JACET Journal of Advances in Computer Engineering and Technology, 1(2) 2015

Adaptive Rule-Base Influence Function Mechanism for Cultural Algorithm

Vahid Seydi Ghomsheh[1], Mohamad Teshnehlab[2], Mehdi Aliyari Shoordeli[3]

Received (2015-01-31)
Accepted (2015-03-12)

Abstract - **This study proposes a modified version of cultural algorithms (CAs) which benefits from rule-based system for influence function. This rule-based system selects and applies the suitable knowledge source according to the distribution of the solutions. This is important to use appropriate influence function to apply to a specific individual, regarding to its role in the search process. This rule based system is optimized using Genetic Algorithm (GA). The proposed modified CA algorithm is compared with several other optimization algorithms including GA, particle swarm optimization (PSO), especially standard version of cultural algorithm. The obtained results demonstrate that the proposed modification enhances the performance of the CA in terms of global optimality.**

Keywords - **Cultural Algorithm (CA), global optimization, knowledge Sources, rule-based system.**

1- Faculty of Computer Engineering, Artificial Intelligence Department, Islamic Azad University, Science and Research Branch, Tehran, Iran (v_seydi@azad.ac.ir).
2- Faculty of Electrical Engineering, Control Department, K. N. Toosi University of Tech., Tehran, 19697, Iran (teshnehlab@eetd.kntu.ac.ir).
3- Faculty of the Department of Mechatronics, K. N. Toosi University of Tech., Tehran, 19697, Iran (aliyari@eetd.kntu.ac.ir).

1. INTRODUCTION

Optimization is an important issue in different scientific applications. Many researches dedicated to algorithms that can be used to find an optimal solution for different applications. Intelligence optimizations which are generally classified as, evolutionary computations techniques like Genetic Algorithm, evolutionary strategy, and evolutionary programming, and swarm intelligence algorithms like particle swarm intelligence algorithm and ant colony optimization, etc are powerful tools for solving optimization problems [1]-[4]. Similar to particle swarm optimization and Ant algorithm in which members share their experiences, cultural algorithms (CAs) try to model social intelligence based on natural cultural evolution to solve the optimization problems [5]-[6][13].

When facing an engineering optimization problem with extensive domain knowledge that cannot be easily integrated into the population level, the CA is a preferred optimization method. In order to study the performance of CA, this algorithm is applied to a number of benchmark optimization problems [7] as well as a number of diverse application areas. For example the efficacy of a variant of CA is tested on Loney's solenoid design which is an electromagnetic engineering problem [8]. In addition, in [9] authors applied the CA to the problem of optimizing the design of a Pressure Vessel which is a benchmark engineering design problem to evaluate the performance of particle swarm. CA is also applied to some other engineering applications such as prediction by functional link-based neural fuzzy network [10], optimizing the tension/ compression of a spring weight [11] and modeling the evolution of agriculture [12].

Three major components of CA are population space, belief space and a protocol that defines the relationship between the population and the belief space. Population space can be modeled based on any population-based computing models such as genetic algorithm, evolutionary programming and particle swarm optimization method [10]. The belief space which is made up of a set of knowledge sources is used to guide the population towards the optimal solution. By taking the feedback from the population, the belief space is updated. This mutual interaction between population and belief space continues until the stop criteria of the algorithm is visited. The mentioned protocol consists of acceptance function and influence function. Acceptance function determines which individuals are to impact the belief space, while influence function which encode the beliefs, evolutes the individuals.

An intelligent optimization algorithm should be able to use the information obtained from the search process in the state space in order to determine an appropriate role for the individuals. By determination, we mean that, when an individual is better to explore, exploit or jumping out of the locals. In CAs, each of the knowledge sources, cause different roles for individuals, and while the next individual position is determined based on one of the influence functions of knowledge sources, it is of a great importance to be able to choose the appropriate knowledge source.

The algorithm proposed for selecting the knowledge sources in cultural algorithms so far are based on the roulette wheel mechanism. In that mechanism, the likelihood of using one of the knowledge sources is based on size of the area under the wheel and the area for a knowledge source is adjusted by the predator based upon the performance of those individuals that are influenced [9].

In this study, we try to select the appropriate knowledge source to update the positions of individuals by using a rule-based system and evolutionary state estimation (ESE) factor proposed by Zhi-Hui Zhan [13]. The ESE factor presents the distribution of the individuals in the search space. Using the obtained information from the behavior of the individuals and a rule based system determines which knowledge source should be selected. The most important feature of this rule-based table is that it is updated using the feedback gained from the search space. This update mechanism continues until the rule-based system learns the features of behavior of individuals in the search space and is parallel to the search mechanism. The simulation results show that convergence of the update of rule-base is faster than the search mechanism. In other words, this rule based finds its optimal values before the search process ends. As soon as the rule-based converges to its optimal state, the update process stops and the search mechanism uses the optimized rule-based to find the optimal solution. The update algorithm for the rule base is based on GA and is presented in details in section 3.

The proposed cultural algorithm is evaluated on six unimodal and multimodal benchmark functions. The algorithm is compared with several other optimization algorithms such as previous version of CA, PSO and GA. The obtained results show that CA which benefits from a rule-based system outperforms these algorithms in terms of global optimality. In section 2 the components of CA are reviewed. Section 3 presents the optimization procedure of rule based system in details. In section 4 we experimentally compare the proposed algorithm with existing CA and various optimization algorithms using a set of benchmark functions. Discussions and further investigation on the proposed method are made in this section. Finally conclusion points are pointed out in section 5.

2. CULTURAL ALGORITHM OVERVIEW

In this section, we describe the traditional CA. The key idea behind CAs is to store and update the problem solving knowledge with the feedback from the population and guide the search using this knowledge [13]. The components of CA are the population space, the belief space, the acceptance function, and the influence function. These major components of CA are depicted in Fig. 1.

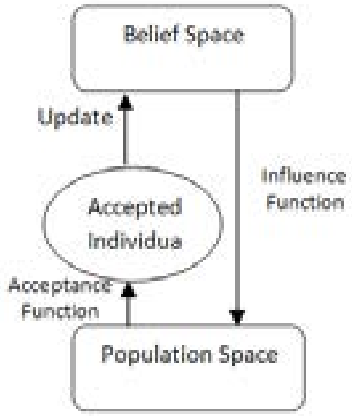


Figure 1. The CA framework

A. The belief space

The experience of individuals are used and stored in information repository called belief space. These experiences can be used by other individuals. In other words, the members of the population share their experiences in the belief space and the knowledge is extracted from these experiences. The benefit of CA over other evolutionary algorithms is that other than sharing the information with offspring the information is shared with other members of the group. Cultural Algorithms employ a set of knowledge sources which are characterized by their appearance in the problem solving process. Reynold [9] identified five basic categories of Knowledge. Each of which are added in different time to achieve a various problem solving capabilities [7], [15, 16]. These five knowledge sources are normative knowledge, situational knowledge, domain knowledge, history knowledge and topographical knowledge. The range of acceptable behaviors in each generation is represented by normative knowledge [7]. Situational knowledge keeps exemplars of successful solutions. Relationships and interactions between the objects in the domain are kept in the domain knowledge source [15]. Temporal and special patterns of behavior are stored in history and topographical knowledge sources respectively [15], [16]. Any cultural knowledge can be expressed as some combination of the five [9].

B. The population space

The population space consists of a set of possible solutions to the problem, and can be modeled using any population-based approach. The population model used here is a simple evolutionary algorithm where each individual is a vector of real-valued variables. In each generation individuals are evolved by using evolutionary operators of a specific knowledge source. In CA, only mutation is used while recombination is taken place in the Belief Space as a result of the updating process. Each knowledge source specifies a different mutation operator. It is also possible to define a mutation operator with the combination of several different knowledge sources.

C. The Acceptance Function

The acceptance function determines which individuals and their behaviors can impact the belief space knowledge. The number of individuals which impact the belief space can range between 1% and 100% of the population size, based on the selected criteria.

D. Influence function

Individuals can be mutated by the influence function. The influence function is composed of methods which each method belong to a specific knowledge source. These methods define the mutation operator in the population space. The strategies proposed to select the proper influence functions for the CAs so far are based on the roulette wheel by adjusting the wheel area assigned to each type of influence function.

In the first stage roulette wheel is divided into equal spaces by different types of influence function. Each area in the roulette wheel is updated using the fitness obtained by the individuals which mutated by the corresponding influence function. A specific influence function is used with the probability computed by the area of the roulette wheel. The division of the roulette wheel is updated based upon performance of the influence function. The performance of an influence function is defined as follow:

$$avr_i = \frac{\sum_{j=1}^{k} f_{j(x)}}{k} \quad (1)$$

where k is the number of individuals generated by the influence function i; $f\ i(x)$ is the fitness value of the individual i. Next, division of roulette wheel for each influence operator is adjusted by the average performance computed for all the influence functions:

$$p_i = \frac{avr_i}{\sum_{j=1}^{n} avr_j} \quad (2)$$

where pi is the probability of the influence operator i and n is the number of influence operators.

3. Rule Base Influence function

In this section we introduce the proposed adaptive rule-based system for influence function. As mentioned earlier, in the CA the next position of individuals are determined based on the influence functions regarding to the knowledge sources, we mean that in CA, each of the knowledge sources, cause different roles for individuals and it is important to choose the suitable roles for individuals based on the information obtained from the search process. This section presents the ESE approach with a fuzzy classification proposed by Zhi-Hui Zhan [13] which extracts the information of search process. Then, we discuss how the ESE approach can be applied to rule-based system to determine the appropriate influence function in CA and how this rule-based system can be updated by obtained feedback of individuals.

A. The ESE approach

The way in which individuals are scattered in the space in called evolutionary state [13].

As an example at the early stage of the CA process individuals are typically scattered in a diverse manner by the topographical knowledge, therefore the evolutionary state is dispersive wherein the role of individuals are exploratory.

While the individuals are being evolutes the population distribution dynamically varies. Now if one can extract information of these distributions of the individuals, then this information can be effectively uses to guide individuals by designating the suitable influence function. The notation of evolutionary states and clustering analysis method was first proposed in [17-18].

The first phase of the evolutionary state estimation is the calculation of the average distance of each particle with respect to all other particles. Estimation of the states can be arranged as the following steps:

Step 1: Calculate the mean Euclidean distance of each particle with respect to other particles:

$$d_i = \frac{1}{N-1} \sum_{j=1, j\neq 1}^{N} \sqrt{\sum_{k=1}^{D} \left(x_i^k - x_j^k\right)^2} \quad (3)$$

where N and D are the population size and the number of dimensions, respectively.

Step 2: Calculate the “evolutionary factor” as follow:

$$f = \frac{d_0 - d_{min}}{d_{max} - d_{min}} \in [0.1] \quad (4)$$

where d_0 is Euclidean distance of the overall best, d_{min} , d_{max} are the minimum and maximum distance among all particles.

B. the rule based system for influence function

In section we introduce how the rule-base system can be determine the type of influence function regarding to evolutionary factor f .in order to classification the evolutionary factor f , Four fuzzy sets s1, s2, s3, s4 are defined in [0, 1], where represent exploration, exploitation, convergence, and jumping out states, respectively. The membership functions are shown in Fig. 2. The mathematical formulations for fuzzy sets to determine membership value of in fuzzy sets and more details are in [13].

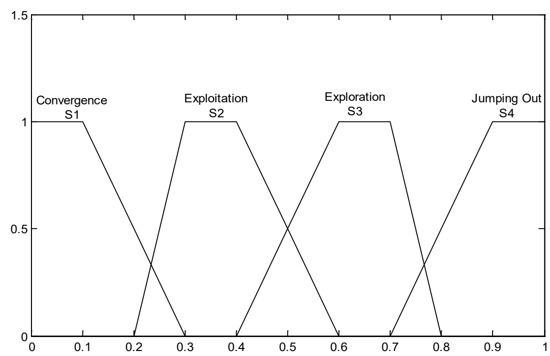


Figure 2.Fuzzy membership functions for the four evolutionary states.

Here two different states may be occurred, on member ship function is activated or two membership functions are activated. When two memberships will be activated at the same time, the decision-making rule base involves both the pervious state (PS) and the "change of state" (COS) variables in a 2-D table as seen in Table 1. For example one of the rules (bolded cells) in this table is as follow:

IF *"COS"* IS *"s1-s2"* AND *"PS"* IS *"S3"* THEN, KNOWLEDGE SOURCE IS *"x_7"*,

Table.1. rule base table based on a chromosome in GA.in the case when two membership function activated.

PS	S4	x_4	x_8	x_{12}
	S3	x_3	$\mathbf{x_7}$	x_{11}
	S2	x_2	x_6	x_{10}
	S1	x_1	x_5	x_9
		S1-S2	S2-S3	S3-S4
		COS		

When only one membership function is activated, the "change of state" converts to "current state estimation" and the rule table is simplified as Table 2.

Table 2. Rule base table based on a chromosome in GA.in the case when one membership function activated.

Knowledge source	$\mathbf{x_{13}}$	x_{14}	x_{15}	x_{16}
CSE	S_1	S_2	S_3	S_4

For instance the bolded cell in the Table 2 represents the following rule:

IF *"CSE"* IS *"s1"* THEN, KNOWLEDGE SOURCE IS "x_{13}",

where x_i $1 \le i \le 16$ can be one of five knowledge source.

In order to optimize the rule-base system, binary Genetic Algorithm is employed. The phenotype of a chromosome is represented as follows:

x_1	x_2	x_3	...	x_{16}

where $x_i \in \{N, S, D, H, T\}$.

At first, the populations of the chromosomes are generated using binary random values. The next and important step is how to evaluate the chromosomes. There exists a rule table which determines knowledge source in influence function of CA which corresponds to each chromosome in GA. Figure 3 illustrate that, each chromosome represents a rule table which creates its corresponding generation in CA. Hence for each chromosome in GA, there exists a different offspring population and the number of offspring groups are is the same as the number of chromosomes in GA.

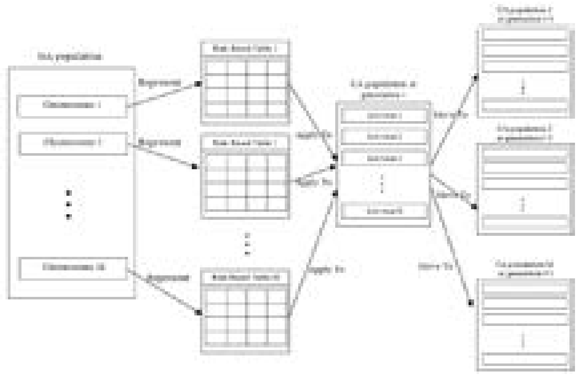


Figure 3 .the block diagram of proposed algorithm

The best individual in each different group of offspring is a representative of corresponding group and its fitness is assigned to chromosome in GA. Now that the fitness of each chromosome is calculated, it is possible to find the best table of rules and based on calculated fitness the GA can evolutes one generation ahead. The best table of the rules based is used as influence function guide to update the population of CA for one generation. This update procedure of GA is iterated until the algorithm is converged. The algorithm is said to be converged if in tGA consecutive number of generations, the best rule-base remains unchanged. When the GA is converged, it is stopped and the CA will be updated using the best rule base found. The Pseudo code of the described algorithm can be briefly presented as follows.

1. Initialize P_{CA} ، P_{GA} randomly (is population of CA and P_{GA} is population of GA).

2. Iterate the following steps until the GA will be converge.

a. Sort the P_{CA} based on fitness function.

b. Select individuals from P_{CA} considering the acceptance function and update the belief space.

c. Calculate the evolutionary factor *f*.

d. for *i=1* to | P_{GA} | (all chromosome in P_{GA}) iterate the following steps:

i. Determine type of knowledge source as influence function by the rule based table corresponding to its chromosome (*i-th* chromosome).

ii. The population of P^{t}_{CA} is mutated by the specific influence function and generate $P^{t+1}_{CA_i}$ where is generation counter and *i* denote to *i-th* offspring of P^{t}_{CA} .

iii. Calculate the fitness function of all individuals in $P^{t+1}_{CA_i}$.

iv. Find the best fitness function in $P^{t+1}_{CA_i}$ and assigned it to current chromosome (chromosome *i*) in P_{GA} as fitness value.

e. Corresponding to best chromosome in P_{GA} , select the P^{t+1}_{CA} as next generation of P_{CA} .

f. Perform the crossover, mutation and selection operators P_{GA} on and the chromosomes evolutes for one generation.

g. Iterate the following steps until the stop criteria is visited

i. Sort the based on fitness function.

ii. Select individuals from P_{CA} considering to the acceptance function and update the belief space.

iii. Calculate the evolutionary factor *f*.

iv. Determine the influence function by the achieved optimal rule based table and mutate the individuals in P_{CA} .

4. Experimental Result

In this section, the proposed algorithm is tested on a number of benchmark functions, and the obtained results are compared with the standard version of CA and well-known optimization methods such as particle swarm optimization (PSO) [23]-[25], genetic algorithms (GA) [26-28].

We have chosen six test functions which are widely used in the nonlinear global optimization literature [20-22]. The function names, formulas, range of variables and the global optima are listed in Table 3. These benchmark functions have a wide variety of different landscapes and present a significant challenge to optimization methods. The Sphere, Schwefel's 2.21, and Quadric Noise are unimodal functions. Rastrigin, Ackley, and Griewank are difficult multimodal functions and their number of local optima increases exponentially with the problem dimension.

In the simulations, the optimization of the rule-based table by the GA is performed in parallel to the CA algorithm until the rule based convergences to its optimal value. While we set the max generations to 500, in practice rule-based table usually converges in less than 200 generations. After convergence of the rule-based table, its optimization process is stopped and CA continues the search process. In the simulations, the dimensions of the benchmark functions are set to 5 and the population size in both CA and GA are set to 10.

Table 4 demonstrates the resulting optimized rule-based tables for difference benchmarks. In the Table 4, N, S, D, H, T are CA knowledge sources which are normative, situational, domain, history and topographical respectively are used by influence function.

For the conditions when only one membership function is activated, the following rule-based table is obtained.

In order to show the convergence of the rule-based in each generation, the absolute value of the difference between the best ruled-based table of the current generation and the previous generation is computed. The convergences of the ruled-based tables are shown in Fig. 4. As can be seen from the figure, in the subfigures relating to f_1, f_3, f_4 and f_6 the error converges before 50 generations, but in the subfigure relating to f_2 this number is 117 and f_5 converges in 62 generations.

In order to compute the convergence of the rule based, the following assignments are considered:

N=1, S=2, D=3, H=4, T=5.

Figure 5 depicts the convergence process of the best individual in each generation. The simulation results of the comparison between proposed CA and the previous methods in five

Table 3. Six Benchmark functions used in this paper. The first three functions are unimodal and the remaining are multimodal

Name of function	Test function	Search space	Global f_{min}
Sphere	$f_1(x)=\sum_{i=1}^{D} x_i^2$	$[-100,100]^D$	$\vec{x}=0, f(\vec{x})=0$
Schwefel2.21	$f_2(x)=\max_i(\lvert x_i \rvert)$	$[-100,100]^D$	$\vec{x}=0, f(\vec{x})=0$
Quadric Noise	$f_3=\sum_{i=1}^{D} i x_i^4 + rand[0,1)$	$[-1.28,1.28]^D$	$\vec{x}=0, f(\vec{x})=0$
Rastrigin	$f_4=\sum_{i=1}^{D}\left[x_i^2-10\cos(2\pi x_i)+10\right]$	$[-5.12,5.12]^D$	$\vec{x}=0, f(\vec{x})=0$
Ackley	$f_5(x)=-20e^{-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2}}-e^{\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)}+20+e$	$[-32,32]^D$	$\vec{x}=0, f(\vec{x})=0$
Griewank	$f_6(x)=\frac{1}{4000}\sum_{i=1}^{D}x_i^2-\prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})+1$	$[-600,600]^D$	$\vec{x}=0, f(\vec{x})=0$

Table 4. Rule base table for six benchmark function.

F1: Sphere

PS \ COS	S1-S2	S2-S3	S3-S4
S4	D	D	H
S3	H	N	S
S2	T	H	N
S1	N	H	S

F2: Schwefel 2.21

PS \ COS	S1-S2	S2-S3	S3-S4
S4	T	H	D
S3	S	H	T
S2	S	D	D
S1	H	N	S

F3: Quadric i.e Noise

PS \ COS	S1-S2	S2-S3	S3-S4
S4	D	S	H
S3	H	H	H
S2	S	H	D
S1	S	S	D

F4: Rastrigin

PS \ COS	S1-S2	S2-S3	S3-S4
S4	H	D	N
S3	T	T	T
S2	S	S	H
S1	N	H	T

F5: Ackley

PS \ COS	S1-S2	S2-S3	S3-S4
S4	N	D	T
S3	N	H	D
S2	H	D	N
S1	H	S	D

F6: Griewank

PS \ COS	S1-S2	S2-S3	S3-S4
S4	H	N	H
S3	S	S	N
S2	S	S	N
S1	S	H	H

Table 5. Rule base table for six benchmark function (one membership function activated).

Sphere

KLG	T	S	H	S
CSE	S1	S2	S3	S4

Schwefel 2.21

KLG	S	S	T	D
CSE	S1	S2	S3	S4

Quadric i.e Noise

KLG	H	S	H	H
CSE	S1	S2	S3	S4

Rastrigin

KLG	S	T	H	S
CSE	S1	S2	S3	S4

Ackley

KLG	S	S	T	T
CSE	S1	S2	S3	S4

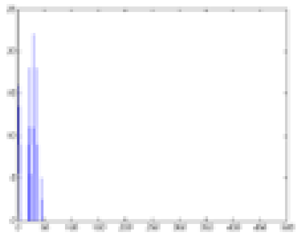
Griewank

KLG	T	N	D	H
CSE	S1	S2	S3	S4

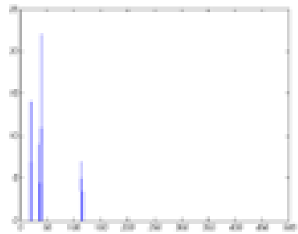
Table 6. The Comparison of the results of the optimizations of six different test functions obtained by proposed CA, CA, PSO and GA. The dimension of the functions is set to 5 and the simulations are repeated for 15 times.

Algorithm		Proposed CA	CA	PSO	GA
f_1	Best	**8.03E-07**	7.11E-06	0.032255	0.001423
	Average	**3.31E-06**	5.07E-05	274.0826	0.015141
	STD	5. 61E-05	4.56E-05	407.2212	0.010216
f_2	Best	**0.000125**	0.003641	2.127886	0.03354
	Average	0.059203	**0.050537**	12.32818	0.099375
	STD	0.070476	0.084336	10.30336	0.041735
f_3	Best	**8.09E-05**	0.000214	0.004779	0.000136
	Average	**0.000906**	0.00154	0.0587	0.00185
	STD	0.000733	0.001077	0.093555	0.001511
f_4	Best	**0.00017**	0.00868	3.98124	0.00039
	Average	**0.00128**	2.729888	14.02944	0.004877
	STD	0.005792	2.352015	6.5129	0.00422
f_5	Best	**0.004678**	0.00869	2.276672	0.007101
	Average	**0.017419**	0.099689	7.307761	0.024232
	STD	0.031176	0.113161	4.51294	0.01357
f_6	Best	**0.065999**	0.409445	0.196867	0.086692
	Average	**0.11139**	8.30541	1.269178	0.138902
	STD	0.186894	6.113189	1.225937	0.040966

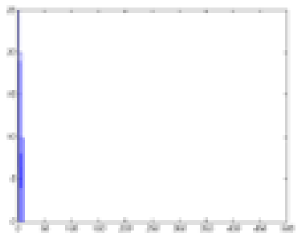
f1: Sphere



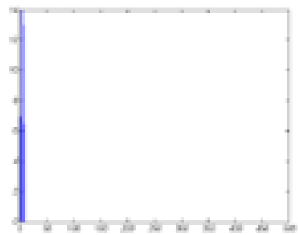
f2: Rastrigin



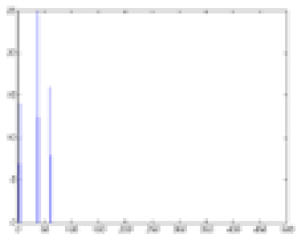
f3: Schwefel 2.21



f4: Ackley



f5: Quadric i.e Noise



f6: Griewank

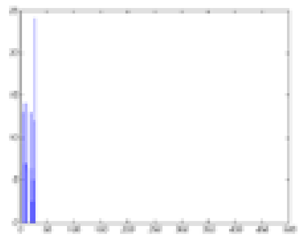
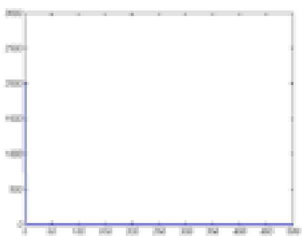
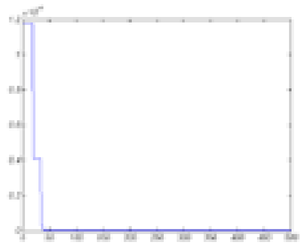


Figure 4: convergence of rule base table.

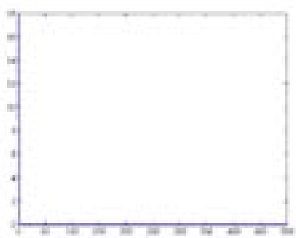
f1: Sphere



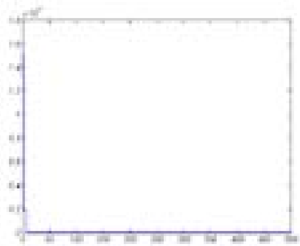
f2: Rastrigin



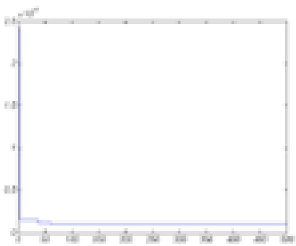
f3: Schwefel 2.21



f4: Ackley



f5: Quadric i.e Noise



f6: Griewank

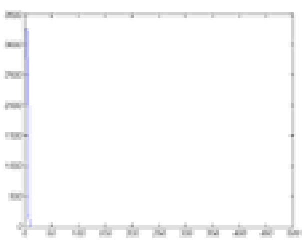


Figure 5: The trend of the convergence of the best individual in CA.

dimensions are given in Table 6. This table illustrates the results of 15 independent runs of each algorithm in terms of the best, mean and overall SD of the solutions. Boldface in the table 6 indicates the best result among those obtained by all contenders. As can be seen from the table, the proposed algorithm outperforms other studied algorithms including GA, PSO and CA.

5. CONCLUSION

In this study, a modified version of CA is introduced and tested. The proposed modified version of CA benefits from a rule-based system in its influence function. The use of the adaptive rule-based system in the influence function of CA makes it possible to update the position of individuals by means of experiences obtained during the optimization process. The adaptation capability makes it possible for the optimization process to identify the optimization environment and makes the moves in a more effective way. In order to optimize the values of the rule-based system used in the proposed algorithm, Genetic Algorithm is utilized. Based on the fact that whether the optimization process is fully within one state or it is in a transition state, different rule is activated. This is the first time this rule-based system is employed to the CA influence function. Moreover, this rule-based system is not dependent upon any expert knowledge and is achieved during the optimization based on the requirement of the optimization environment. This is another major contribution of the current algorithm over what can be seen in the literature.

In order to show the efficacy and superior performance of the proposed algorithm, it is compared with three other well-known optimization methods namely GA, PSO, and existing version of CA using several unimodal and multimodal optimization problems. The obtained results show that the proposed CA can successfully jump out of the local minima of all unimodal and multimodal functions and surpasses all the other algorithms.

REFERENCES

[1] D. E. Goldberg, “Genetic algorithms in search, optimization, and machine learning”. Reading, MA: Addison-Wesley, 1989.

[2] Fogel, D. B. “Evolving Artificial Intelligence. “ Ph. D. Thesis, San Diego, CA: University of California.1992

[3] J. Kennedy and R.C. Eberhart, “Particle swarm optimization,” in Proceeding of the IEEE International Conference on Neural Networks, Perth, Australia, IEEE Service Center, pp. 12–13, 1995.

[4] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant system: Optimization by a colony of cooperating agents,” in IEEE Transactions on Systems, Man, and Cybernetics, vol. 26, no. 1, pp 29–41, 1996.

[5] R.G. Reynolds, “An introduction to cultural algorithms,” in Proceedings of the 3rd Annual Conference on Evolutionary Programming, World Scientific Publishing, pp 131–139, 1994.

[6] R.G. Reynolds, “On modeling the evolution of Hunter-Gatherer decision-making systems,” in Geographical Analysis, vol. 10, no. 1, pp 31–46, 1978.

[7] C. Chung and R.G. Reynolds, “CAEP: An evolution-based tool for real-valued function optimization using cultural algorithms,” in International Journal on Artificial Intelligence Tools, vol. 7, no. 3, pp. 239–291, 1998.

[8] L.S Coelho, P. Alotto,“Electromagnetic Optimization Using a Cultural Self-Organizing Migrating Algorithm Approach Based on Normative Knowledge”, in IEEE Transactions on Magnetics, vol. 45, no. 3, pp 1446-1449, 2009.

[9] R.G. Reynolds, M. Ali, “computing with social fabric The Evolution of Social Intelligence within a Cultural Framework”,In IEEE computational intelligence magazine, 2008

[10] C.J. Lin, C.H. Chen, C.T. Lin,“A Hybrid of Cooperative Particle Swarm Optimization and Cultural Algorithm for Neural Fuzzy Networks and Its Prediction Applications”, in IEEE Transactions on Systems, Man, and Cybernetics —part C, vol. 39, no. 1, pp 55-68,2009

[11] R. G. Reynolds, B. Peng, “Cultural Algorithms: Modeling of How Cultures Learn to Solve Problems” ,in Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI), 2004

[12] R.G. Reynolds, “An adaptive computer model of plant collection and early agriculture in the eastern valley of Oaxaca,” in Guila Naquitz: Archaic Foraging and Early Agriculture in Oaxaca, Mexico, K. V. Flannery Ed, Academic Press, pp. 439–500.

[13] R.Iacoban, R. G. Reynolds, J. Brewste, “Cultural swarms: modeling the impact of culture on social interaction and problem solving,” Proc IEEE Swarm Intelligence Symposiu, Indianapolis,MI, USA, PP. 205-211,2003

[14] Zhi-Hui Zhan, Jun Zhang, Yun Li, and Henry Shu-Hung Chung, “Adaptive Particle Swarm Optimization,” IEEE Transaction on SYSTEMS, MAN, AND CYBERNETICS—PART B: CYBERNETICS, VOL. 39, NO. 6, DECEMBER 2009

[15] R.G. Reynolds and S.M. Saleem, “The impact of environmental dynamics on cultural emergence,” in Perspectives on Adaptions in Natural and Artificial Systems. Oxford University Press, pp. 253–280, 2001.

[16] X. Jin and R.G. Reynolds, “Using knowledge-based

evolutionary computation to solve nonlinear constraint optimization problems: a cultural algorithm approach," in Proceeding of the 1999 Congress on Evolutionary Computation, pp. 1672–1678, 1999.

[17] J. Zhang, H. S.-H. Chung, and W.-L. Lo, "Clustering-based adaptive crossover and mutation probabilities for genetic algorithms," IEEE Trans. Evol. Comput., vol. 11, no. 3, pp. 326–335, Jun. 2007.

[18] Z. H. Zhan, J. Xiao, J. Zhang, and W. N. Chen, "Adaptive control of acceleration coefficients for particle swarm optimization based on clustering analysis," in Proc. IEEE Congr. Evol. Comput., Singapore, Sep. 2007, pp. 3276–3282.

[19] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, Neuro-Fuzzy and Soft Computing. Englewood Cliffs, NJ: Prentice-Hall, 1997.

[20] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," IEEE Trans. Evol. Comput., vol. 10, no. 3, pp. 281–295, Jun. 2006.

[21] X. Yao, Y. Liu, and G. M. Lin, "Evolutionary programming made faster," IEEE Trans. Evol. Comput., vol. 3, no. 2, pp. 82–102, Jul. 1999.

[22] N. Shafiabady, M. Teshnehlab and M. A. Nekoui "Some Applications of S.T. (Shafiabady-Teshnehlab) Evolutionary Optimization Algorithm", International Journal of Innovative Computing, Information and Control (IJICIC), Vol 8, no 2, Feb 2012.

[23] G. Onwubolu and B. Babu, "New Optimization Techniques in Engineering, " Berlin, Germany: Springer-Verlag, 2004.

[24] R. Eberhart, Y. Shi, and J. Kennedy, "Swarm Intelligence, " San Mateo,CA: Morgan Kaufmann, 2001.

[25] Alireza Alfi ,"Particle Swarm Optimization Algorithm with Dynamic Inertia Weight for Online Parameter Identification Applied to Lorenz Chaotic System" ", International Journal of Innovative Computing, Information and Control (IJICIC), Vol 8, no 2, Feb 2012.

[26] T. Back, "Evolutionary Algorithms in Theory and Practice," Oxford,U.K.: Oxford Univ. Press, 1996.

[27] Z. Michalewicz, "Genetic Algorithms +Data Structures = Evolution Programs," New York: Springer, 1992.

[28] D. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine," Learning. Reading, MA: Addison-Wesley, 1989