HASIL CEK_Chicken Swarm_IJESI

by Chicken Swarm_ijesi

International Journal of Engineering Science Invention
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 6 Issue 1||January 2017 || PP. 08-14

Chicken Swarm as a Multi Step Algorithm for Global Optimization

Nursyiva Irsalinda[1], Aris Thobirin[2], Dian Eka Wijayanti[3]
[1,2,3]*(Mathematics Department, Ahmad Dahlan University, Indonesia)*

Abstract: *A new modified of Chicken Swarm Optimization (CSO) algorithm called multi step CSO is proposed for global optimization. This modification is reducing the CSO algorithm's steps by eliminates the parameter roosters, hens and chicks. Multi step CSO more efficient than CSO algorithm to solve optimization problems. Experiments on seven benchmark problems and a speed reducer design were conducted to compare the performance of Multi Step CSO with CSO algorithms and the other algorithms based population such as Cuckoo Search (CS),Particle Swarm Optimization (PSO), Differential Evolution (DE) and Genetic Algorithm (GA). Simulation results show that Multi step CSO algorithm performs better than those algorithms. Multi step CSO algorithm has the advantages of simple, high robustness, fast convergence, fewer control.*

Keywords: *Benchmark function, Chicken swarm optimization, Metaheuristic algorithm, Multi swarm optimization, single swarm optimization,*

I. Introduction

Bio-inspired meta-heuristic algorithms have shown proficiency of solving a great many optimization applications [1],[2]. In fact, in addition to genetic algorithms and neural networks, there is a class of metaheuristic algorithms which are inspired by some successful characteristics of biological systems in nature [3]. In particular, a metaheuristic algorithm is called as a robust only if it fulfils two requirements: intensification and diversification [3],[4]. Intensification consists of exploring the current local search position to find the best quality of solution. Diversification consists of ensuring that the entire search space can be covered during search for new solutions. Therefore, the ability of a metaheuristic to find the global optima is in correlation with its capability to find an optimal balance between the intensification (exploitation) and the diversification (exploration) of the search.

Bio-inspired meta-heuristic algorithms find near-optimal solutions to the difficult optimization problems by motivation from nature [5]. The previous bio-inspired meta-heuristic algorithms has been introduced i,e. Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC) algorithm, Differential Evolution (DE)[6]. Those algorithms have been developed for solving difficult optimization problem. Researchers have shown that algorithms based on swarm intelligent have great potential and have attracted much attention [7].

PSO is a heuristic global optimization method put forward originally by Doctor Kennedy and Eberhart in 1995 [8]. It is developed from swarm intelligence and based on the research of bird and fish flock movement behavior [8,9].

ABC algorithm is proposed by Karaboga in Erciyes University of Turkey in 2005 [5,10,11]. This Algorithm mimicking the foraging behaviour of honey bee colony. As in [10], there are three essential components of ABC algorithm: food sources, employed foragers and unemployed foragers and two important basic behaviour: recruitment and abandonment the food sources [5, 12]. In ABC algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. ABC algorithm has many advantages but it has two major weaknesses: one is slower convergence speed; the other is getting trapped in local optimal value early [13].

The DE algorithm is a population-based algorithm like genetic algorithms using the similar operators; crossover, mutation and selection [14]. The main difference in constructing better solutions is that genetic algorithms rely on crossover while DE relies on mutation operation. This main operation is based on the differences of randomly sampled pairs of solutions in the population. The algorithm uses mutation operation as a search mechanism and selection operation to direct the search toward the prospective regions in the search space. The DE algorithm also uses a non-uniform crossover that can take child vector parameters from one parent more often than it does from others. By using the components of the existing population members to construct trial vectors, the recombination (crossover) operator efficiently shuffles information about successful combinations, enabling the search for a better solution space. In DE, a population of solution vectors is randomly created at the start. This population is successfully improved by applying mutation, crossover and selection operators. In the DE algorithm, each new solution produced competes with a mutant vector and the better one wins the

competition. Differential evolution is a very simple but very powerful stochastic global optimizer. Since its inception, it has proved to be very efficient and robust technique for function optimization and has been used to solve problems in many scientific and engineering fields [15].
The previous algorithms are single swarm optimization algorithm. Their common essence is to simulate and reveal some natural phenomena and processes developed according to the system initializing a set of initial solution, the operation iterative rules specific for a group of solutions combined with the search mechanism itself are iterative, and finally get the optimal solution [16]. Algorithm to obtain better performance is still being developed. Therefore, in 2014 Xianbing Meng et.al. proposed multi swarm optimization algorithm called Chicken Swarm Optimization (CSO). CSO can achieve optimization results both accuracy and robustness optimization in terms compared to previous single swarm optimization algorithms.
However, as a multi swarm optimization, there are so many parameters should be set. To reduce the number of parameters, in this paper, we modify the chicken optimization as a multi swarm optimization be a multi step optimization**.**

II. Chicken Swarm Optimization

Chicken Swarm Optimization (CSO) based on the chicken behavior was proposed by meng et al [1]. As in [1], there are at least four rules in the chicken behavior, as follows

a. In the chicken swarm, there exist several groups. Each group comprises a dominant rooster, a couple of hens, and chicks.
b. How to divide the chicken swarm into several groups and determine the identity of the chickens (roosters, hens and chicks) all depend on the fitness values of the chickens themselves. The chickens with best several fitness values would be acted as roosters, each of which would be the head rooster in a group. The chickens with worst several fitness values would be designated as chicks. The others would be the hens. The hens randomly choose which group to live in. The mother-child relationship between the hens and the chicks is also randomly established.
c. The hierarchal order, dominance relationship and mother-child relationship in a group will remain unchanged. These statuses only update every several (G) time steps.
d. Chickens follow their group-mate rooster to search for food, while they may prevent the ones from eating their own food. Assume chickens would randomly steal the good food already found by others. The chicks search for food around their mother (hen). The dominant individuals have advantage in competition for food.

The roosters with better fitness values have priority for food access than the ones with worse fitness values. For simplicity, this case can be simulated by the situation that the roosters with better fitness values can search for food in a wider range of places than that of the roosters with worse fitness values. This can be formulated below.

$$x_{i,j}^{t+1} = x_{i,j}^{t} * \left(1 + Randn(0, \sigma^2)\right). \tag{1}$$

$$\sigma^2 = \begin{cases} 1 & f_i \le f_k \\ \exp\left(\dfrac{f_k - f_i}{|f_i| + \varepsilon}\right) & otherwise \end{cases} \quad k \in [1, N], k \ne i. \tag{2}$$

where *Randn* $(0, \sigma^2)$ is a Gaussian distribution with mean 0 and standard deviation σ^2. ε, which is used to avoid zero-division-error, is the smallest constant in the computer. *k*, a rooster's index, is randomly selected from the roosters group, *f* is the fitness value of the corresponding *x*.
As for the hens, they can follow their group-mate roosters to search for food. Moreover, they would also randomly steal the good food found by other chickens, though they would be repressed by the other chickens. The more dominant hens would have advantage in competing for food than the more submissive ones. These phenomena can be formulated mathematically as follows.

$$x_{i,j}^{t+1} = x_{i,j}^{t} + S1 * Rand * \left(x_{r1,j}^{t} - x_{i,j}^{t}\right) + S2 * Rand * \left(x_{r2,j}^{t} - x_{i,j}^{t}\right). \tag{3}$$

$$S1 = \exp\left(\frac{f_i - f_{r1}}{abs(f_i) + \varepsilon}\right). \tag{4}$$

$$S2 = \exp(f_{r2} - f_i). \tag{5}$$

where *Rand* is a uniform random number over [0, 1], $r1 \in [1 \ldots N]$ is an index of the rooster, which is the *i*th hen's group-mate, while $r2 \in [1 \ldots N]$ is an index of the chicken (rooster or hen), which is randomly chosen from the swarm $r1 \ne r2$.
The chicks move around their mother to forage for food. This is formulated below.

$$x_{i,j}^{t+1} = x_{i,j}^{t} + FL * \left(x_{m,j}^{t} - x_{i,j}^{t}\right). \tag{6}$$

Where $x_{m,j}^{t}$, stands for the position of the i-th chick's mother ($m \in [1, N]$). FL ($FL \in (0,2)$) is a parameter, which means that the chick would follow its mother to forage for food. Consider the individual differences, the FL of each chick would randomly choose between 0 and 2.

III. Chicken Swarm as A Multi Step Algorithm

Original CSO requires at least six parameters should be set, i.e RN (number of rooster), CN (number of Chick), MN (number of mother), HN (number of hen) , G (maximum generation) and FL (the interval random number). The superiority of CSO over PSO, ABC and DE should be the case as follows

a. If we set $RN = CN = 0$, and let $S1$, $S2$ be the parameters like $c1$ and $c2$ in PSO, thus CSO will be similar to the standard PSO. Hence CSO can inherit many advantages of PSO
b. If we set RN and MN at 0, thus CSO essentially becomes the basic mutation scheme of DE. Hence the partial conclusions from the DE [2] can be used. In practice, FL [0.4, 1] usually perform well.
c. If we set RN amd HN at 0, thus CSO will be similar to ABC.

Thus, as a multi swarm optimization, each group of CSO has the different range search ability. For example, Hen group has wide range exploration, thus the movement of Hen group have global search ability. Other hand, the range of movement of chick and Rooster group is at the Neighbor of current position. To this, chick and rooster group have local search ability. That's mean that after ranking the fitness value, only the first RN and the last CN will exploit the local optima from the current position and HN will explore to find the new solution. Whereas, the next best optimal may be occur at the neighbor of the one of Hen group current position. Because of this, the original CSO has been modified be a multi step CSO by running all groups step by step for all population. Not only reducing the number parameter but also the exploration and exploitation of search space can be done by all individual of population not separately. The multi steps are separated be two steps. The first step is diversification (exploration). In this step, the hen group which will be the first step is reduced. Because of having the largest area search ability, the reduced form is used to exploring the global optima. Each individual of chicken population move to the other position by the best chicken and the other chicken. Obviously, all individual chicken is treated as a hen. The second one is intensification (exploitation) step. This step evaluates the value from the first step. Since the rooster and chick group have the local search ability, the both group will be used to exploit the current position from the first step, respectively. Similar to the first step, each individual of chicken population is considered as a rooster then as a chicken respectively.

IV. The Procedure Of Multi Step Chicken Swarm Optimization (MCSO) Algorithm

The procedure of multi step CSO could be described in the following steps.

4.1 Inizialization of MCSO Population :

Individu of multi step chicken swarm population are inizialized by using the following formula

$$x_{i,j} = lb + Rand\ (ub - lb) \tag{7}$$

with lb and ub are lower bound and upper bound of the search space. It is given so that the obtained candidate solutions located in the search space.

4.2 Diversification (exploration) Step

Exploration step reduces the hens step in the CSO because it has the most wide search space. The reduction formula is used to explore the global optimum by eliminating the hen group parameter. Each individu of chicken population repair their position againts two other individu in that population. The formed formula is as follows:

$$x_{i,j}(*) = x_{i,j} + S1 * Rand\ * (x_{l,j} - x_{i,j}) + S2 * Rand\ * (x_{n,j} - x_{i,j}) \tag{8}$$

with

$$S1 = \exp\left(\frac{(f_i - f_l)}{|f_i| + \varepsilon}\right) \tag{9}$$

and

$$S2 = \exp(f_n - f_i). \tag{10}$$

$x_l, x_n \in [1, \ldots, N]$ is randomly chosen from the chicken swarm with $x_i \neq x_l \neq x_n$.

After $x_{i,j}(*)$ obtained, the objective value (fitness value) compared with the fitness value of $x_{i,j}$. The solution that have the best fitness value is choosen as an individu of new population that called individu of the global population $\left(x_{i,j}(g)\right)$.

4.3 Intensification (exploitation) Step

Candidate solution (chicken individu) that have been obtained by exploration step will be repaired again by exploite the neighbourhood using the reduction of rooster and chicken formulas. Similar with exploration step, this step also eliminates rooster and chicken groups. Local optimum search carried out in two steps, the first step using the reduction rooster formula as follows.

$$x_{i,j}(**) = x_{i,j}(g) * \left(1 + Randn\ (0,\sigma^2)\right) \qquad (11)$$

with

$$\sigma^2 = \begin{cases} 1 & f_i(g) \le f_l(g) \\ \exp\left(\dfrac{f_l(g) - f_i(g)}{|f_i(g)| + \varepsilon}\right) & otherwise \end{cases} \quad l \in [1, \dots, N](g), l \neq i. \qquad (12)$$

The first local optimum solution obtained by exploite the global optimum population using Eq (11). After the first local optimum obtained, the next step is compare its fitness value with fitness value of previous global optimum solution. The solution that have the best fitness value is choosen as individu of the first renewal population that called Local population I $\left(x_{i,j}(l_1)\right)$.

After new local population I $\left(x_{i,j}(l_1)\right)$ obtained, the next step as the final step of Multi step CSO is find the more local optimum (the second local optimum) by using the reduction chicken formula of CSO as follows :

$$x_{i,j}(***) = x_{i,j}(l_1) + C * \left(x_{n,j}(l_1) - x_{i,j}(l_1)\right). \qquad (13)$$

$x_n \in [1, \dots, N]$ is randomly chosen from the local population I with $x_i \neq x_n$ and $C\left(C \in (0,2)\right)$ is a parameter (see CSO).

After the second local optimum obtained, the next step is compares its fitness value with the previous local optimum solution fitness value. The solution that have the best fitness value is choosen as individu of the second renewal population that called local population II $\left(x_{i,j}(l_2)\right)$. This population is used as the initial population for the next iteration until the stopping criteria are met. In this research the stopping criteria is amount of iteration.

From above explanation, then Multi Step CSO Algorithm is Fig. 1 :

```
Initialize a population of N chicken using (7)
Evaluate the N chicken fitness value, t = 0.
While t < G.
For i = 1:N
        4.1. Step 1: explore the global optimum using (8)
      Selection of individu global population (x_{i,j}(g))
        4.2. Step 2: Exploitation local optimum
              4.2.1. The first local optimum using (11)
   Selection of individu local population I (x_{i,j}(l_1))
              4.2.2. The second local optimum using (13)
             Selection of individu local population II (x_{i,j}(l_2))
End For
        End While
```

Figure 1. Pseudocode of Multi-step CSO

V. Experiment and Ddiscussion

Recent related studies show that global optimization algorithms have been adopted in a wide range of applications. In all domains, the role of an optimization algorithm consists of reducing the cost or increases the outputs such as profit or performance. Many different studies show that Cuckoo Search (CS) performs better than Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and many other optimization algorithms. The CS high performance can be justified by the fewer parameters used. Furthermore, CS convergence and search for obtaining new solutions is faster than PSO and GA, which affects the optimization cost. In order to evaluate the cost value of MCSO with CSO, GA and PSO, seven popular test functions are used. The implementation of these test functions is realized by using Matlab R2009B. Comparisons are carried out for ten-dimensional case, that is, $n = 10$ for all test functions. 30 particles are included in the population. Change of average means that an average of the best particle in 30 particles at the iteration for 20 trials are shown.

Table 1: The benchmarks function

No.	Function	Formula	Global optima and boundary
1	De jong	$f(x) = \sum_{i=1}^{n} x_i^2$	$f_* = 0$ $-500 \le x_i \le 500,$
2	Rastrigin	$f(x) = 10n + \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i)]$	$f_*=0$ $-5.12 \le x_i \le 5.12.$
3	Rosenbrock	$f(x) = \sum_{i=1}^{n-1} 100(x_{i+1} + x_i^2)^2 + (1 - x_i)^2$	$f_*=0$ $-2.048 \le x_i \le 2.048$
4	Griewangks	$f(x) = \frac{1}{4000}\sum_{i=1}^{n} {x_i}^2 - \prod_{i=1}^{n} cos(\frac{x_i}{\sqrt{i}}) + 1$	$f_* = 0$ $-600 \le x_i \le 600,$
5	Ackley	$f(x) =$ $-20.\exp\left(\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\sqrt{\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)} + \right.$ $(20 + e)$	$f_* = 0$ $-32.768 \le x_i \le 32.768,$
6	Shuberts	$f(x,y) = \sum_{i=1}^{5} icos[(i+1)x + 1] \sum_{i=1}^{5} cos\,[(i+1)y$ $+ 1]$	$f_* = -186.730$ $(x,y) \in [-10,10] \times [-10,10]$
7	Michaelwiz d modal	$f(x) = -\sum_{i=1}^{n} \sin(x_i)\left[\sin\frac{ix_i^2}{\pi}\right]^{2m}$	$f_* = -9.66$ $0 \le x_i \le \pi$

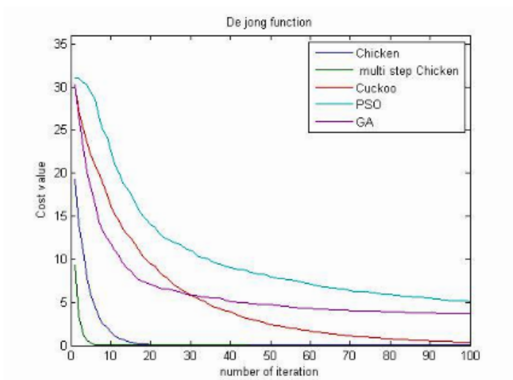

Figure 2. Cost minimization for De Jong function

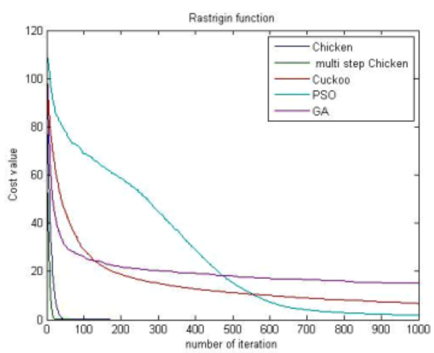

Figure 3. Cost minimization for Rastrigin function

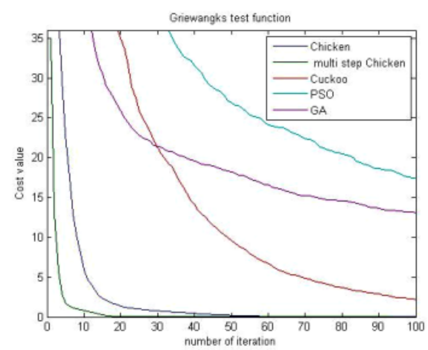

Figure 4. Cost minimization for Griewangks function

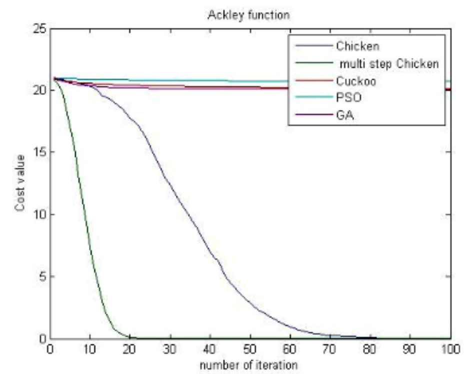

Figure 5. Cost minimization for Ackley function

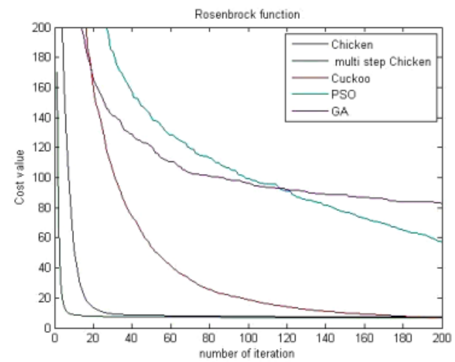

Figure 6. Cost minimization for Rosenbrock function

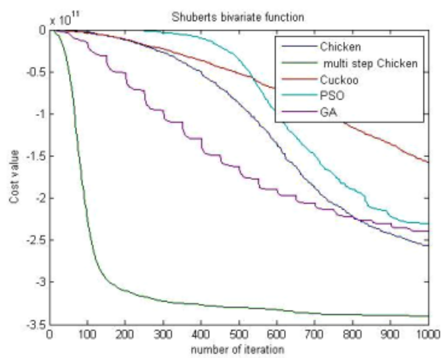

Figure 7. Cost minimization for Shubers function

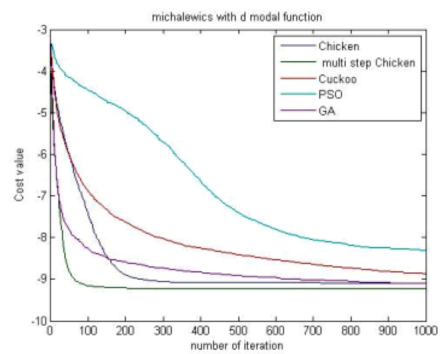

Figure 8. Cost minimization for d modal Michalewiz function

The Figure 2- Figure 8 show that MCSO obtain the best minimum value than the others and fast convergence. It is shown from the number of iteration in each experiment. In the first number of experiment MCSO algorithm directly produces minimum value that close to minimum value of each benchmark function. It is happens because in MCSO every individu of population on chicken swarm through the diversification and intensification steps so this algorithm is more simple because it have fewer control parameter. Therefore, the MCSO algorithm has high robustness and fast convergences.

VI. Conclusions

In this work, the performance of Multi step CSO algorithm was compared with the original CSO, CS, PSO and GA on a large set of unconstrained test functions. Simulation results show that Multi step CSO algorithm performs better than those algorithms (Fig. 2-8). Multi step CSO algorithm has the advantages of simple, high robustness, fast convergence, fewer control parameters.

References

[1]. X. Meng, Y. Liu, X. Gao, and H. Zhang, A New Bio-inspired Algorithm: Chicken Swarm Optimization, Advances in Swarm Intelligence SE , 8794. 2014, 86–94.

[2]. X.-S. Yang and X. He, Bat Algorithm: Literature Review and Applications, Int. J. Bio-Inspired Comput., 5(3). 2013, 141–149.

[3]. X. S. Yang, S. Deb, and S. Fong, Metaheuristic algorithms: Optimal balance of intensification and diversification, Appl. Math. Inf. Sci., 8(3). 2014, 977–983.

[4]. Y. Rochat and É. Taillard, Probabilistic diversification and intensification in local search for vehicle routing," J. Heuristics, 1(1). 1995, 147–167.

[5]. D. Karaboga and B. Akay, A comparative study of Artificial Bee Colony algorithm, Applied Mathematics and Computation , 214. 2009,108-132.

[6]. T.K. Sharma, M. Pant and V.P. Sighn., Improved Local Search in Artificial Bee Colony using Golden Section Search, Journal of Engineering, 1(1). 2012, 14-19.

[7]. W.C. Feng, L. Kui and S.P. Ping., Research Article Hybrid Artificial Bee Colony Algorithm and Particle Swarm Search for Global Optimization, Hindawi Publishing Corporation Mathematical Problems in Engineering, 2014(Article ID. 832949). 2014,1-8.
[8]. Q. Bai, Analysis of Particle Swarm Optimization Algorithm, Computer and Information Science, 3(1), 2010,180-184.
[9]. M. Clerc, Particle Swarm Optimization, (UK: ISTE ltd. 2006)
[10]. D. Karaboga and B. Basturk, On the Performance of Artificial Bee Colony Algorithm, Applied Soft Computing, 8. 2008, 687-697.
[11]. W. Gao, S. Liu, and L. Huang, A Novel Artificial Bee Colony Algorithm with Powell's Method, Applied Soft Computing, 13. 2013, 3763-3775.
[12]. D. Karaboga, An Idea Based On Honey Bee Swarm For Numerical Optimization, Technical Report-Tr06, Erciyes University, Engineering Faculty, Computer Engineering Department, Kayseri/Türkiye. 2005.
[13]. J.Qiu, J. Wang, D. Yang and J. Xie, An Artificial Bee Colony Algorithm with Modified Search Strategies for Global Numerical Optimization, Journal of Theoretical and Applied Information Technology, 48(1). 2013, 294-302.
[14]. R. Storn and K. Price, Differential Evolution – A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, Journal of Global Optimization, 11.1997, 341-359.
[15]. D.P. Reddy and M.C.V Suresh, Differential Evolution Algorithm For Optimal Power Flow, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2(8). 2013, 4043-4047.
[16]. H. Li, Q. Zhang, and Y. Zhang, Improvement and Application of Particle Swarm Optimization Algorithm based on the Parameters and the Strategy of Co-Evolution, Appl. Math. Inf. Sci., 9(3). 2015,1355–1364

HASIL CEK_Chicken Swarm_IJESI

ORIGINALITY REPORT

13%	13%	16%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	jwcn-eurasipjournals.springeropen.com Internet Source	4%
2	Sanchari Deb, Xiao-Zhi Gao, Kari Tammi, Karuna Kalita, Pinakeswar Mahanta. "Recent Studies on Chicken Swarm Optimization algorithm: a review (2014–2018)", Artificial Intelligence Review, 2019 Publication	4%
3	Wu, Dinghui, Fei Kong, Wenzhong Gao, Yanxia Shen, and Zhicheng Ji. "Improved chicken swarm optimization", 2015 IEEE International Conference on Cyber Technology in Automation Control and Intelligent Systems (CYBER), 2015. Publication	3%
4	journals.plos.org Internet Source	3%

Exclude quotes	On	Exclude matches	< 3%

Exclude bibliography On