Applied Soft Computing 11 (2011) 5508–5518

Contents lists available at ScienceDirect

Applied Soft Computing

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

Cuckoo Optimization Algorithm

Ramin Rajabioun *

Control and Intelligent Processing Centre of Excellence (CIPCE), School of Electrical and Computer Engineering, University of Tehran, Tehran, Iran

ARTICLE INFO

Article history:
Received 17 September 2009
Received in revised form 28 August 2010
Accepted 1 May 2011
Available online 13 May 2011

Keywords:
Cuckoo Optimization Algorithm (COA)
Evolutionary algorithms
Nonlinear optimization

ABSTRACT

In this paper a novel evolutionary algorithm, suitable for continuous nonlinear optimization problems, is introduced. This optimization algorithm is inspired by the life of a bird family, called Cuckoo. Special lifestyle of these birds and their characteristics in egg laying and breeding has been the basic motivation for development of this new evolutionary optimization algorithm. Similar to other evolutionary methods, Cuckoo Optimization Algorithm (COA) starts with an initial population. The cuckoo population, in different societies, is in two types: mature cuckoos and eggs. The effort to survive among cuckoos constitutes the basis of Cuckoo Optimization Algorithm. During the survival competition some of the cuckoos or their eggs, demise. The survived cuckoo societies immigrate to a better environment and start reproducing and laying eggs. Cuckoos' survival effort hopefully converges to a state that there is only one cuckoo society, all with the same profit values. Application of the proposed algorithm to some benchmark functions and a real problem has proven its capability to deal with difficult optimization problems.

1. Introduction

Optimization is the process of making something better. In other words, optimization is the process of adjusting the inputs to or characteristics of a device, mathematical process, or experiment to find the minimum or maximum output or result. The input consists of variables: the process or function is known as the cost function, objective function, or fitness function; and the output is the cost or fitness [1]. There are different methods for solving an optimization problem. Some of these methods are inspired from natural processes. These methods usually start with an initial set of variables and then evolve to obtain the global minimum or maximum of the objective function. Genetic Algorithm (GA) has been the most popular technique in evolutionary computation research. Genetic Algorithm uses operators inspired by natural genetic variation and natural selection [2,3]. Another example is Particle Swarm Optimization (PSO) which was developed by Eberhart and Kennedy in 1995. This stochastic optimization algorithm is inspired by social behavior of bird flocking or fish schooling [3–5]. Ant Colony Optimization (ACO) is another evolutionary optimization algorithm which is inspired by the pheromone trail laying behavior of real ant colonies [3,6,7]. On the other hand Simulated Annealing simulates the annealing process in which a substance is heated above its melting temperature and then gradually cools to produce the crystalline lattice, which minimizes its energy probability distribution [1,8,9]. Besides these well known methods, the investigations on nature inspired optimization algorithms are still being done and new methods are being developed to continually solve some sort of nonlinear problems. In [10], making use of the ergodicity and internal randomness of chaos iterations, a novel immune evolutionary algorithm based on the chaos optimization algorithm and immune evolutionary algorithm is presented to improve the convergence performance of the immune evolutionary algorithm. The novel algorithm integrates advantages of the immune evolutionary algorithm and chaos optimization algorithm. [11] introduces a new optimization technique called Grenade Explosion Method (GEM) and its underlying ideas, including the concept of Optimal Search Direction (OSD), are elaborated. In [12] a new particle swarm optimization method based on the clonal selection algorithm is proposed to avoid premature convergence and guarantee the diversity of the population.

The main advantages of evolutionary algorithms are [3]:

(1) *Being robust to dynamic changes*: Traditional methods of optimization are not robust to dynamic changes in the environment and they require a complete restart for providing a solution. In contrary, evolutionary computation can be used to adapt solutions to changing circumstances.
(2) *Broad applicability*: Evolutionary algorithms can be applied to any problems that can be formulated as function optimization problems.
(3) *Hybridization with other methods*: Evolutionary algorithms can be combined with more traditional optimization techniques.
(4) *Solves problems that have no solutions*: The advantage of evolutionary algorithms includes the ability to address problems for

* Correspondence address: Faculty of Engineering, Campus #2, University of Tehran, Kargar Shomali St., P.O. Box 14395-515, Tehran, Iran. Tel.: +98 9144045713.
E-mail addresses: r.rajabioun@ece.ut.ac.ir, ramin4251@gmail.com

1568-4946/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2011.05.008

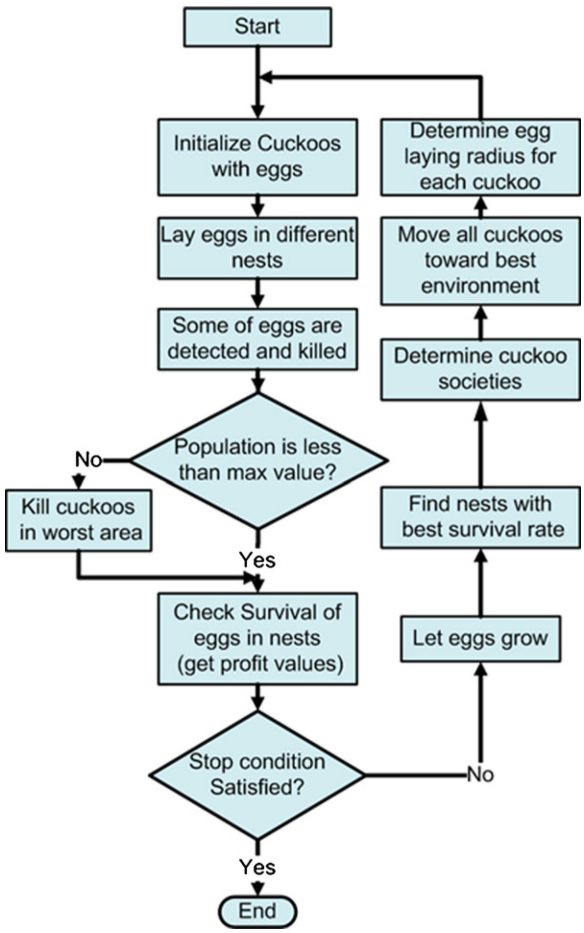


Fig. 1. Flowchart of Cuckoo Optimization Algorithm.

which there is no human expertise. Even though human expertise should be used when it is needed and available; it often proves less adequate for automated problem-solving routines.

Considering these features, evolutionary algorithms can be applied to various applications including: Power Systems operations and control [13,19,20], NP-Hard combinatorial problems [14,15], Chemical Processes [16], Job Scheduling problems [17], Vehicle Routing Problems, Mobile Networking, Batch process scheduling, Multi-objective optimization problems [18], Modeling optimized parameters [21], Image processing and Pattern recognition problems.

In this paper we introduce a new evolutionary optimization algorithm which is inspired by lifestyle of a bird family called cuckoo. Specific egg laying and breeding of cuckoos is the basis of this novel optimization algorithm. Cuckoos used in this modeling exist in two forms: mature cuckoos and eggs. Mature cuckoos lay eggs in some other birds' nest and if these eggs are not recognized and not killed by host birds, they grow and become a mature cuckoo. Environmental features and the immigration of societies (groups) of cuckoos hopefully lead them to converge and find the best environment for breeding and reproduction. This best environment is the global maximum of objective functions. This paper illustrates how the life method of cuckoos is modeled and implemented.

Section 2 investigates the birds called cuckoo and reviews their amazing life characteristics. In Section 3, the Cuckoo Optimization Algorithm (COA) is proposed and its different parts are studied in details. The proposed algorithm is tested with some benchmark functions and also with a controller design of a Multi-Input Multi-Output (MIMO) process as a real case study in Section 4. Finally the conclusions are presented in Section 5.

2. Cuckoos and their special lifestyle for reproduction

All 9000 species of birds have the same approach to motherhood: every one lays eggs. No bird gives birth to live young. Birds quickly form and lay an egg covered in a protective shell that is then incubated outside the body. The large size of an egg makes it difficult for the female to retain more than a single one egg at a time – carrying eggs would make flying harder and require more energy. And because the egg is such a protein-rich high-nutrition prize to all sorts of predators, birds must find a secure place to hatch their eggs. Finding a place to safely place and hatch their eggs, and raise their young to the point of independence, is a challenge birds have solved in many clever ways. They use artistry, intricate design and complex engineering. The diversity of nest architecture has no equal in the animal kingdom. Many birds build isolated, inconspicuous nests, hidden away inside the vegetation to avoid detection by predators. Some of them are so successful at hiding their nests that even the all-seeing eyes of man has hardly ever looked on them.

There are other birds that dispense with every convention of home making and parenthood, and resort to cunning to raise their families. These are the "brood parasites," birds which never build their own nests and instead lay their eggs in the nest of another species, leaving those parents to care for its young. The *cuckoo* is the best known brood parasite, an expert in the art of cruel deception. Its strategy involves stealth, surprise and speed. The mother removes one egg laid by the host mother, lays her own and flies off with the host egg in her bill. The whole process takes barely ten seconds. Cuckoos parasitize the nests of a large variety of bird species and carefully mimic the color and pattern of their own eggs to match that of their hosts. Each female cuckoo specializes on one particular host species. How the cuckoo manages to lay eggs to imitate each host's eggs so accurately is one of nature's main mysteries. Many bird species learn to recognize a cuckoo egg dumped in their own nest and either throw out the strange egg or desert the nest to start afresh. So the cuckoo constantly tries to improve its mimicry of its hosts' eggs, while the hosts try to find ways of detecting the parasitic egg. The struggle between host and parasite is akin to an arms race, each trying to out-survive the other [22].

For the cuckoos suitable habitat provides a source of food (principally insects and especially caterpillars) and a place to breed, for brood parasites the need is for suitable habitat for the host species. Cuckoos occur in a wide variety of habitats. The majority of species occur in forests and woodland, principally in the evergreen rainforests of the tropics. In addition to forests some species of cuckoo occupy more open environments; this can include even arid areas like deserts. Temperate migratory species like the Common Cuckoo inhabit a wide range of habitats in order to make maximum use of the potential brood hosts, from reed beds to treeless moors.

Most species of cuckoo are sedentary, but several species of cuckoo undertake regular seasonal migrations, and several more undertake partial migrations over part of their range. The migration may be Diurnal, as in the Channel-billed Cuckoo, or nocturnal, as in the Yellow-billed Cuckoo. For species breeding at higher latitudes food availability dictates that they migrate to warmer climates during the winter, and all do so. The Long-tailed Koel which breeds in New Zealand flies migrates to its wintering grounds in Polynesia, Micronesia and Melanesia, a feat described as "perhaps the most remarkable over water migration of any land bird" [23]; and the Yellow-billed Cuckoo and Black-billed Cuckoo breed in North America and fly across the Caribbean Sea, a non-stop flight of 4000 km. Other long migration flights include the Lesser Cuckoo, which flies from India to Kenya across the Indian Ocean (3000 km) and the Common Cuckoos of Europe which fly non-stop over the Mediterranean Sea and Sahara Desert on their voyage to southern Africa. Within Africa 10 species make regular intra-continental migrations that are described as polarized, that is they spend the non-breeding season in the tropical centre of the continent and move north and south to breed in the more arid and open savannah and deserts [24].

About 56 of the Old World species and 3 of the New World species are brood parasites, laying their eggs in the nests of other birds [25]. These species are obligate brood parasites, meaning that they only reproduce in this fashion. The cuckoo egg hatches earlier than the host's, and the cuckoo chick grows faster; in most cases the chick evicts the eggs or young of the host species. The chick has no time to learn this behavior, so it must be an instinct passed on genetically. The chick encourages the host to keep pace with its high growth rate with its rapid begging call [26] and the chick's open mouth which serves as a sign stimulus [27]. Female parasitic cuckoos specialize and lay eggs that closely resemble the eggs of their chosen host. This has been produced by natural selection, as some birds are able to distinguish cuckoo eggs from their own, leading to those eggs least like the host's being thrown out of the nest [27]. Host species may engage in more direct action to prevent cuckoos laying eggs in their nest in the first place – birds whose nests are at high risk of cuckoo-contamination are known to mob cuckoos to drive them out of the area [28]. Parasitic cuckoos are grouped into gents, with each gent specializing in a particular host. There is some evidence that the gents are genetically different from one another. Host specificity is enhanced by the need to imitate the eggs of the host.

3. The proposed Cuckoo Optimization Algorithm (COA)

Fig. 1 shows a flowchart of the proposed algorithm. Like other evolutionary algorithms, the proposed algorithm starts with an initial population of cuckoos. These initial cuckoos have some eggs to lay in some host birds' nests. Some of these eggs which are more similar to the host bird's eggs have the opportunity to grow up and become a mature cuckoo. Other eggs are detected by host birds and are killed. The grown eggs reveal the suitability of the nests in that area. The more eggs survive in an area, the more profit is gained in that area. So the position in which more eggs survive will be the term that COA is going to optimize.

Cuckoos search for the most suitable area to lay eggs in order to maximize their eggs survival rate. After remained eggs grow and turn into a mature cuckoo, they make some societies. Each society has its habitat region to live in. The best habitat of all societies will be the destination for the cuckoos in other societies. Then they immigrate toward this best habitat. They will inhabit somewhere near the best habitat. Considering the number of eggs each cuckoo has and also the cuckoo's distance to the goal point (best habitat), some egg laying radii is dedicated to it. Then, cuckoo starts to lay eggs in some random nests inside her egg laying radius. This process continues until the best position with maximum profit value is obtained and most of the cuckoo population is gathered around the same position.

3.1. Generating initial cuckoo habitat

In order to solve an optimization problem, it's necessary that the values of problem variables be formed as an array. In GA and PSO terminologies this array is called "*Chromosome*" and "*Particle Position*", respectively. But here in Cuckoo Optimization Algorithm (COA) it is called "***habitat***". In a N_{var}-dimensional optimization problem, a habitat is an array of $1 \times N_{\text{var}}$, representing current living position of cuckoo. This array is defined as follows:

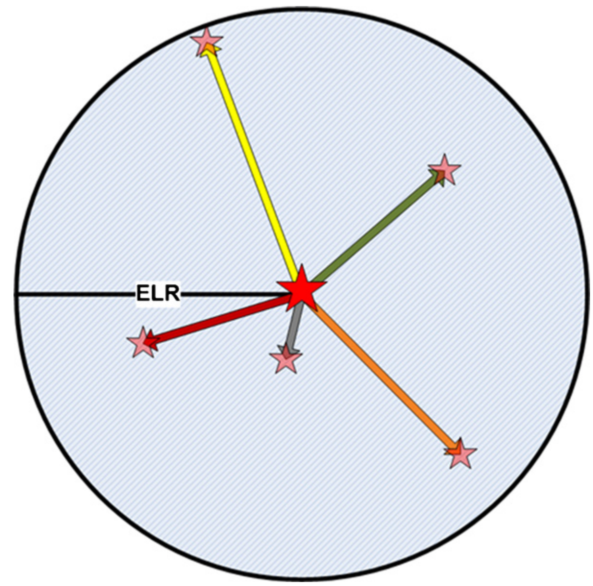


Fig. 2. Random egg laying in ELR, central red star is the initial habitat of the cuckoo with 5 eggs; pink stars are the eggs' new nest.

$$\text{habitat} = [x_1, x_2, \ldots, x_{N_{var}}] \quad (1)$$

Each of the variable values $(x_1, x_2, \ldots, x_{N_{var}})$ is floating point number. The profit of a habitat is obtained by evaluation of profit function f_p at a habitat of $(x_1, x_2, \ldots, x_{N_{var}})$. So

$$\text{Profit} = f_p(\text{habitat}) = f_p(x_1, x_2, \ldots, x_{N_{var}}) \quad (2)$$

As it is seen COA is an algorithm that maximizes a profit function. To use COA in cost minimization problems, one can easily maximize the following profit function:

$$\text{Profit} = -\text{Cost}(\text{habitat}) = -f_c(x_1, x_2, \ldots, x_{N_{var}}) \quad (3)$$

To start the optimization algorithm, a candidate habitat matrix of size $N_{pop} \times N_{\text{var}}$ is generated. Then some randomly produced number of eggs is supposed for each of these initial cuckoo habitats. In nature, each cuckoo lays from 5 to 20 eggs. These values are used as the upper and lower limits of egg dedication to each cuckoo at different iterations. Another habit of real cuckoos is that they lay eggs within a maximum distance from their habitat. From now on, this maximum range will be called "*Egg Laying Radius (ELR)*". In an optimization problem with upper limit of var_{hi} and lower limit of var_{low} for variables, each cuckoo has an *egg laying radius* (ELR) which is proportional to the total number of eggs, number of current cuckoo's eggs and also variable limits of var_{hi} and var_{low}. So ELR is defined as:

$$\text{ELR} = \alpha \times \frac{\text{Number of current cuckoo's eggs}}{\text{Total number of eggs}} \times (var_{hi} - var_{low}) \quad (4)$$

where α is an integer, supposed to handle the maximum value of *ELR*.

3.2. Cuckoos' style for egg laying

Each cuckoo starts laying eggs randomly in some other host birds' nests within her ELR. Fig. 2 gives a clear view of this concept.

After all cuckoos' eggs are laid in host birds' nests, some of them that are less similar to host birds' own eggs, are detected by host birds and though are thrown out of the nest. So after egg laying process, ***p%*** of all eggs (usually 10%), with less profit values, will be

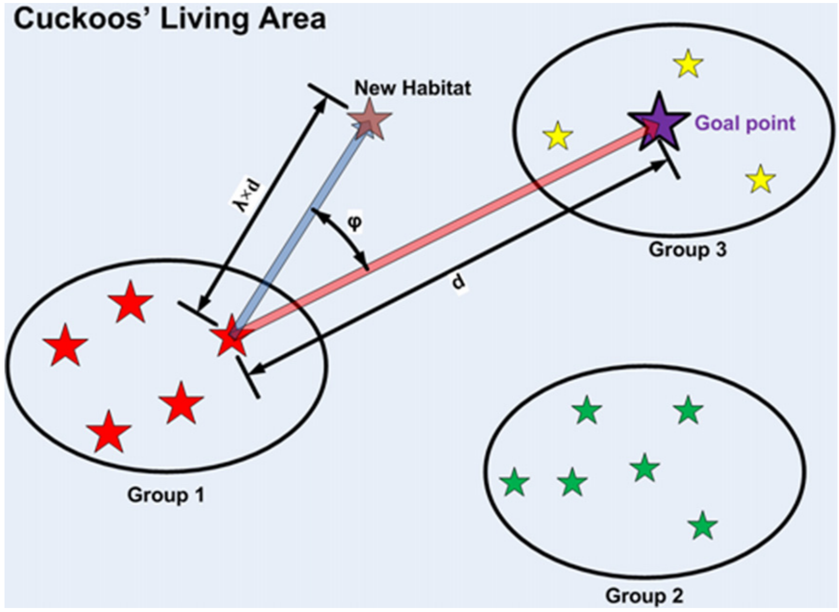


Fig. 3. Immigration of a sample cuckoo toward goal habitat.

killed. These eggs have no chance to grow. Rest of the eggs grow in host nests, hatch and are fed by host birds. Another interesting point about laid cuckoo eggs is that only one egg in a nest has the chance to grow. This is because when cuckoo egg hatches and the chicks come out, she throws the host bird's own eggs out of the nest. In case that host bird's eggs hatch earlier and cuckoo egg hatches later, cuckoo's chick eats most of the food host bird brings to the nest (because of her 3 times bigger body, she pushes other chicks and eats more). After couple of days the host bird's own chicks die from hunger and only cuckoo chick remains in the nest.

3.3. Immigration of cuckoos

When young cuckoos grow and become mature, they live in their own area and society for sometime. But when the time for egg laying approaches they immigrate to new and better habitats with more similarity of eggs to host birds and also with more food for new youngsters. After the cuckoo groups are formed in different areas, the society with best profit value is selected as the goal point for other cuckoos to immigrate. When mature cuckoos live in all over the environment it's difficult to recognize which cuckoo belongs to which group. To solve this problem, the grouping of cuckoos is done with K-means clustering method (a $\boldsymbol{k}$ of 3–5 seems to be sufficient in simulations). Now that the cuckoo groups are constituted their mean profit value is calculated. Then the maximum value of these mean profits determines the goal group and consequently that group's best habitat is the new destination habitat for immigrant cuckoos.

When moving toward goal point, the cuckoos do not fly all the way to the destination habitat. They only fly a part of the way and also have a deviation. This movement is clearly shown in Fig. 3.

As it is seen in Fig. 3, each cuckoo only flies λ% of all distance toward goal habitat and also has a deviation of φ radians. These two parameters, λ and φ, help cuckoos search much more positions in all environment. For each cuckoo, λ and φ are defined as follows:

$$\begin{aligned} &\lambda \sim U(0, 1) \\ &\varphi \sim U(-\omega, \omega) \end{aligned} \tag{5}$$

where $\lambda \sim U(0,1)$ means that λ is a random number (uniformly distributed) between 0 and 1. ω is a parameter that constrains the deviation from goal habitat. An ω of $\pi/6$ (rad) seems to be enough

1. Initialize cuckoo habitats with some random points on the profit function
2. Dedicate some eggs to each cuckoo
3. Define ELR for each cuckoo
4. Let cuckoos to lay eggs inside their corresponding ELR
5. Kill those eggs that are recognized by host birds
6. Let eggs hatch and chicks grow
7. Evaluate the habitat of each newly grown cuckoo
8. Limit cuckoos' maximum number in environment and kill those who live in worst habitats
9. Cluster cuckoos and find best group and select goal habitat
10. Let new cuckoo population immigrate toward goal habitat
11. if stop condition is satisfied stop, if not go to 2

Fig. 4. Pseudo-code for Cuckoo Optimization Algorithm.

for good convergence of the cuckoo population to global maximum profit.

When all cuckoos immigrated toward goal point and new habitats were specified, each mature cuckoo is given some eggs. Then considering the number of eggs dedicated to each bird, an ELR is calculated for each cuckoo. Afterward new egg laying process restarts.

3.4. Eliminating cuckoos in worst habitats

Due to the fact that there is always equilibrium in birds' population so a number of N_{max} controls and limits the maximum number of live cuckoos in the environment. This balance is because of food limitations, being killed by predators and also inability to find proper nest for eggs. In the modeling proposed here in this paper, only those N_{max} number of cuckoos survive that have better profit values, others demise.

3.5. Convergence

After some iterations, all the cuckoo population moves to one best habitat with maximum similarity of eggs to the host birds and also with the maximum food resources. This habitat will produce the maximum profit ever. There will be least egg losses in this best habitat. Convergence of more than 95% of all cuckoos to the same habitat puts an end to Cuckoo Optimization Algorithm (COA). The main steps of COA are presented in Fig. 4 as a pseudo-code. In the next part, COA is applied to some benchmark optimization problems.

Theoretical proofs for convergence to asymptotic probability laws in all stochastic optimization algorithms, considering the Markovian nature of the underlying processes, require some sort of detailed balance or reversibility condition which means the algorithm loses much of its efficiency. Furthermore, if one insists on eventual convergence to the global optima in the strong or even weak sense, very slow annealing is also called for. The strength of stochastic algorithms stem from the fact that their very probabilistic nature ensures that the algorithms will not necessarily get stuck at local optima, and there is no need for using any information on objective gradients, further requiring differentiability conditions.

4. Benchmarks on Cuckoo Optimization Algorithm

In this section the proposed Cuckoo Optimization Algorithm (COA) is tested with 4 benchmark functions from Ref. [1], one 10-dimensional Rastrigin function and a real case study.

4.1. Test cost functions

All the benchmark functions are minimization problems. These functions are listed below:

Function F1:

$$f = x \times \sin(4x) + 1.1y \times \sin(2y)$$
$$0 < x, \quad y < 0, \quad \text{minimum}: \quad f(9.039,\ 8.668) = -8.5547 \tag{6}$$

Function F2:

$$f = 0.5 + \frac{\sin^2\left(\sqrt{x^2+y^2} - 0.5\right)}{1 + 0.1(x^2+y^2)}$$
$$0 < x, \quad y < 2, \quad \text{minimum}: \quad f(0, 0.5) = 0.5 \tag{7}$$

Function F3:

$$f = (x^2+y^2)^{0.25} \times \sin\{30[(x+0.5)^2 + y^2]^{0.1}\} + |x| + |y|$$
$$-\infty < x, \quad y < +\infty, \quad \text{minimum}: \quad f(-0.2, 0) = -0.2471 \tag{8}$$

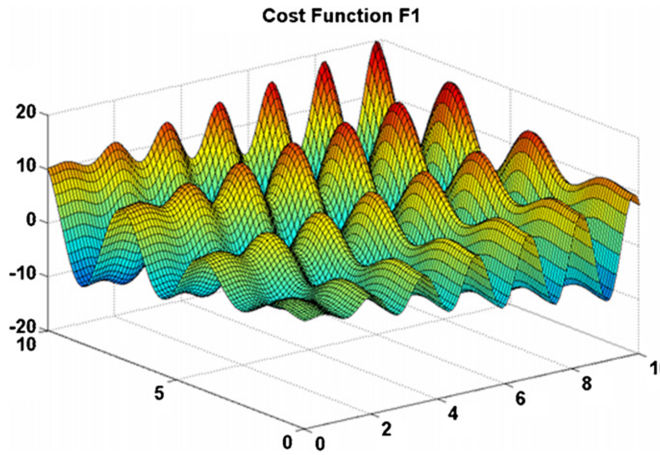


Fig. 5. A 3D plot of cost function F1.

Function F4:

$$f = J_0(x^2+y^2) + 0.1\left|1-x\right| + 0.1\left|1-y\right|$$
$$-\infty < x, \quad y < +\infty, \quad \text{minimum}: \quad f(1, 1.6606) = -0.3356 \tag{9}$$

Function F5 (10-dimensional Rastrigin function):

$$f = 10n + \sum_{i=1}^{n}(x_i{}^2 - 10\ \cos(2\pi x_i)), \quad n = 9$$
$$-5.12 \le x_i \le 5.12, \quad f(0, 0, \ldots, 0) = 0 \tag{10}$$

First function F1 is studied. This function has the global minimum of −18.5547 at $(x, y) = (9.039, 8.668)$ in interval $0 < x$, $y < 10$. Fig. 5 shows the 3D plot of this function.

The initial number of cuckoos is set only to 20. Each cuckoo can lay between 5 and 10 eggs. Fig. 6 shows initial distribution of cuckoos in problem environment.

Figs. 7–12 show the cuckoo population habitats in consequent iterations. Convergence is gained at iteration 7. The COA has obtained the global minimum just in 7 iterations.

As it is seen in Figs. 7–12, cuckoos have found 2 minima at iteration 4. Then in iteration 5 it is seen that one group of cuckoos is immigrating toward the global minimum. In iteration 6 most of cuckoos are in global minimum. And finally at iteration 7 nearly all of cuckoos are on the best habitat, which is the global minimum of the problem. This habitat is (9.0396, 8.6706) with the cost value −18.5543. Fig. 13 depicts the cost minimization for test function F1.

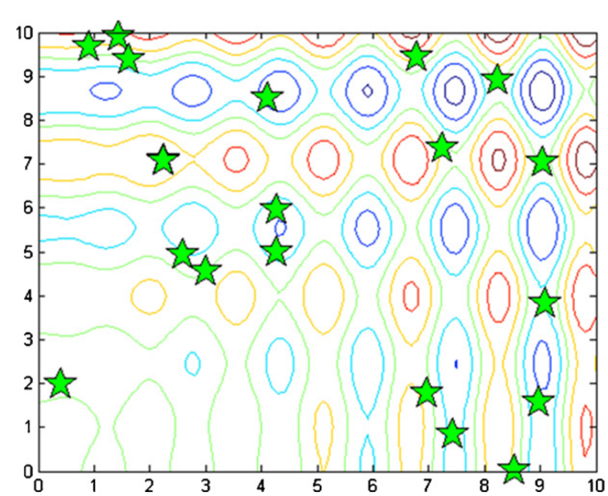


Fig. 6. Initial habitats of cuckoos.

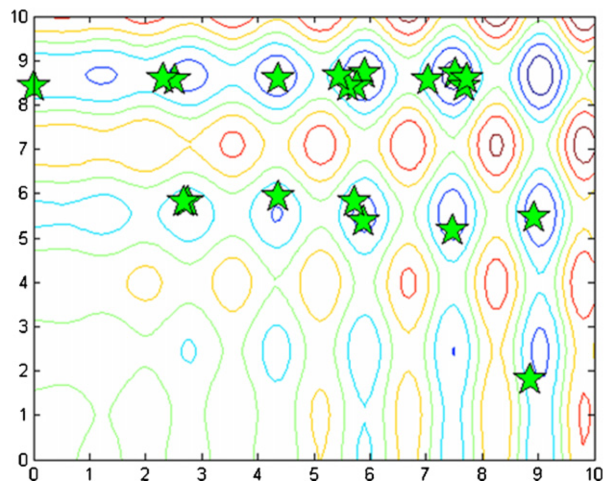


Fig. 7. Habitats of cuckoos in 2nd iteration.

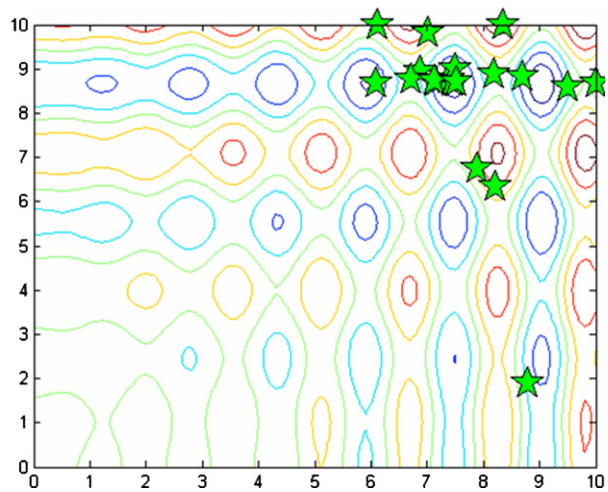


Fig. 8. Habitats of cuckoos in 3rd iteration.

In order to do a comparison, PSO and continuous GA with *Roulette wheel* selection, uniform cross-over are applied to this function too. The initial population of GA is also set to 20, mutation and selection rates are set to 0.2 and 0.5, respectively. For PSO cognitive and social parameters are both set to 2. Due to the fact that different initial populations of each method affect directly to the final result and the speed of algorithm, a series of test runs is done to have a mean expectance of performance for each method.

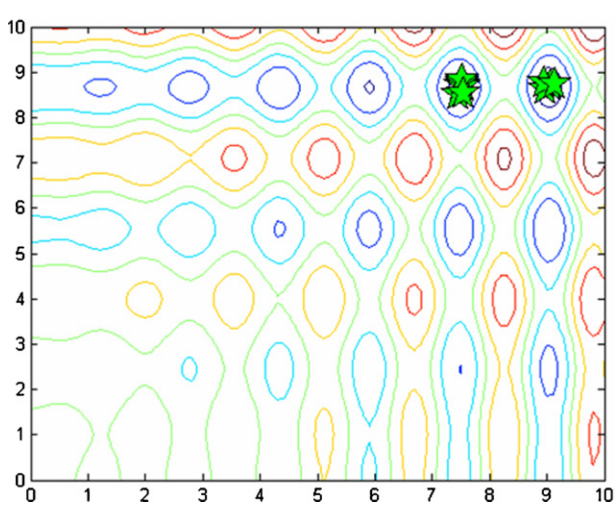


Fig. 9. Habitats of cuckoos in 4th iteration.

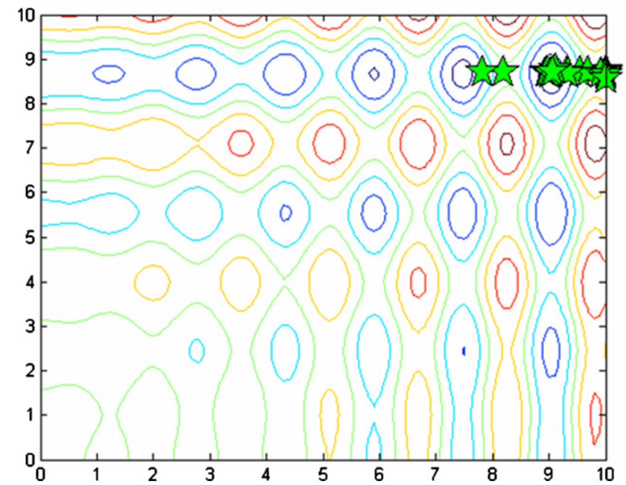


Fig. 10. Habitats of cuckoos in 5th iteration.

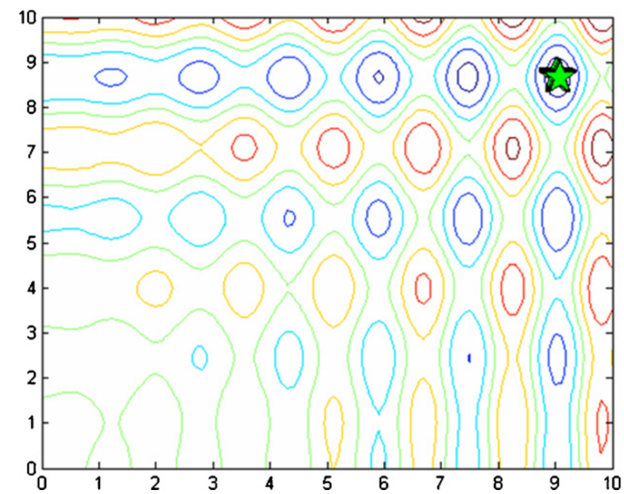


Fig. 11. Habitats of cuckoos in 6th iteration.

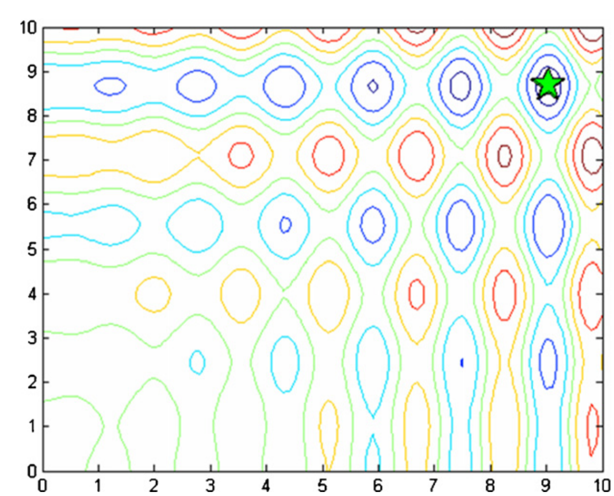


Fig. 12. Habitats of cuckoos in 7th iteration.

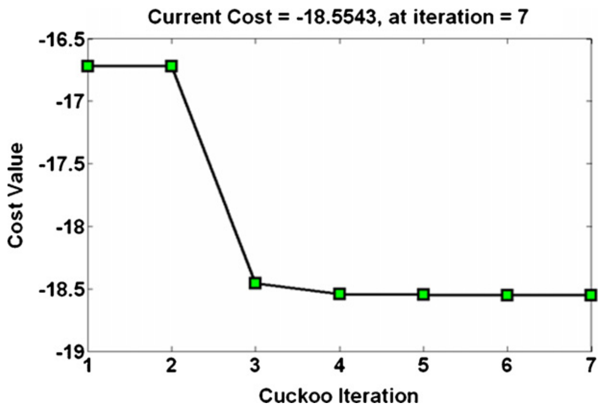


Fig. 13. Cost minimization for test function F1.

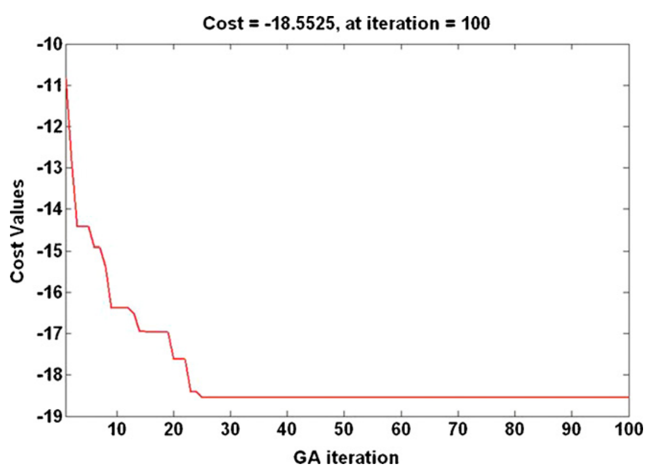


Fig. 14. Cost minimization using GA.

Running the simulations for 30 times produces a mean of 45.9, 38.7 and 6.8 stopping iterations for GA, PSO and COA.

Fig. 14 shows a sample cost minimization plot of function F1 for GA in 100 iterations.

As it is seen from Fig. 14, GA has reached to global minimum at 24th iteration. Best chromosome is (9.0434, 8.6785) and the cost value is −18.5513. Fig. 15 depicts cost minimization of function F1 using PSO.

As it is seen from Fig. 15, PSO has reached to global minimum at 19th iteration. Best particle position is (9.0390, 8.6682) and the cost value is −18.5547. Considering Table 1 it can be seen that while GA and PSO need a mean of 46.8 and 39.1 iterations, COA reaches to the goal point in a mean of 6.9 (approximately 7) iterations. Until now it can be concluded that COA has out performed GA and PSO.

For more test we apply these three optimization algorithms on test functions F2, F3 and F4.

Figs. 16–18 show the cost minimization plot of all three algorithms for test functions F2, F3 and F4 in a random run. Table 1

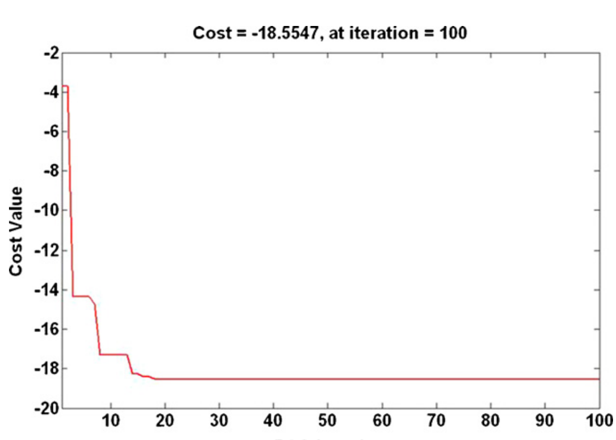


Fig. 15. Cost minimization using PSO.

Table 1
Mean stopping iterations of GA, PSO and COA in 30 runs.

	F2	F3	F4
GA	12.6	52.2	44.1
PSO	10.3	24.8	38.6
COA	5.2	6.9	6.3

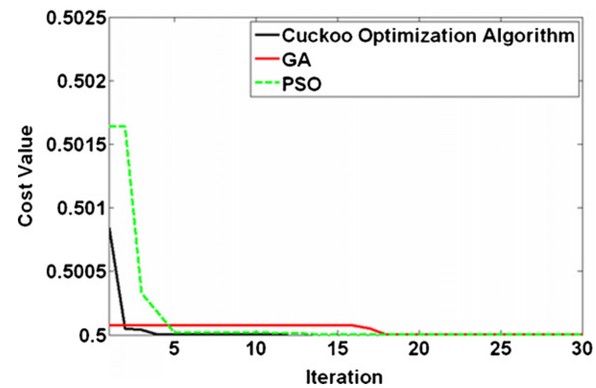


Fig. 16. Cost minimization plot of function F2.

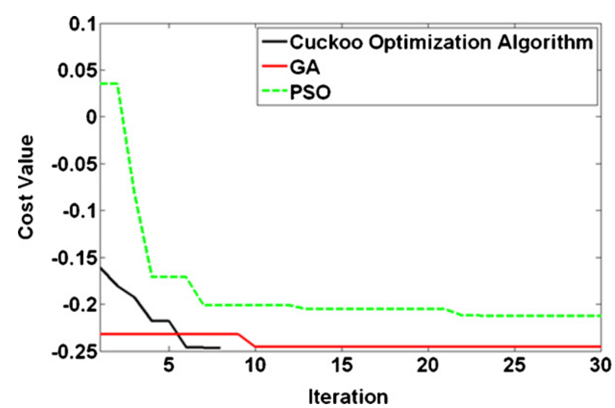


Fig. 17. Cost minimization plot of function F3.

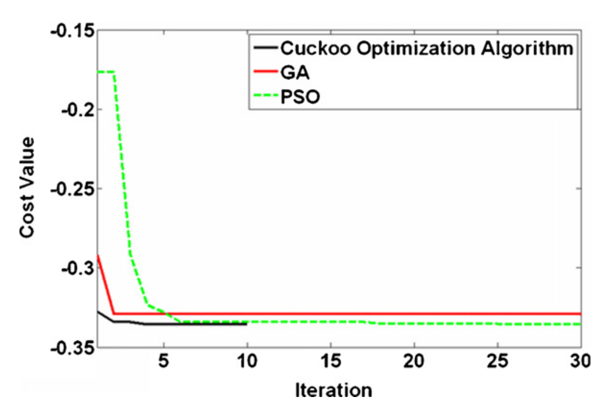


Fig. 18. Cost minimization plot of function F4.

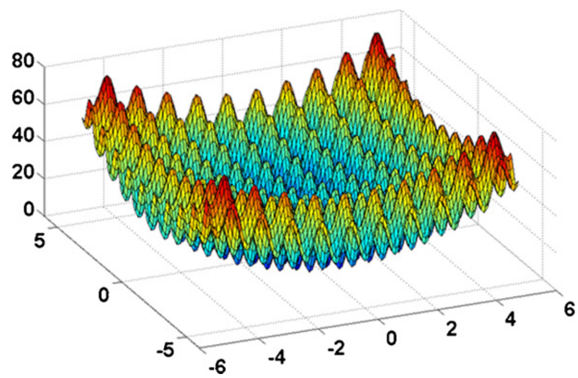


Fig. 19. 3D plot of Rastrigin function.

shows the mean stopping iterations for aforementioned test functions.

The most interesting point seen in Figs. 16–18 and also in Table 1, is *faster convergence* of Cuckoo Optimization Algorithm.

Considering the results obtained for test functions F1, F2, F3 and F4 it can be seen that all three methods have been able to find the global minimum. The only interesting point of Cuckoo Optimization Algorithm (COA) might be its faster convergence. But to show the superiority of COA over GA and PSO, the 10-dimensional Rastrigin function is chosen as test function F5. This function has lots of local minima and is one of the difficult problems to solve, even in 3-dimensional case. Fig. 19 shows the 3-dimensional Rastrigin function.

As it is seen even in 3-dimensional case, the Rastrigin function is a really challenging optimization problem. But to see the real performance of COA, GA and PSO the 10-dimensional Rastrigin function is selected as last benchmark function. Fig. 20 depicts the cost minimization results for all three algorithms. For all three methods, the initial population size and the maximum number of iterations are set to 20 and 100, respectively.

Now it is clearly seen that GA and PSO have not been able to find the global minimum in 100 iterations, but COA has converged in only 66 iterations to almost the global minimum of $f(x^*)=0$. In this benchmark function, COA has stunningly out performed and has found a very good estimation of the real global minimum.

After that the great performance of COA is proven in test cost functions it is needed to investigate its performance in real problems. For this, a Multi-Input Multi-Output (MIMO) distillation column process is chosen in order to be controlled by means of multivariable PID controller. The parameters of PID controller are designed using COA, GA and the method proposed in [29]. Before illustrating the design process a brief description is given about multivariable controller design.

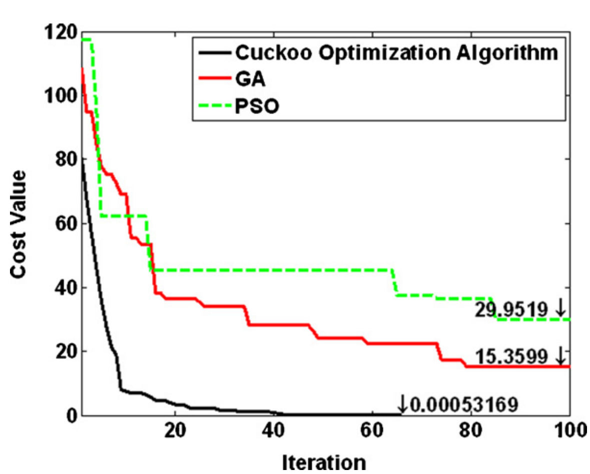


Fig. 20. Cost minimization for 10-dimensional Rastrigin function.

4.2. Multivariable controller design

4.2.1. PID controller for MIMO processes

Consider the multivariable PID control loop in Fig. 21.

In Fig. 21, multivariable process $\mathbf{P}(s)$ could be demonstrated as follows:

$$\mathbf{P}(s) = \begin{bmatrix} p_{11}(s) & \dots & p_{1n}(s) \\ \vdots & \ddots & \vdots \\ p_{n1}(s) & \dots & p_{nn}(s) \end{bmatrix} \tag{11}$$

where $p_{ij}(s)$ is the transfers function between y_i and u_j. In Fig. 21, vectors $\mathbf{Y}_d$, $\mathbf{Y}$, $\mathbf{U}$ and $\mathbf{E}$ are in following form:

$$\mathbf{Y}_d = [\, y_{d1} \quad y_{d2} \quad \cdots \quad y_{dn} \,]^T$$

$$\mathbf{Y} = [\, y_1 \quad y_2 \quad \cdots \quad y_n \,]^T$$

$$\mathbf{U} = [\, u_1 \quad u_2 \quad \cdots \quad u_n \,]^T$$

$$\mathbf{E} = \mathbf{Y}_d - \mathbf{Y} = [\, e_{11} \quad e_{22} \quad \cdots \quad e_{nn} \,]^T$$

Multivariable PID controller $\mathbf{C}(s)$ in Fig. 21, is in the following form:

$$\mathbf{C}(s) = \begin{bmatrix} c_{11}(s) & \dots & c_{1n}(s) \\ \vdots & \ddots & \vdots \\ c_{n1}(s) & \dots & c_{nn}(s) \end{bmatrix} \tag{12}$$

where $c_{ij}(s)$ that $i, j \in \{1,2,\ldots,n\}$ is as follows:

$$c_{ij}(s) = K_{Pij} + K_{Iij}\frac{1}{s} + K_{Dij}s \tag{13}$$

where K_{Pij} is the proportional, K_{Iij} is the integral and K_{Dij} is the derivative gains of the PID controller $c_{ij}(s)$.

4.2.2. Evolutionary PID design

In designing PID controllers, the goal is to tune proper coefficients K_P, K_I and K_d so that the output has some desired characteristics. Usually in time domain, these characteristics are given in terms of overshoot, rise time, settling time and steady state error. Two kinds of performance criteria in output tracking, usually considered in the controller designing, are the integral squared error (ISE) and integral absolute error (IAE) of the desired output.

In design of a multivariable controller, one of the major aims is diagonally domination of the control process. That is the controller be designed in such a way that $y_i(t)$ be able to track the desired input $y_{di}(t)$ and to reject the response of other inputs $y_{dj}(t)$, for $i, j \in \{1,2,\ldots,n \mid i \neq j\}$.

Considering the decupling aim, IAE is defined in the following form:

$$\mathrm{IAE} \triangleq \sum_{i=1}^{n}\sum_{j=1}^{n}\mathrm{IAE}_{ij} \triangleq \sum_{i=1}^{n}\sum_{j=1}^{n}\int_0^{\infty}\left(\left|e_{ij}(t)\right|\right)dt \tag{14}$$

where $|e_{ii}(t)|$ is absolute error of the output $y_i(t)$ when tracking input $y_{di}(t)$ and $|e_{ij}(t)|$ is the absolute error caused by the effect of the input $y_{dj}(t)$, on the output $y_i(t)$, $(i \neq j)$. Also IAE_{ij} is defined as integral of absolute error $e_{ij}(t)$ over time. The source of $|e_{ij}(t)|$ is the coupling problem.

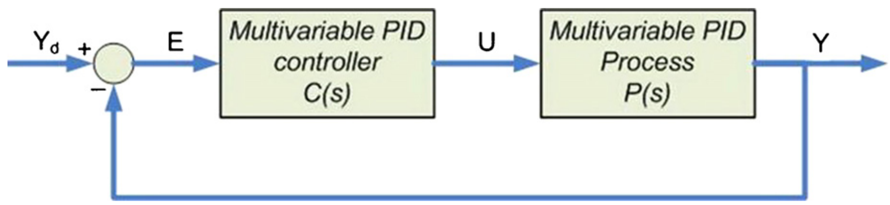


Fig. 21. Block diagram of a multivariable controlled process.

Another performance criteria used in controller design is the Percentage of Overshoot (PO) and Undershoot (PU) which is defined as follows:

$$\mathrm{POU} \triangleq \sum_{i=1}^{n}\sum_{j=1}^{n}\mathrm{POU}_{ij} \triangleq \sum_{i=1}^{n}\sum_{j=1}^{n} Max\{\mathrm{PO}_{ij}, \mathrm{PU}_{ij}\} \quad (15)$$

The aim is to design a controller to track the desired outputs and to decuple controlled process as much as possible. Because POU usually has small values compared with IAE and also to accentuate on POU, we added up 10 times of POU to IAE to build the objective function.

$$\mathrm{Cost} = \mathrm{IAE} + 10 \times \mathrm{POU} \quad (16)$$

Using the proposed evolutionary optimization algorithm, the PID controller parameters are tuned for a typical distillation column process. Obtained results are compared with that of Genetic Algorithm (GA) and with the method introduced in [29], called decentralized relay feedback (DRF).

4.2.2.1. Experimental case study. Here a multivariable PID controller is designed for a MIMO chemical system. This system is a typical 2×2 model of distillation column [30]. A simple schematic of Distillation Column System (DCS) is shown in Fig. 22.

The matrix transfer function of DCS is defined as:

$$\begin{bmatrix} X_D(s) \\ X_B(s) \end{bmatrix} = \begin{bmatrix} \dfrac{12.8\ e^{-s}}{1+16.7s} & \dfrac{-18.9\,e^{-3s}}{1+21s} \\ \dfrac{6.6\,e^{-7s}}{1+10.9s} & \dfrac{-19.4\,e^{-3s}}{1+14.4s} \end{bmatrix} \cdot \begin{bmatrix} R(s) \\ S(s) \end{bmatrix} \quad (17)$$

where $X_D(s)$ and $X_B(s)$ are percentage of methanol in the distillate and percentage of methanol in the bottom products, respectively. Also $R(s)$ and $S(s)$ are reflux flow rate and steam flow rate in the reboiler, respectively.

DCS is a 2×2 MIMO system with strong interactions between inputs and outputs. The four transfer functions in multivariable process have first-order dynamics and significant time delays. The control objectives are (a) tracking the control inputs y_{1d} and y_{2d} by the outputs y_1 and y_2 and (b) diagonally domination of the controlled process as much as possible. In [29] a multivariable PID controller for DCS is designed using decentralized relay feedback (DRF) method. The diagonal and off-diagonal elements of this controller are designed in PI and PID forms, respectively. This controller is as follows:

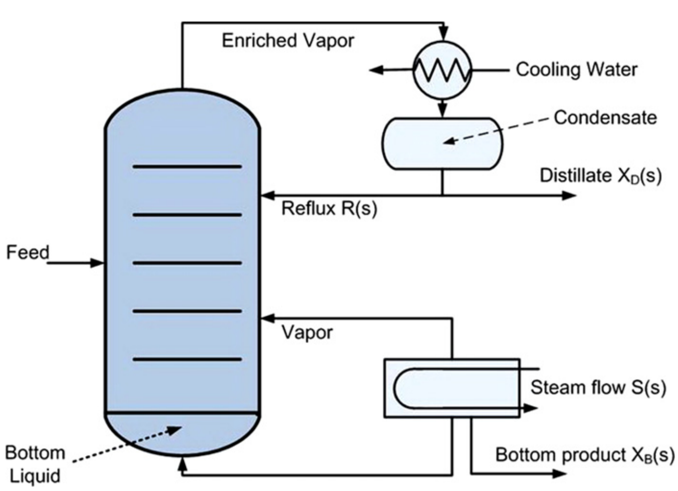


Fig. 22. A simple schematic of distillation column system.

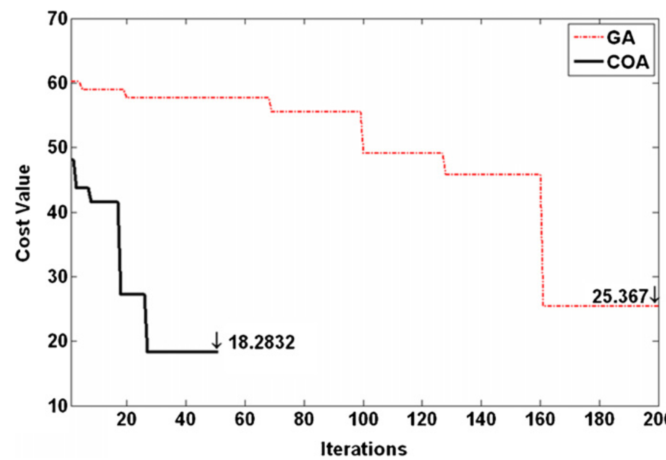


Fig. 23. Minimum cost of COA and GA versus iteration.

$$\mathbf{C}(s) = \begin{bmatrix} 0.184 + 0.0469\dfrac{1}{s} & -0.0102 - 0.0229\dfrac{1}{s} + 0.0082s \\ -0.0674 + 0.0159\dfrac{1}{s} - 0.0536s & -0.066 - 0.0155\dfrac{1}{s} \end{bmatrix} \quad (18)$$

To compare the results of COA and GA with DRF method, in tuning parameters of the PID controller for the plant defined by (17), controller $\mathbf{C}(s)$ is considered as the following form.

$$\mathbf{C}(s) = \begin{bmatrix} K_{P11} + K_{I11}\dfrac{1}{s} & K_{P12} + K_{I12}\dfrac{1}{s} + K_{D12}s \\ K_{P21} + K_{I21}\dfrac{1}{s} + K_{D21}s & K_{P22} + K_{I22}\dfrac{1}{s} \end{bmatrix} \quad (19)$$

So the objective will be a 10 dimensional optimization problem of determining the optimal coefficients $[K_{P11}\ \ K_{I11}\ \ K_{P12}\ \ K_{I12}\ \ K_{D12}\ \ K_{P21}\ \ K_{I21}\ \ K_{D21}\ \ K_{P22}\ \ K_{I22}]$

Table 2
Parameters of PID controller obtained by COA, GA and DRF.

PID parameters	Method		
	COA	GA	DRF
K_{P11}	0.2751	0.1763	0.184
K_{I11}	0.0803	0.0592	0.0469
K_{P12}	−0.0675	−0.0418	−0.0102
K_{I12}	−0.0290	−0.0246	−0.0229
K_{D12}	0.0835	0.1037	0.0082
K_{P21}	−0.0522	0.0404	−0.0673
K_{I21}	0.0330	0.0227	0.0159
K_{D21}	−0.0680	−0.0425	−0.0536
K_{P22}	−0.1243	−0.1127	−0.066
K_{I22}	−0.0210	−0.019	−0.0155

Table 3
Different parts of cost function that are optimized by methods, COA, GA and DRF.

Criteria	Method		
	COA	GA	DRF
IAE_{11}	*3.849*	6.5688	4.9278
IAE_{12}	*0.99608*	1.1735	1.0625
IAE_{21}	*2.6465*	3.8051	4.4716
IAE_{22}	*7.1225*	7.2117	9.0288
IAE	**14.614**	18.759	19.4907
POU_{11}	*9.1797%*	16.118%	*9.91%*
POU_{12}	7.2331%	8.8628%	*4.07%*
POU_{21}	*10.967%*	31.293%	22.05%
POU_{22}	*9.3256%*	9.8398%	9.86%
POU	**36.705%**	66.114%	45.89%
Cost	***18.283***	25.376	24.0791

to minimize the cost function (16). Both COA and GA are applied to this problem 10 times and the best result of each is given and studied in this section.

A COA with 20 cuckoos and with maximum egg laying value of 5 is used in order to tune controller parameters. As for GA, the maximum iterations of the COA is set to 200 but it reached to the total cost of 18.28 in 51 iterations and the algorithm stopped.

A GA with 100 initial population, tournament selection, Gaussian mutation and scattered crossover was used to tune the parameters of the multivariable PID controller for the process. To fully exploit GA's potential in cost minimization it was equipped with a hybrid function.

Fig. 23 depicts the minimum costs for the best results of 10 different runs of COA and GA. As shown in this figure, the steady state convergence value of COA is 18.283, which is smaller than that of GA, 25.367.

Parameters of PID controller and their relevant cost values obtained by COA, GA and DRF methods are demonstrated in Tables 2 and 3. According to Table 2, the controller obtained by DRF has only resulted in the least POU_{12}. Considering all other parameters it can be clearly seen that the controller designed with COA is the best of all three methods. The values in Table 3 shows that using the controller with parameters designed by COA both outputs will have best tracking and the least coupling. The total cost obtained in Table 3 demonstrates the better performance of the controller designed by COA.

Fig. 24 shows the response of controlled distillation column process to step inputs using different controllers obtained by COA, GA and DRF. To have a better view of decoupling created by different controllers, step inputs are applied with delays at time step 110 s.

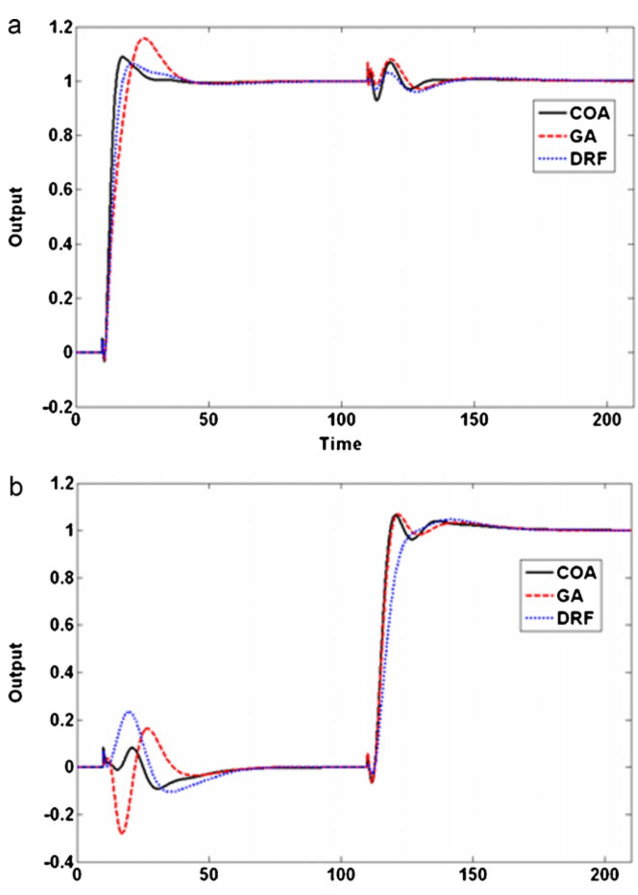


Fig. 24. The response of distillation column process to different delays in step inputs: (a) first output, (b) second output.

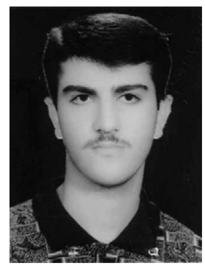
5. Conclusions

In this paper, a new optimization algorithm was proposed which was inspired by lifestyle of a bird called Cuckoo. Special characteristics of cuckoos in egg laying and breeding had been the basic motivation for development of this new optimization algorithm. Each individual in the algorithm has a habitat around which she starts to lay eggs. In case the eggs survive, they grow and become mature cuckoos. Then for reproduction purposes cuckoos immigrate toward best habitat, found up to now. The diversion occurred when moving toward goal habitat makes the population search more area than the case population moves straight forward on a line. After some immigrations all cuckoo population gather the same habitat which is the area's best position. The introduced algorithm was tested on 5 benchmark cost functions. The comparison of COA with standard versions of PSO and GA with *Roulette wheel* selection, uniform cross-over, showed the superiority of COA in fast convergence and global optima achievement. In the first 4 test functions all methods have found the global minima but COA has converged faster in less iterations. But in the last test function (10-dimensional Rastrigin function) GA and PSO could not converge to even a close value of global optima. But COA had found a very good and acceptable estimation of global minimum in just 66 iterations. Off course, it should be noted that the higher performance of COA in reaching better results for these 5 benchmark functions and a real case study does not necessarily mean that COA is the ever best evolutionary method developed. It just can be considered as a successful mimicking of nature; suitable for some sort of optimization problems.

References

[1] R.L. Haupt, S.E. Haupt, Practical Genetic Algorithms, second ed., John Wiley & Sons, New Jersey, 2004.
[2] M. Melanie, An Introduction to Genetic Algorithms, MIT Press, Massachusetts, 1999.
[3] S.N. Sivanandam, S.N. Deepa, Introduction to Genetic Algorithms, Springer-Verlag, Berlin, Heidelberg, 2008.
[4] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Proceedings of IEEE International Conference on Neural Networks, vol. 4, 1995, pp. 1942–1948.
[5] A.P. Engelbrecht, Fundamentals of Computational Swarm Intelligence, John Wiley & Sons, New Jersey, 2005.
[6] M. Dorigo, C. Blum, Ant colony optimization theory: a survey, Theoretical Computer Science 344 (2005) 243–278.
[7] M. Dorigo, L.M. Gambardella, Ant Colony System: a cooperative learning approach to the traveling salesman problem, IEEE Transactions on Evolutionary Computation 1 (1997) 53–66.
[8] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimization by simulated annealing, Science: New Series 220 (1983) 671–680.
[9] J. De Vicente, J. Lanchares, R. Hermida, Placement by thermodynamic simulated annealing, Physics Letters A 317 (2003) 415–423.
[10] G. Zilong, W. Sun'an, Z. Jian, A novel immune evolutionary algorithm incorporating chaos optimization, Pattern Recognition Letters 27 (2006) 2–8.

[11] A. Ahrari, M. Shariat-Panahi, A.A. Atai, G.E.M.: a novel evolutionary optimization method with improved neighborhood search, Applied Mathematics and Computation 210 (2009) 376–386.
[12] Lu Hong, A novel particle swarm optimization method using clonal selection algorithm, in: International Conference on Measuring Technology and Mechatronics Automation, vol. 2, 2009, pp. 471–474.
[13] V. Chellaboina, M.K. Ranga, Reduced order optimal control using genetic algorithms, in: American Control Conference, Portland, USA, vol. 2, 2005, pp. 1407–1412.
[14] D.B. Fogel, An evolutionary approach to the traveling salesman problem, Biological Cybernetics 60 (2004) 139–144.
[15] B. Bontoux, D. Feillet, Ant colony optimization for the traveling purchaser problem, Computers & Operations Research 35 (2008) 628–637.
[16] R.L. Johnston, H.M. Cartwright, Applications of Evolutionary Computation in Chemistry, Springer-Verlag, Berlin, Heidelberg, 2004.
[17] C.-C. Wu, K.-C. Lai, R.-Y. Sun, GA-based job scheduling strategies for fault tolerant grid systems, in: IEEE Asia-Pacific Services Computing Conference, 2008, pp. 27–32.
[18] L.V. Arruda, M.C. Swiech, M.R. Delgado, F. Neves Jr., PID control of MIMO process based on rank niching genetic algorithm, Applied Intelligence 29 (2008) 290–305.
[19] H.A. Varol, Z. Bingul, A new PID tuning technique using ant algorithm, in: Proceeding of the IEEE 2004 American Control Conference, vol. 3, 2004, pp. 2154–2159.
[20] H.H. Balaci, J.F. Valenzuela, Scheduling electric power generators using particle swarm optimization combined with Lagrangian relaxation method, International Journal of Applied Mathematics and Computer Science 14 (2004) 411–421.
[21] C. Andrieu, A. Doucet, Simulated annealing for maximum a posteriori parameter estimation of hidden Markov models, IEEE Transactions on Information Theory 46 (2000) 994–1004.
[22] The Life of Birds, Parenthood. http://www.pbs.org/lifeofbirds/home/index.html (retrieved 05.09.09).
[23] C. Ellis, C. Kepler, A. Kepler, K. Teebaki, Occurrence of the longtailed cuckoo Eudynamis Taitensis on Caroline Atoll, Kiribati 90 (1990) 202.
[24] P. Hockey, Patterns and correlates of bird migrations in Sub-Saharan Africa, Emu 100 (2000) 401–417.
[25] Payne, B. Robert, The Cuckoos, Oxford University Press, 2005.
[26] http://www.telegraph.co.uk/earth/wildlife/4109282/Cuckoo-chicks-dupe-foster-parents-from-the-moment-they-hatch.html.
[27] N.A. Campbell, Fixed action patterns, in: Biology, fourth ed., Benjamin Cummings, New York, 1996, p. 1179.
[28] D. Wheatcroft, Co-evolution: a behavioral ‘spam filter’ to prevent nest parasitism, Current Biology 19 (2009) R170–R171.
[29] Q.G. Wang, B. Zou, T.H. Lee, B. Qiang, Auto-tuning of multivariable PID controllers from decentralized relay feedback, Automatica 33 (1997) 319–330.
[30] W.L. Luyben, A simple method for tuning SISO controllers in a multivariable system, Industrial and Engineering Chemistry Product Research and Development 25 (1986) 654–660.



Ramin Rajabioun was born in Tabriz, Iran February 15th 1982. He received his B.S. degree in Biomedical Engineering from Sahand University of Technology, Tabriz, Iran, 2001. He has now graduated the Control Systems Engineering at Control and Intelligent Processing Center of Excellence at University of Tehran. His research interests are Evolutionary Computation, Optimization, Image processing, System Identification, Computer Vision, Computer Programming, Game Theory and Stochastic Control.