4.46E-29	3.99E-01	1.20E+00
LBA	50		1.55E-38	4.45E-29	1.22E-5
BA	50	Ackley	5.02E-14	6.70E-14	6.80E-15
	100		0.00E+00	0.00E+00	0.00E+00
ABC	50		5.72E-14	6.40-14	4.24E-15
	100		2.55E-11	1.78E-15	2.55E-11
EXP	50		7.55E-15	3.42E-02	1.84E-01
	100		8.57E-01	1.47E-14	8.57E-01

LBA	50	3.76E-16	6.28E-18	6.11E-15
	100	1.84E-11	1.14E-09	1.47E-07

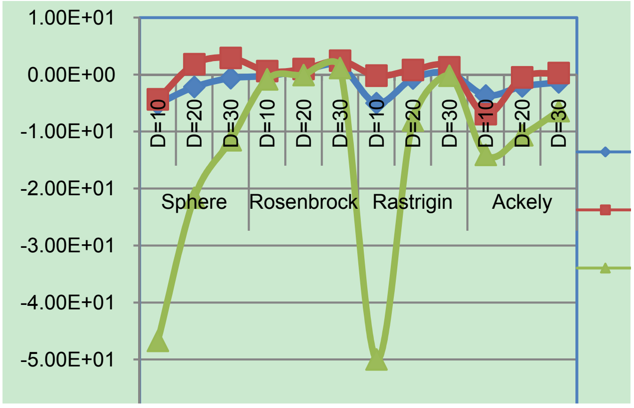

Figure 4. Comparison chart of logarithm of mean values of algorithms in low dimensional spaces

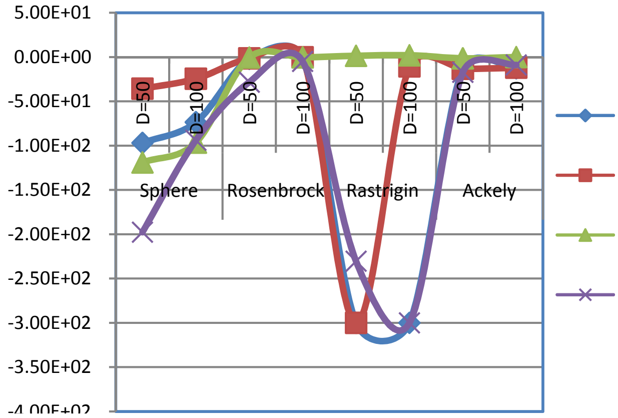

Figure 5. Comparison chart of logarithm of mean values of algorithms in high dimensional spaces

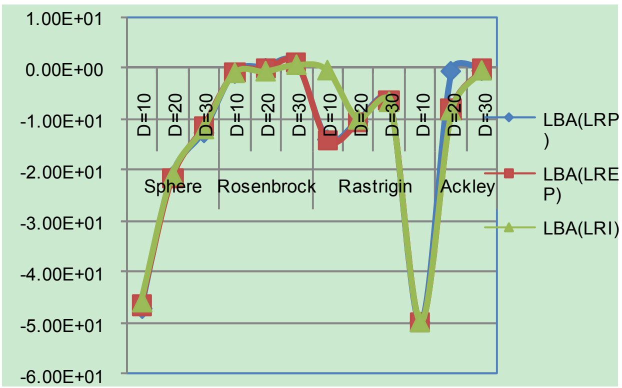

Figure 6. Comparison chart of logarithm of mean values of proposed LBA algorithms with different values of a and b parameters

7. Conclusion

In this paper the bees algorithm based on Learning Automata which we called LBA has been presented that uses a Learning Automata for controlling bee's behavior.

In the proposed model while performing local searches, selection of a bee in order to move toward it, is done by Learning Automata. Learning Automata selects one of its functions based on possibility vector of functions. By considering the selected function, the way for updating the position of bees determined and then the bees update their position. The algorithm tested by the series of well known problem instances. The simulation results shows that the proposed algorithm overcomes others in both unimodal and multimodal cases for all dimensions.

8. References

[1] R. C. Eberhart, Y. Shi and J. Kennedy, "Swarm Intelligence", In Proceedings of the Morgan Kaufmann Series on Evolutionary Computation, USA, 1st Edition, 2001.

[2] D. T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim and M. Zaidi, "The Bees Algorithm", Technical Note, Manufacturing Engineering Centre, Cardiff University, UK, 2005.

[3] D. T. Pham, E. Kog, A. Ghanbarzadeh, S. Otri, S. Rahim and M. Zaidi, "The Bees Algorithm–A Novel Tool for Complex Optimization Problems", In Proceedings of the Intelligent Production Machines and Systems (IPROMS) Conference, Pages 454–461, 2006.

[4] A. Nowe, K. Verbeeck, M. Peeters, "Learning Automata as a Basis for Multi-Agent Reinforcement Learning", Proceedings of First International Workshop on Learning and Adaptation in Multi-Agent Systems (LAMAS), Utrecht, the Netherlands, Pages 71-85, 2006.

[5] M. R. Shirazi, M. R. Meybodi, "Application of Learning Automata to Cooperation in Multi-Agent Systems", Proceedings of First International Conference on Information and Knowledge Technology (IKT2003), Pages 338-349, 2003.

[6] O. Martin, A. Odluzko and S. Wolfram, "Algedric Properties of Cellular Automata", Journal of Communications in Mathematical Physics on Springer, 1984.

[7] M. Lessani and M. R. Meybodi, "Rearrangement Mechanism for Genetic Cellular Automata", In Proceedings of 12th Annual CSI Computer Conference of Iran, 2007.

[8] D. Karaboga and B. Akay, "A Comparative Study of Artificial Bee Colony Algorithm", Journal of Applied Mathematics and Computation on SinceDirect, Pages 108-132, 2009.

[9] H. Li, K. Liu and X. Li, "A Comparative Study of Artificial Bee Colony, Bees Algorithms and Differential Evolution on Numerical Benchmark Problems", Journal of Computational Intelligence and Intelligent Systems on Springer, Pages 198-207, 2010.

[10] D. T. Pham, M. Negm and S. Otri, "Using the Bees Algorithm to solve a stochastic optimization problem", in 4th International Virtual Conference on Intelligent Production Machines and Systems (IPROMS), Whittles, Dunbeath, Scotland, 2008.

[11] T. Blackwell and J. Branke, "Multiswarms, Exclusion, and Anti-Convergence in Dynamic Environments", IEEE Transactions on Evolutionary Computation, 2006.

[12] Z. Golmirzaee and M. R. Meybodi, "Cellular Honey Bee Colony", In Proceedings of the 15th Annual CSI Computer Conference (CSICC'10), 2010.

[13] T. Ghanbarzadeh, "Multi-Objective Optimization using the Bees Algorithm", In Proceedings of International Virtual Conference on Intelligent Production Machines and Systems (IPROMS), Cardiff, UK, 2007.

[14] D. Karaboga and B. Basturk, "A powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony (ABC) Algorithm", Journal of Global Optomization, Pages 459-471, 2007.

[15] D. Karaboga and B. Basturk, "On the Performance of Artificial Bee Colony (ABC) Algorithm", Journal of Soft computing, Pages 687-697, 2008.