Commun Nonlinear Sci Numer Simulat 17 (2012) 1312–1319

Contents lists available at SciVerse ScienceDirect

Commun Nonlinear Sci Numer Simulat

journal homepage: www.elsevier.com/locate/cnsns

Imperialist competitive algorithm combined with chaos for global optimization

S. Talatahari [a,*], B. Farahmand Azar [b], R. Sheikholeslami [b], A.H. Gandomi [c]

[a] *Marand Faculty of Engineering, University of Tabriz, Tabriz, Iran*
[b] *Department of Civil Engineering, University of Tabriz, Tabriz, Iran*
[c] *Department of Civil Engineering, University of Akron, Akron, OH, USA*

ARTICLE INFO

Article history:
Received 14 January 2011
Received in revised form 1 August 2011
Accepted 1 August 2011
Available online 25 August 2011

Keywords:
Imperialist competitive algorithm
Chaos
Meta-heuristic algorithms
Global optimization

ABSTRACT

A novel chaotic improved imperialist competitive algorithm (CICA) is presented for global optimization. The ICA is a new meta-heuristic optimization developed based on a socio-politically motivated strategy and contains two main steps: the movement of the colonies and the imperialistic competition. Here different chaotic maps are utilized to improve the movement step of the algorithm. Seven different chaotic maps are investigated and the Logistic and Sinusoidal maps are found as the best choices. Comparing the new algorithm with the other ICA-based methods demonstrates the superiority of the CICA for the benchmark functions.

1. Introduction

The imperialist competitive algorithm (ICA) is one of the recent meta-heuristic optimization techniques. This novel optimization method developed based on a socio-politically motivated strategy. The ICA is a multi-agent algorithm in which each agent is a country and can be either a colony or an imperialist. These countries form some empires in the search space. Movement of the colonies toward their related imperialist and imperialistic competition among the empires forms the basis of the ICA. During these movements, the powerful imperialists are reinforced and the weak ones are weakened and gradually collapsed, directing the algorithm towards optimum points [1]. Imperialistic competition is the main part of the ICA and hopefully causes the colonies to converge to the global minimum of the cost function. This algorithm is proposed by Atashpaz-Gargari and co-workers [2,3]. Kaveh and Talatahari improved the ICA by defining two new movement steps and investigated the performance of this algorithm to optimize the design of skeletal structures [1] and engineering optimization problems [4].

Characterizing the irregular behavior that can be caused either by deterministic chaos or by stochastic processes is not an easy task to perform and it is still an unsolved problem to distinguish among these two types of phenomena. However, the interest in studying the use of chaotic systems instead of random ones arises when the theme of chaos reaches a high interdisciplinary level involving not only mathematicians, physicians and engineers but also biologists, economists and scientists from different areas [5,6]. One of these fields is based on the idea of using chaotic systems for stochastic optimization algorithms [7].

Although chaos and random signals share the property of long term unpredictable irregular behavior and many of random generators in programming softwares as well as the chaotic maps are deterministic; however chaos can help order

* Corresponding author.
E-mail address: siamak.talat@gmail.com (S. Talatahari).

1007-5704/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cnsns.2011.08.021

to arise from disorder. Similarly, many optimization algorithms are inspired from biological systems where order arises from disorder. In these cases disorder often indicates both non-organized patterns and irregular behavior, whereas order is the result of self-organization and evolution and often arises from a disorder condition or from the presence of dissymmetries. Self-organization and evolution are two key factors of many stochastic optimization techniques. Due to these common properties between chaos and optimization algorithms, simultaneous use of these concepts seems to improve the performance. Utilizing chaotic sequences for particle swarm optimization [8], harmony search [9], bee colony [10], and Big Bang – Big Crunch [11] are some familiar examples of this combination. Seemingly the benefits of such combination is a generic for other stochastic optimization and experimental studies confirmed this; although, this has not mathematically been proved yet [12].

This paper adds such positive benefits of the chaos to the ICA algorithm. In this way, generating different chaotic systems substitute random numbers for different parameters of the ICA. Thus different methods that use chaotic maps as efficient alternatives to pseudorandom sequences have been proposed. In order to evaluate these algorithms, some mathematical benchmark examples are utilized. The results reveal the improvement of the new algorithm due to the application of the chaotic signals in place of random sequences.

The rest of the paper is organized as follows. Review of the ICA is presented in Section 2. The chaotic maps that generate chaotic sequences for the ICA are described in Section 3. In Section 4, we introduce different variants of the proposed method which are called chaotic imperialist competitive algorithms (CICA). In Section 5, the proposed methods are tested through benchmark problems, and the simulation results are compared to each other and the standard ICA. Finally, Section 6 is devoted to some conclusions based on the reported comparison analysis.

2. Imperialist competitive algorithm

The meta-heuristic optimization techniques have widely been investigated to reach global optimum of difficult problems. Many of these methods are created by the simulation of the natural processes. Genetic algorithms, particle swarm optimization [13], ant colony optimization [14], harmony search [15] and charged system search [16] are some familiar examples of meta-heuristic algorithms. This paper improves the performance of the imperialist competitive algorithm by using the chaotic maps as a new optimization algorithm contrary to the above mentioned methods which is not based on phenomena from the nature.

2.1. General aspects [2]

The ICA simulates the social political process of imperialism and imperialistic competition. The agents of this algorithm are called “countries”. There are two types of countries; some of the best countries (in optimization terminology, countries with lower cost) are selected to be the “imperialist” states and the remaining countries form the “colonies” of these imperialists. All the colonies of initial countries are divided among the imperialists based on their “power”. The power of each country is inversely proportional to its cost. The imperialist states together with their colonies form some “empires”.

After forming initial empires, the colonies in each empire start moving toward their relevant imperialist country. This movement is a simple model of assimilation policy which was pursued by some of the imperialist states. The total power of an empire depends on both the power of the imperialist country and the power of its colonies. This fact is modeled by defining the total power of an empire as the power of the imperialist country plus a percentage of mean power of its colonies.

Then the imperialistic competition begins among all the empires. Any empire that is not able to succeed in this competition and cannot increase its power (or at least prevent losing its power) will be eliminated from the competition. The imperialistic competition will gradually result in an increase in the power of the powerful empires and a decrease in the power of weaker ones. Weak empires will loose their power and ultimately they will collapse. The movement of colonies toward their relevant imperialist states along with competition among empires and also the collapse mechanism will cause all the countries to converge to a state in which there exist just one empire in the world and all the other countries are colonies of that empire. In this ideal new world, colonies will have the same position and power as the imperialist.

2.2. Original ICA

Each country is formed of an array of variable values and the related cost of a country is found by evaluation of the cost function f_{cost} of the corresponding variables considering the related objective function. Total number of initial countries is set to $N_{country}$ and the number of the most powerful countries to form the empires is taken as N_{imp}. The remaining N_{col} of the initial countries will be the colonies each of which belongs to an empire. In this paper, 10% of countries belong to empires and the remaining is used as colonies. To form the initial empires, the colonies are divided among imperialists based on their power. To fulfill this aim, the normalized cost of an imperialist is defined as

$$C_n = f_{\text{cost}}^{(imp,n)} - \max_i \left(f_{\text{cost}}^{(imp,i)} \right) \tag{1}$$

where $f_{\cos t}^{(imp,n)}$ is the cost of the nth imperialist and C_n is its normalized cost. The initial colonies are divided among empires based on their power or normalized cost, and for the nth empire it will be as follows

$$NC_n = Round\left(\left|\frac{C_n}{\sum_{i=1}^{N_{imp}} C_i}\right| \cdot N_{col}\right) \tag{2}$$

where NC_n is the initial number of the colonies associated to the nth empire which are selected randomly among the colonies. These colonies along with the nth imperialist form the nth empire.

In the ICA, the assimilation policy is modeled by moving all the colonies toward the imperialist. This movement is shown in Fig. 1 in which a colony moves toward the imperialist by a random value that is uniformly distributed between 0 and $\beta \times d$:

$$\{x\}_{new} = \{x\}_{old} + U(0, \beta \times d) \times \{V_1\} \tag{3}$$

where β is a control parameter and d is the distance between colony and imperialist. $\{V_1\}$ is a vector which its start point is the previous location of the colony and its direction is toward the imperialist locations. The length of this vector is set to unity.

In the original ICA, to increase the searching around the imperialist, a random amount of deviation is added to the direction of movement. Fig. 1 shows the new direction which is obtained by deviating the previous location of the country as great as θ which is a random number with uniform distribution.

If the new position of a colony is better than that of the corresponding imperialist (considering the cost function), the imperialist and the colony change their positions and the new location with the lower cost becomes the imperialist.

Imperialistic competition is another strategy utilized in the ICA methodology. All empires try to take the possession of colonies of other empires and control them. The imperialistic competition gradually reduces the power of weaker empires and increases the power of more powerful ones. The imperialistic competition is modeled by just picking some (usually one) of the weakest colonies of the weakest empires and making a competition among all empires to possess these (this) colonies. Based on their total power, in this competition, each of empires will have a likelihood of taking possession of the mentioned colonies.

Total power of an empire is affected by the power of imperialist country and the colonies of an empire as

$$TC_n = f_{\cos t}^{(imp,n)} + \xi \cdot \frac{\sum_{i=1}^{NC_n} f_{\cos t}^{(col,i)}}{NC_n} \tag{4}$$

where TC_n is the total cost of the nth empire and ξ is a positive number. Similar to Eq. (1), the normalized total cost is defined as

$$NTC_n = TC_n - \max_i(TC_i) \tag{5}$$

where NTC_n is the normalized total cost of the nth empire. Having the normalized total cost, the possession probability of each empire is evaluated by

$$P_n = \left|\frac{NTC_n}{\sum_{i=1}^{N_{imp}} NTC_i}\right| \tag{6}$$

When an empire loses all its colonies, it is assumed to be collapsed. In this model implementation where the powerless empires collapse in the imperialistic competition, the corresponding colonies will be divided among the other empires.

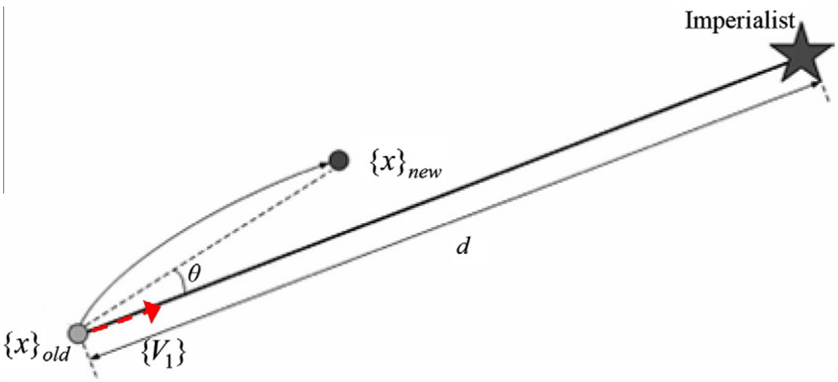

Fig. 1. Movement of colonies to its new location in the original ICA.

Moving colonies toward imperialists are continued and imperialistic competition and implementations are performed during the search process. When the number of iterations reaches a pre-defined value, the search process is stopped.

2.3. Improved ICA

Recently, Kaveh and Talatahari [1] presented an improved ICA. This algorithm is obtained by modifying the movement stage of the original algorithm. Considering the movement process of the ICA, a point out of the colony-imperialistic contacting line can be obtained as indicated in Fig. 2. This algorithm not only uses different random values, but also utilizes the orthogonal colony-imperialistic contacting line instead of θ for deviating the colony as follows

$$\{x\}_{new} = \{x\}_{old} + \beta \times d \times \{rand\} \otimes \{V_1\} + U(-1,+1) \times \tan(\theta) \times d \times \{V_2\}, \quad \{V_1\} \cdot \{V_2\} = 0, \quad \|\{V_2\}\| = 1 \tag{7}$$

where $\{V_2\}$ is perpendicular to $\{V_1\}$, and therefore here after this algorithm is called orthogonal imperialist competitive algorithm (OICA). Since this vector must be crossed the point obtained from the two first terms, we use a random value by using $U(-1,+1)$ for the third term of the Eq. (7) which changes its value in addition to its direction by using negative values.

3. Chaotic maps

Currently, chaos as a kind of dynamic behavior of nonlinear systems has raised enormous interest in different fields of sciences such as chaos control, synchronization, pattern recognition, optimization theory and so on [17]. In random-based optimization algorithms, the methods using chaotic variables instead of random variables are called chaotic optimization algorithm (COA). Optimization algorithms based on the chaos theory are stochastic search methodologies that differ from any of the existing evolutionary computation and swarm intelligence methods. Due to the non-repetition of chaos, it can carry out overall searches at higher speeds than stochastic searches that depend on probabilities [18].

When a random number is needed by the ICA algorithm, it can be generated by iterating one step of the chosen chaotic map (cm) being started from a random initial condition at the first iteration of the ICA. One-dimensional noninvertible maps are the simplest systems with capability of generating chaotic motion [19]. In following subsections, we review some of well-known one-dimensional maps.

3.1. Logistic map

The equation of this map appears in nonlinear dynamics of biological population evidencing chaotic behavior [20],

$$x_{k+1} = a x_k (1 - x_k) \tag{8}$$

In this equation, x_k is the kth chaotic number, with k denoting the iteration number. Obviously, $x \in (0,1)$ under the conditions that the initial $x_0 \in (0,1)$ and that $x_0 \notin \{0.0, 0.25, 0.75, 0.5, 1.0\}$. In the experiments $a = 4$ is used.

3.2. Tent map

The tent map, similar to the logistic map, displays some very specific chaotic effects. This map is defined by the following equation [19]

$$x_{k+1} = \begin{cases} 2x_k & x_k < 0.5 \\ 2(1 - x_k) & x_n \geqslant 0.5 \end{cases} \tag{9}$$

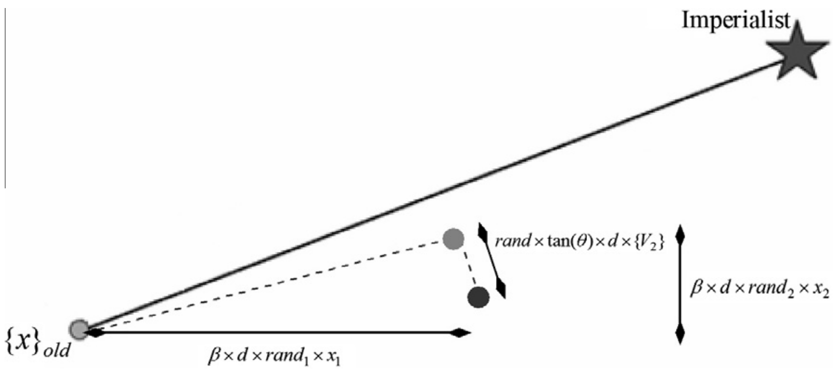

Fig. 2. Movement of colonies to its new location in the improved ICA.

3.3. ICMIC map

The iterative chaotic map with infinite collapses (ICMIC) is defined by

$$x_{k+1} = \sin\left(\frac{a}{x_n}\right) \tag{10}$$

where $a \in (0, \infty)$ is an adjustable parameter [21].

3.4. Sinusoidal map

This iterator [20] is represented by

$$x_{k+1} = a x_k^2 \sin(\pi x_k) \tag{11}$$

For $a = 2.3$ and $x_0 = 0.7$, it has the following simplified form

$$x_{k+1} = \sin(\pi x_k) \tag{12}$$

3.5. Circle map

The Circle map [22] is represented by

$$x_{k+1} = x_k + b - (a/2\pi)\sin(2\pi x_k) \bmod (1) \tag{13}$$

With $a = 0.5$ and $b = 0.2$, it generates chaotic sequence in $(0, 1)$.

3.6. Gauss map

The following equations define Gaussian map [23]

$$x_{k+1} = \begin{cases} 0 & x_k = 0 \\ 1/x_k \bmod (1) & \mathit{otherwise} \end{cases}$$

$$1/x_k \bmod (1) = \frac{1}{x_k} - \left[\frac{1}{x_k}\right] \tag{14}$$

This map also generates chaotic sequences in $(0, 1)$.

3.7. Sinus map

Sinus map is defined as

$$x_{k+1} = 2.3(x_k)^{2\sin(\pi x_k)} \tag{15}$$

4. Chaotic imperialist competitive algorithm

Taking properties of chaos like ergodicity, some new searching algorithms called chaos optimization algorithms (COAs) were presented [24–26]. COA can more easily escape from local minimum point than the stochastic optimization algorithms. The random-based algorithms often escape from local minimum point by admitting some unacceptable solutions with a certain probability. On the contrary, a COA using a chaotic motion can escape from local minimum point. In addition to quasi-stochastic property, the other property of chaos that can be advantageous in the optimization is the sensitivity to the initial condition [12]. Therefore random parameters of the ICA may affect the algorithm performance. As mentioned in Section 2, imperialists countries started to improve their colonies. We have modeled this fact by moving all the colonies toward the imperialist using Eq. (7). In order to increase the searching around the imperialist we use chaotic variables instead of random variables in Eq. (7).

In this paper, sequences generated from chaotic systems substitute the random numbers for the ICA parameters, where it is necessary to make a random-based choice. In this way, it is intended to improve the global convergence and to prevent being trapped in a local solution.

New chaotic ICA (CICA) algorithms may simply be classified and described as follows:

4.1. CICA 1

The vector $\{rand\}$ of Eq. (7) is modified by the selected chaotic maps and the assimilation (moving the colonies of an empire toward the imperialist) equation is modified by

$$\{x\}_{new} = \{x\}_{old} + \beta \times d \times \{cm\} \otimes \{V_1\} + U(-1,+1) \times \tan(\theta) \times d \times \{V_2\}, \quad \{V_1\} \cdot \{V_2\} = 0, \quad \|\{V_2\}\| = 1 \tag{16}$$

where $\{cm\}$ is a chaotic vector based on the selected map.

4.2. CICA 2

Parameter $U(-1,+1)$ of Eq. (7) is modified by the selected chaotic maps and the assimilation equation is modified by

$$\{x\}_{new} = \{x\}_{old} + \beta \times d \times \{rand\} \otimes \{V_1\} + cm \times \tan(\theta) \times d \times \{V_2\}, \quad \{V_1\} \cdot \{V_2\} = 0, \quad \|\{V_2\}\| = 1 \tag{17}$$

where cm is a chaotic variable based on the selected map.

4.3. CICA3

CICA-1 and CICA-2 are combined, that is chaotically modification of $\{rand\}$ and $U(-1,+1)$ values when needed, as

$$\{x\}_{new} = \{x\}_{old} + \beta \times d \times \{cm\} \otimes \{V_1\} + cm \times \tan(\theta) \times d \times \{V_2\}, \quad \{V_1\} \cdot \{V_2\} = 0, \quad \|\{V_2\}\| = 1 \tag{18}$$

5. Numerical examples

To evaluate the efficiency and performance of the proposed algorithms, we used four benchmark functions. These are some well-known mathematical examples presented in Table 1. The algorithms contain the ICA, its improved variant (OICA) and the new variants as described in the previous section.

In order to discover the potential of the algorithms, we define success rate in Eq. (19) as

$$S_r = 100 \frac{N_{successful}}{N_{all}}\bigg|_Q. \tag{19}$$

N_{all} is the number of all trials, $N_{successful}$ is the number of trials which found the solution on the Q, Q is the stopping condition of the algorithm, when it converges into Q tolerance and it is defined as:

$$|f_{\text{cost}}(X_t) - f_{\text{cost}}(X^*)| \leqslant Q \tag{20}$$

where $f_{\text{cost}}(X_t)$ is the cost function in tth iteration and $f_{\text{cos}}(X^*)$ is the global minimum of f. The algorithms were run for 100 times and maximum iteration number was set to 1000.

Using all three variants of the CICA and considering all different chaotic maps described in Section 3, Griewank function with $N = 30$ are solved 100 times for each case. The results of the success rates are shown in Table 2. The results identify that for the all chaotic maps, the performance of CICA-3 is better than two others. CICA-2 takes the second place, however it is weaker than the CICA-3, considerably. Comparing the results of different chaotic maps for each of the variants, it can be concluded that CICA have somewhat shown better performance when Logistic and Sinusoidal maps have been used for generating chaotic signals.

As another investigation and for testing the degree of consistency of the algorithm, the CICA-3 is utilized for all the numerical examples described in Table 1 and compared with ICA and OICA. The simulations are performed 30 times for each algorithm. From recorded results statistical analyses are carried out when maximum iteration number was set to 2000. Tables 3–6 show the best, the worst, the mean of results and the standard deviation for the ICA, the OICA, and the CICA-3 (Sinusoidal map have been used), for Griewank, Ackley, Rosenborck, and Rastring functions. For these examples, the number of variables is set to 10. From the tables, it can be seen that the standard deviation of the results by CICA-3 in 30 independent

Table 1
Specifications of the benchmark problems.

Function name	Definition	Interval	Global minimum
Griewank	$f(X) = 1 + \frac{1}{4000}\sum_{i=1}^{30} x_i^2 - \prod_{i=1}^{30} \cos\left(\frac{x_i}{\sqrt{i}}\right)$	$[-150, 150]$	0.0
Rastring	$f(X) = \sum_{i=1}^{10}(x_i^2 - 10.\cos(2\pi x_i) + 10)$	$[-10, 10]$	0.0
Rosenbrock	$f(X) = \sum_{i=1}^{10-1} 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2$	$[-5, 5]$	0.0
Ackley	$f(X) = -20\exp\left(-0.2\sqrt{\frac{1}{10}\sum_{i=1}^{10} x_i^2}\right) - \exp\left(\sqrt{\frac{1}{10}\sum_{i=1}^{10}\cos(2\pi x_i)}\right) + 20 + e$	$[-32, 32]$	0.0

Table 2
Success rate of CICA algorithms using different chaotic maps for Griewank function ($Q = 10^{-5}$).

	CICA-1	CICA-2	CICA-3
Logistic map	33	49	76
Tent map	18	23	32
ICMIC map	20	34	55
Sinusoidal map	51	68	89
Circle map	10	18	25
Gauss map	8	14	23
Sinus map	14	21	29

Table 3
Statistical results of different methods for Griewank function ($N = 10$).

Methods	Min (best)	Mean	Max (worst)	SD
ICA	2.6990e−11	1.0341e−10	2.6780e−8	8.1404e−10
OICA	1.8260e−14	2.3681e−12	6.6786e−11	1.2168e−11
Present work (CICA3)	1.1707e−16	3.4777e−14	2.5794e−12	5.0708e−15

Table 4
Statistical results of different methods for Ackley function ($N = 10$).

Methods	Min (best)	Mean	Max (worst)	SD
ICA	8.3538e−7	7.1169e−5	9.7544e−5	8.2014e−6
OICA	2.8449e−7	3.3425e−6	6.7409e−5	4.5647e−7
Present work (CICA3)	5.7959e−8	1.0239e−7	5.1388e−6	1.2366e−7

Table 5
Statistical results of different methods for Rosenbrock function ($N = 10$).

Methods	Min (best)	Mean	Max (worst)	SD
ICA	0.001296	0.201608	1.217682	0.362075
OICA	0.000901	0.053585	0.175631	0.043873
Present work (CICA3)	0.000182	0.024174	0.07179	0.021891

Table 6
Statistical results of different methods for Rastring function ($N = 10$).

Methods	Min (best)	Mean	Max (worst)	SD
ICA	0	1.66667e−06	0.00005	9.12871e−06
OICA	0	1.27964e−06	3.83815e−05	7.00742e−06
Present work (CICA3)	0	9.34269e−09	1.06847e−07	3.42961e−08

runs is very small. For all examples, the present algorithm finds better results and also the related standard deviation is far better than two other ICA-based algorithms.

6. Conclusion

A novel improved imperialist competitive algorithm (ICA) using chaotic maps is presented for global optimization. The ICA is a new meta-heuristic optimization developed based on a socio-politically motivated strategy. The ICA contains countries, either a colony or an imperialist, which form some empires. Movement of the colonies and imperialistic competition are the two main steps of the ICA. Modifying each of these steps will affect the performance of the algorithm. Here, we utilize the positive properties of the chaotic maps to improve the orthogonal imperialist competitive algorithm (OICA). The OICA is an improved ICA which defines two movement steps, in direction toward the imperialist locations, and orthogonal to the pervious vector.

Three different variants of the new methodology are defined by adding the chaos to the OICA. In this way, different chaotic systems are utilized instead of different random numbers available in the movement steps of the OICA. For the first variant (CICA-1), the random coefficient vector of $\{V1\}$ is replaced with the chaotic maps. In the second one, the random parameter in the orthogonal vector is replaced with the chaotic maps and finally in CICA-3, both of previous conditions are considered, simultaneously.

Seven different chaotic maps are investigated to recognize the adaptive one for the present algorithm. The results show that Logistic and Sinusoidal maps cause better performance of the CICA than others. In addition, when all random parameters are replaced with the chaotic maps, its performance is far better than the state that only some of them are changed. This means the CICA-3 has better performance than two others.

In order to evaluate the algorithm with its original (ICA) and improved (OICA) variants, some other mathematical benchmark examples are utilized. The results reveal the improvement of the new algorithm due to the application of deterministic chaotic signals in place of random sequences. Comparing the best, the worst and the mean of results as well as the standard deviation for the results of the ICA, the OICA, and the CICA identify that the new algorithm not improves the reliability property due to decrease in the standard deviation but also enhances the quality of the results due to the decrease in the best and the mean of results.

References

[1] Kaveh A, Talatahari S. Optimum design of skeletal structures using imperialist competitive algorithm. Comput Struct 2010;88:1220–9.
[2] Atashpaz-Gargari E, Lucas C. Imperialist competitive algorithm: an algorithm for optimization inspired by imperialistic competition. IEE Cong Evol Comput 2007:4661–7.
[3] Atashpaz-Gargari E, Hashemzadeh F, Rajabioun R, Lucas C. Colonial competitive algorithm: a novel approach for PID controller design in MIMO distillation column process. Int J Intell Comput Cybernet 2008;3(1):337–55.
[4] Kaveh A, Talatahari S. Imperialist competitive algorithm for engineering design problems. Asian J Civil Eng 2010;11(6):675–97.
[5] Ott E, Grebogi C, Yorke J. Controlling chaos. Phys Rev Lett 1990;64(11):1196–9.
[6] Pecora L, Carroll T. Synchronization in chaotic system. Phys Rev Lett 1990;64(8):821–4.
[7] Yang D, Li G, Cheng G. On the efficiency of chaos optimization algorithms for global optimization. Chaos Soliton Fract 2007;34:1366–75.
[8] Alatas B, Akin E, Ozer A. Chaos embedded particle swarm optimization algorithms. Chaos Soliton Fract 2009;40:1715–34.
[9] Alatas B. Chaotic harmony search algorithm. Appl Math Comput 2010;29(4):2687–99.
[10] Alatas B. Chaotic bee colony algorithms for global numerical optimization. Expert Syst Appl 2010;37:5682–7.
[11] Alatas B. Uniform big bang-chaotic big crunch optimization. Commun Nonlinear Sci Numer Simul 2011;16(9):3696–703.
[12] Tavazoei MS, Haeri M. An optimization algorithm based on chaotic behavior and fractal nature. Comput Appl Math 2007;206:1070–81.
[13] Eberhart RC, Kennedy J. A new optimizer using particle swarm theory. In: Proceedings of the sixth international symposium on micro machine and human science, Nagoya, Japan; 1995.
[14] Dorigo M, Maniezzo V, Colorni A. The ant system: optimization by a colony of cooperating agents. IEEE Trans Systems Man Cybernet B 1996;26(1):29–41.
[15] Geem ZW, Kim JH, Loganathan GV. A new heuristic optimization algorithm; harmony search. Simulation 2001;76:60–8.
[16] Kaveh A, Talatahari S. A novel heuristic optimization method: Charged system search. Acta Mech 2010;213(3–4):267–89.
[17] He Y, Zhou J, Xiang X, Chen H, Qin H. Comparison of different chaotic maps in particle swarm optimization algorithm for long-term cascaded hydroelectric system scheduling. Chaos Solitons Fract 2009;42:3169–76.
[18] Coelho L, Mariani V. Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization. Expert Syst Appl 2008;34:1905–13.
[19] Ott E. Chaos in dynamical systems. Cambridge UK: Cambridge University Press; 2002.
[20] May RM. Simple mathematical models with very complicated dynamics. Nature 1976;261:459–74.
[21] He D, He C, Jiang L, Zhu L, Hu G. Chaotic characteristic of a one-dimensional iterative map with infinite collapses. IEEE Trans Circuits Syst 2001;48(7):900–6.
[22] Zheng WM. Kneading plane of the circle map. Chaos Soliton Fract 1994:1221–33.
[23] Bucolo M, Caponetto R, Fortuna L, Frasca M, Rizzo A. Does chaos work better than noise? IEEE Circuits Syst Mag 2002;2(3):4–19.
[24] Yang J. A chaos algorithm based on progressive optimality and Tabu search algorithm. In: Proceedings of the fourth conference on machine learning and cybernetics; 2005. p. 2977–81.
[25] Xu H, Zhu H, Zhang T, Wang Z. Application of mutation scale chaos algorithm in power plant and units economics dispatch. J Harbin Inst Tech 2000;32(4):55–8.
[26] Xu L, Zhou S, Zhang H. A hybrid chaos optimization method and its application. System Eng Electron 2003;25(2):226–8.