ASIAN JOURNAL OF CIVIL ENGINEERING (BUILDING AND HOUSING) VOL. 9, NO. 6 (2007)
PAGES 563-575

A DISCRETE PARTICLE SWARM ANT COLONY OPTIMIZATION FOR DESIGN OF STEEL FRAMES

A. Kaveh*[a] and S. Talatahari[b]
[a]*Department of Civil Engineering, Iran University of Science and Technology, Tehran-16, Iran*
[b]*Department of Civil Engineering, University of Tabriz, Tabriz, Iran*

Abstract

A discrete version of Particle Swarm Ant Colony Optimization (DPSACO) for design of frame structures is presented. DPSACO, similar to continous PSACO, is based on a particle swarm optimizer, worked as a global search, and ant colony optimization employed as a local search. In the DPSACO, the nearest permitted discrete value is replaced with any value selected by agents (Particles or ants). Therefore, the positions of all agents always contain the permitted discrete values. In order to improve the exploration of the proposed method, a new formula for particles' velocity is defined. Two design examples are tested using the new method and their results are compared with the results of other PSO-based algorithms to demonstrate the effectiveness of the presented method.

Keywords: Ant colony; discrete optimum; frame design; optimization; particle swarm

1. Introduction

Particle Swarm Optimization (PSO) is a stochastic optimization method capable of handling non differentiable, nonlinear, and multi module objective functions. The PSO approach is motivated from the social behavior of bird flocking and fish schooling [1]. PSO has a population of individuals that move through search space and each individual has a velocity that acts as an operator to obtain a new set of individuals. Individuals, called particles, adjust their movements depending on both their own experience and the population's experience. At each iteration, a particle moves towards a direction computed from the best visited position and the best visited position of all particles in its neighborhood. In this approach, except the particle that is the best experience of particles, the effect of other particles is ignored. So the probability of becoming trapped in the local points is increased [2]. Recently, authors have presented the Particle Swarm Ant Colony Optimization approach (PSACO) [3] to avoid this problem. In PSACO, the PSOPC algorithm (a hybrid PSO with passive congregation [4]) is combined with the ant colony algorithm. The PSACO applies

*E-mails address of the corresponding author: alikaveh@iust.ac.ir (A. Kaveh)

the PSOPC for global optimization and ant colony approach is employed as a local search, wherein, ants apply pheromone-guided mechanism to update the positions found by the particles in the earlier stage.

The PSACO, similar to PSO and PSOPC, is a continuous approach while, in practical structural optimization problems, industrial cross sections are used which have discrete values and as a result a discrete solution is better than continuous one for this kind of the optimization problems [5]. This paper presents a discrete version of PSACO (DPSACO) for design of frame structures. In the proposed method, particles (or ants) are allowed to select discrete values from the permissible list of cross sections, and if any one of particles (ants) selects another value for a design variable, the DPSACO changes the amount of it with the value of the nearest discrete cross section. In addition, in this paper the formula of particles' velocity is changed to improve the performance of the proposed method. Two design examples are tested using the new method and their results are compared with the results of PSO, PSOPC and primary PSACO. The remaining sections of this paper are organized as follows:

The problem formulation is given in Section 2. Section 3 includes a brief review of the PSACO algorithm. Section 4 describes a discrete PSACO algorithm, and Section 5 contains two illustrative examples. Section 6 includes the concluding remarks.

2. Discrete Optimum Design of Steel Frames

Optimal design of frame structures can be formulated as

$$\begin{aligned} &Find \qquad X = [x_1, x_2, ..., x_{ng}], \\ &\qquad\qquad x_i \in D_i, D_i = \{d_{i,1}, d_{i,2}, ..., d_{i,r(i)}\} \\ &to\ \ minimize \quad f(X) = \sum_{i=1}^{nm} \gamma_i \cdot x_i \cdot L_i \end{aligned} \tag{1}$$

subjected to the following constraints [6]:

Stress constraints

$$\left| \frac{\sigma_i}{\sigma_i^a} \right| \le 1, \qquad i = 1,2,...,nm \tag{2}$$

Maximum lateral displacement

$$\frac{\Delta_T}{H} \le R \tag{3}$$

Inter-story displacement constraints

$$\frac{\Delta_j}{h_j} \le R_I, \qquad j = 1,2,...,ns \tag{4}$$

where X is the vector containing the design variables; D_i is an allowable set of discrete values (a set of 267 W-sections from the AISC database in this paper) for the design variable x_i; ng is the number of design variables or the number of groups; $r(i)$ is the number of available discrete values for the ith design variable; $f(X)$ is the cost function which is taken as the weight of the structure; γ_i is the material density of member i; L_i is the length of member i; σ_i is the stress in member i; σ_i^a is the allowable stress in member i; nm is the number of members making up the frame; Δ_T is the maximum lateral displacement; H is the height of the frame structure; R is the maximum drift index; Δ_j is the inter-story drift; h_j is the story height of the jth floor; ns is the total number of stories; and R_I is the inter-story drift index permitted by the code of the practice.

If the code of the practice is selected AISC 2001 [7], the allowed inter-story drift index is 1/300 and the LRFD interaction formula constraints (AISC 2001, Equation H1-1a,b) is defined as

$$\frac{P_u}{2\phi_c P_n} + \left(\frac{M_{ux}}{\phi_b M_{nx}} + \frac{M_{uy}}{\phi_b M_{ny}} \right) \le 1 \quad For \quad \frac{P_u}{\phi_c P_n} < 0.2 \tag{5}$$

$$\frac{P_u}{2\phi_c P_n} + \frac{8}{9}\left(\frac{M_{ux}}{\phi_b M_{nx}} + \frac{M_{uy}}{\phi_b M_{ny}} \right) \le 1 \quad For \quad \frac{P_u}{\phi_c P_n} \ge 0.2 \tag{6}$$

where P_u is the required strength (tension or compression); P_n is the nominal axial strength (tension or compression); ϕ_c is the resistance factor (ϕ_c =0.9 for tension, ϕ_c =0.85 for compression); M_{ux} and M_{uy} are the required flexural strengths in the x and y directions, respectively; M_{ux} and M_{ny} are the nominal flexural strengths in the x and y directions (for two-dimensional structures, $M_{ny} = 0$); and ϕ_b is the flexural resistance reduction factor (ϕ_b=0.90).

In this paper, the constraints are handled by using a fly-back mechanism. Compared with other constraint-handling techniques, fly-back mechanism is relatively simple and easy to implement. In this method, the particles are initialized in the feasible region. When the particles fly in the search space, if any one of them flies into the infeasible region, it will be forced to fly back to the previous position to guarantee a feasible solution [8].

3. Review of Particle Swarm Ant Colony Optimization

The implementation of PSACO algorithm consists of two stages. In the first stage, PSOPC is applied, and ACO is implemented in the second stage [3].

First stage involves a number of particles, which are initialized randomly in the feasible space. These particles fly through the search space and their positions are updated based on the best positions of individual particles, the best position among all particles in the search space

[9], and the position of a particle selected randomly from the swarm [4] in each iteration.

The update moves a particle by adding a change velocity V_i^{k+1} to the current position X_i^k as follows:

$$X_i^{k+1} = X_i^k + V_i^{k+1} \tag{7}$$

$$V_i^{k+1} = \omega V_i^k + c_1 r_1 (P_i^k - X_i^k) + c_2 r_2 (P_g^k - X_i^k) + c_3 r_3 (R_i^k - X_i^k) \tag{8}$$

where ω is an inertia weight to control the influence of the previous velocity; r_1, r_2 and r_3 are three random numbers uniformly distributed in the range of (0, 1); c_1 and c_2 are two acceleration constants; c_3 is the passive congregation coefficient; P_i^k is the best position of the ith particle up to iteration k; P_g^k is the best position among all particles in the swarm up to iteration k; and R_i is a particle selected randomly from the swarm [10].

In the second stage, ACO works as a local search, wherein, ants apply a pheromone-guided mechanism to refine the positions found by particles in the PSOPC stage. The ACO algorithm handles P ants equal to the number of particles in the PSOPC stage, and each ant generates a solution around P_g^k which can be written as

$$Z_i^k = N(P_g^k, \sigma) \tag{9}$$

where, Z_i^k is the solution constructed by ant i in the stage k; $N(P_g^k, \sigma)$ denotes a random number normally distributed with mean value P_g^k and variance σ, where

$$\sigma = (A_{max} - A_{min}) \times \eta \tag{10}$$

Where η is the step size; $A_{\min}$=2.51 in^2 (W6×8.5); and $A_{\max}$=249 in^2 (W36×848).

Then, objective function value for each ant, $f(Z_i^k)$, is computed and the current position of ant i, Z_i^k, is replaced with the position X_i^k, the current position of particle i in the swarm, if $f(X_i^k) > f(Z_i^k)$ and current ant is in the feasible space.

The above process continues, until the absolute value of every component of the velocity vector is greater than the exactitude of the solutions, shown as A^*. This can be written as

$$\text{Terminatin criterion: } max(|V_{ij}^k|) < A^* \begin{cases} \forall i = 1,2,\ldots,P \\ \forall j = 1,2,\ldots,ng \end{cases} \tag{11}$$

By using this criterion, the extra iterations are eliminated and optimum solution is reached earlier [3].

4. A Discrete Particle Swarm Ant Colony Optimization

In 1997, Kennedy and Eberhart [11] proposed a kind of discrete particle swarm optimization algorithm (DPSO) on the basis of the primary continuous PSO algorithm. In DPSO, the movement of the particle is realized by flip of bit value, the position of any particle is expressed as a binary bit vector composed of 0 and 1. The velocity of the particle is no longer a change ratio of its position but a change probability of its position. That is to say, the velocity of any particle is a probability in which its position is 1 or 0, the bigger the velocity, the bigger the probability in which its position is 1.

Many discrete PSO algorithms in literature are based on DPSO or relatively utilized DPSO principles. Instead, this paper presents a new discrete PSO-based algorithm, called discrete PSACO (DPSACO). The framework of DPSACO algorithm is illustrated in Figure 1. In the DPSACO, new position of each agent is defined as following:

For particles

$$X_i^{k+1} = Fix(X_i^k + V_i^{k+1}) \tag{12}$$

For ants

$$Z_i^k = Fix\left(N(P_g^k, \sigma)\right) \tag{13}$$

where $Fix(X)$ is a function which rounds each elements of X to the nearest permissible discrete value. Using this position updating formula, the agents will be permitted to select discrete values. Although this change is simple and efficient, it may be reduce exploration (global investigation of the search space) of the algorithm. Therefore, in this paper, the velocity of particles is redefined as following to increase the exploration:

$$V_i^{k+1} = \omega V_i^k + c_1 r_1 (P_i^k - X_i^k) + c_2 r_2 (P_g^k - X_i^k) + c_3 r_3 (R_i^k - X_i^k) + c_4 r_4 (Rd_i^k - X_i^k) \tag{14}$$

where c_4 is the exploration coefficient; r_4 is a uniformly distributed random number in the range of (0, 1); and Rd^k is a vector generated randomly from the search domain.

For DPSACO, A^* (in Eq. (11)) is defined as

$$A^{*(i)} = \begin{cases} 0.05 & P_g^{k(i)} \le 9.12 \\ 0.1 & 9.12 < P_g^{k(i)} \le 98.8 \\ 1.0 & P_g^{k(i)} > 98.8 \end{cases} \tag{15}$$

where $A^{*(i)}$ is the exactitude of the ith design variable; $P_g^{k(i)}$ is the ith element of the P_g^k.

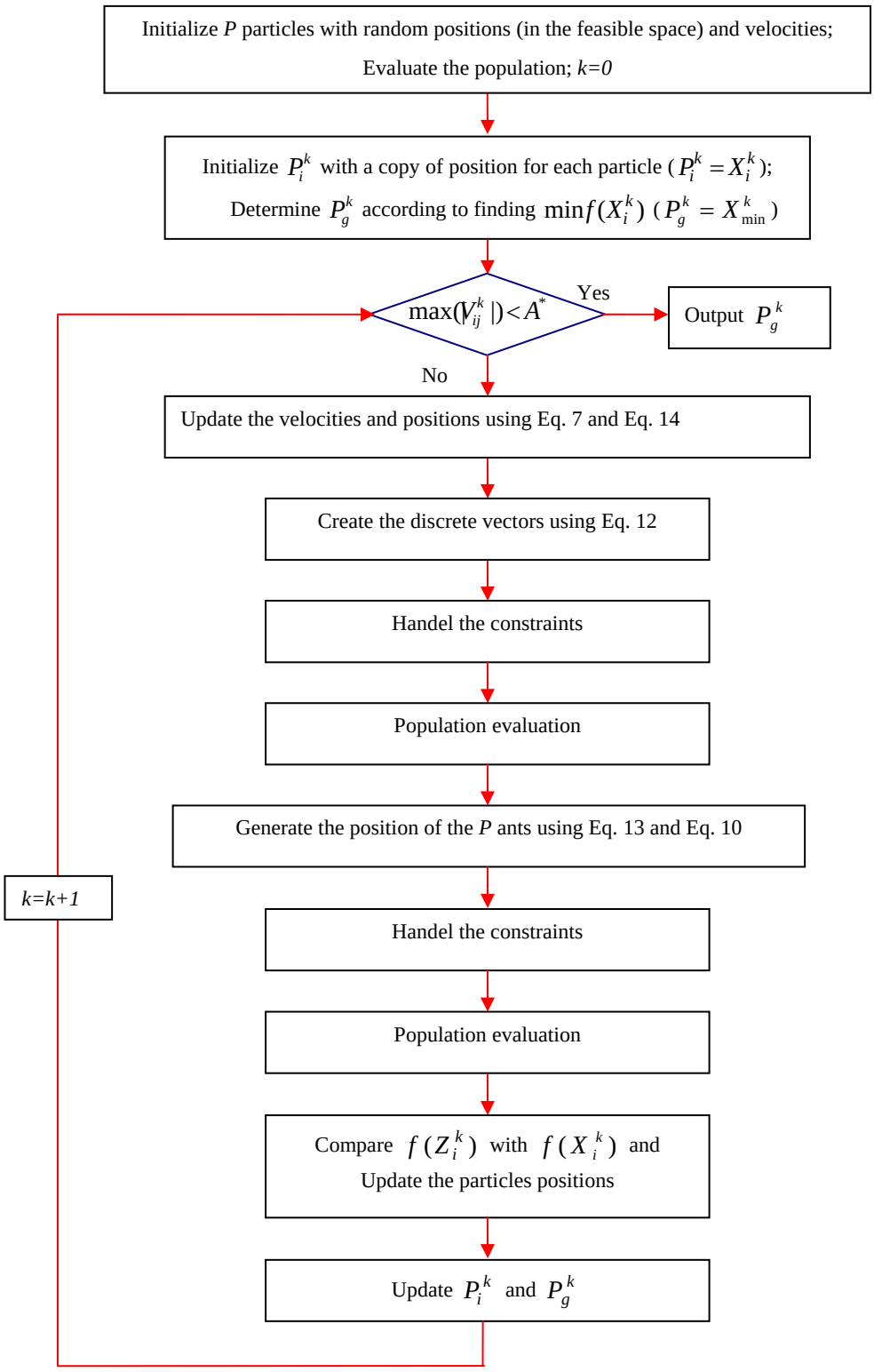

Figure 1. The flow chart for DPSACO

5. Design Examples

In this section, two frame structures are optimized with the proposed method. The LRFD specification (AISC 2001) and inter-story drift are considered as constraints for the second example but only lateral drift at the top of the structure is considered as the constraint for the first example. To demonstrate the effectiveness of this research, the final results of the DPSACO algorithm (using Eqs. 10, 12-14) are compared with solutions of other PSO-based algorithms containing: simple PSO, PSOPC (using Eqs. 7, 8), PSOPC+ACO (using Eqs. 7-10). In addition, the result of the first example is compared with the result of a genetic algorithm and ACO from the literature.

The steel members used for the design exercises are 267 W-shaped sections from the AISC database. A population of 50 individuals is used for both particles and ants; the value of constants c_1 and c_2 are set to 0.8, the passive congregation coefficient c_3 and the exploration coefficient c_4 are given 0.6 and 0.1, respectively. η is set to 0.01.

5.1 One-bay eight-story steel frame

Figure 2 shows the configuration and applied loads of one-bay eight-story framed structure. The 24 members of the structure have been categorized into eight groups, as indicated in the Figure. The lateral drift at the top of the structure is the only performance constraint (no more than 2 in.). The modulus of elasticity is taken as E=200 GPa (29000 ksi).

The DPSACO algorithm found the optimal weight of the one-bay eight-story frame to be 30.91 kN (6.95 kips). Authors [6] used an improved ACO (IACO) to design this frame resulting in a weight 31.05 kN (6.98 kips). Also, Kaveh and Shojaee [12] obtained a frame with 31.68 kN (7.12 kips) weight using ACO. Camp *et al.* [13] achieved a frame with 32.83 kN (7.38 kips) weight by utilizing a genetic algorithm. Table 6 lists the optimal values of the eight design variables obtained by this research, and compares them with other results.

Figure 3 compares the convergence rate of the various PSO-based algorithms and DPSACO. It takes about 2640 and 2350 iterations for the PSOPC and the PSO algorithms to converge, respectively. However the DPSACO and PSOPC+ACO algorithms take 627 and 588 iterations to converge, respectively. Because of doing more explorations, the number of required iterations for reaching a solution by DPSACO is greater than PSOPC+ACO. Instead, the last result of DPSACO (30.91 kN) is better than the result of PSOPC+ACO (32.29 kN).

5.2 Three-bay ten-story steel frame

A 10-storey frame, shown in Figure 4, was first analyzed by Saka and Kameshki [14] under displacement and AISC combined strength constraints. The dimensions of the frame, the applied loading system, and the grouping of the members, are shown in the figure. The sway of the top storey was limited to 4.66 in (118.3 mm). The material has a modulus of elasticity E=200 GPa (29000 ksi) and a yield stress of f_y=248.2 MPa (36 ksi).

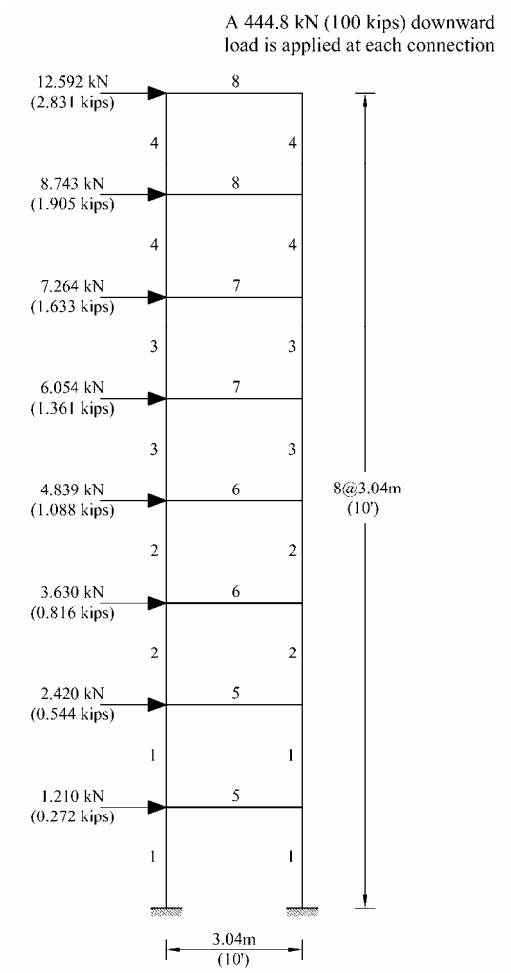

Figure 2. Topology of the one-bay eight-story frame

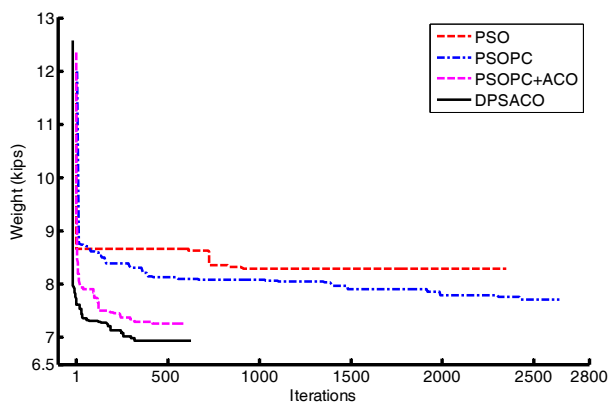

Figure 3. Comparison of the convergence rates between various PSO-based algorithms for the one-bay eight-story frame

Table l. Optimal design comparison for the one-bay eight-story frame

	AISC W-shapes						
Element group No.	**Camp *et al.* [13] GA**	**Kaveh and Shojaee [12] ACO**	**Kaveh and Talatahari [6] IACO**	**PSO**	**PSOPC**	**PSOPC+ACO**	**DPSACO**
1	W18×46	W21×50	W18×40	W14×34	W18×35	W18×35	W18×35
2	W16×31	W16×26	W16×26	W16×26	W14×26	W16×31	W16×31
3	W16×26	W16×26	W16×26	W16×26	W16×26	W14×22	W16×26
4	W12×16	W12×14	W12×14	W12×26	W14×26	W12×16	W14×22
5	W18×35	W16×26	W21×44	W21×44	W24×62	W21×48	W16×31
6	W18×35	W18×40	W18×35	W14×34	W18×35	W18×40	W18×40
7	W18×35	W18×35	W18×35	W21×83	W16×31	W16×31	W16×26
8	W18×26	W14×22	W12×22	W12×30	W12×30	W16×36	W14×22
Weight kN (kips)	32.83 (7.38)	31.68 (7.12)	31.05 (6.98)	36.89 (8.29)	34.21 (7.69)	32.29 (7.26)	30.91 (6.95)

Note: GA= Genetic Algorithm; IACO= Improved Ant Colony Optimization

The effective length factors of the members are calculated as $K_x \geq 0$ for a sway-permitted frame and the out-of-plane effective length factor is specified as $K_y = 1.0$. Each column is considered unbraced along its length, and the unbraced length for each beam member is specified as one-fifth of the span length.

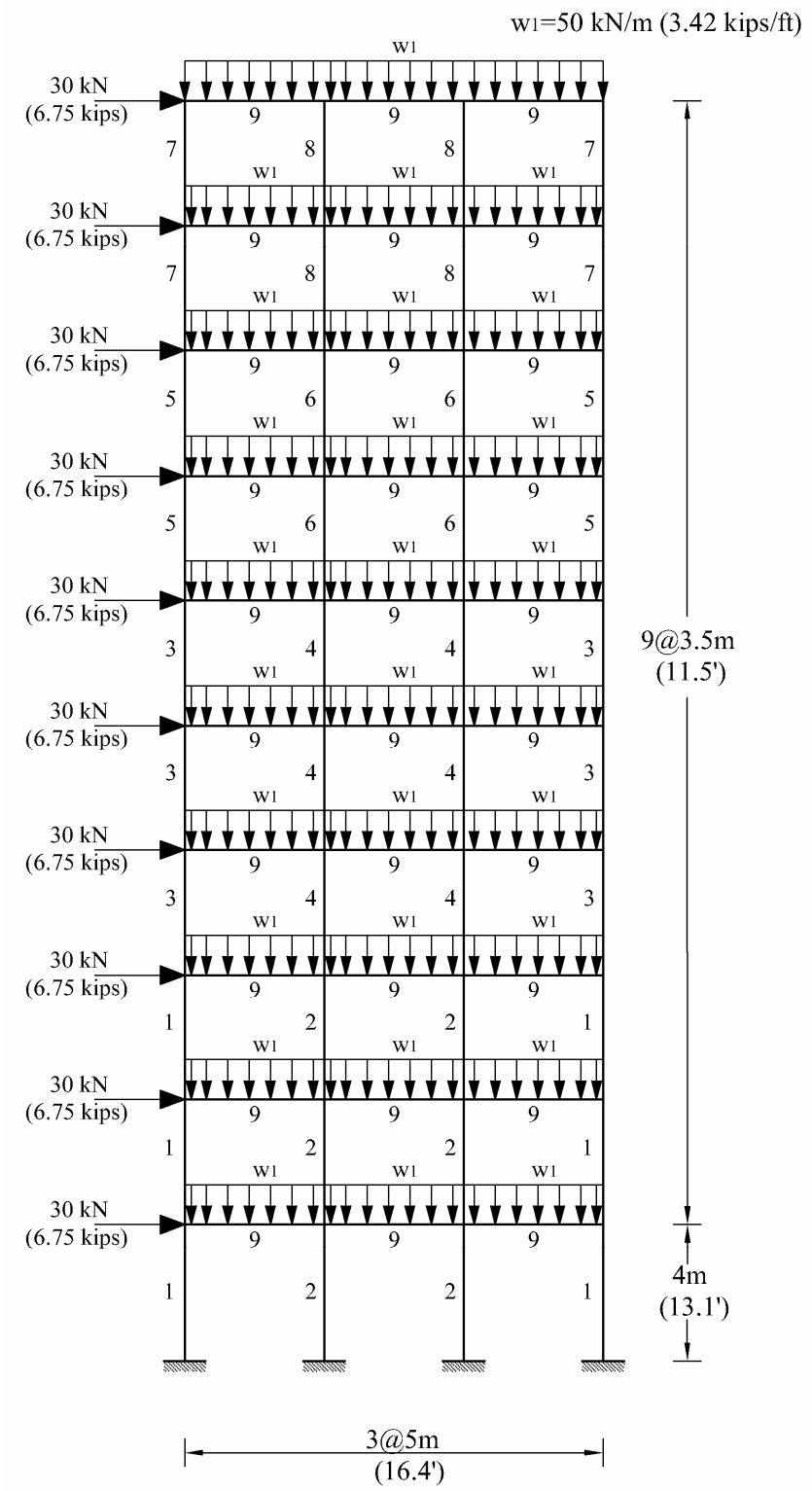

Figure 4. Topology of the three-bay ten-story frame

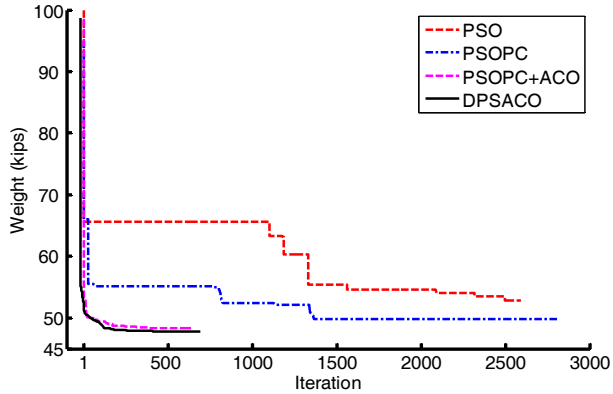

Figure 5. Comparison of the convergence rates between various PSO-based algorithms for the three-bay ten-story frame

The optimum design of the frame was obtained after 692 iterations by using DPSACO, which had the minimum weight of 211.41 kN (47.53 kips). The optimum designs for PSOPC+ACO, PSOPC and simple PSO had the weight of 215.19 kN (48.38 kips), 221.69 kN (49.84 kips) and 235.08 kN (52.85 kips), respectively. Table 2 summarizes the optimal results and the number of the required iterations for the various algorithms. Figure 5 shows the convergence history for the various PSO-based algorithms.

Table 2. Optimal design comparison for the three-bay ten-story frame

Element group No.	**AISC W-shapes**			
	PSO	**PSOPC**	**PSOPC+ACO**	**DPSACO**
1	W30×99	W14×74	W30×99	W12×72
2	W12×96	W14×90	W12×87	W21×101
3	W14×82	W10×49	W14×43	W21×48
4	W18×76	W12×72	W21×83	W24×68
5	W8×31	W14×43	W10×39	W14×43
6	W14×53	W18×65	W14×53	W18×65
7	W8×28	W16×36	W8×28	W8×28
8	W8×24	W12×30	W8×24	W8×24
9	W21×44	W21×44	W18×40	W18×40
Weight kN (kips)	235.08 (52.85)	221.69 (49.84)	215.19 (48.38)	211.41 (47.53)
Number of iterations	2590	2820	635	679

6. Concluding Remarks

Particle Swarm Optimization is a stochastic optimization method capable of handling various objective functions. Although the PSO is a simple and effective method, the probability of becoming trapped in the local optimums is high. Particle Swarm Ant Colony Optimization approach (PSACO) prevents this problem by using the PSOPC algorithm for global optimization and ant colony approach as a local search.

This paper presents a discrete version of PSACO (DPSACO) for design of frame structures. In this method, particles (or ants) are allowed to select discrete values from the permissible list of cross sections. A new formula for the velocity of particles is also defined to improve the performance of the presented method. The algorithm is tested on two frames. The comparison of results with those of PSO-based optimization algorithms, prove the robustness of the proposed method in optimizing frame structures.

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

References

1. Eberhart RC, Kennedy J. A new optimizer using particle swarm theory, *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, 1995.
2. Angeline P. Evolutionary optimization versus particle swarm optimization: philosophy and performance difference, *Proceedings of the Evolutionary Programming Conference*, San Diego, USA, 1998.
3. Kaveh A, Talatahari S. A hybrid particle swarm and ant colony optimization for design of truss structures, *Asian Journal of Civil Engineering*, No. 4, **9**(2008)329-48.
4. He S, Wu, QH, Wen JY, Saunders JR, Paton RC. A particle swarm optimizer with passive congregation, *Biosystem*, **78**(2004)135-47.
5. Kaveh A, Farhmand Azar B, Talatahari S. Ant colony optimization for design of space trusses, *Inernational Journal of Space Structures*, No. 3, **23**(2008).
6. Kaveh A, Talatahari S. An improved ant colony optimization for design of steel frames, submitted for publication, 2008.
7. American Institute of Steel Construction (AISC) Manual of Steel Construction-Load Resistance Factor Design, 3rd ed., AISC, Chicago, 2001.
8. He S, Prempain E, Wu QH. An improved particle swarm optimizer for mechanical design optimization problems, *Engineering Optimization*, No. 5, **36**(2004)585–605.
9. Kennedy J, Eberhart RC. Swarm Intelligence. Morgan Kaufman Publishers, 2001.
10. Shi Y, Eberhart RC. A modified particle swarm optimizer. In: *Proceeding of IEEE Inc. Conference on Evolutionary Computation*, 1997, pp. 303-8.
11. Kennedy J, Eberhart RC. A discrete binary version of the particle swarm optimisation algorithm, *in: Proceedings of the IEEE International Conference on Neural Networks*, Perth, Australia, 1997, pp. 4104-8.

12. Kaveh A, Shojaee S. Optimal design of skeletal structures using ant colony optimisation, *International Journal for Numerical Methods in Engineering*, No. 5, **70**(2007)563-81.
13. Camp CV, Pezeshk S, Cao G. Optimized design of two dimensional structures using a genetic algorithm, *Journl of Structural Engineering, ASCE*, No. 5, **124**(1998)551-9.
14. Saka MP, Kameshki ES. Optimum design of unbraced rigid frames, *Computers and Structures*, **69**(1998)433-42.