International Journal of Engineering & Technology, 7 (4) (2018) 4146-4150

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet. v7i4.19991
Research paper

Chaotic multi-swarm particle swarm approach for solving numerical optimization problems

Asmaa Hussein Alwan [1] *, Abdullah Hasan Ali [2]

[1] *Department of Computer Center, Ibn Rushd, University of Baghdad, Baghdad, Iraq*
[2] *Supervision and Scientific Evaluation Apparatus, Ministry of Higher Education and Scientific Research, Baghdad, Iraq*
**Corresponding author E-mail: asmaa.hussien@ircoedu.uobaghdad.edu.iq*

Abstract

Different fields of study are faced with several optimization problems which can either be discrete, nonlinear, linear, continuous, non-smooth, or non-convex in nature. The continuously differentiable problems can be handled using several conventional methods such as the gradient-based methods, but such methods may not be ideal for the complex problems such as the non-convex or non- differentiable problems. Despite the existing number of methods for solving complex optimization problems, achieving optimal results is still difficult without much computational effort and cost input. The Particle Swarm Optimization (PSO) algorithm is a common optimization algorithm which is still suffering from an unbalanced local search (exploitation) and global search (exploration). The Meeting Room Approach (MRA) was recently developed as a multi-swarm model which for enhancing the exploration and exploitation in the PSO algorithm. In proposed Multi-swarm approach, the algorithm starts from a uniformly generated positions, which may start from not good positions. In other words, the algorithm may have a slow convergence due to the initial positions. In this paper, a Logistic map was used to initiate a multi-swarm PSO to enable it to start from better positions. The performance of the proposed algorithm was evaluated on several numerical optimization problems and its convergence was found to be faster compared to the original model.

Keywords: *Optimization; Operational Research; Numerical Problems; Chaotic Maps; Meeting Room Approach.*

1. Introduction

Optimization problems are a unique type of problem which seeks to either maximize or minimize the mathematical function of several variables with respect to some constraints. Most of the real-world or theoretical optimization problems can be generally modeled in this framework. Mathematical models can be generally represented as [1–3]:

$$\text{Max or Min } f_i(x), (i = 1, 2, 3, \ldots, M) \tag{1}$$

$$\text{subject to } h_j(x) = 0, (j = 1, 2, 3, \ldots, J), \tag{2}$$

$$g_k(x) \le 0, (k = 1, 2, 3, \ldots, K) \tag{3}$$

Where x = decision variables, $f_i(x)$, $h_j(x)$ and $g_k(x)$ = functions of the design vector:

$$x = (x_1, x_2, x_3, \ldots, x_n)^T \tag{4}$$

Hence, each optimization problem consists of an objective function $f_i(x)$, the variables (x_i), and the constraints of the problems.
All forms of life are accountable to nature, including the stellar, galactic, and planetary systems. Nature is mainly characterized by its ability to ensure its equilibrium through various means (both known and unknown). This is simply illustrated by the concept of seeking optimum in all aspects of life [4], [5]. There are certain goals which must be achieved, as well as certain demands that must be met while searching for optimum [6–9]. This process of searching for the optimum can be expressed as an optimization problem [10–12]; it can be simply put as a process of finding the optimum solution to a problem with respect to a performance matrix often referred to as an objective function (OF) (problem-specific) in most engineering and computing applications [13–15].
The past few years witnessed the development of several mathematical methods and the commonest of these methods is the metaheuristics which are regarded as an efficient method for achieving acceptable solutions to a complex optimization problem in a reasonable computational time using trial and error. The extent of a solution is dependent on the nature of the problem at hand but finding the best solution within an appropriate time frame is a major aim. There is no guarantee that a given approach or a chosen algorithm will proffer the best solution even though the basic components which can make it work may be known. However, the major aim is to have a reliable and efficient framework algorithm which can provide the best solution at any time. Among the achieved quality solutions, some are expected to be near optimal even though no such assurance for such optimality exists [16].
Several metaheuristics were proposed within the last two decades and the PSO is one of the popular metaheuristics; it was developed as a swarm intelligence technique with inspiration drawn from the social flocking and schooling behaviors of birds and fishes [17], [18]. For each swarm, there are movements with variable velocities to better locations with a better food or experience compared to the previously explored locations. The PSO has no explicit selection function and this lack is compensated using leaders as guides during exploration. Each particle in the swarm is considered as a potential solution in the solution space, and a solution update is achieved by updating the position of each particle.

A new multi-swarm model called Meeting Room Approach (MRA) has been proposed for enhancing the exploration and exploitation balancing capabilities of the PSO [19–21]. The MRA consists of several swarms referred to as 'clans' and each clan has a leader (representing the best solution in the clan or the local best solution). The clan leaders meet periodically to select an overall best leader among them (the global best solution). The overall best leader has authority over all the other leaders and can control them or lead them to the best positions. The balance between exploration and exploitation is influenced by the interaction between the overall best leader (global best) and the normal leaders (local best). This interaction also maintains a suitable population diversity even when approaching the global solution; therefore, the risk of being trapped at the local sub-optimal is reduced.
The initial positions of all particles in of Multi-swarm PSO (MPSO) are generated by using a uniform distribution equation, which may lead to starting from a not good positions, thus, MPSO may have a slow convergence. In this paper, MPSO is enhanced by using a chaotic map, meaning that, all particles are started from positions generated by using different randomization technique which enhance the searching process of MPSO.
The remaining parts of this paper are organized as follows: Section 2 presented the original PSO and the MRA, including the initialization of MRA, while section 3 presented the explanation of the proposed enhancement to the MRA. Section 4 presented the findings of the experimental evaluations while the conclusion of the study was presented in section 5.

2. Particle swarm optimization

2.1. The standard version

The PSO is one of the nature-inspired metaheuristics which was developed based on inspiration drawn from the flocking behavior of birds. In the PSO, it is assumed that a flock of birds is distributed randomly in an area with just a piece of food (Fig. 1a). The available food is represented by the dot on the tree, with an unknown position to each bird even though their distance from the food is known. Furthermore, a bird that is nearest to the food can signal the other birds to fly towards the food. Here, the food is considered as the optimal value (Fig. 1b) and each bird is considered as a particle. The distance between the food and each bird is a value of the OF. Hence, the birds' flocking process can be defined as a function optimization process. Xi in Fig. 1b is the nearest particle to the goal, and as such, is considered as the current global optimal particle with a distance from goal expressed as Nbesti, representing the global optimal value [17], [18]. The PSO is conceptualized on the idea that each particle is defined with a position and velocity while searching for the global optimum of an NP-hard problem. The position of the particles is iteratively updated based on their respective local and global optimal positions so far visited. The updated position of each particle (e.g., particle i) is defined as:

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (1)$$

where t = current (temporal) status, $t+1$ = post-update status, $X_i(t+1)$ = new particles' velocity. It should be noted that the time difference $\Delta t = (t+1) - 1$ is a time unit.
The velocity of particle i is expressed thus:

$$V_i(i+t) = \omega V_i(t) + c_i r_i \left(X_i^P - X_i(t)\right) + c_2 r_2 \left(X^G - X_i(t)\right) \quad (2)$$

where $v_i(t)$ = current particles' velocity, X_i^P= local best position of the particle, X^G = global best position of the particle at the swarm level, and ω, c_1, and c_2 = constants that determines the importance of each velocity component, r_1 and r_2 = random values in the range of [0, 1].

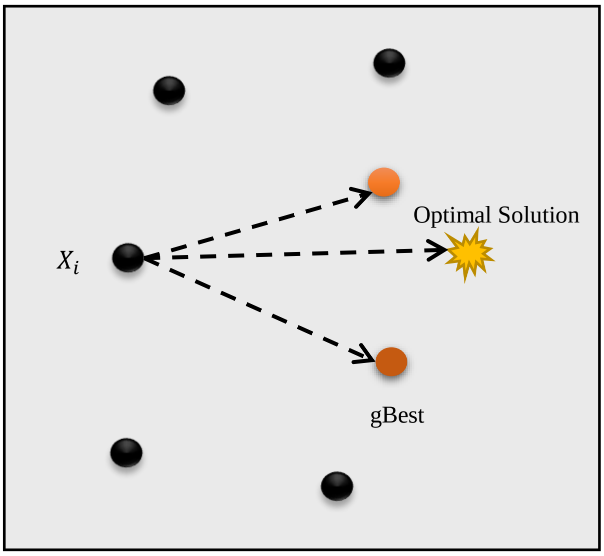

Fig. 1: Particle Swarm Optimization.

Despite the numerous modifications of the PSO, it is still prone to several problems which demand attention. Such problems include: its premature convergence (where it ends up searching for the early best solutions especially in multimodal functions); its convergence speed (gets trapped while exploiting for the global solution despite establishes the best solution in the early search stage); the quality of its solution is low due to the inherent complexity, discontinuity, and multimodality problems; the uncertainty of its solutions due to its stochastic nature which makes it difficult to produce different solutions in different runs; it has a simple solution update strategy which makes it hard to achieve better solutions in complex situations.

2.2. Meeting room approach

The major concept of a multi-swarm is the inter-group interaction that exists between groups during a solution search. Numerous multi-swarm techniques have been developed, with each idea drawing inspiration from natural processes. This paper presents a novel cooperative multi-swarm technique whose inspiration was drawn from the human social behavior. It was inspired by the interaction between human groups (referred to as 'Clans') and their leaders. The scheme is made up of several clans and each clan has several solutions (as represented by the clan members). In each clan, the best member is chosen as the clan leader and this leader controls the activities of the members of the clan in terms of where and when to move to a better location[19–21].
The clan leaders meet in each generation to select the overall best leader whose positional information will be transmitted to the normal clan leaders for them to update the positions. This positional information dissemination helps to balance the exploration and exploitation stages of the PSO. Figure 2 depicts the model of the proposed multi-swarm scheme called Meeting Room Approach (MRA). In this figure, each clan executes a single PSO search (including velocity and positional updating) and generates a new local population. After generating the new populations for each clan, the leader of each clan delegates the leader (best solution) to the meeting room where the overall best leader is selected. The position of the newly selected overall best leader is shared with the ordinary leaders to update their positions using the following equations:

$$w^{Ln} = \left(\frac{w^{Lg} - w^{Ln}}{Itr}\right) \times rand() \quad (3)$$

$$v_i^{Ln}(t+1) = w^{Ln} \times v_i^{Ln}(t) + rc\left(P_g^{\,L} - P_n^{\,L}(t)\right) \quad (4)$$

$$x_i^{Ln}(t+1) = x_i^{Ln}(t) + v_i^{Ln}(t) \quad (5)$$

where Ln = normal leaders, Lg = overall best leader, $x_i{}^L$ = normal leaders' position, $v_i{}^{Ln}$ = normal leaders' velocity, w^{Lg} and w^{Ln} = best and normal leaders' inertia weight, respectively.
A new leader is elected for each swarm after each generation because of the changes in the positions of the clan members. The exploration of the PSO is controlled by the new equation of the inertia in the meeting room. Figure 3 presents the pseudo-code of the proposed Multi-Swarm Particle Swarm Optimisation (MPSO) algorithm.

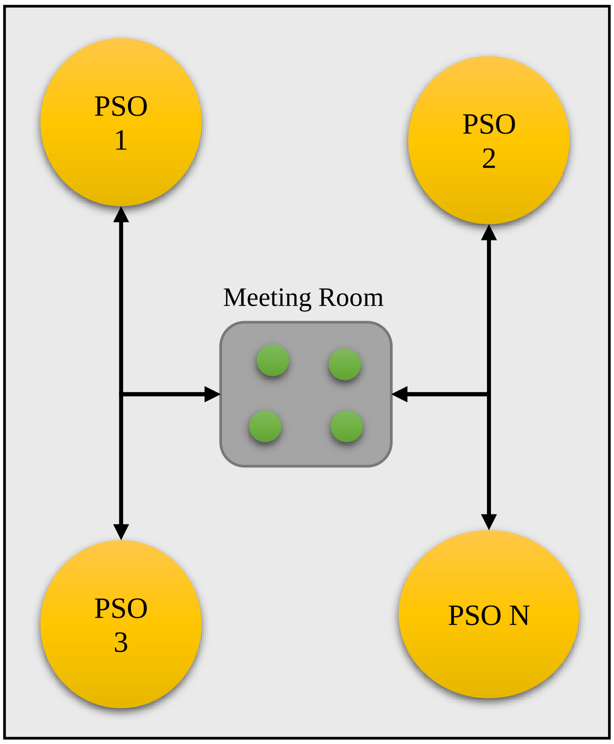

Fig. 2: The structure of Meeting Room Approach.

The MRA with PSO (MPSO) has shown a superior performance in solving several numerical problems. However, the initialization step of the MPSO is based on uniform distribution equations which may start from inappropriate positions. Thus, the particles may explore a wrong area in the search space, leading to being trapped in local optima. The main contribution of this study is the initialization of the particles in each swarm using a well-known chaotic map (logistic map) to enhance the starting positions of all the particles, as well as the convergence of the algorithm.

3. Chaotic meeting room approach

The proposed Chaotic Multi-Swarm Particle Swarm Optimization (CMPSO) is different from the original version of MPSO in the initialization step while the remaining steps are the same, as given below:
Step 1: Each particle (i.e., the position) is initialized using Logistic Map, which is given in the following equation:

$$X_{i+1} = \mu X_i\,(1 - X_i) \quad (6)$$

Where X_i= a real value in range 0 and 1, which represents a single dimension of any given problem, and μ = the control parameter – or mutation – of logistic map, which is in range 0 and 4.
Step 2: Calculate the fitness function of each particle in each swam based on their generated positions.
Step 3: For each clan, and for each particle in the clan, update the velocity using equation 2; then, update the position using equation 1. After updating the positions, evaluate the new fitness function for them. If the new position is better than the previous one, then, update the$pBest$.
Step 4: Determine the best particle in each clan as the leader of that clan.
Step 5: Update the controlling parameters of each particle using equations 3, 4, and 5.
Step6: Update the best leader ever.
It is worth mentioning that all the random values in the proposed algorithm are generated using equation 6. The flowchart of the proposed CMPSO is given in Figure 3.

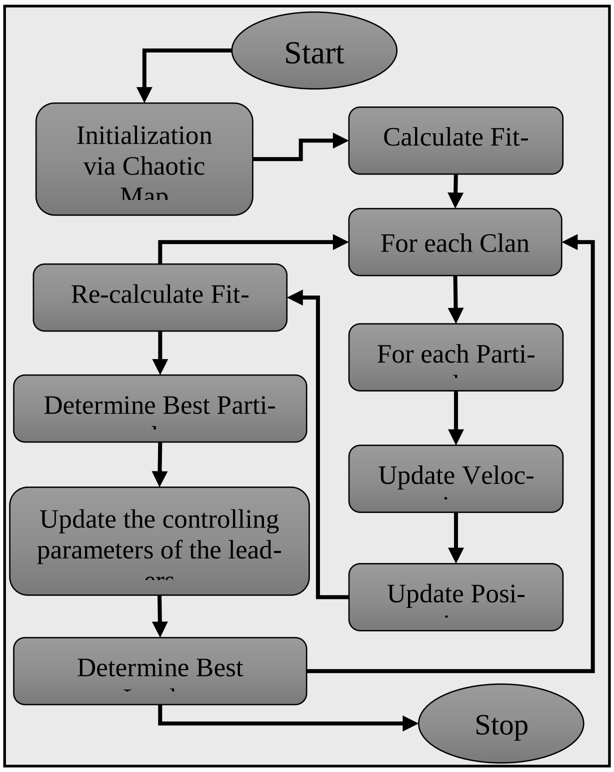

Fig. 3: The Flowchart of CMPSO.

4. Results and discussion

The results of the benchmarking evaluations commonly used in the evolutionary literature are presented in this section [22]. Each test function varies in terms of modality (unimodal and multimodal) and the number dimensions (fixed and dynamic). In this study, CPSO has been examined on the exact same test function used in [19]. They are:

1) Sphere Function (Continuous, Separable, Unimodal)

$$f(x_i) = \sum_{i=1}^{D} x^2 \;\; D = 30 \quad (7)$$

2) Griewank Function (Continuous, Non-Separable, Unimodal)

$$f(x_i) = \sum_{i=1}^{D} \frac{x_i^2}{4000} - \prod cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \;\; D = 30 \quad (8)$$

3) Rastrigin Function (Continuous, Separable, Multimodal)

$$f(x_i) = 10d + \sum_{i=1}^{d}[x_i^2 - 10\,cos(2\pi x_i)] \;\; D = 30 \quad (9)$$

4) Ackely Function (Continuous, Non-Separable, Multimodal)

$$f(x) = -20e^{-0.02\sqrt{D^{-1}\sum_{i=1}^{D} x_1^2}} - e^{D^{-1}\sum_{i=1}^{D} cos(2\pi x_i)} + 20 + e \quad (10)$$

The performance of CMPSO was evaluated by comparing with that of the original PSO[23], Master-Slave PSO (MCPSO)[24], and the original MPSO [19]. The values of the controlling parameters are given in Table 1.

Table 1: The Values of the Controlling Parameters

Algorithm	Parameter	Value
PSO	Inertia weight ω	0.9-0.4
	No. of Swarms	1
	c_1, c_2	1.42
	Swarm Size	50
MCPSO	Inertia weight ω	0.9-0.4
	No. of Swarms	5
	c_1, c_2, c_3	1.42
	Swarm Size	50
MPSO	Inertia weight ω^{Ln}	0.8-0.5
	Inertia weight ω^{Lg}	0.9-0.7
	c_1, c_2	1.42
	No. of Clans	5
	Clan Size	10
CMPSO	Inertia weight ω^{Ln}	0.8-0.5
	Inertia weight ω^{Lg}	0.9-0.7
	c_1, c_2	1.42
	No. of Clans	5
	Clan Size	10
	Initial Value X_0	0.11123
	Mutation μ	3.99999856

Table 2 presents the best and mean fitness values of the particles after 30 experimental runs over benchmark functions. Evidently, the MPSO outperformed the other benchmarking algorithms virtually in all the cases. A general analysis of the table shows the MPSO to contain 5 swarms, with each swarm consisting of 10 particles out of which only 5 are interacting in the meeting room. It can, therefore, be stated that the MPSO has less computational complexity and performed better when establishing the best solution. The ability of the CMPSO to evolve in situations where the algorithms may have converged is presented in Figures 4. Figure 4 portrays the comparison between the convergences of the all metaheuristics based on Ackely Test function.

Table 2: Results

Algorithm	Test	Average	Std. Deviation
PSO	Sphere	2.5457521	0.01485
	Griewank	0.0884741	0.97485
	Rastrigin	21.695847	0.34871
	Ackely	16.4875218	0.01348
MCPSO	Sphere	0.9854126	0.0014784
	Griewank	0.0078414	0.0009874
	Rastrigin	2.0018977	0.0078487
	Ackely	1.9984722	0.0084578
MPSO	Sphere	0.0007845	0.0000148
	Griewank	0.0000897	0.0000668
	Rastrigin	0.0004687	0.0000159
	Ackely	0.0002648	0.0000588
CMPSO	Sphere	0.0007798	0.0000140
	Griewank	0.0000899	0.0000765
	Rastrigin	0.0004690	0.0000621
	Ackely	0.0002649	0.0000489

From Figure 4, it can be seen that CMPSO is much faster than the original version (i.e., MPSO) in the first 50 function evaluation. Meaning that, CMPSO has initial positions better than the uniformly generated positions. However, there is no much difference between MPSO and CMPSO in terms of the optimal solutions and standard deviation.

Although the statistical results presented in Table 2 provide a first insight into the performance of the algorithms, a pair-wise statistical test is typically used for a better comparison. For this purpose, by using the results obtained from 30 runs of each algorithm, a Wilcoxon Signed-Rank Test is performed with a statistical significance value (α = 0.05) The null-hypothesis is "There is no difference between the median of the solutions produced by algorithm A and the median of the solutions produced by algorithm B for the same benchmark problem". In table 3, the statistical analysis of CMPSO algorithm compared to the other three algorithms are given. In this table, α indicates the p-values, while the R column indicates whether CMPSO is better than MPSO (+) or it is equal (=).

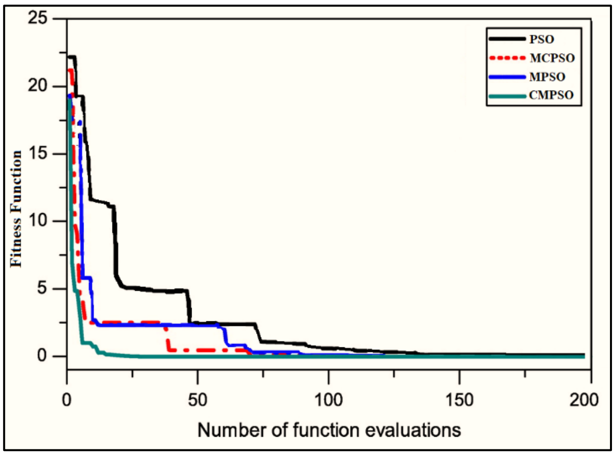

Fig. 4: Convergence Analysis.

Table 3: Statistical Results

Test	Vs. PSO		Vs. MCPSO		Vs. MPSO	
	α	R	α	R	α	R
Sphere	0.0004	+	0.0022	+	1	=
Griewank	0.0027	+	0.0018	+	1	=
Rastrigin	0.0012	+	0.0005	+	1	=
Ackely	0.0024	+	0.0003	+	1	=

Figure 5 below shoes a visual comparison between the new enhanced version CMPSO algorithm with the original MPSO algorithm. The box plot shows that both algorithms have almost the same range, however, CMPSO has a better mean and the distance between the max and min values are less than the values of MPSO which proofs that CMPSO is more stable than MPSO when solving the numerical optimization problems.

5. Conclusion

Metaheuristics are faced with several problems and one of such problems is striking a balance between their exploration and exploitation capabilities. The Meeting Room Approach (MRA) has been proposed for the enhancement of this balance in the Particle Swarm Optimization (PSO). However, this study proposed the enhancement of the MRA using a chaotic map where the particles are initialized based on a chaotic sequence (which is better than the uniform distribution). From the benchmarking results, the MRA was enhanced in terms of convergence speed when solving several numerical optimization problems. For future studies, the enhanced version of MPSO can be applied on different optimization problems such as training neural networks, or for selecting the best subset features from a known dataset.

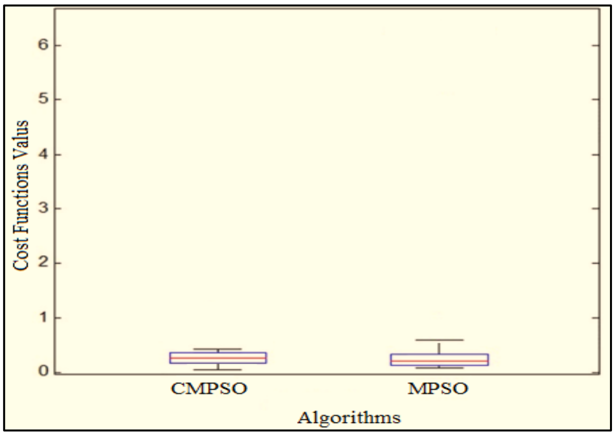

Fig. 5: Comparasion between CMPSO and MPSO.

References

[1] S. Koziel, X.S. Yang, Computational optimization, methods and algorithms, 2011. https://doi.org/10.1007/978-3-642-20859-1.

[2] T. Weise, M. Zapf, R. Chiong, A.J. Nebro, Why Is Optimization Difficult? In: R. Chiong (Ed.), Nature-Inspired Algorithms Optim. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009: pp. 1–50. https://doi.org/10.1007/978-3-642-00267-0_1.

[3] S. Mirjalili, SCA: A Sine Cosine Algorithm for solving optimization problems, Knowledge-Based Syst. 96 (2016) 120–133. https://doi.org/10.1016/j.knosys.2015.12.022.

[4] C. Arango, P. Cortes, L. Onieva, A. Escudero, Simulation-optimization models for the dynamic berth allocation problem, Comput. Civ. Infrastruct. Eng. 28 (2013) 769–779 https://doi.org/10.1111/mice.12049.

[5] J.Y.J. Chow, Activity-based travel scenario analysis with routing problem reoptimization, Comput. Civ. Infrastruct. Eng. 29 (2014) 91–106. https://doi.org/10.1111/mice.12023.

[6] X. Chen, L. Zhang, X. He, C. Xiong, Z. Li, Surrogate-Based Optimization of Expensive-to-Evaluate Objective for Optimal Highway Toll Charges in Transportation Network, Comput. Civ. Infrastruct. Eng. 29 (2014) 359–381. https://doi.org/10.1111/mice.12058.

[7] L. Jia, Y. Wang, L. Fan, Multiobjective bilevel optimization for production-distribution planning problems using hybrid genetic algorithm, Integr. Comput. Aided. Eng. 21 (2014) 77–90. https://doi.org/10.3233/ICA-130452.

[8] R. Faturechi, E. Miller-Hooks, a Mathematical Framework for Quantifying and Optimizing Protective Actions for Civil Infrastructure Systems, Comput. Civ. Infrastruct. Eng. 29 (2014) 572–589. https://doi.org/10.1111/mice.12027.

[9] M. Aldwaik, H. Adeli, Advances in optimization of highrise building structures, Struct. Multidiscip. Optim.50 (2014) 899–919. https://doi.org/10.1007/s00158-014-1148-1.

[10] H. Adeli, O. Kamal, Efficient optimization of space trusses, Comput. Struct.24 (1986) 501–511. https://doi.org/10.1016/0045-7949(86)90327-5.

[11] R. Smith, E. Ferrebee, Y. Ouyang, J. Roesler, Optimal Staging Area Locations and Material Recycling Strategies for Sustainable Highway Reconstruction, Comput. Civ. Infrastruct. Eng. 29 (2014) 559–571. https://doi.org/10.1111/mice.12089.

[12] F. Peng, Y. Ouyang, Optimal clustering of railroad track maintenance jobs, Comput. Civ. Infrastruct. Eng. 29 (2014) 235–247. https://doi.org/10.1111/mice.12036.

[13] H. Adeli, K.C. Sarma, Cost Optimization of Structures: Fuzzy Logic, Genetic Algorithms, and Parallel Computing, 2006. https://doi.org/10.1002/0470867353.

[14] S. Boyd, L. Vandenberghe, M. Grant, Advances in Convex Optimization, in 2006 Chinese Control Conf., 2006: p. PL-40-PL-40. https://doi.org/10.1109/CHICC.2006.280567.

[15] D. Luo, Z. Ibrahim, Z. Ismail, B. Xu, Optimization of the geometries of biconical tapered fiber sensors for monitoring the early-age curing temperatures of concrete specimens, Comput. Civ. Infrastruct. Eng. 28 (2013) 531–541. https://doi.org/10.1111/mice.12022.

[16] X.S. Yang, Nature-Inspired Metaheuristic Algorithms, 2010. https://doi.org/10.1016/B978-0-12-416743-8.00005-1.

[17] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, MHS'95. Proc. Sixth Int. Symp. Micro Mach. Hum. Sci. (1995) 39–43. https://doi.org/10.1109/MHS.1995.494215.

[18] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Micro Mach. Hum. Sci. 1995. MHS'95. Proc. Sixth Int. Symp., 1995: pp. 39–43. https://doi.org/10.1109/MHS.1995.494215.

[19] S.Q. Salih, A.A. Alsewari, Solving large-scale problems using multi-swarm particle swarm approach, Int. J. Eng. Technol. 7 (2018) 1725–1729.

[20] S.Q. Salih, A.A. Alsewari, B. Al-Khateeb, M.F. Zolkipli, And Novel Multi-Swarm Approach for Balancing Exploration and Exploitation in Particle Swarm Optimization, in: 3rd Int. Conf. Reliab. Inf. Commun. Technol. 2018 (IRICT 2018), Springer, 2018.

[21] S.Q. Salih, A.A. Alsewari, B. Al-Khateeb, M.F. Zolkipli, Novel Multi-swarm Approach for Balancing Exploration and Exploitation in Particle Swarm Optimization, in: Recent Trends Data Sci. Soft Comput Springer International Publishing, Cham, 2019: pp. 196–206. https://doi.org/10.1007/978-3-319-99007-1_19.

[22] M. Jamil, X.S. Yang, H.J.D. Zepernick, Test Functions for Global Optimization: A Comprehensive Survey, in: Swarm Intell. Bio-Inspired Comput. 2013: pp. 193–222. https://doi.org/10.1016/B978-0-12-405163-8.00008-9.

[23] M. Clerc, Standard PSO 2011 (SPSO), Part. Swarm Cent. (2011). https://www.particleswarm.info/Programs.html.

[24] B. Niu, Y. Zhu, X. He, H. Wu, MCPSO: A multi-swarm cooperative particle swarm optimizer, Appl. Math. Comput.185 (2007) 1050–1062. https://doi.org/10.1016/j.amc.2006.07.026.