Structural Engineering and Mechanics, Vol. 42, No. 6 (2012) 783-797

DOI: http://dx.doi.org/10.12989/sem.2012.42.6.783

A hybrid CSS and PSO algorithm for optimal design of structures

A. Kaveh*[1] and S. Talatahari[2a]

[1]*Centre of Excellence for Fundamental Studies in Structural Engineering, Iran University of Science and Technology, Narmak, Tehran-16, Iran*
[2]*Marand Faculty of Engineering, Tabriz University, Tabriz, Iran*

(Received December 16, 2010, Revised April 20, 2012, Accepted May 1, 2012)

Abstract. A new hybrid meta-heuristic optimization algorithm is presented for design of structures. The algorithm is based on the concepts of the charged system search (CSS) and the particle swarm optimization (PSO) algorithms. The CSS is inspired by the Coulomb and Gauss's laws of electrostatics in physics, the governing laws of motion from the Newtonian mechanics, and the PSO is based on the swarm intelligence and utilizes the information of the best fitness historically achieved by the particles (local best) and by the best among all the particles (global best). In the new hybrid algorithm, each agent is affected by local and global best positions stored in the charged memory considering the governing laws of electrical physics. Three different types of structures are optimized as the numerical examples with the new algorithm. Comparison of the results of the hybrid algorithm with those of other meta-heuristic algorithms proves the robustness of the new algorithm.

Keywords: charged system search; particle swarm optimization; structural optimum design; hybrid methods; truss structures; frames; grillage systems

1. Introduction

Recently, many meta-heuristic algorithms have been developed for optimization problems, and because of their high potential for modeling engineering problems in environments which have been resistant to solution by classic techniques, these methods have attracted a great deal of attention (Kaveh and Talatahari 2009a). Although these methods do not require gradient information and possess better global search abilities than the conventional optimization algorithms (Coello 2002), however, hybridization can strengthen and improve the searching abilities of these algorithms. The contribution of this paper is to develop a new hybrid algorithm using the concepts of the charged system search (CSS) and the particle swarm optimization (PSO) algorithms.

The charged system search is the most recently introduced meta-heuristic algorithm (Kaveh and Talatahari 2010a), which has been utilized for optimum design of different types of structures consisting of trusses, frames and grillage systems (Kaveh and Talatahari 2010b, c). The governing

*Corresponding author, Professor, E-mail: alikaveh@iust.ac.ir
[a]Assistant Professor, E-mail: siamak.talat@gmail.com

laws from the physics initiate the base of the CSS algorithm. CSS is a multi-agent algorithm in which each agent is considered as a charged sphere. Since these agents are treated as charged particles that can affect each other according to the Coulomb and Gauss's laws from electrostatics, they are called Charged Particles (CPs). After determining the resultant force affected on each CP, the Newtonian motion law is utilized to determine the movement of the agents. The successive moving of CPs considering the resultant forces directs the agents toward optimum solutions.

The particle swarm optimization algorithm is based on the swarm intelligence and its ability is underpinned by the fact that decentralized biological creatures can often accomplish complex goals by cooperation. The particle swarm optimization algorithm is initialized with a population (swarm) of random potential solutions (particles). Each particle iteratively moves across the search space and is attracted to the position of the best fitness historically achieved by the particle itself (local best) and by the best among the neighbors of the particle (global best), (Kennedy *et al*. 2001). Compared to other evolutionary algorithms based on heuristics, the advantages of PSO consist of easy implementation and a smaller number of parameters to be adjusted. However, its practical use in solving engineering optimization problems is severely limited due to the high computational cost of slow convergence rate (Smith 1998) and the PSO had difficulties in controlling the balance between exploration and exploitation (Angeline 1998).

The remainder of this paper is organized as follows. Section 2 presents the statement of optimum design of structures. A brief review of the CSS and PSO algorithms are presented in Section 3. The new hybrid method is explained in section 4. Section 5 studies various numerical examples to verify the efficiency of the new algorithm. The final section will contain the concluding remarks.

2. Statement of optimum design of structures

The aim of optimizing structures is to reach at a set of design variables that has the minimum weight satisfying certain constraints. This can be expressed as

$$\begin{aligned} &\text{Find} \quad \mathbf{X} = [x_1, x_2, \ldots, x_{ng}] \\ &\qquad\quad x_i \in D_i \\ &\text{to minimize} \quad W(\mathbf{X}) = \sum_{i=1}^{nm} \gamma_i \cdot x_i \cdot L_i \\ &\text{subject to:} \quad g_j(\mathbf{X}) \le 0 \qquad j = 1, 2, \ldots, n \end{aligned} \tag{1}$$

where $\mathbf{X}$ is a set of design variables; ng is the number of groups (number of design variables); D_i is an allowable set of values for the design variable x_i; $W(\mathbf{X})$ denotes the weight of the structure; nm is the number of members making up the structure; γ_i is the material density of member i; L_i is the length of member i; A_i represents the cross-sectional area of member i; $g_j(\mathbf{X})$ is the design constraints; and n is the number of the constraints. Some structural examples are selected from literature and their related constraints are as follows.

2.1 Constraint conditions for truss structures

For truss structures, the stress limitations for the members, and the displacement constraints for the nodes are imposed as

$$\delta_i \le \delta_i^u \qquad i = 1, 2, \ldots, nn \tag{2}$$

$$\sigma_i \le \sigma_i^u \qquad i = 1, 2, \ldots, nm \tag{3}$$

where nm is the number of members making up the structure; nn is the number of nodes; σ_i and δ_i represent the stress and nodal deflection, respectively; σ_i^b and δ_i^b are the allowable values.

2.2 Constraint conditions for steel frames

Optimal design of frame structures is subjected to the following constrains according to AISC-ASD provisions (1989)

$$\frac{f_a}{F_a} + \frac{f_{bx}}{F_{bx}} + \frac{f_{by}}{F_{by}} \le 1 \quad \text{For} \quad \frac{f_a}{F_a} \le 0.15 \tag{4}$$

$$\frac{f_a}{F_a} + \frac{C_{mx} f_{bx}}{\left(1 - \frac{f_a}{F'_{ex}}\right) F_{bx}} + \frac{C_{my} f_{by}}{\left(1 - \frac{f_a}{F'_{ey}}\right) F_{by}} \le 1 \quad \text{For} \quad \frac{f_a}{F_a} > 0.15 \tag{5}$$

$$\frac{f_a}{0.6 F_y} + \frac{f_{bx}}{F_{bx}} + \frac{f_{by}}{F_{by}} \le 1 \quad \text{For} \quad \frac{f_a}{F_a} > 0.15 \tag{6}$$

where f_a ($=P/A_i$) represents the computed axial stress. The computed flexural stresses due to the bending of the member about its major (x) and minor (y) principal axes are denoted by f_{bx} and f_{by}, respectively. F'_{ex} and F'_{ey} denote the Euler stresses about the principal axes of the member that are divided by a factor of safety of 23/12. The allowable bending compressive stresses about the major and minor axes are designated by F_{bx} and F_{by}. C_{mx} and C_{my} are the reduction factors, introduced to counterbalance overestimation of the effect of secondary moments by the amplification factors $(1 - f_a/F'_{ex})$ and F_a stands for the allowable axial stress under axial compression force alone.

The slenderness ratio limitation is considered as follows

$$\begin{cases} \lambda_i = \dfrac{k_i l_i}{r_i} \le 300 & \text{For tension members} \\ \lambda_i = \dfrac{k_i l_i}{r_i} \le 200 & \text{For compression members} \end{cases} \tag{7}$$

Eq. (7) represents the slenderness limitations imposed on all members such that the maximum slenderness ratio is limited to 300 for members under tension, and to 200 for the members under compression loads.

Geometric constraints are considered between beams and columns framing into each other at a common joint for practicality of an optimum solution generated as described in Kaveh and Talatahari (2009b).

The maximum lateral displacement limitations are considered as

$$\frac{\Delta_T}{H} \leq R \tag{8}$$

$$\frac{d_i}{h_i} \leq R_I, \quad i = 1, 2, \ldots, ns \tag{9}$$

where Δ_T is the maximum lateral displacement; H is the height of the frame structure; R is the maximum drift index (= 1/400); d_i is the inter-story drift; h_i is the story height of the ith floor; ns represents the total number of stories; and R_I is the inter-story drift index permitted by the code of the practice (= 1/400).

2.3 Constraint conditions for grillage systems

The displacement limitations is defined by Eq. (2), and the strength constraints are required to be imposed for grillage systems according to LRFD-AISC provisions (1999) in which the strength constraints are defined as follows

$$\frac{M_u}{\phi_b M_n} \leq 1 \tag{10}$$

$$\frac{V_u}{\phi_v V_n} \leq 1 \tag{11}$$

where $M_{u,i}$ is the required flexural strengths of member i; $M_{n,i}$ denotes the nominal flexural strengths; ϕ_b is flexural resistance reduction factor (ϕ_b = 0.90); $V_{u,i}$ is the factored service load shear for member i; $V_{n,i}$ is the nominal strength in shear; and ϕ_v represents the resistance factor for shear given as 0.9.

3. A review of the CSS and PSO

3.1 Charged system search algorithm

The Charged System Search (CSS) algorithm is based on the Coulomb and Gauss laws from electrical physics and the governing laws of motion from the Newtonian mechanics. This algorithm can be considered as a multi-agent approach, where each agent is a Charged Particle (CP). Each CP is considered as a charged sphere with radius a, having a uniform volume charge density and is equal to (Kaveh and Talatahari 2010a)

$$q_j = \frac{W_j - W_{worst}}{W_{best} - W_{worst}}, \qquad j = 1, 2, \ldots, N \tag{12}$$

where W_{best} and W_{worst} are the minimum and the maximum weight among all the particles; W_j represents the weight of the agent i, and N is the total number of CPs.

CPs can impose electrical forces on the others. The kind of the forces is attractive and its magnitude for the CP located in the inside of the sphere is proportional to the separation distance between the CPs, and for a CP located outside the sphere is inversely proportional to the square of the separation distance between the particles

$$\mathbf{F}_j = q_j \sum_{i,i \neq j} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2 \right) p_{ij} (\mathbf{X}_i - \mathbf{X}_j) , \quad \begin{cases} j = 1, 2, \ldots, N \\ i_1 = 1, i_2 = 0 \Leftrightarrow r_{ij} < a \\ i_1 = 1, i_2 = 1 \Leftrightarrow r_{ij} \geq a \end{cases} \tag{13}$$

where $\mathbf{F}_j$ is the resultant force acting on the jth CP; r_{ij} is the separation distance between two charged particles which is defined as follows

$$r_{ij} = \frac{\|\mathbf{X}_i - \mathbf{X}_j\|}{\|(\mathbf{X}_i + \mathbf{X}_j)/2 - \mathbf{X}_{best}\| + \varepsilon} \tag{14}$$

where $\mathbf{X}_i$ and $\mathbf{X}_j$ are the positions of the ith and jth CPs, respectively; $\mathbf{X}_{best}$ is the position of the best current CP with the minimal weight; and ε is a small positive number. The initial positions of CPs are determined randomly in the search space and the initial velocities of charged particles are assumed to be zero. P_{ij} determines the probability of moving each CP toward the others as

$$p_{ij} = \begin{cases} 1 & \dfrac{W_i - W_{best}}{W_j - W_i} > rand \text{ or } W_j > W_i \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

The resultant forces and the motion laws determine the new location of the CPs. At this stage, each CP moves toward to its new position considering the resultant forces and its previous velocity, as

$$\mathbf{X}_{j,new} = rand_{j1} \cdot k_a + \frac{\mathbf{F}_j}{m_j} \cdot \Delta t^2 + rand_{j2} \cdot k_v \cdot \mathbf{V}_{j,old} \cdot \Delta t + \mathbf{X}_{j,old} \tag{16}$$

$$\mathbf{V}_{j,new} = \frac{\mathbf{X}_{j,new} - \mathbf{X}_{j,old}}{\Delta t} \tag{17}$$

where k_a is the acceleration coefficient; k_v is the velocity coefficient to control the influence of the previous velocity; and $rand_{j1}$ and $rand_{j2}$ are two random numbers uniformly distributed in the range of (0,1). If each CP exits from the allowable search space, its position is corrected using the harmony search-based handling approach as described by Kaveh and Talatahari (2009a). In addition, to save the best design, a memory (Charged Memory) is considered containing the *CMS* number of positions for the so far best agents.

3.2 Particle swarm optimization

The Particle Swarm Optimization (PSO) is motivated from the social behavior of bird flocking and fish schooling which has a population of individuals, called particles, that adjust their movements depending on both their own experience and the population's experience (Kennedy *et al.* 2001). In other words, each particle in the PSO algorithm continuously focuses and refocuses on the effort of its search according to both local best and global best. In PSO, the position of each agent, $\mathbf{X}_i^k$, and its velocity, $\mathbf{V}_i^{k+1}$, are calculated as

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \mathbf{V}_i^{k+1} \tag{18}$$

$$\mathbf{V}_i^{k+1} = \omega \mathbf{V}_i^k + c_1 r_1 \circ (\mathbf{P}_i^k - \mathbf{X}_i^k) + c_2 r_2 \circ (\mathbf{P}_g^k - \mathbf{X}_i^k) \tag{19}$$

where ω is an inertia weight to control the influence of the previous velocity, r_1 and r_2 are two random vectors uniformly distributed in the range of (0,1), and c_1 and c_2 are two acceleration constants, and the sign "∘" denotes element-by-element multiplication. The above mentioned formulations of the PSO algorithm can be combined and rewritten as

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \omega \mathbf{V}_i^k + c_1 r_1 \circ (\mathbf{P}_i^k - \mathbf{X}_i^k) + c_2 r_2 \circ (\mathbf{P}_g^k - \mathbf{X}_i^k) \tag{20}$$

In some previous studies, to improve the performance of the algorithm, another term is added to above formulae as

$$\mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \omega \mathbf{V}_i^k + c_1 r_1 \circ (\mathbf{P}_i^k - \mathbf{X}_i^k) + c_2 r_2 \circ (\mathbf{P}_g^k - \mathbf{X}_i^k) + \sum_{j=1}^{ne} c_j r_j \circ (\mathbf{P}_j^k - \mathbf{X}_i^k) \tag{21}$$

where c_j, similar to c_1 and c_2, is a constant value, and r_j is a random vector. *ne* denotes the number of extra terms considered in the algorithm and $\mathbf{R}_j^k$ is defined based on the type of the algorithm being used. For example in the particle swarm with passive congregation (PSOPC), (He *et al.* 2004), the number of the extra terms is equal to unity, and $\mathbf{R}_j^k$ is one particle's location selected randomly from the current swarm; for improved PSO suggested by the authors (Kaveh and Talatahari 2009c), the number of these terms is equal to two and the first $\mathbf{R}_j^k$ is defined similar to the PSOPC and the second one is generated from the search space randomly. As the third example in the work of Xu and Xin (2005), $\mathbf{R}_j^k$ is selected as the best position of neighboring particle, where the definition of the neighborhood particle may be changed in different implementations of the approach. Often the aim of the third term is to increase the exploration ability of the algorithm.

4. A hybrid charged system search and particle swarm optimization algorithm

Both CSS and PSO are multi-agent algorithms in which the position of each agent is obtained by adding the agent's movement to its previous position; however the movement strategies are different. Though each algorithm has some positive characters that direct the searching process, however there are some disadvantages which make some problems in finding the optimum point or decrease the speed of the algorithm. This paper collects the advantages of both optimization methods to develop an efficient hybrid algorithm.

The PSO algorithm utilizes a velocity term which is a combination of the previous velocity, $\mathbf{V}_i^k$, movement in the direction of the local best, $\mathbf{P}_i^k$, and movement in the direction of the global best, $\mathbf{P}_g^k$. This means that at each iteration, a particle moves towards a direction computed from the best visited position (local best) and the best visited position of all particles in its neighborhood (global best). One of the greatest disadvantages of the PSO approach is the existence some difficulties in controlling the balance between the exploration and exploitation due to ignoring the effect of other agents in calculating the direction of agents (Angeline 1998), and this increases the probability of losing the favorite space containing the optimum solution. The potency of the PSO is summarized to find the direction of movements of agents, and therefore determining the acceleration constants (c_1 and c_2) become important.

Similar to the PSO, the CSS algorithm utilizes a term of the previous velocities, however, the CSS can determine the amount and the direction of a charged particle' movement. Since in the CSS, the movements are calculated based on the overall forces resulted by the agents and the movement updating is performed by considering the quality of the solutions, so not only the directions but also the amount of movements are determined.

In the present hybrid algorithm, the advantage of the PSO containing utilizing the local best and the global best is added to the CSS algorithm. The charged memory (**CM**) for the hybrid algorithm is treated as the local best in the PSO, and the **CM** updating process is defined as

$$\mathbf{CM}_{i,new} = \begin{cases} \mathbf{CM}_{i,old} & W(\mathbf{X}_{i,new}) \geq W(\mathbf{CM}_{i,old}) \\ \mathbf{X}_{i,new} & W(\mathbf{X}_{i,new}) < W(\mathbf{CM}_{i,old}) \end{cases} \tag{22}$$

in which the first term identifies that when the new position is not better that the previous one, the updating does not perform while when the new position is better than the stored so far good position, the new solution vector is replaced. In the first iteration, the vector stored in **CM** and the first positions of the agents will be identical. Considering the above mentioned new charged memory, the electric forces generated by agents are modified as

$$\mathbf{F}_j = \sum_{i \in S_1} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2\right)(\mathbf{CM}_{i,old} - \mathbf{X}_j) + \sum_{i \in S_2} \left(\frac{q_i}{a^3} r_{ij} \cdot i_1 + \frac{q_i}{r_{ij}^2} \cdot i_2\right) ar_{ij} p_{ij} (\mathbf{X}_i - \mathbf{X}_j) \tag{23}$$

where S_1 and S_2 are defined as follows

$$S_1 = \{t_1, t_2, \ldots, t_n | q(t) > q(j), j = 1, 2, \ldots, N, j \neq i, g\} \tag{24}$$

$$S_2 = S - S_1 \tag{25}$$

in which S_1 determines the set of agents utilized from **CM**; n denotes the number of **CM**' agents; S is utilized as a set of all agents' number and thus S_2 will be the set of current agents used for directing the agent j. Here, in the primary iterations n is set to two continuing the number of the best stored so far agent among all CPs (global best) and jth agent stored in the CM which is treated as local best. Then the number of used agents from **CM** is increased linearly and finally it reached to N in the last iterations. In this hybrid algorithm, $\mathbf{CM}_{i,old}$ will be treated similar to $\mathbf{P}_i^k$ in the PSO. The other modification is that the forces can be attractive or repulsive, and ar_{ij} is added to fulfill this aim which determines the kind of the force as

$$ar_{ij} = \begin{cases} +1 & \text{w.p.} \quad k_t \\ -1 & \text{w.p.} \quad 1 - k_t \end{cases} \tag{26}$$

where "*w.p.*" represents the abbreviation for "with the probability"; k_t is a parameter to control the effect of the kind of forces. Comparing to Eq. (13), this new formulae (Eq. (23)) considers the best so far location of agents and the best local position of the current agent in addition to the location of other agents. Also, here m_j is assumed to be q_j and therefore Eq. (16) is simplified as

$$\mathbf{X}_{j,new} = k_a \cdot r_1 \cdot \mathbf{F}_j + k_v \cdot r_2 \cdot \mathbf{V}_{j,old} + \mathbf{X}_{j,old} \tag{27}$$

The pseudo-code of the hybrid algorithm can be summarized as follows:

Step 1: Initialization. The magnitude of the charge for each CP is defined by Eq. (12). The initial positions of the CPs are determined randomly and the initial velocities of charged particles are assumed to be zero.

Step 2: CM creation. The position of the initial agents and the values of their corresponding objective functions are saved in the Charged Memory (**CM**).

Step 3: The forces determination. The probability of moving each CP towards the others (p_{ij}), the kind of forces (a_{ij}) are determined using Eq. (15) and Eq. (26), respectively, and the resultant force vector for each CP is calculated using Eq. (23).

Step 4: Solution construction. Each CP moves to the new position according to Eq. (27).

Step 5: CM updating. CM updating is performed according to Eq. (22).

Step 6: Terminating criterion control. Steps 3-5 are repeated for a predefined number of iterations.

6. Numerical examples

Three examples containing a truss, a frame and a grillage system are optimized utilizing the new hybrid method. These examples are those that have been solved by the CSS and PSO previously and therefore are selected for this study to compare the solutions of other advanced heuristic methods with the new algorithm and to examine the efficiency of this work. For the CSS algorithm, a population of 50 CPs is used for the first two examples and a population of 20 candidates is selected for the last example. k_a and k_v will be different for different populations.

Here, k_v and k_a are defined as (Kaveh and Talatahari 2010b)

$$k_v = c(1 - iter/iter_{\max}), \quad k_a = c(1 + iter/iter_{\max}) \tag{28}$$

where $iter$ is the iteration number; $iter_{\max}$ is the maximum number of the iterations; and c is set to 0.5 and 0.2 when the population of 20 and 50 CPs are selected, respectively.

5.1 A 942-bar spatial truss

A 26-story-tower space truss containing 942 elements and 244 nodes is considered as the truss example. Fifty-nine design variables are used to represent the cross-sectional areas of 59 element groups in this structure, employing the symmetry of the structure. Fig. 1 shows the geometry and the 59 element groups. The detailed information is presented in Kaveh and Talatahari (2010b).

This example has been optimized using 5 meta-heuristic algorithms, previously. The CSS method achieved a good solution after 15,000 analyses and found an optimum weight of 47,371 lb (210,716 N), (Kaveh and Talatahari 2010b). The best weights for the GA, PSO, BB–BC and HBB–BC were

56,343 lb (250,626 N), 60,385 lb (268,606 N), 53,201 lb (236,650 N) and 52,401 lb (233,091 N), respectively (Kaveh and Talatahari, 2009b). The new algorithm can find the best result among others as shown in Table 1. The best result of this hybrid algorithm is equal to 46,310 lb (205,997 N). The new algorithm has better performance in terms of the optimization time, standard deviation

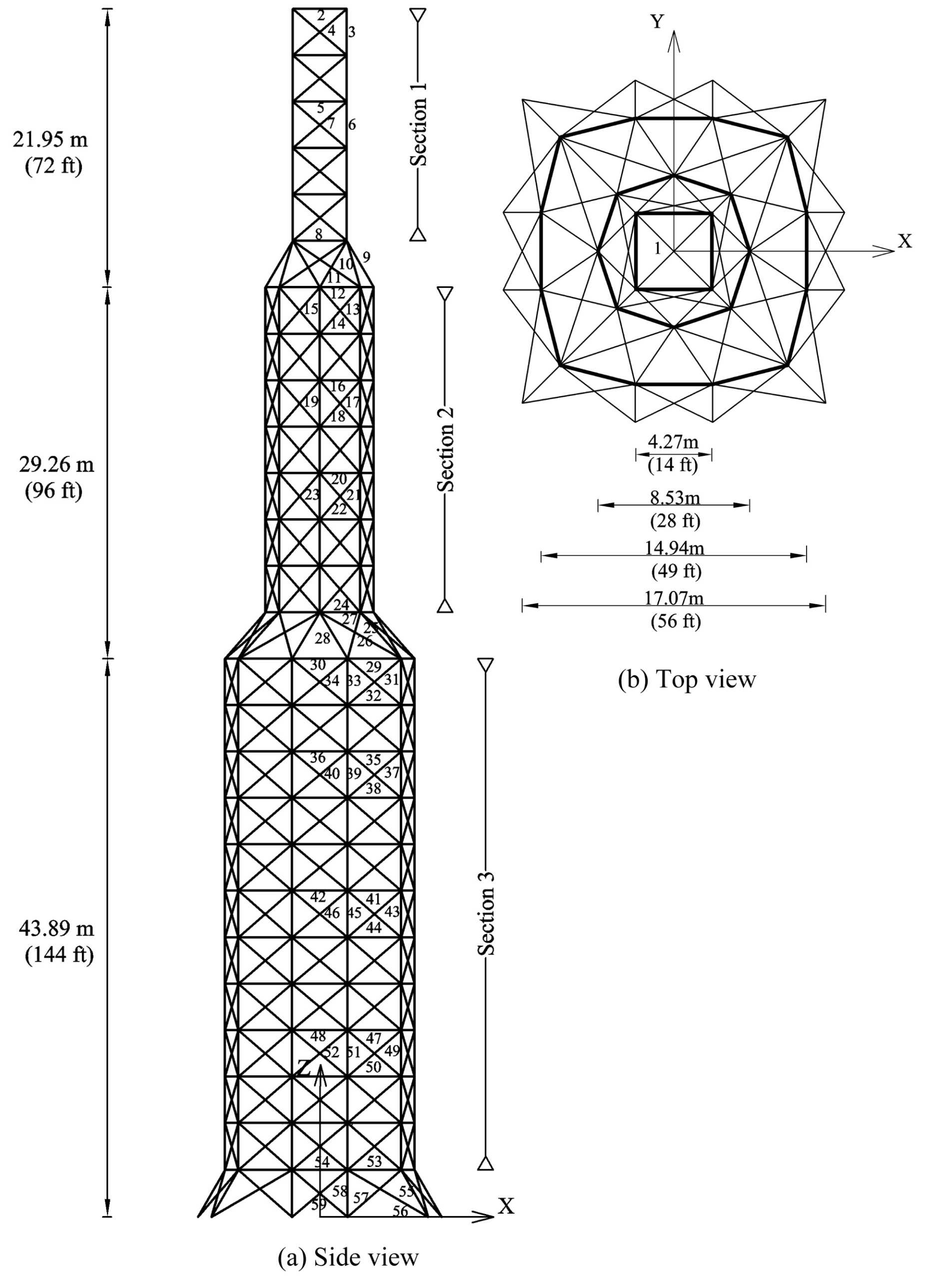


Fig. 1 A 942-bar spatial truss

and the average weight. It converges to a solution after 13,500 analyses of structures in average. Table 2 provides the statistic information for this example and the convergence history is shown in Fig. 2.

Table 1 The optimum design of the hybrid algorithm result for the truss example

Optimal cross-sectional areas (cm^2)								
	Members	Area		Members	Area		Members	Area
1	A_1	1.00129	21	A_{21}	2.517844	41	A_{41}	0.487918
2	A_2	2.172368	22	A_{22}	0.345915	42	A_{42}	0.652938
3	A_3	1.507987	23	A_{23}	3.318067	43	A_{43}	19.57336
4	A_4	0.519582	24	A_{24}	4.811102	44	A_{44}	0.520678
5	A_5	0.652481	25	A_{25}	19.44032	45	A_{45}	1.591111
6	A_6	18.36865	26	A_{26}	0.525286	46	A_{46}	0.481946
7	A_7	0.360254	27	A_{27}	2.653218	47	A_{47}	0.570354
8	A_8	3.002	28	A_{28}	19.54679	48	A_{48}	1.28328
9	A_9	2.234142	29	A_{29}	4.641994	49	A_{49}	19.63833
10	A_{10}	3.664224	30	A_{30}	4.763839	50	A_{50}	0.843581
11	A_{11}	0.784325	31	A_{31}	14.3183	51	A_{51}	4.131012
12	A_{12}	1.077496	32	A_{32}	0.853898	52	A_{52}	0.403386
13	A_{13}	2.860273	33	A_{33}	0.890297	53	A_{53}	11.7018
14	A_{14}	0.493023	34	A_{34}	1.26838	54	A_{54}	18.50203
15	A_{15}	19.24458	35	A_{35}	0.136246	55	A_{55}	18.25599
16	A_{16}	1.325013	36	A_{36}	0.195457	56	A_{56}	3.376078
17	A_{17}	1.980365	37	A_{37}	18.57036	57	A_{57}	2.774705
18	A_{18}	0.529124	38	A_{38}	0.639793	58	A_{58}	4.884796
19	A_{19}	18.10974	39	A_{39}	1.396738	59	A_{59}	0.264933
20	A_{20}	0.323562	40	A_{40}	0.355912			
Weight		205997 N						

Table 2 Performance comparison for the truss example

	GA	PSO	BB–BC	HBB–BC	CSS	CSS+PSO
Best weight (lb)	56343 (250,626 N)	60385 (268606 N)	53201 (236650 N)	52401 (233091 N)	47371 (210716 N)	46310 (205997 N)
Average weight (lb)	63223 (281230 N)	75242 (334693 N)	55206 (245568 N)	53532 (238122 N)	48603 (216197 N)	47953 (213305 N)
Std Dev (lb)	6640.6 (29,539 N)	9906.6 (44,067N)	2621.3 (11660 N)	1420.5 (6318 N)	950.4 (4,227 N)	874.3 (3,889 N)
No. of analyses	50,000	50,000	50,000	30,000	15,000	13,500
Optimization time (sec.)	4,450	3,640	3,162	1,926	1,340	1,190

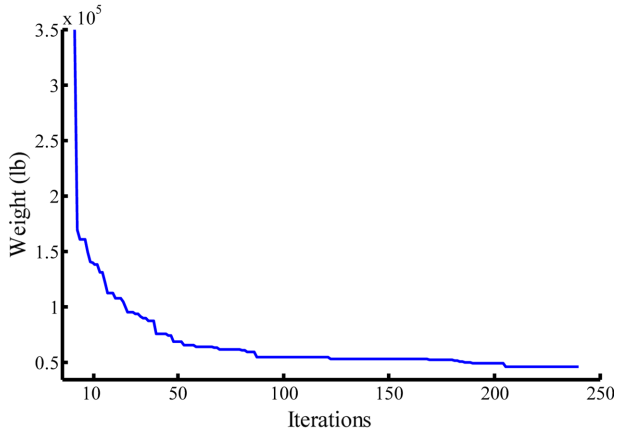
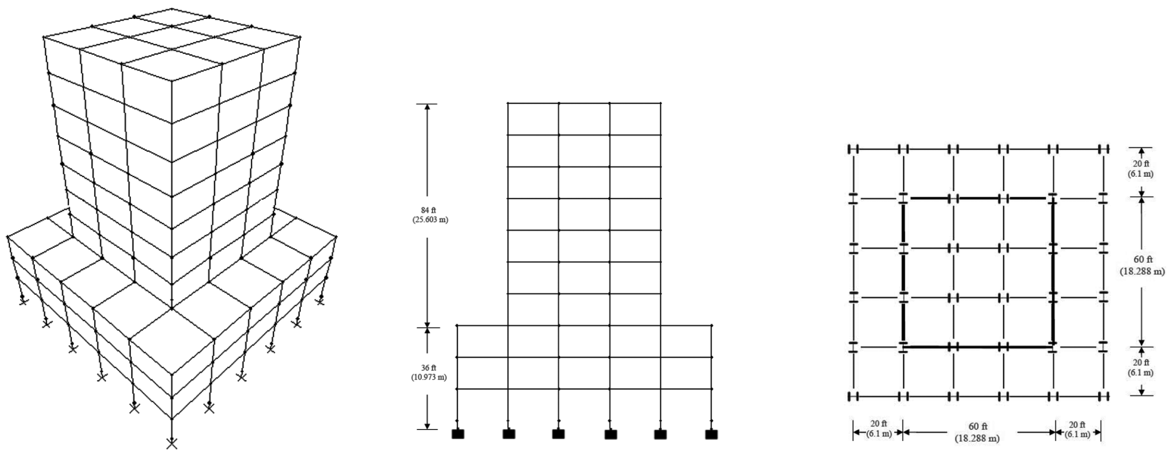


Fig. 2 The convergence history of the truss example for the hybrid algorithm

5.2 A 10-story spatial frame

A 10-story space steel frame consisting of 256 joints and 568 members is considered as shown in Fig. 3. The detailed information about the grouping of elements and loading conditions are presented by Saka and Hasançebi (2009).

The optimum design of this space frame is carried out using the CSS, the simulated annealing (SA), evolution strategies (ESs), particle swarm optimizer (PSO), tabu search optimization (TSO), simple genetic algorithm (SGA), ant colony optimization (ACO), and harmony search (HS) methods. In each optimization technique the number of iterations has been taken as 50,000 for all methods except the CSS algorithm in which 12,500 is sufficient as the maximum number of analyses. The hybrid algorithm could find the optimum solution after 10,800 analyses. The design history of hybrid algorithm is shown in Fig. 4. The optimum design attained by the new hybrid



(a) 3D view (b) Elevation (c) Plan

Fig. 3 A spatial frame example

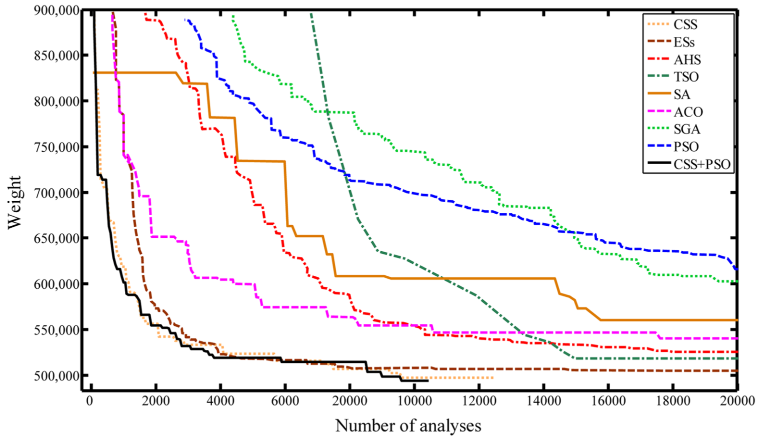


Fig. 4 The convergence history of the frame example

Table 3 Optimal design for the frame example

Element group	Optimal W-shaped sections			
	PSO	ESs	CSS	CSS+PSO
1	W14X159	W14X193	W14X132	W14X145
2	W24X76	W8X48	W21X55	W21X48
3	W10X39	W10X39	W12X40	W18X35
4	W10X22	W10X22	W10X33	W16X31
5	W24X55	W21X50	W21X50	W21X44
6	W12X72	W10X54	W12X65	W18X65
7	W27X146	W14X109	W14X99	W14X99
8	W27X217	W14X176	W14X120	W14X120
9	W18X40	W18X40	W21X44	W21X44
10	W18X40	W18X40	W21X44	W21X44
11	W18X71	W10X49	W14X61	W14X53
12	W12X101	W14X90	W10X88	W18X86
13	W14X176	W14X109	W14X99	W21X101
14	W14X34	W14X30	W18X35	W16X36
15	W21X44	W16X36	W12X50	W18X40
16	W12X65	W12X45	W21X68	W24X62
17	W10X68	W12X65	W16X57	W12X65
18	W12X35	W10X22	W24X68	W8X67
19	W12X79	W12X79	W16X36	W10X33
20	W14X38	W14X30	W16X31	W14X38
21	W10X39	W8X35	W10X33	W14X26
22	W8X31	W10X39	W16X31	W14X38
23	W12X96	W8X31	W8X28	W8X31
24	W12X26	W8X18	W8X18	W8X21
25	W12X26	W14X30	W10X26	W10X26
Weight (kg)	253,260.2	228,588.3	225,654.0	218,971.0

method for this example is 218,971.0 kg, while it is 225,654.0 kg, 228,588.3 kg for the CSS and ESs which are the best ones among the others. The minimum weights as well as W-section designations obtained by the PSO, ESs, CSS and the new algorithm are provided in Table 3. For the present algorithm, maximum stress ratio is equal to 98.39%, and the maximum drift is 0.86 cm, while the allowable value is set to 0.91 cm.

5.3 A 12 m × 12 m grillage system

As the last example, a grillage system with a 12 m × 12 m square area is considered. The system is supposed to carry a 15 kN/m^2 uniformly distributed load (total load is 2,160 kN). The grillage system that can be used to cover the area has the longitudinal beams of length 12 m and the transverse beams of length 12 m. This system is composed of 2m beams as shown in Fig. 5, where the system has 60 members. The total external load is distributed on the joints of the grillage system as point loads.

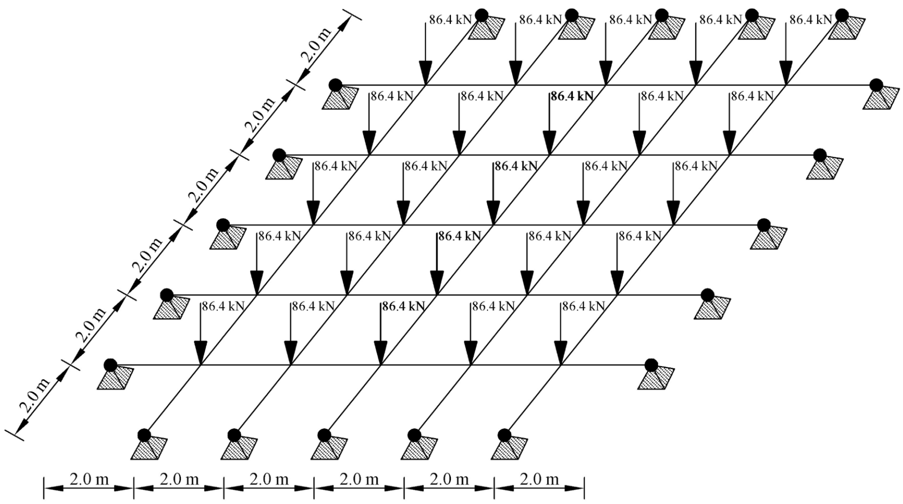


Fig. 5 A 60-elements grillage system

Table 4 Optimal design for the grillage system example

Element group	Optimal W-shaped sections	
	CSS	CSS+PSO
1	W6X9	W10X12
2	W36X135	W36X135
3	W12X14	W8X10
4	W12X22	W14X22
Weight (kg)	9,251	9,211
δ^u (mm)	24.3	23.4
Maximum Strength Ratio	99.0%	99.2%

The vertical displacements of middle joints are restricted to 25 mm. Four group designs are considered in such a way that the outer and inner longitudinal beams belong to group 1 and 2, respectively, while the outer and inner transverse beams are taken as group 3 and 4, respectively. The weight obtained by the new algorithm is 9,211 kg while it has been 9,251 kg for the CSS method (Kaveh and Talatahari 2010c). The optimum results obtained by the hybrid algorithm are summarized in Table 4. The number of required structural analyses for this example was equal to 2,560 which is less than 3,000 analyses required for the CSS.

6. Conclusions

A hybrid algorithm is developed by adding positive characters of the particle swarm optimization into the charged system search algorithm. The first change is to redefine the charged memory (**CM**) in a way that it is treated as the local best in the PSO, and the size of **CM** is taken as the number of agents. The **CM** updating process is performed when the new position of an agent is better than the so far stored good position of this agent. The second modification is the utilization of a new **CM** in determining the affected forces. In this way, not only the global and local best agents from the **CM** but also some other stored points are utilized. In addition, some of the locations of the current agents are also employed to determine the resultant forces. Utilizing the **CM** increases the exploitation of the algorithm while using the current agents enlarges the exploration ability of the algorithm. As a result, for the present algorithm the number of agents utilized from **CM** is increased gradually and the number of used current agents is reduced, simultaneously. As the third difference between the hybrid algorithm and the original CSS, one should mention the utilization of the repulsive forces in addition to attractive ones in the algorithm.

The new algorithm has high power in the searching level because of high exploration ability due to utilizing current agents distributed all over the search space in the primary iterations. In addition using the best so far CPs as the one point existing in the PSO increases the exploitation ability of the algorithm. Finally putting together of these properties improve the balance between exploration and exploitation of the resulted method.

In order to investigate the efficiency of the new algorithm, three different skeletal structures are optimized. The results are compared to those of some advanced meta-heuristic algorithm especially to the PSO and CSS algorithms. For all three examples, the present algorithm can find better designs in most cases with smaller computational costs. This demonstrates that the presented new algorithm is a powerful optimization method which can easily be employed in optimum design of structures.

Acknowledgements

The first author is grateful to the Iran National Science Foundation for the support.

References

Angeline, P. (1998), "Evolutionary optimization versus particle swarm optimization: philosophy and performance

difference", *Proceedings of the Evolutionary Programming Conference,* San Diego, USA.
American Institute of Steel Construction (AISC) (1989), *Manual of Steel Construction-allowable Stress Design,* 9th Ed., Chicago, IL.
Coello, C.A.C. (2002), "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art", *Comput. Meth. Appl. Mech. Eng.*, **191**(11-12), 245-287.
He, S., Wu, Q.H., Wen, J.Y., Saunders, J.R. and Paton, R.C. (2004), "A particle swarm optimizer with passive congregation", *Biosystem*, **78**, 135-147.
Kaveh, A. and Talatahari, S. (2009a), "Particle Swarm Optimizer, Ant colony strategy and harmony search scheme hybridized for optimization of truss structures", *Comput. Struct.*, **87**(5-6), 267-283.
Kaveh, A. and Talatahari, S. (2009b), "Size optimization of space trusses using Big Bang–Big Crunch algorithm", *Comput. Struct.*, **87**(17-18), 1129-1140.
Kaveh, A. and Talatahari, S. (2009c), "A particle swarm ant colony optimization algorithm for truss structures with discrete variables", *J. Constr. Steel Res.*, **65**(8-9), 1558-1568.
Kaveh, A. and Talatahari, S. (2010a), "A novel heuristic optimization method: charged system search", *Acta Mech.*, **213**(3-4), 267-289.
Kaveh, A. and Talatahari, S. (2010b), "Optimal design of skeletal structures via the charged system search algorithm", *Struct. Multidiscip. O.*, **41**(6), 893-911.
Kaveh, A. and Talatahari, S. (2010c), "Charged system search for optimum grillage systems design using the LRFD-AISC code", *J. Constr. Steel Res.*, **66**(6), 767-771.
Kennedy, J., Eberhart, R.C. and Shi, Y. (2001), *Swarm Intelligence*, Morgan Kaufman Publishers, San Francisco.
LRFD-AISC, Manual of Steel Construction, Load and Resistance Factor Design (1999), Metric Conversion of the Second Edition, Vol. 1, 2. AISC, Chicago.
Saka, M.P. and Hasançebi, O. (2009) "Design code optimization of steel structures using adaptive harmony search algorithm", *Stud. Comput. Intel.*, **239**, 79-120.
Smith, S. (1998), "The simplex method and evolutionary algorithms", *Proceedings of the IEEE International Conference on Evolutionary Computation*, 799-804.
Xu, J.J. and Xin, Z.H. (2005), "An extended particle swarm optimizer", *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 6.*