

## IMPROVED PSO ALGORITHM FOR SHAPE AND SIZING OPTIMIZATION OF TRUSS STRUCTURE

Yancang LI, Yang PENG, Shujing ZHOU

*College of Civil Engineering, Hebei University of Engineering, Handan, 056038, China*

Received 11 Jan. 2013; accepted 25 Feb. 2013

**Abstract.** In order to overcome the premature convergence defect of the basic particle swarm optimization (PSO) algorithm and provide an effective method for shape and sizing optimization of truss structure, an improved PSO was proposed. The random direction method was employed to produce high-quality initial population, the fuzzy system was applied in the dynamic adaptive adjustment of parameters of the PSO, and the Metropolis criteria were used to improve the performance of PSO. Then, the improved PSO was introduced to the truss structure shape and sizing optimization design. Engineering practice and comparison with the other optimization algorithms show that the algorithm has good convergence and global searching capability. The study provides a promising algorithm for the structural optimization.

**Keywords:** structural optimization; improved particle swarm optimization algorithm; truss structure; shape optimization; sizing optimization.

**Reference** to this paper should be made as follows: Li, Y.; Peng, Y.; Zhou, S. 2013. Improved PSO algorithm for shape and sizing optimization of truss structure, *Journal of Civil Engineering and Management* 19(4): 542–549. <http://dx.doi.org/10.3846/13923730.2013.786754>

### Introduction

The engineering structure optimization can improve the design quality, shorten the design cycle, and cut down the engineering cost (Cai *et al.* 2011; Gandomi, Yang 2011). The traditional structure optimization algorithms are mainly classified into the mathematical programming (MP) algorithms and the mechanics criterion (MC) algorithms. The former requires the complex analysis, and for multi-variable optimization problems, it needs large amount of computation, which limits its use in large complex engineering structure design optimization. Though the MC has clear concepts, it is not stable because it easily falls into premature convergence with the increase of iterations (Huang, Meng 2009; Šešok, Belevičius 2008; Wang *et al.* 2002b).

How to solve the problem and provide novel methods for the structures optimization are urgent questions for researchers and engineers. Up to date, lots of works have been done on this, by the attention and efforts of researchers in corresponding fields, many algorithms and models were proposed (Hadidi *et al.* 2011; Hagishita, Ohsaki 2009; Kaveh, Talatahari 2010, 2011, 2012; Kaveh *et al.* 2008; Sonmez 2011; Takada 2012; Talatahari *et al.* 2012; Wang, Ohmori

2012; Xia *et al.* 2013; Zhou, Gao 2012). In some degree, all these studies give rise to the structural optimization, but we still have a long way to go. This has signified the need for the related study.

According to this, a new method – the improved PSO algorithm – was introduced to solve the problem. As a population-based stochastic approach for solving continuous and discrete optimization problems, especially the complicated combinational optimization problems, the particle swarm optimization (PSO) algorithm, proposed by Reeves (1983), is a new swarm intelligence technique, inspired by the social behavior of bird flocking or fish schooling. In PSO, simple software agents, called particles, move in the search space of an optimization problem. The position of a particle represents a candidate solution to the optimization problem at hand. Each particle searches for better positions in the search space by changing its velocity according to rules originally inspired by behavioral models of bird flocking.

Within the field of computer graphics, the first antecedents of PSO can be traced back to the work of Reeves (1983), who proposed particle systems to model objects that are dynamic and cannot be easily

represented by polygons or surfaces. Examples of such objects are fire, smoke, water, and clouds. In these systems, particles are independent of each other and their movements are governed by a set of rules. Then, Reynolds (1987) used a particle system to simulate the collective behavior of a flock of birds. In a similar kind of simulation, Heppner and Grenander (1990) included a roost that was attractive to the simulated birds. Both models inspired the set of rules that were later used in the original PSO algorithm. Then, PSO was introduced by Kennedy and Eberhart (1995). They pointed out that the rules which govern the movement of the particles in a problem's search space can also be seen as a model of human social behavior in which individuals adjust their beliefs and attitudes to conform with those of their peers. It has roots in the simulation of social behaviors, in particular the dynamic theory of social impact (Kennedy 2006; Nowak *et al.* 1990), using tools and ideas taken from computer graphics and social psychology research.

The PSO algorithm is simple and easy to understand, and it has the higher search efficiency with occupying fewer programming resources. With these advantages, PSO finds a number of applications in the areas of optimization, including the unconstrained, single-objective optimization problems and constrained problems, multi-objective optimization problems, and problems with dynamically changing landscapes. To date, there have been hundreds of publications reporting applications of PSO algorithms (Poli 2008; Zhang, Zhan 2009). But, like other population-based stochastic approaches, the biggest bottleneck of limiting its application is the premature convergence. By the attention and efforts of researchers in corresponding fields, PSO algorithm is improved and expanded based on the initial model (Bu *et al.* 2010; Li *et al.* 2009; Reng, Li 2008; Tang *et al.* 2009; Zhou *et al.* 2011). Although the improved PSO algorithm's performance is improved comparing with the basic algorithm, there still exists the drawback of easily to premature convergence (Shi, Eberhart 2001; Wang *et al.* 2002a; Xie *et al.* 2011). Here, we proposed an improved PSO based on chaos theory and metropolis criteria, and introduced it to the optimization of the shape of space truss. Engineering practice and comparison with the other algorithms showed its efficiency.

The rest of the paper is organized as follows. First, the attention was paid to the modification of the algorithm after the brief introduction of the basic knowledge of PSO. In the following part, application study and performance comparison with other algorithms on the structural optimization were introduced

and the advantage of the improved algorithm was analyzed.

## 1. Improvement of basic PSO algorithm

### 1.1. Basic knowledge of PSO algorithm

The PSO algorithm is a global stochastic search algorithm by simulating the nature of birds' activities and swarm intelligence. Particles, with no size and quality, always maintain two performances expressed by vectors in the search process, one is the speed expressed by  $\vec{v}_i^j = [v_i^1, v_i^2, \dots, v_i^n]$ , the other is the position expressed by  $\vec{x}_i^j = [x_i^1, x_i^2, \dots, x_i^n]$ . The speed vector represents the search direction of the flight, and position vector is the location of the solution in the search process. Each particle can maintain its own history optimal position  $H_{best}$  and global optimal position vector  $G_{best}$  in flight process. The velocity and location updating rules are as follows:

$$\vec{v}_i^j = w \times \vec{v}_i^j + c_1 \times rand_1^j \times (H_{best}^j - \vec{x}_i^j) + c_2 \times rand_2^j \times (G_{best}^j - \vec{x}_i^j), \quad (1)$$

$$x_i^j = x_i^j + v_i^j, \quad (2)$$

where  $i$  is the number of a particle,  $n$  is the dimension of the problem solved,  $j$  is the number of routine generation,  $w$  is the inertia weight,  $c_1$  and  $c_2$  are acceleration coefficients,  $rand_1^j$  and  $rand_2^j$  are the random numbers in the range of  $[0, 1]$ .

### 1.2. Improvement of initial population

The basic PSO has the shortcoming of premature convergence, which is related to the quality of the initial population. Here, the idea of random direction was employed to produce high-quality initial population to ensure that the algorithm can find the optimal solution and avoid stagnating while the normal particles are in the search midway. The procedure producing high-quality initial population is as follows: Step 1: Determining the initial search space according to the problem to be solved:

$$S = \prod_{i=1}^n [x_{i \min}, x_{i \max}]. \quad (3)$$

Step 2: Initializing of the number of an individual in the population  $j=1$  and selecting a random position  $x_j$  in the search space as base position.

Step 3: Initializing array  $X(k)$  for storing population individuals and  $k$  is the total number of individuals in the population.

Step 4: Producing a random vector  $\vec{v}_j$  in the given search space.

Step 5: Producing individual  $x_{j+1} = x_j + \vec{v}_j$  in sequence.

Step 6: Comparing  $x_{j+1}$  with  $x_j$  to judge whether they are beyond the search space, if  $x_{j+1}$  is better than  $x_j$ ,  $x_{j+1}$  is reserved,  $j=j+1$  and return to Step 5. Otherwise, return to Step 4 until the particle is generated which meets the terminate conditions.

### 1.3. Parameters improvement of PSO algorithm

#### Improvement of inertia parameter $w$

The process of birds searching for food in the nature is a continuous learning procedure. With the increase of searching time, their experience and the ability of acclimatization increases rapidly. It can be controlled by adjusting  $w$ . In Shi and Eberhart (2001),  $w$  was set to a constant and the abilities of the population study and environment sensitivity were ignored. Although then some scholars proposed linear  $w$ , they still ignore other constrains. Here, the accelerations of exploration experience and environmental suitability were considered to adjust parameters dynamically.  $w$  was defined as a nonlinear variable. In the early search phase, particles should fly with the lower speed around the initial position to avoid destroying the optimal value found. In the later search period, the particles should fly with rapid speed away from the initial

position to explore the solution space as much as possible. We define  $w$  as follows:

$$w = (w_{ini} - w_{end}) \times \left( \frac{g_{max} - g}{g_{max}} \right)^2 + w_{end}, \quad (4)$$

where  $w_{ini}$  and  $w_{end}$  represent initial and final inertia weight value, respectively,  $g$  is the value of current iteration times and  $g_{max}$  is the maximum iterating times.

#### Improvement of acceleration factors $c_1$ and $c_2$

The velocity direction of individual is incessantly changing with the surrounding environment changing. Here, the fuzzy reasoning was employed to adjust the acceleration coefficients adaptively. In the early search process, the flight direction should be adjusted constantly for obtaining a high acceleration factor. In the middle search process, the acceleration factors should be moderate. In the later process, acceleration factor should be small because of the larger flight rate. The acceleration factors adjusted by the fuzzy reasoning can make proper adjustment according to the experience and environment which can help the algorithm to find the better solution. The acceleration factors improved by fuzzy reason membership functions are shown in the (5) equation:

$$\left\{ \begin{array}{l} c_1 = c_2 = c \frac{w_{min} - w}{w_{min} - w_{end}} \\ c_1 = c_2 = c \left( \frac{w_{min} - w}{w_{min} - w_{end}} + \frac{2 \times (w - w_{max})}{w_{max} - w_{min}} + \frac{w - w_{min}}{w_{ini} - w_{min}} \right) \\ c_1 = c_2 = c \left( \frac{w_{max} - w}{w_{max} - w_{end}} + \frac{2 \times (w_{max} - w)}{w_{max} - w_{min}} + \frac{w - w_{min}}{w_{ini} - w_{min}} \right) \\ c_1 = c_2 = c \frac{w - w_{max}}{w_{ini} - w_{max}} \end{array} \right. \quad \begin{array}{l} w_{end} \leq w \leq w_{min} \\ w_{min} < w \leq \frac{w_{min} + w_{max}}{2} \\ \frac{w_{min} + w_{max}}{2} < w \leq w_{max} \\ w_{max} < w \leq w_{ini} \end{array}, \quad (5)$$

where  $c$  is a constant between [2, 3],  $w_{min}$  is the minimum inertia weight according to experts' experience and  $w_{max}$  is the maximum inertia weight according to experts' experience.

### 1.4. Improvement based on metropolis criteria

As a conversion function in the simulated annealing algorithm, the metropolis criteria can ensure the algorithm to jump out of local optimal solutions and possibly find the global optimal solution. PSO algorithm only accepts better solution position but ignores the bad solution positions having potential to find the global optimal solution, which makes the algorithm easily fall into premature convergence. By employing the metropolis criteria, PSO algorithm can accept some bad solutions, which improves its performance. The steps of the improved PSO algorithm are as follows.

Step 1: Initialization. According to the given problem, determine the value of space dimension  $n$ , inertia weight  $w_{ini}$ ,  $w_{end}$ ,  $w_{min}$  and  $w_{max}$ , acceleration factor  $c$ , the number of a routine population  $k$  and other parameters. Then, produce initial population according to the former section of "Improvement of initial population".

Step 2: Judge whether the algorithm should stop searching according to the fitness value  $fit(i)$ . If the terminate condition is met, stop searching, otherwise, go to Step 3.

Step 3: Remember the history optimal position  $H_{best}$  of every individual and global optimal position vector  $G_{best}$  of all particles.

Step 4: Producing next new generation population.

Step 5: Calculating the fitness value  $fit(j)^{k+1}$  of the new population and comparing with their parent's  $fit(j)$ , if  $fit(j)^{k+1} > fit(j)^k$ , the new generation will

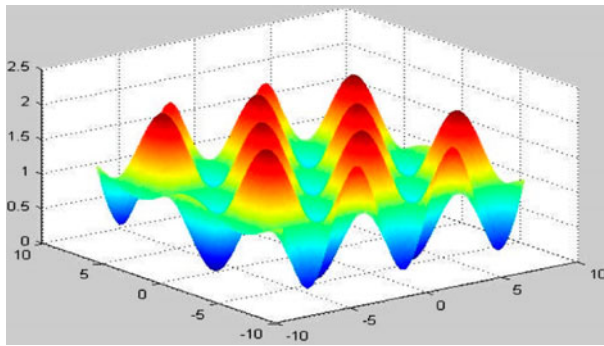


Fig. 1. Griewank function

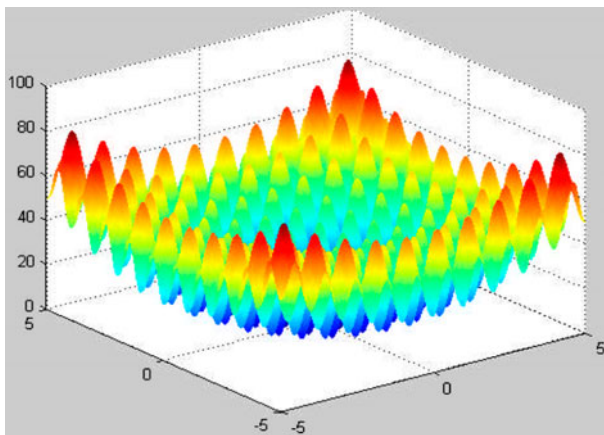


Fig. 2. Rastrigrin function

replace their parents, or else calculate the acceptable probability of the new population according to the following metropolis criterion:

$$p(g) = \exp - \left( \frac{fit(j) - fit(i)}{g} \right). \quad (6)$$

Step 6: Producing a random number  $rand(0,1)$ , if  $P(g) \geq rand(0,1)$ , replace their parents with the new generation, or else return to Step 2.

In order to validate the efficiency of the improved PSO we proposed, we employed it to optimize two typical functions with multiple extreme values and compared the results with the original PSO algorithm. The two chosen functions with different characteristics are the Griewank function and Rastrigrin function shown in Figures 1 and 2. The reason we chose the two functions is that both they have multiple local

minima and their optimization quality is easier to examine. The parameters are as follows:  $w_{ini}=0.9$ ;  $w_{end}=0.2$ ;  $w_{max}=0.7$ ;  $w_{min}=0.4$ ;  $c=2$ ;  $g_{max}=30$  and  $n=20$ . Because the algorithms are stochastic, in order to overcome the shortage that taking results of different numerical experiments would show completely different results, the comparison was based on a number of numerical experiments.

The comparison results based on 20 numerical experiments were shown in Table 1 and the average convergence curves were shown in Figures 3 and 4.

Through the results above, we know that the improved PSO algorithm can find the optimal of both functions. It performs better than the original PSO algorithm. From the average convergence curves, we see that the improved algorithm is obviously faster than the basic algorithm in finding the optimal solution. In Figure 3, the improved algorithm needs 10 iterations, yet the original PSO needs to circulate 18 iterations to find the solution. From Figure 4, we observe that the original PSO algorithm is difficult to jump out of the multiple local peaks. The improved algorithm can easily jump out the local optimal solution and find the global optimal solution.

Through the above comparison, we know that the performance of the algorithm has been improved and it can be employed to solve the structure shape optimization problems.

## 2. Engineering practice

In order to verify the efficiency of the improved PSO in truss structure optimization design, we employed it to solve the problem of truss structure shape optimization shown in Wang *et al.* (2002a).

The initial shape of the selected 37-bar plane truss structure is shown in Figure 5. Suppose that the position of lower chord panel points keeps without change, the upper panel points can move in the vertical direction and the structure remains symmetrical in the optimization. The load on the bottom chord node is  $P=10$  kN. The vertical direction displacement of the 10<sup>th</sup> node is restricted to  $[0, 10]$  mm. The minimum cross-sectional area is  $50 \text{ mm}^2$ , elastic modulus of the material is  $E=210$  GPa, material density is  $\rho=7800 \text{ kg/m}^3$  and the allowable stresses of all bars are restricted to  $[-240, 240]$  Pa.

Table 1. Comparison of original PSO algorithm and the improved PSO algorithm on Griewank function and Rastrigrin function

Function	Function name	Formulas	Test outcome value	
			Original PSO	Improved PSO
$f_1$	Griewank	$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos \frac{x_i}{\sqrt{i}} + 1$	0.3172	0.0000
$f_2$	Rastrigrin	$f(x) = \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i) + 10]$	9.1205	0.0001



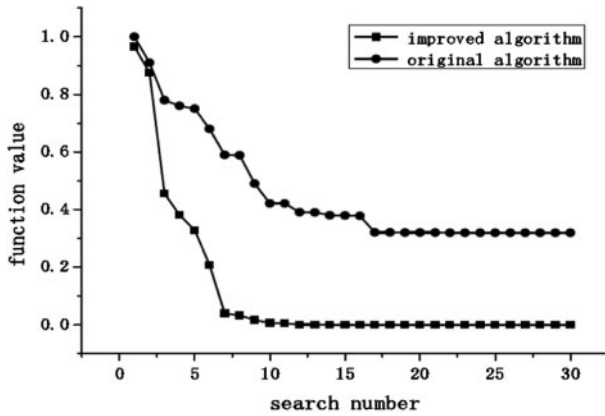


Fig. 3. Average convergence curve for Griewank function

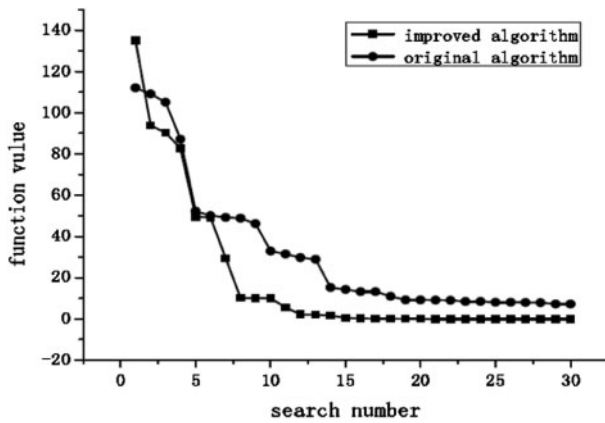


Fig. 4. Average convergence curve for Rastrigin function

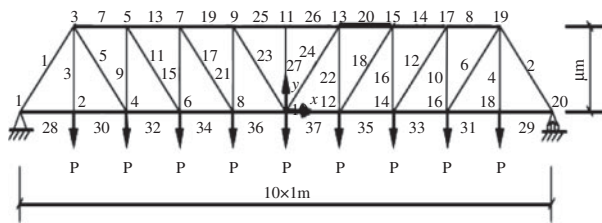


Fig. 5. Shape of the 37-bar plane truss before optimization

Before the application of the improved PSO, we set up the following mathematical model: Objective function:

$$\min W = \rho \sum_{i=1}^n A_i l_i + \lambda M. \quad (7)$$

Stress constraint conditions:

$$\sigma_{\min} \leq \frac{FN_i}{A_i} \leq \sigma_{\max}. \quad (8)$$

Displacement constraint condition:

$$\sum \frac{FN_i \times FL_i \times l_i}{E \times A_i} \leq D_{\max}. \quad (9)$$

The optimized node displacement constraint conditions:

$$\begin{cases} X_j^{\min} \leq X_j \leq X_j^{\max} \\ Y_j^{\min} \leq Y_j \leq Y_j^{\max}, \\ j = (1, 2, \dots, m) \end{cases} \quad (10)$$

where  $W$  represents the structure quality;  $M$  is multiplied by a penalty;  $\lambda$  is a punish factor, if the optimized variables satisfy the restriction condition, its value equals to zero, or else, it is 1.  $A_i$  and  $l_i$  ( $i = 1, 2, \dots, n$ ) represent the cross-sectional area and length of the  $i^{\text{th}}$  bar, respectively.  $FN_i$  is the calculated axial force on  $i^{\text{th}}$  bar;  $\sigma_{\min}$  and  $\sigma_{\max}$  represent the minimum and maximum of stress of bar, respectively;  $FL_i$  is an axial force in  $i^{\text{th}}$  bar resulting from a positive unit load;  $E$  is the value of material elastic modulus;  $D_{\max}$  is the maximum displacement value;  $Y_j^{\min}$  and  $Y_j^{\max}$  represent the minimum and maximum displacement of the  $j^{\text{th}}$  optimized node in the  $Y$  direction, respectively.

The improved PSO was used to solve this problem and the parameters selected are as follows: the particles number  $PN = 25$ , the maximum iteration times  $INT = 1000$ , the acceleration parameter  $c = 2$  the inertial weight coefficients:  $w_{ini} = 1.0$ ;  $w_{end} = 0.2$ ;  $w_{\min} = 0.4$ , and  $w_{\max} = 0.8$ . Numerical simulation shows under these values of coefficients, the algorithm performs best. To keeping symmetry in the process of the truss structure optimization, the position optimization variables are

$$[A_1, A_3, A_5, A_7, A_9, A_{11}, A_{13}, A_{15}, A_{17}, A_{19}, A_{21}, A_{23}, A_{25}, A_{27}, A_{29}, A_{31}, A_{33}, A_{35}, A_{37}, Y_3, Y_5, Y_7, Y_9, Y_{11}]^T$$

and the design variables are shown in Table 2. Combined with finite element method (FEM), the improved PSO was employed to the optimization and the programs were written in Matlab7.1 language.

Table 2. Symmetric groups of bars and nodes

Symmetric bars	1, 2	3, 4	5, 6	7, 8	9, 10	11, 12	13, 14	15, 16	17, 18	19, 20
Optimized bars	1	3	5	7	9	11	13	15	17	19
Symmetric bars	21, 22	23, 24	25, 26	27	28, 29	30, 31	32, 33	34, 35	36, 37	
Optimized bars	21	23	25	27	29	31	33	35	37	
Symmetric nodes	1, 20	2, 18	3, 19	4, 16	5, 17	6, 14	7, 15	8, 12	9, 13	10, 11
Optimized nodes			3		5		7		9	11

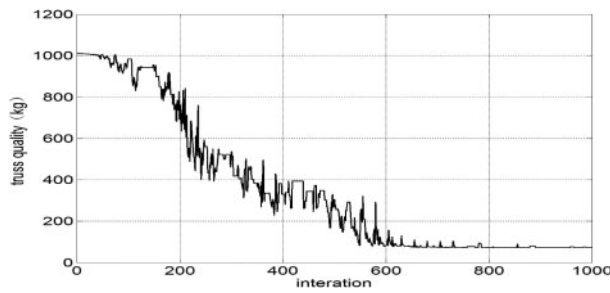


Fig. 6. Convergence curve of the algorithm we proposed for the 37-bar structure optimization

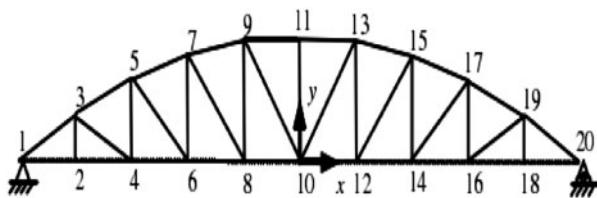


Fig. 7. Optimized shape of the 37-bar plane truss

The optimized results of structure quality are shown in Table 2 and convergence curve is shown in Figure 6. And the final optimization shape is shown in Figure 7.

In Figure 6, the convergence curve shows that in the early searching process, the objective function is around 1000 kg which is approximately 15 times of the optimal solution. And the iterations account for a little over one percent of the total, which means the random direction algorithm can obviously produce the high-quality initial population. The curve has no stagnation in the first 600 iterations, and this is made possible because of the dynamic adjustment by fuzzy system. In all search process, the curve shows concussion adjustment phenomenon, which is due to the function of the embedded metropolis criterion.

Then, the results were compared with the results obtained from Tang *et al.* (2009) and Wang *et al.* (2002a). The comparison was shown in Tables 3 and 4.

From Table 3, we know that the displacement of the tenth node is 9.93 mm in the *Y* direction, which meets the displacement constraint conditions. The total quality of the optimized structure is 69.27 kg, which is better than the outcome of Tang *et al.* (2009) and Wang *et al.* (2002a). The improved PSO we proposed has the better global search performance with the stronger ability to jump out the local optimal solution.

Figure 7 shows that the optimized truss shape is an arch that can decompose the vertical load to horizontal direction and reduce the stress concentration effect. This is more reasonable than the original trapezoid shape according to the mechanical knowledge. From Table 4, we see that both the mean and

Table 3. Comparison of different algorithms on the 37-bar plane truss optimization

Variable	Improved PSO we proposed	Results of Tang <i>et al.</i> (2009)	Results of Wang <i>et al.</i> (2002a)	Improved PSO we proposed	Results of Tang <i>et al.</i> (2009)	Results of Wang <i>et al.</i> (2002a)
$A_1$ (mm <sup>2</sup> )	567.8	870.7	883.1	205.4	50.5	50.0
$A_3$ (mm <sup>2</sup> )	158.7	51.9	50.0	65.3	67.4	183.7
$A_5$ (mm <sup>2</sup> )	120.9	51.3	50.0	50.0	50.5	183.7
$A_7$ (mm <sup>2</sup> )	760.1	818.6	715.4	52.3	50.2	194.0
$A_9$ (mm <sup>2</sup> )	60.2	52.0	50.0	51.2	51.3	192.8
$A_{11}$ (mm <sup>2</sup> )	56.2	50.1	115.3	54.2	50.2	187.4
$A_{13}$ (mm <sup>2</sup> )	564.8	776.2	646.1	496.5	508.2	1021
$A_{15}$ (mm <sup>2</sup> )	87.5	50.3	50.0	897.6	904.4	1718
$A_{17}$ (mm <sup>2</sup> )	50.0	50.3	348.1	1186.2	1178.1	2269
$A_{19}$ (mm <sup>2</sup> )	565.4	754.0	553.8	1370.4	1346.1	2669
$A_{21}$ (mm <sup>2</sup> )	98.6	50.1	54.1	1406.8	1363.4	2734
$A_{23}$ (mm <sup>2</sup> )	67.8	50.9	50.0	69.27	77.46	105.2
$A_{25}$ (mm <sup>2</sup> )	598.1	746.8	528.2	-9.93	-8.05	-2.23
			Node10 displacement (mm)			
			Quality (kg)			

Table 4. Comparison of stresses in all bars (MPa)

Bar number	Improved PSO we proposed	Results of Tang et al. (2009)	Results of Wang et al. (2002a)	Bar number	Improved PSO we proposed	Results of Tang et al. (2009)	Results of Wang et al. (2002a)
1 <sup>st</sup>	178.22	114.08	71.33	23 <sup>rd</sup>	-32.24	-83.55	-43.32
3 <sup>rd</sup>	-63.0	-192.68	-200	25 <sup>th</sup>	148.66	122.79	86.74
5 <sup>th</sup>	13.92	1.998	-71.21	27 <sup>th</sup>	-31.49	-62.82	-118.87
7 <sup>th</sup>	126.34	116.23	79.34	29 <sup>th</sup>	-21.01	3.38	5.94
9 <sup>th</sup>	-178.55	-193.20	-149.13	31 <sup>st</sup>	-27.44	4.51	5.94
11 <sup>th</sup>	14.55	-18.04	5.00	33 <sup>rd</sup>	2.60	6.35	-7.22
13 <sup>th</sup>	163.12	119.05	81.78	35 <sup>th</sup>	14.54	-6.85	-5.76
15 <sup>th</sup>	-120.53	-186.76	-209.96	37 <sup>th</sup>	31.31	-7.39	1.09
17 <sup>th</sup>	29.55	-0.61	9.37	Mean	78.47	82.59	76.76
19 <sup>th</sup>	157.48	119.89	87.44	Standard deviation	64.83	75.43	75.44
21 <sup>st</sup>	-112.88	-199.13	-240.00				

standard deviation of the stress obtained by our algorithm are smaller than the results of Tang et al. (2009). Although the mean stress we obtained is bigger than results of Wang et al. (2002a), the displacement of the latter is 2.23 mm and quality is 105.2 kg which are too conservative. The above analysis shows that improved PSO we proposed performs better than the other two algorithms. Engineering practice also verifies the applicability of the results we obtained.

## Conclusions

It is theoretically and practically significant to study the algorithms for the structure optimization. In order to overcome the shortcoming of the premature convergence in the common algorithms, we proposed a novel PSO and introduced it to the shape optimization of a truss structure. The application and comparison with other algorithms show that the new approach using combination of stochastic direction algorithm, fuzzy system, and metropolis criterion has significant implementation advantages over the earlier techniques. This study provides a promising method for the structure optimization.

## Acknowledgements

This work was supported by the Natural Science Foundation of Hebei Province, China (No. E2012402030) and Foundation of Hebei Educational Committee, China (ZD2010222).

## References

- Bu, Y.; Jiang, J.; Sun, Y. 2010. The study progress of particle swarm optimization algorithm, *Journal of Inner Mongolia University for Nationalities* 25(6): 630–633.
- Cai, X.; Li, H.; Wu, Y.; Zhu, J. 2011. Research progress of optimal design engineering structures, *Journal of Hohai University* 39(3): 269–276.

- Gandomi, A. H.; Yang, X. S. 2011. Benchmark problems in structural optimization, in Koziel, S.; Yang, X.-S. (Eds.). *Computational Optimization, Methods and Algorithms*. Berlin: Springer, 259–281.
- Hadidi, A.; Kaveh, A.; Azar, B. F.; Talatahari, S.; Farahmandpour, C. 2011. An efficient hybrid algorithm based on particle swarm and simulated annealing for optimal design of space trusses, *Iranian Journal of Optimization in Civil Engineering* 1(3): 377–395.
- Hagishita, T.; Ohsaki, M. 2009. Topology optimization of trusses by growing ground structure method, *Structural and Multidisciplinary Optimization* 37: 377–393.  
<http://dx.doi.org/10.1007/s00158-008-0237-4>
- Heppner, F.; Grenander, U. 1990. A stochastic nonlinear model for coordinated bird flocks, in Krasner, E. (Ed.). *The Ubiquity of Chaos*. Washington: AAAS Publications, 233–238.
- Huang, P.; Meng, Y. 2009. *Optimal theories and methods*. Beijing: Tsinghua University Press. 59 p.
- Kaveh, A.; Azar, B. F.; Talatahari, S. 2008. Ant colony optimization for design of space trusses, *International Journal of Space Structures* 23(3): 167–181.  
<http://dx.doi.org/10.1260/026635108786260956>
- Kaveh, A.; Talatahari, S. 2010. Optimum design of skeletal structures using imperialist competitive algorithm, *Computers & Structures* 88(21–22): 1220–1229.  
<http://dx.doi.org/10.1016/j.compstruc.2010.06.011>
- Kaveh, A.; Talatahari, S. 2011. An enhanced charged system search for configuration optimization using the concept of fields of forces, *Structural and Multidisciplinary Optimization* 43: 339–351.  
<http://dx.doi.org/10.1007/s00158-010-0571-1>
- Kaveh, A.; Talatahari, S. 2012. Charged system search for optimal design of frame structures, *Applied Soft Computing* 12(1): 382–393.  
<http://dx.doi.org/10.1016/j.asoc.2011.08.034>
- Kennedy, J. 2006. Swarm intelligence, in Zomaya, A. Y. (Ed.). *Handbook of Nature-inspired and Innovative Computing: Integrating Classical Models with Emerging Technologies*. Secaucus, NJ, US: Springer, 187–219.

- Kennedy, J.; Eberhart, R. 1995. *Particle swarm optimization*, in *Proc. of the IEEE International Conference on Neural Networks*. Piscataway, NJ: IEEE Press, 1942–1948.
- Li, F.; He, S.; Xue, S. 2009. Application of a particle swarm optimization algorithm in truss structure optimal design, *Journal of Civil, Architecture & Environmental Engineering* 31(1): 7–12.
- Nowak, A.; Szamrej, J.; Latane, B. 1990. From private attitude to public opinion: a dynamic theory of social impact, *Psychological Review* 97(3): 362–376. <http://dx.doi.org/10.1037/0033-295X.97.3.362>
- Poli, R. 2008. Analysis of the publications on the applications of particle swarm optimization, *Journal of Artificial Evolution and Applications*, ID 685175, 10 p.
- Reeves, W. T. 1983. Particle systems – a technique for modeling a class of fuzzy objects, *ACM Transactions on Graphics* 2(2): 91–108. <http://dx.doi.org/10.1145/357318.357320>
- Reng, F.; Li, L. 2008. An improved particle swarm optimization (IPSO) method and its application in truss design, *Journal of Guangzhou University* 7(3): 82–85.
- Reynolds, C. W. 1987. Flocks, herds, and schools: a distributed behavioral model, *ACM Computer Graphics* 21(4): 25–34. <http://dx.doi.org/10.1145/37402.37406>
- Sonmez, M. 2011. Artificial bee colony algorithm for optimization of truss structures, *Applied Soft Computing* 11(2): 2406–2418. <http://dx.doi.org/10.1016/j.asoc.2010.09.003>
- Šešok, D.; Belevičius, R. 2008. Global optimization of trusses with a modified genetic algorithm, *Journal of Civil Engineering and Management* 14(3): 147–154. <http://dx.doi.org/10.3846/1392-3730.2008.14.10>
- Shi, Y.; Eberhart, R. 2001. Fuzzy adaptive particle swarm optimization, in *Proc. of the IEEE Conference on Evolutionary Computation*, Seoul, 2001, 103–106.
- Takada, T. 2012. Multiobjective optimization of truss topology by linear/sequential linear programming method, *Journal of Mechanics Engineering and Automation* 2: 585–593.
- Talatahari, S.; Kheirollahi, M.; Farahmandpour, C.; Gandomi, A. H. 2012. A multi-stage particle swarm for optimum design of truss structures, *Neural Computing & Applications*. <http://dx.doi.org/10.1007/s00521-012-1072-5>
- Tang, H.; Li, F.; Wang, Y.; Xue, S.; Cheng, R. 2009. Particle swarm optimization algorithm for shape optimization of truss structures, *Journal of Harbin Institute of Technology* 41(12): 94–99.
- Wang, D.; Hong, W.; Jiang, J. 2002a. Combinatorial optimization of the truss structure shape and size, *Chinese Journal of Applied Mechanics* 19(3): 72–77.
- Wang, D.; Zhang, W. H.; Jiang, J. S. 2002b. Combined shape and sizing optimization of truss structures, *Computational Mechanics* 29: 307–312. <http://dx.doi.org/10.1007/s00466-002-0343-x>
- Wang, H.; Ohmori, H. 2012. Truss optimization using genetic algorithm considering ultimate resistance, *Journal of the International Association for Shell and Spatial Structures* 53(1): 9–17.
- Xia, Q.; Wang, M. Y.; Shi, T. 2013. A method for shape and topology optimization of truss-like structure, *Structural and Multidisciplinary Optimization* 47(5): 687–697.
- Xie, Z.; Zhong, S.; Wei, Y. 2011. Improved particle swarm algorithm and convergence analysis, *Computer Engineering and Application* 47(1): 46–49.
- Zhang, J.; Zhan, Z. 2009. *Computational intelligence*. Beijing: Tsinghua University Press. 140 p.
- Zhou, S.; Gao, Y. 2012. The application of an improved particle swarm algorithm based on the information entropy in truss structure optimization design, *Chinese Journal of Engineering Design* 19(5): 340–344.
- Zhou, S.; Bo, T.; Shi, S. 2011. Application of hybrid algorithm in optimization design of light steel structure, *Journal of Hebei University of Engineering* 28(2): 71–74.

**Yancang LI.** Received the PhD degree in Project Management from Tianjin University, China in 2008. Currently he is a researcher at Hebei University of Engineering, China. His research interests include artificial intelligence and its application in the engineering practice.

**Yang PENG.** He is studying at Civil Engineering College, Hebei University of Engineering, majored in artificial intelligence and its application in the engineering practice.

**Shujing ZHOU.** He is a researcher at Hebei University of Engineering, China. His research interests include artificial intelligence and its application in the engineering practice.