Journal of Information Hiding and Multimedia Signal Processing ©2018 ISSN 2073-4212
Ubiquitous International Volume **9**, Number **2**, March **2018**

Symbiotic Organisms Search with Mixed Strategy

Yan-Jiao Wang, Huan-Huan Tao

College of Information Engineering
Northeast Electric Power University
No.169, Changchun Rd., Chuanying, 132012, Jilin, China
563274435@qq.com, taociguan@hotmail.com

Ning Lan

International School of Economics and Management,
Capital University of Economic and Business
No.121, Huaxiangzhangjia Rd., Fengtai, 100071, Beijing, China
308073146@qq.com

Xiao-Jie Li

Far East Holding Group Co., LTD
No.6, Yuandong Rd., Yixing, 214257, Jiangsu, China
635967767@qq.com

Received March, 2017; revised December, 2017

Abstract. *Considering the drawbacks of slow convergence and the ease of falling into a local optimum in Symbiotic Organisms Search (SOS), Symbiotic Organisms Search with Mixed Strategy (ISOS) is proposed. In the first stage, to accelerate the speed of population convergence, the variance of fitness value is used as the evaluation criterion in the mutualism and commensalism phase. In the second stage, to avoid the problem of easily reaching local optimization state in the later period of evolution, a new population-updating formula is introduced to replace the blindness of the random search for the parasitic mechanism. This improvement increases the diversity of the population to guarantee the search ability of the algorithm. Experimental results on eight benchmark functions show that ISOS improves convergence and robustness compared to SOS, and can avoid premature.*

Keywords: Swarm intelligence; Symbiotic Organisms Search; Function optimization

1. **Introduction.** Symbiotic Organisms Search (SOS) is a new optimization algorithm that simulates the interactive behavior seen among organisms in nature [1]. It is proposed by Min-Yuan Cheng and Doddy Prayogo in 2014. Lots of experiments have proved that SOS has the advantages of fast search speed and less parameter setting. Tests on standard benchmark functions indicate that the performance of SOS is better than lots of algorithm, such as Genetic Algorithm (GA) [2], Particle Swarm Optimization (PSO) [3], Differential Evolution (DE) [4-5] and Bees Algorithm (BA) [6]. It is one of the most outstanding function optimization algorithms. However, similar to other evolutionary algorithms, the problem of slow convergence and easy falling into local optimum still remains [7-9]. This recently developed algorithm has not been well-studied. For this reason, it has not yet undergone significant performance improvement and extensive application, so the theoretical system remains imperfect.

To improve the performance of the SOS, Symbiotic Organisms Search with Mixed Strategy (ISOS) is proposed. On the one hand, ISOS uses the variance of fitness value as the criterion to realize the fusion of random update and adaptive update to speed up the convergence. On the other hand, to improve the search ability of SOS, ISOS introduces a new individual update formula in parasitism phase. Experimental results on eight benchmark functions show that the ISOS is better, and the performance of the SOS is improved.

2. **Symbiotic Organisms Search(SOS) algorithm.** SOS deals with function optimization problems by simulating the symbiotic relationship between different organisms in nature. SOS corresponds to three different symbiotic relationships in nature, the mutualism phase, the commensalism phase and the parasitism phase. In the process of simulating this characteristic, the solution of the optimization problem corresponds to the individual, and the fitness function shows the adaptability of the organism to the natural world.

2.1. **Mutualism phase.** Similar to legume and root nodules, SOS algorithm establishes mutualism mechanism through simulating the interaction of these organisms that can work together to benefit each other. According to the Eq. (1), the individual update is carried out.

$$\begin{cases} X_{inew} = X_i + rand(0,1) * (Xbest - Mutual - Vector * BF_1) \\ X_{jnew} = Xj + rand(0,1) * (Xbest - Mutual - Vector * BF_1) \end{cases} \tag{1}$$

$$Mutual - Vector = \frac{X_i + X_j}{2} \tag{2}$$

In SOS, X_i is an organism matched to the i^{th} member of ecosystem. Another organism X_j is then selected randomly from the ecosystem to interact with X_i. X_{best} is representing the highest degree of adaptation, and *rand(0,1)* is a vector of random numbers. Eq. (2) shows a vector called "*Mutural_Vector*" that represents the relationship characteristic between X_i and X_j. Benefit factors(BF_1 and BF_2) are determined randomly as either 1 or 2,which shows two types of organisms benefit to each other.

2.2. **Commensalism phase.** Similar to remora fish and sharks, SOS algorithm establishes commensalism phase through simulating the interaction of these organisms that benefit to one of the organisms and nothing to the other. According to the Eq. (3), the individual update is carried out.

$$X_{inew} = X_i + rand(-1,1) * (X_{best} - X_j) \tag{3}$$

X_j is selected randomly from ecosystem to interact X_i. The part of formula, $X_{best} - X_j$, is reflecting as the beneficial advantage provided by X_j to help X_i increasing its survival advantage in ecosystem to the highest degree in current organism. If the new individual's fitness value is better than the original one, then to update the original individual.

2.3. **Parasitism phase.** Similar to roundworms and humans, SOS establishes commensalism phase through simulating the interaction of these organisms that benefit to one of the organisms and harm to the other.

Parasite_Vector is created by duplicating organism X_i, then modifying the randomly selected dimensions using a random number. To compare Parasite_Vector and organism X_i, the better one can be saved.

3. **Symbiotic Organisms Search with Mixed Strategy (ISOS).** It is well known that the individual update strategy of swarm intelligence optimization algorithm affects the convergence speed and convergence precision directly [10,11]. In SOS, in order to speed up the convergence rate, mutualism phase and commensalism phase are established. In order to ensure the diversity of population, parasitism phase is established. A great quantity studies demonstrated that SOS algorithm has slow convergence rate and it is easily trapping in local optimum. The fundamental cause of the problem is the irrational individual renewal strategy. So, this paper proposes two improved strategies for individual renewal.

3.1. **The improvement of mutualism and commensalism phase.** We know from Eqs (1) and (2) that the essence of the individual renewal formula in the mutualism and commensalism phases is: New individual=original individual+ learning from optimal individual*random weight. In this way, learning from the optimal individual introduces better evolutionary information that can ensure that individuals move to a better location. At the same time, random weights can maintain the diversity of the population. In the latter part of the algorithm, because the degree of similarity of individuals is high, the optimal individual is unable to provide more information than their own. This can cause an enhancement of the function of the random weighting part, causing the search step to become random. This updated mode causes the algorithm to run slowly at the later stage, because it is difficult for individuals to explore more excellent new individuals by random searching in their own vicinity. Thus, SOS has a slow convergence rate during the posterior evolving process. Overall, the convergence speed of the SOS should be improved.

As can be seen from the above analysis, the difference between individuals is small, and all of them converge to a global optimum solution in the posterior evolving process.

Therefore, this paper adopts the following measures. First, smaller search steps should be taken to allow fine search in the local vicinity. Second, considering the influence of the iteration on the search scope, the random weights can be changed into adaptive weights. With more iterations, the random weight becomes smaller, and local search can be changed into search in the vicinity of the best individual to increase the rate of convergence. In summary, the new individual updating formula is introduced as Eqs (4) and (5). However, the original strategy is partially preserved as Eqs (1) and (3) to avoid individuals becoming almost the same as the result of convergence that is too fast. At this point, the learning part from other individuals approaches 0. Many experiments have shown that if too fast convergence is required during the anterior evolving process, the algorithm can be easily trapped in local optima. Thus, this method only increases the convergence of the algorithm during the anterior evolving process, and proceeds as far as possible to ensure the diversity of population in the posterior evolving process. In summary, this paper proposes a new fusion updating strategy in the posterior evolving process. The pseudo-code of the mutualism phase is shown below.

$$\begin{cases} new\_X_{inew} = X_{best} + (1\text{-}(\frac{\mathrm{g}}{G}) \wedge 2) * (X_{best} - Mutual\_Vector(1,:) * BF_1) \\ new\_X_{jnew} = X_{best} + (1\text{-}(\frac{\mathrm{g}}{G}) \wedge 2) * (X_{best} - Mutual\_Vector(1,:) * BF_2) \end{cases} \tag{4}$$

$$new\_X_{inew} = X_{best} + (1 - (\frac{g}{G}) \wedge 2) * (X_{best} - X_j) \tag{5}$$

The pseudo-code of mutualism phase.

Eq. (1) is replaced by Eq. (3), and Eq. (4) is replaced by Eq. (5) in commensalism phase, and the rest is the same.

Input: The population(X) size is S $X=\{X_1, X_2, \ldots, X_N\}$;
Output: New populations(X')=$\{X'_1, X'_2, \ldots, X'_N\}$;
if $s < q$
if $rand < r$
Updating individuals according Eq. (4);
else
Updating individuals X_i , X_j according Eq. (1)
end
else
Updating individuals X_i , X_j according Eq. (1)
end
The new individuals'(X'_i, X'_j) fitness values are calculated and compared. Individuals with poor fitness were eliminated

In the pseudo-code, the 's' is the variance of individual fitness value that can be used to measure individual differences to distinguish the anterior and posterior evolving process. The 'r' is a random number between 0 and 1 that allows the improved strategy to increase the convergence rate to be selected with greater probability. At the same time the original strategy of maintaining population diversity also has a certain probability of being selected. This allows reaching convergence as quickly as possible without significant reduction of the diversity of the population.

3.2. **The improvement of parasitism phase.** The parasitic mechanism of SOS selects dimensions randomly, and searches randomly in the domain of its definition. Although this approach can complement diversity to some extent, blind random search is likely to destroy good individuals. It is difficult to update a more outstanding individual, and it is difficult to maintain the diversity of the population. For unimodal functions the diversity requirement is not high, and the convergence rate is slow but does not reach the local optimum when it only relies on mutualism and commensalism to update the individuals. For multimodal functions that the diversity requirement is high, and SOS will still fall into a local optimal even with the parasitism phase. Therefore, not only does the population need to increase diversity, but should increase the convergence rate. Based on this idea, a new strategy of individual update in the parasitism phase is introduced, as shown in Eq. (6).

$$new\_X_{jnew} = a * X_j + b * (X_{best} - X_j) + c * (X_k - X_j) \tag{6}$$

Of this, X_j, X_k and X_p are selected randomly from the population, and the selected individuals are different. The new formula of individual renewal consists of two parts: $a * X_j$is a random changing part, and the $b * (X_{best} - X_j)$ and $c * (X_k - X_j)$are the learning part. In learning, the first term indicates that the individual learns the best individual near the optimal position, and the second term indicates that the individual learns from the rest of the population to increase population diversity and to avoid the algorithm reaching a local optimum. The 'a', 'b' and 'c' are control parameters, as shown in Eq. (7).

$$\begin{cases} a = a_{min} + (a_{max} - a_{min}) * ((1 - (\frac{g}{G}) \wedge 2) * w_1 + w_2 * (f_i - f_{\min})/(f_{\max} - f_{\min})); \\ b = b_{min} + (b_{max} - b_{min}) * (((\frac{g}{G}) \wedge 2 - 1) * w_1 + w_2 * (f_i - f_{\min})/(f_{\max} - f_{\min})) \\ c = c_{min} + (c_{max} - c_{min}) * ((1 - (\frac{g}{G}) \wedge 2) * w_1 + w_2 * (f_{\max} - f_i)/(f_{\max} - f_{\min})) \end{cases} \tag{7}$$

Equation (13) is the result of learning from the individual to the others. The quality and the front part of the multiplication result can be utilized to determine the proportion of the overall learning; this can be regarded as the overall difference in front of the weight, which has a certain algebraic relation with the fitness value. To efficiently algebraically calculate the function, we developed the following linear formula.

The range of values of 'a', 'b' and 'c' is from 0 to 1. 'G' reflects the maximum number of iterations and 'g' reflects the current iteration. The 'fi' reflects ith individuals' fitness. The 'f_{min}' and 'f_{max}' represent the minimum fitness and the maximum fitness in the current iterations. The 'w_1'equals $\frac{1}{2}$ and 'w_2' equals $\frac{1}{2}$. We can see from the formulas that the parameters in the adaptive adjustment of SOS take into account the number of iterations as well as the fitness value. This setting does not require human intervention.

4. **Experiments and Results.**

Laboratory experiments were conducted in order to determine the efficiency of the proposed evolutionary algorithm ISOS. All experiments were performed on an Intel (R) Core (TM) CPU 1.80GHz i5-3337U, 4G memory computer, with the Matlab 7.11software running environment.

4.1. **Benchmark functions.**

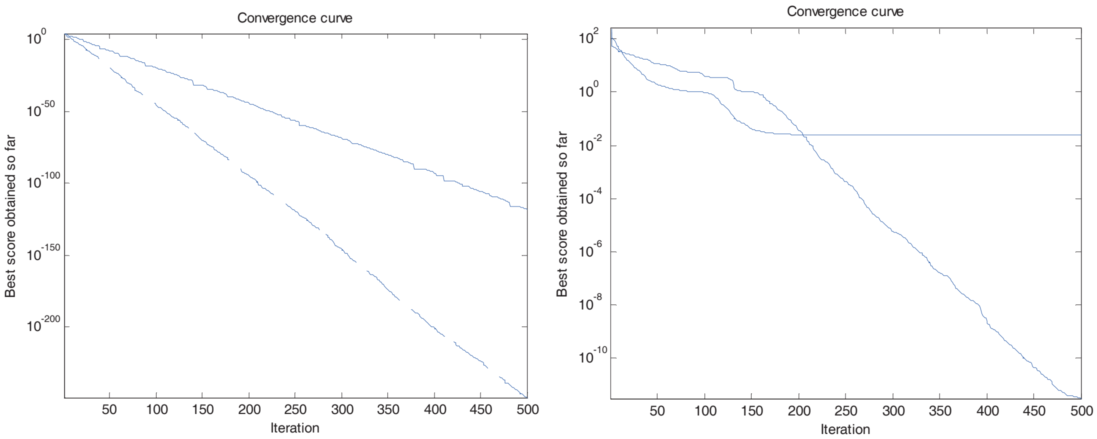
We used a test bed of 8 benchmark functions (Table 1), most of which were taken from the list of CEC2005 benchmarks, to evaluate the performance of ISOS [12]. The theoretical minimum value of f5 is -29.6309, and the minimum value of the other functions is 0.Functions f5, f6 and f8 are multimodal functions that are typically used to test the global searching performance and the ability to avoid local optimum [13]. The other functions are unimodal functions that are used to test the performance of convergence speed and precision.

Table 1. Test Function

Function	Formulation	Range
f1	$F_1(X) = \sum_{i=1}^{D} x_i^2$	$[-100, 100]$
f2	$f(x) = \sum_{i=1}^{D} ixi^2$	$[-10, 10]$
f3	$f(x) = \sum_{i=1}^{D} \lvert x_i\rvert + \prod_{i=1}^{D} \lvert x_i\rvert$	$[-10, 10]$
f4	$f(x) = \sum_{i=1}^{D} (\sum_{j=1}^{i} x_j)^2$	$[-100, 100]$
f5	$f(x) = -\sum_{i=1}^{D} \sin(xi)(\sin(ixi^2/\pi))^{20}$	$[0, \pi]$
f6	$f(x) = \sum_{i=1}^{D-1} 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2$	$[-30, 30]$
f7	$f(x) = (x_1 - 1)^2 + \sum_{i=2}^{D} i(2x_i^2 - x_i - 1)^2$	$[-10, 10]$
f8	$f(x) = \sum_{i=1}^{D} \frac{(x_i-100)^2}{4000} - \prod_{i=1}^{D} \cos\left(\frac{x_i-100}{\sqrt{i}}\right) + 1$	$[-600, 600]$

4.2. **Performance test of two kinds of improvement.**

In this study, only one unimodal function, f1, was selected to verify the improvement of mutualism phase and commensalism phase and only one multimodal function, f8, was selected to verify the improvement of parasitism phase due to limitations of space. The population size is 50, and the dimension is 30. The maximum evaluation number of the benchmark function is 150,000. The experimental results are shown in Fig 1.



(a) the comparison of the improvement of mutualism and commensalism

(b) the comparison of the improvement of parasitism phase

FIGURE 1. Comparison of the Improvement

TABLE 2. The Result of Test Function

Name	Algorithm	Minimum	Mean	Maximum	Variance
f1	DE	3.55e-045	3.62e-044	1.06e-043	6.57e-088
	SOS	1.62e-184	4.23e-176	6.53e-175	0
	ISOS	**0**	**0**	**0**	**0**
f2	DE	6.17e-045	4.51e-044	2.14e-043	1.74e-087
	SOS	4.322e-184	3.48e-176	7.28e-175	0
	ISOS	**0**	**0**	**0**	**0**
f3	DE	1.75e-026	5.792e-026	1.24e-025	8.62e-052
	SOS	9.52e-090	3.13e-087	3.74e-086	7.2285e-087
	ISOS	**0**	**0**	**0**	**0**
f4	DE	2.69e-045	3.96e-044	1.92e-043	1.54e-087
	SOS	8.12e-183	2.31e-175	3.31e-174	0
	ISOS	**0**	**0**	**0**	**0**
f5	DE	-7.2493	-5.1471	-1.8231	1.4180
	SOS	**-9.8293**	**-9.6983**	**-9.3850**	**1.1912e-01**
	ISOS	-18.6871	-18.4215	-18.0163	1.0674e-01
f6	DE	19.1454	22.4776	24.0148	1.7308
	SOS	9. 901e-001	9.901e-001	9.901e-001	0
	ISOS	**1.62e-004**	**5.83e-004**	**9.41e-004**	**2.5e-004**
f7	DE	**6.1058**	**17.2493**	**26.4788**	**38.1277**
	SOS	4.06e-022	4.50e-018	1.24e-014	2.25e-018
	ISOS	**6.73e-033**	**2.89e-029**	**3.47e-025**	**7.44e-029**
f8	DE	**0**	**0**	**0**	**0**
	SOS	1.1102e-016	2.36e-002	1.102e-001	2.57e-002
	ISOS	**0**	**4.03E-004**	**7.403e-004**	**3.99E-003**

TABLE 3. Comparison of Convergence Rate of the Algorithm

Name	Algs	Times				
		20000	40000	60000	80000	100000
f1	DE	4.74e-002±3.61e-004	1.68e-008±6.95e-017	5.44e-015±9.41e-030	2.32e-021±3.33e-042	8.47e-028±4.55e-055
	SOS	8.03e-020±1.68e-019	4.92e-044±1.13e-043	1.18e-068±2.78e-068	4.96e-092±9.83e-092	5.71e-115± 1.20e-115
	ISOS	**1.01e-097±4.42e-097**	**8.75e-204±0e+00**	**572e-310±0e+00**	**0**	**0**
f2	DE	7.07e+002±5.43e+002	1.13e-003±1.09e-006	3.34e-010±4.87e-020	1.31e-016±9.71e-033	4.63e-023±6.52e-046
	SOS	4.57e-021±5.52e-021	5.26e-045±1.33e-044	7.12e-069±1.53e-068	3.77e-092±1.79e-091	3.69e-117±1.01e-116
	ISOS	**1.46e-098±5.76e-098**	**1.10e-204±0e+00**	**4.35e-310±0e+00**	**0**	**0**
f3	DE	6.26e-002±5.98e-004	1.22e-005±1.53e-011	2.89e-009±1.18e-018	5.75e-013±7.50e-026	1.16e-016±4.61e-033
	SOS	1.83e-010±1.80e-010	1.89e-022±3.02e-022	3.99e-034±6.07e-034	3.59e-046±4.49e-046	8.06e-058±2.24e-057
	ISOS	**1.56e-047±5.64e-047**	**1.01e-100±4.23e-100**	**6.34e-153±294e-152**	**2.85e-205±0e+00**	**1.15e-256±0e+00**
f4	DE	3.17e-001±2.02e-002	1.32e-007±6.96e-015	3.89e-014±9.50e-028	1.53e-020±7.28e-041	5.49e-027±1.43e-053
	SOS	4.18e-019±7.06e-019	4.18e-043±1.08e-042	3.76e-067±7.13e-067	2.44e-091±6.99e-091	7.62e-115±3.48e-114
	ISOS	**3.78e-095±1.98e-094**	**507e-202±0e+00**	**1.02e-303±0e+00**	**0**	**0**
f5	DE	-4.94±2.09	-4.94±1.45	-4.69±2.08	-5.12±2.58	-4.86±2.69
	SOS	-8.70±339e-001	-9.36±2.64e-001	-9.48±1.55e-001	-9.58±1.95e-001	-9.61±1.69e-001
	ISOS	**-12.02±9.08e-001**	**-12.80±8.35e-001**	**-12.93±981e-001**	**-13.44±1.37e-001**	**-14.10±1.50e-001**
f6	DE	**0.42e+003±0.65e+002**	**26.57±0.47**	**25.42±1.89**	**24.21±0.43**	**23.64± 0.75**
	SOS	9.90e-001±1.32e-006	9.901e-01±2.26e-009	9.9e-001±2.59e-013	9.9e-001±3.88e-016	9.9e-001±4.84e-016
	ISOS	**5.67e-002±7.62e-002**	**8.4e-003±207e-002**	**2.5e-003±104e-002**	**2.2e-003±8.6e-003**	**3.21e-003±874e-003**
f7	DE	**99.94±0.13e+003**	**60.29±0.13e+003**	**46.79±62.62**	**35.67±63.26**	**26.57±55.75**
	SOS	2.85e-001±3.37e-001	4.78e-004±20e-03	1.89e-006±7.84e-006	4.20e-009±1.71e-008	2.19e-013±8.39e-013
	ISOS	**4.08e-003±1.4e-003**	**2.85e-008±7.04e-008**	**2.09e-011±4.16e-011**	**1.99e-015±8.4e-010**	**9.91e-019±5.52e-018**
f8	DE	**3.08e-001±2.35e-001**	**6.39e-009±2.08e-016**	**2.00e-014±4.29e-027**	**0**	**0**
	SOS	9.55e-001±115e-001	6.79e-002±5.93e-002	4.58e-002±5.32e-002	2.71e-002±2.27e-002	1.23e-002±1.82e-002
	ISOS	**4.32e-001±115e-001**	**3.93e-002±7.99e-002**	**1.34e-002±1.47e-002**	**9.83e-003±8.54e-003**	**2.48e-003±4.67e-003**

4.3. **Test of the performance of ISOS.** The research results of the related improved algorithms have not been published, so we only compared ISOS with SOS and DE for convergence rate and convergence accuracy.

4.3.1. *Analysis of convergence accuracy of ISOS.* To ensure fair comparison, the same population size, the same dimension and the same termination criterion in each run are used. Each function was run 30 times independently. The performance of ISOS was investigated by the maximum, minimum, mean and variance of the test results. The related data are shown in Table 2.

As shown in Table 2, at the same evaluation number, the optimal value of ISOS was better than SOS for all the test functions. In addition to function 8, the SOS is superior to the DE on the rest of the functions. Although the performance of ISOS in function 8 is not as good as DE, it is better than SOS. The f1, f2, f3, f4, and f8 values converge to the theoretical optimal value. Although the average of the f8 function is not ideal, the optimal value was found 21 times in 30 replicates. ISOS was significantly improved compared with SOS model. Additionally, compared with SOS, ISOS exhibits smaller variance, indicating that the ISOS algorithm is more stable.

4.3.2. *Analysis of convergence accuracy of ISOS.* To compare the convergence speed of the two algorithms, we set the function evaluation times to 20000, 40000, 60000, 80000, and 100000. The population size is 50, and the dimension is 30. Each function is run 30 times independently. The related data are shown in Table 3.

It can be seen from Table 3 that ISOS has better convergence accuracy than SOS on all benchmark functions under the same evaluation times of function, which indicates that the convergence rate of ISOS algorithm is improved.

In summary, compared with ISOS and SOS, the convergence rate and convergence accuracy are significantly improved, indicating that the two improvements are effective, for a balance of the exploration and development capabilities of the algorithm.

5. **Conclusions.** In this paper, Symbiotic Organisms Search with Mixed Strategy is proposed. A mixed population regeneration strategy is introduced in the mutualism and commensalism phase, and a new population-updating formula is introduced to replace the blindness of the random search for the parasitic mechanism in the parasitism phase. The former speeds up the convergence rate of the population, and the latter increases the diversity of the population and avoids the local optimization. Experimental results on eight benchmark functions show that the ISOS improves convergence and robustness compared to the SOS, and can avoid premature to an extent.

Acknowledgment. This work was supported by the National Natural Science Foundation of China under Grant NO.61501107, the Education Department of Jilin province science and technology research project of “13th Five-Year” and Innovation Fund for graduate students of Northeast Electric Power University.

REFERENCES

[1] M. Y. Cheng, D Prayogo, Symbiotic Organisms Search: A new Metaheuristic Optimization Algorithm, *Journal of Computers and Structures*, vol. 139, pp. 98–112, 2014

[2] H. Li, Z. Y. Chen, M. J. Zhou, Genetic Algorithm Application on Optimal Design of Strip Foundation, *Journal of Open Cybernetics & Systemics Journal*, vol. 9, no. 8 pp. 335–339, 2015.

[3] J. Kennedy, R. Eberhart, Particle Swarm Optimization, *Proceedings of the IEEE international conference on neural networks. Perth,* Australia, 1942–1948, 1995.

[4] R. Storn, K. Price, Differential Evolution –A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces, *Journal of Global Optimization,* vol. 11, pp. 341–59. 1997.

[5] Y. Lou, J. L. Li, A Differential Evolution Algorithm based on Ordering of Individuals, *The 2nd Int Conf on Industrial Mechatronics and Automation.* Wuhan, China, 105-108, 2010.

[6] Tsai, C. Hsing, Novel Bees Algorithm: Stochastic Self-adaptive Neighborhood, *Journal of Elsevier Inc.*, vol. 247, pp. 1161-1172, 2014.

[7] K. Željko, M R Rapaić Z D, G Jeličić Particle Swarm Optimization Algorithm- Theoretical and Empirical Analysis with Application in Fault Detection, *Journal of Applied Mathematics & Computation*, vol. 217 pp. 10175-10186,2011.

[8] Z. L. Meng, J. S. Pan, H. R. Xu, QUasi-Affine TRansformation Evolutionary (QUATRE) algorithm: A cooperative swarm based algorithm for global optimization, *Knowl.-Based Syst*, vol. 109, pp.104-121, 2016.

[9] Z. Y. Meng, J. S. Pan, Monkey King Evolution: A new memetic evolutionary algorithm and its application in vehicle fuel consumption optimization, *Knowledge Based Systems*, vol. 97, pp. 144-157, 2016

[10] X. Yao, Y. Liu, G. Lin. Evolutionary Programming Made Faster, *Journal of IEEE Trans. on Evolutionary Computer*, vol. 3, no. 2 pp. 82-102, 1999.

[11] S. C. Chu, P. W. Tsai and J. S Pan, Cat Swarm Optimization, 9th Pacific Rim International *Conference on Artificial Intelligence, LNAI* 4099, pp. 854-858, 2006

[12] E. Khorram, H. Zarei, Multi-objective Optimization Problems with Fuzzy Relation Equation Constraints Regarding Max-average Composition, *Journal of Math. Comput. Modelling,* vol. 49, no. 5–6, pp. 856–867, 2009.

[13] M. Hu, T. Wu, J. D. Wei, An Intelligent Augmentation of Particle Swarm Optimization with Multiple Adaptive Methods, *Journal of Information Sciences*, vol. 213, no. 5, pp. 68-83, 2012.