A quantum-inspired particle swarm optimization for sizing optimization of truss structures

Zhiqiang Xu 1, 2, a, Yunxian Cui 1, b and Baoliang Li 1, c

1 School of Mechanical Engineering, Dalian Jiaotong University, Dalian 116028, China
2 School of Chemical and Mechanical Engineering, Eastern Liaoning University, Dandong 118003, China.

a xzq3171057@126.com, b dlcyx007@126.com, c libl@djtu.edu.cn

Abstract. Particle swarm optimization (PSO) algorithm has some advantages such as simple principle, less model parameters. However, it has some flaws for dealing with sizing optimization of truss structures, such as low accuracy, slow astringency and poor population diversity in the later evolution stage. In order to overcome these shortcomings and deal with optimization problem of truss structures effectively, a new quantum-inspired particle swarm optimization (QPSO) algorithm with quantum angle encoding and a perturbation operator of the population is proposed in this paper. Then, QPSO is used for optimization of truss structures by means of penalty function method. In the end, two typical numerical tests for truss design are performed to calibrate and verified the proposed algorithm. The results of the QPSO compared with PSO and other optimization algorithms show that the proposed QPSO has a better performance in terms of both the optimum solutions and the convergence capability than PSO and is a powerful optimization technique for solving optimization problem of truss structures.

Keywords: Particle swarm optimization algorithm; population; truss; quantum angle; penalty function.

1. Introduction

Truss is the most common structural type in modern structural engineering, which is widely used in large span grid (shell) engineering, bridge engineering, transmission line engineering and other structural engineering. It can minimize the dead weight of the structure under the condition of satisfying the functions of displacement and stress, which is of great significance to reduce the cost of truss structure. Therefore, the optimal design of truss structure has always been a hot issue in structural engineering [1,2,3].

Truss optimization algorithms have been widely used in structural optimization over the past decade. In general, these algorithms can be divided into two categories: one is the traditional optimization algorithm based on gradient information, and the other is the direct search algorithm which does not rely on gradient information (modern intelligent optimization algorithm). The traditional gradient optimization algorithm needs to calculate the gradient information of the objective function and the constraint function, which requires that the objective function and the constraint function must be
continuous in the solution space, and is strict with the initial value, resulting in the local optimal solution. These defects of gradient optimization algorithm often lead to the failure in multivariable complex truss structure optimization. Different from gradient optimization algorithm, direct search algorithm is a modern intelligent optimization algorithm inspired by the social behavior of biological population (such as particle swarm optimization algorithm), objective physical criteria (such as gravity search algorithm) and biological evolution theory (such as genetic algorithm). The direct search algorithm does not need the gradient information of objective function and constraint function, which is not sensitive to the initial iteration value, and is easy to jump out of the local optimal solution. It can deal with the optimization problem of continuous function and discontinuous function, attracting the attention of structural engineers. Babaei et al studied the optimal design of scallop truss dome structure by using genetic algorithm [1]. Degertekin et al used a teaching optimization algorithm to study the optimal design of truss structures [2]. Kaveh et al proposed a hybrid particle swarm optimization algorithm to study the optimization of truss structures [3]. Angelo et al used ant colony algorithm to study the multi-objective optimization problem of truss structure [4].

Compared with other direct search algorithms, particle swarm optimization (PSO) algorithm has the advantages of simple mathematical principle, less parameters and easy programming. Therefore it has been widely used in various structural optimization design. However, it is prone to premature convergence, with population diversity and poor local search ability [5-7], which seriously affects the computational efficiency and accuracy. In order to improve the optimization ability of PSO and better solve structural optimization problems, this paper first proposes an improved PSO algorithm, the quantum-inspired particle swarm optimization (QPSO) algorithm, then uses two truss optimization examples to verify the effectiveness and reliability of optimization performance and dealing with truss optimization of QPSO.

2. Mathematical model of truss optimization

2.1. Design variable.
The truss optimization problem studied in this paper is to minimize the dead weight of truss by adjusting the cross-sectional area of structural members under the conditions of structural displacement and stress. The design variable is the sectional area of truss members.

\[
\mathbf{x} = \left[ A_1, A_2, L, A_{ng} \right], A_i \in D, \ i = 1, L, ng
\]  

(1)

In the formula, vector \( \mathbf{x} \) is the design variable in solution space \( D \). \( A_i \) is the sectional area of the truss member, and \( ng \) is the number of truss elements.

2.2. Objective function.
The total weight of the truss structure is taken as the objective function of the optimization problem, that is

\[
\text{min} f(\mathbf{x}) = \sum_{i=1}^{ng} \rho_i L_i A_i
\]  

(2)

In the formula, \( f(\mathbf{x}) \) is the objective function. \( \rho_i \) is the material density of truss element, and \( L_i \) is the length of member?

2.3. Constraint condition
1. Stress constraint

\[
g_j(\mathbf{x}) = \left| \frac{\sigma_j}{\sigma_{ij}} \right| -1 \leq 0, j = 1, L, NM
\]  

(3)
In the formula, \( g_j(x) \) is the stress constraint function, \( \sigma_j \) and \( \sigma_d \) are the calculated axial stress and its allowable value respectively, and \( NM \) is the total number of members.

2. Displacement constraint

\[
g_r(x) = \frac{\delta_r}{\delta_{wr}} - 1 \leq 0, r = 1, L, ND
\]  

(4)

In the formula, \( g_r(x) \) is the displacement constraint function. \( \delta_r \) and \( \delta_{wr} \) are the calculated node displacement and its allowable value respectively. \( ND \) is the total number of node degrees of freedom.

2.4. Treatment of constrained condition.

Like other direct search methods, QPSO can’t be directly used in constrained optimization problems. In this paper, the constrained optimization problem of truss structure is transformed into an unconstrained problem (5) by using the penalty function technique shown in equation [3].

\[
W(x) = (1 + \varepsilon_1 \cdot \nu)^{\varepsilon_2} \times f(x),

v = \sum_{j=1}^{NM} \max(0, g_j(x)) + \sum_{r=1}^{ND} \max(0, g_r(x))
\]  

(5)

In the formula, \( W(x) \) is an unconstrained objective function (the objective function after penalty). The global and local search ability of the algorithm should be considered for the value of penalty factors \( \varepsilon_1 \) and \( \varepsilon_2 \). In this paper, the value of \( \varepsilon_1 \) is 0, and \( \varepsilon_2 \) is gradually increased from 1.5 in the early stage of algorithm evolution to 3 at the end of algorithm evolution.

3. Literature References Particle swarm optimization algorithm

PSO solves the optimization problem by simulating the predation behavior of birds. When a bird flies away from the flock to hunt, the birds around will also fly to prey. Once a bird finds food, the birds around will follow, until the entire flock falls on the food. A bird in the flock is regarded as a particle with velocity and position attributes. All particles search for the global optimal value by searching for the current optimal value in the search space.

Suppose that the particle swarm optimizes in \( d \) dimensional space, the position of the \( i \) th particle is \( X_i = (x_{i1}, x_{i2}, \ldots, x_{id}) \), and the velocity is \( V_i = (v_{i1}, v_{i2}, \ldots, v_{id}) \); \( p_i = (p_{i1}, p_{i2}, \ldots, p_{id}) \) is the optimal location it searches. If \( p_g = (p_{g1}, p_{g2}, \ldots, p_{gd}) \) is the optimal position of the population, then the particle uses the formula (6) and (7) to update the speed and position:

\[
v_{ij}^{t+1} = \omega v_{ij}^t + c_1 r_1 [p_g - x_{ij}^t] + c_2 r_2 [p_g - x_{ij}^t]
\]

\[
x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1}, j = 1, 2, \ldots, d
\]

(6)

(7)

In the formula, \( c_1 \) and \( c_2 \) are learning factors; \( r_1 \) and \( r_2 \) are random numbers between \([0,1]\). \( \omega \) is the inertia weight, which is calculated according to formula (8).

\[
\omega = \omega_{\text{max}} - \frac{\omega_{\text{max}} - \omega_{\text{min}}}{\text{iter}_{\text{max}}} \cdot \text{iter}
\]

(8)

In the formula, \( \omega_{\text{max}} \) and \( \omega_{\text{min}} \) represent the maximum weight and the minimum weight respectively. \( \text{iter}_{\text{max}} \) is the maximum iteration algebra of the algorithm, and \( \text{iter} \) is the current algebra.
4. Quantum-inspired particle swarm optimization algorithm

4.1. Quantum-bit encoding
QPSO uses qubits (quantum bit) to encode particles. The chromosomes can represent the superposition of multiple states at the same time by coding the particles with qubits. Therefore QPSO has better population diversity to improve the traversal ability and convergence speed of the algorithm to the solution space [8]. In quantum computing, qubit is the smallest unit of information. A qubit can be in $|0\rangle$ state, $|1\rangle$ state and any superposition state of them, that is:

$$ |\varphi\rangle = \alpha|0\rangle + \beta|1\rangle $$  \hspace{1cm} (9)

In the formula, $\alpha$ and $\beta$ is the probability amplitude of qubit $|\varphi\rangle$; $|\alpha|^2$ represents the probability that $|\varphi\rangle$ collapses to $|0\rangle$ due to observation; $|\beta|^2$ represents the probability of collapse to $|1\rangle$, and it satisfies:

$$ |\alpha|^2 + |\beta|^2 = 1 $$  \hspace{1cm} (10)

A qubit can be defined by its probability amplitude, for example, $|\varphi\rangle$ can be represented as $(\alpha, \beta)^T$.

Then $m$ qubits are expressed as

$$ \begin{bmatrix} \alpha_1, \alpha_2, \ldots, \alpha_m \\ \beta_1, \beta_2, \ldots, \beta_m \end{bmatrix}, \quad \text{and} \quad |\alpha_i|^2 + |\beta_i|^2 = 1, \quad i = 1, L, m. $$

According to equations (9) and (10), the state of a qubit can also be expressed as [9]:

$$ |\varphi\rangle = \sin \theta |0\rangle + \cos \theta |1\rangle, \quad \theta = \arctan \frac{\alpha}{\beta} $$  \hspace{1cm} (11)

$\theta$ is the quantum angle. Obviously, a qubit can also be represented as $[\theta]$, and $m$ qubits can be expressed as $[\theta_1, \theta_2, \ldots, \theta_m]$.

4.2. Particle evolution
In this paper, the rotation of quantum angle is used to describe the motion of particles in solution space, and the velocity evolution formula of particle is:

$$ v_{ij}^{t+1} = \omega v_{ij}^t + c_1 r_1 [\theta_{ij} (pbest) - \theta_{ij}] + c_2 r_2 [\theta_{ij} (gbest) - \theta_{ij}] $$  \hspace{1cm} (12)

The evolution formula of particle position is:

$$ \theta_{ij}^{t+1} = \theta_{ij}^t + v_{ij}^{t+1} $$  \hspace{1cm} (13)

$v_{ij}$, $\theta_{ij}$, $\theta_{ij} (pbest)$ and $\theta_{ij} (gbest)$ represent particle velocity, current position, individual optimal value and global optimal value respectively.

4.3. Perturbation operator
In order to avoid local optimal value and population premature phenomenon, a perturbation operator is proposed in this paper. When the same optimal value is obtained in three successive iterations, the population is disturbed. The specific method is as follows: the optimal individual is retained, and the other individuals are randomly generated.

4.4. Evolution process of QPSO
The flow chart of QPSO algorithm proposed in this paper is as follows:

1. Set the simulation parameters and use the quantum angle to encode particles to obtain the initial population $Q(t)$:

$$ Q(t) = [q_1, q_2, \ldots, q_m], q_j = [\theta_{ij}, \theta_{ij}, \ldots, \theta_{ij}]_m. $$  \hspace{1cm} (14)

2. Obtain the binary solution by observing each individual of the population $Q(t)$. 

The observation method is as follows: a value is randomly generated between \([0, 1]\). If the value is greater than \(|\cos \theta_j|\) (or \(|\sin \theta_j|\)), the observation result is taken as 1, otherwise, it is taken as 0.

(3). Transform the binary solution into decimal solution, and implement fitness evaluation to save the optimal individual.

(4). Decide whether the stop condition is met. If so, output the optimal value; otherwise, turn to (5).

(5). Use equations (12) and (13) to update the population \(Q(t)\).

(6). Determine whether the algorithm needs to implement population disturbance. If necessary, start the perturbation operator, otherwise, enter (7);

(7). \(t=t+1\), turn to (2).

The flow chart of QPSO algorithm is shown in Figure 1.

![Figure. 1 Calculation flow chart of QPSO](image-url)
5. Example analysis

5.1. Example 1: plane structure-10 bar truss

Firstly, use the 10 bar plane truss shown in Figure 2 as the first example. The elastic modulus of the structure is 10,000 ksi and the density is 0.1 lb/in3. The load $P_1 = 150$ kips acts on joints 2 and 4; $P_2 = 50$ kips acts on joints 1 and 3. Besides, the cross-sectional area of the design variable members is in the range of [0.1~35.0] in2, and the displacements of joints 1, 2, 3 and 4 along the X and Y axes are limited within ±2 in. The tensile and compressive stresses of the members are all limited to 25 ksi. QPSO and PSO are used respectively to calculate this example. The calculation parameters are as follows: the number of particles is 50 and the learning factor is 2 with the maximum weight of 0.9 and the minimum weight of 0.4. The number of iteration steps is 400.

![Figure 2. 10 bar truss structure](image)

Table 1 Comparison of optimization results of 10 bar truss

| Design variable | Design value (in²) | Design value (lb) |
|-----------------|--------------------|-------------------|
| No.  | Name  | TBLO[2] | PSO[11] | HPSO[11] | Literature [10] | SAHS [12] | PSO | QPSO |
|------|-------|---------|---------|--------|-----------------|---------|------|------|
| 1    | A1    | 23.524  | 22.935  | 25.353 | 25.810          | 23.525  | 23.371 | 23.524 |
| 2    | A2    | 0.100   | 0.100   | 0.100  | 0.100           | 0.100   | 0.100 | 0.100 |
| 3    | A3    | 25.441  | 25.355  | 25.502 | 27.230          | 25.429  | 24.154 | 25.369 |
| 4    | A4    | 14.479  | 14.373  | 14.250 | 16.650          | 14.488  | 14.505 | 14.380 |
| 5    | A5    | 0.100   | 0.100   | 0.100  | 0.100           | 0.100   | 0.146 | 0.100 |
| 6    | A6    | 1.995   | 1.990   | 1.972  | 2.204           | 1.992   | 1.969 | 1.969 |
| 7    | A7    | 12.334  | 12.346  | 12.363 | 12.780          | 12.352  | 12.549 | 12.368 |
| 8    | A8    | 12.689  | 12.923  | 12.984 | 14.220          | 12.698  | 13.522 | 12.797 |
| 9    | A9    | 20.354  | 20.678  | 20.356 | 22.140          | 20.341  | 20.569 | 20.326 |
| 10   | A10   | 0.100   | 0.100   | 0.101  | 0.100           | 0.100   | 0.100 | 0.100 |
| Weight | (lb) | 4678.31 | 4679.47 | 4677.29 | 5059.70 | 4678.84 | 4692.98 | 4677.01 |
Table 2 Node displacement

| Node | X  | Y  |
|------|----|----|
| 1    | 0.0377 | -0.6024 |
| 2    | 0.2345 | -0.3501 |
| 3    | 1.1000 | -1.9999 |
| 4    | -1.9999 | -0.6583 |
| 1    | -1.5583 | -1.1000 |
| 2    | -1.9999 | -0.6583 |
| 3    | -1.5583 | -1.1000 |
| 4    | -1.1000 | -0.6583 |

The results of QPSO and PSO optimization of 10 bar truss are given in Table 1. Reference [10] adopts traditional optimization algorithm. Reference [11] uses PSO algorithm and improved PSO algorithm HPSO algorithm; reference [2] adopts TBLO algorithm and reference [12] uses SAHS algorithm to study the example. In order to facilitate comparative analysis, the optimization results of these literatures are also shown in Table 1. According to table 1, the self weight of QPSO optimized structure is 467701lb. This calculation result is about 7.6% lighter than that of reference [10], and is also reduced to some extent compared with the optimization results of PSO, HPSO, TBLO and SAHS in reference [11]. The calculated value of PSO is 4692.98 lb, which is about 0.3% heavier than that of QPSO. According to this, the QPSO algorithm proposed in this paper can search for better optimal solution, which shows that QPSO algorithm has better optimization ability than PSO algorithm.

Figure 3 shows the evolution curve of the structure weight with the iteration step when QPSO and PSO algorithms are used to solve the 10 bar truss optimization problem. QPSO algorithm finds the optimal value of structure weight in 187 iteration steps, while PSO algorithm finds its optimal value in 396 iteration steps. Obviously, the convergence speed of QPSO algorithm is obviously faster than that of PSO algorithm. Table 2 and Figure 3 show the joint displacement and bar stress values of 10 bar truss under QPSO optimal design state respectively. It can be seen from Table 2 and Figure 3 that the displacement and stress values of the structure meet the requirements of constraint conditions, which indicates that the design scheme of 10 bar truss calculated by QPSO is feasible.
5.2. Example 2: space truss-25 bar space truss

Take the 25 bar space truss shown in Figure 5 as the second example. The stress of the structure is shown in Table 3. According to the symmetry of the structure, the sectional area of structural bars can be divided into 8 groups. The grouping situation is in Table 4, and the area of bars is not less than 0.1 in². The material properties of the bars are the same. The elastic modulus is 10000 ksi, and the density is 0.1 lb/in³. The displacements in X and Y directions at nodes 1 and 2 are not more than ± 0.35 in. The allowable tensile and compressive stress of each bar is 40 ksi. QPSO and PSO are used to calculate this example. The number of iteration steps is 400, and other parameters are the same as those in example 1.

Table 3. Load table of 25 bar space truss

| Node | X   | Y    | Z    |
|------|-----|------|------|
| 1    | 1.000 | -10.000 | -10.000 |
| 2    | 0   | -10.000 | -10.000 |
| 3    | 0.500 | 0    | 0    |
| 6    | 0.600 | 0    | 0    |

The QPSO and PSO design results of 25 bar truss are given in Table 5. For comparative analysis, the results of Perez, Wu and Kaveh are given in Table 5 at the same time. Table 5 shows that the structure weight calculated by QPSO is 467.59 lb, which is 3.9% and 3.7% lower than that of Wu and Perez respectively. Compared with Kaveh's, the QPSO results still have advantages. The weight of truss calculated by PSO in this paper is 468.39 lb, which is better than that of Wu and Perez, but larger than that of QPSO and Kaveh. This example also indicates that QPSO has better calculation accuracy than PSO, and the optimization ability of QPSO is obviously better than that of PSO.
Table 4 Bar group number of 25 bar truss

| Bar No. | Node (i,1) | Node (i,2) | Group | Bar No. | Node (i,1) | Node (i,2) | Group |
|---------|------------|------------|-------|---------|------------|------------|-------|
| 1       | 1          | 2          | 1     | 14      | 3          | 10         | 6     |
| 2       | 1          | 4          | 2     | 15      | 6          | 7          | 6     |
| 3       | 2          | 3          | 2     | 16      | 4          | 9          | 6     |
| 4       | 1          | 5          | 2     | 17      | 5          | 8          | 6     |
| 5       | 2          | 6          | 2     | 18      | 4          | 7          | 6     |
| 6       | 2          | 4          | 3     | 19      | 3          | 8          | 7     |
| 7       | 2          | 5          | 3     | 20      | 5          | 10         | 7     |
| 8       | 1          | 3          | 3     | 21      | 6          | 9          | 7     |
| 9       | 1          | 6          | 3     | 22      | 6          | 10         | 8     |
| 10      | 6          | 3          | 4     | 23      | 3          | 7          | 8     |
| 11      | 4          | 5          | 4     | 24      | 4          | 8          | 8     |
| 12      | 3          | 4          | 5     | 25      | 5          | 9          | 8     |
| 13      | 6          | 5          | 5     |         |            |            |       |

Table 5 Optimization results of 25 bar truss

| Variable | Design value(in2) |
|----------|-------------------|
| No.      | Name              | Perez [13] | Wu [14] | Kaveh [15] | Kaveh [16] | PSO   | QPSO  |
| 1        | A1                | 0.100      | 0.100   | 0.100      | 0.100      | 0.100 | 0.100 |
| 2        | A2-A5             | 1.023      | 0.500   | 0.100      | 0.120      | 0.107 | 0.104 |
| 3        | A6-A9             | 3.400      | 3.400   | 3.759      | 3.528      | 3.469 | 3.561 |
| 4        | A10-A11           | 0.100      | 0.100   | 0.100      | 0.100      | 0.100 | 0.100 |
| 5        | A12-A13           | 0.100      | 1.500   | 1.855      | 1.871      | 1.732 | 1.879 |
| 6        | A14-A17           | 0.640      | 0.900   | 0.776      | 0.791      | 0.831 | 0.796 |
| 7        | A18-A21           | 2.042      | 0.600   | 0.141      | 0.152      | 0.163 | 0.161 |
| 8        | A22-A25           | 3.400      | 3.400   | 3.846      | 3.970      | 4.027 | 3.938 |

| Weight (lb) | 485.33 | 486.29 | 467.63 | 467.69 | 468.39 | 467.59 |

Table 6 Node displacement of 25 bar space truss

| Direction | X       | Y       | Z       |
|-----------|---------|---------|---------|
| Node      | 1       | 2       | 1       | 2       | 1       | 2       |
| Displacement | 0.1104 | 0.0884 | -0.3499 | -0.3433 | -0.0507 | -0.0463 |
Figure 6 shows the evolution curve of structure weight with iteration step when QPSO and PSO algorithm are used to solve the optimization problem of 25 bar truss structure. QPSO algorithm finds the optimal value of structure weight after 178 iteration steps, while PSO algorithm finds the optimal value after 393 iteration steps. Obviously, the convergence speed of QPSO algorithm is faster than that of PSO algorithm. Table 6 and Figure 7 show the node displacement and bar stress value of 25 bar truss under QPSO optimal design state. It can be seen from table 6 and Figure 7 that the displacement and stress values of the structure meet the design requirements, which indicates that the design scheme of 25 bar truss calculated by QPSO is feasible.

6. Conclusion
In order to solve the optimization problem of truss structure effectively, the QPSO algorithm is constructed by combining quantum angle coding technology and perturbation operator with PSO algorithm. The QPSO algorithm is used to solve the two truss optimization problems, and the results are compared with those of the existing research methods. The analysis result shows that:

1. Both plane and space truss optimization examples show that QPSO algorithm can search for better results than PSO algorithm, which indicates that the QPSO algorithm proposed in this paper has stronger optimization ability.

2. Both plane and space truss optimization examples show that QPSO algorithm takes less iteration steps to converge to the optimal solution compared with PSO algorithm, which indicates that the quantum angle coding used in this paper can accelerate the convergence speed of PSO algorithm.

3. Examples show that the QPSO algorithm can obtain more accurate calculation results, which verifies the effectiveness of QPSO algorithm in solving truss structure optimization problems, and lays a foundation for the study of complex structure optimization problems.

References
[1] M Babaei, M Sheidaii. Optimal design of double layer scallop domes using genetic algorithm [J]. Applied Mathematical Modelling, 2013,37:2127- 2138.
[2] S O Degertekin, M S Hayalioglu. Sizing truss structures using teaching learning based optimization [J]. Computers and Structures, 2013, 119:177-188.
[3] A Kaveh, T Bakhshpoori, E Afshari. An efficient hybrid Particle Swarm and Swallow Swarm Optimization algorithm[J]. Computers and Structures, 2014, 143:40-59.
[4] J S. Angelo, H S Bernardino, H J C Barbosa. Ant colony approaches for multiobjective structural optimization problems with a cardinality constraint [J]. Advances in Engineering Software, 2015, 80: 101-115.
[5] Jiang Yingying, Ji Zhicheng. Hybrid Quantum-Behaved Particle Swarm Optimization for Parameter Identification of DFIG [J]. Journal of System Simulation, 2016, 28(5): 1054- 1062.
[6] ZHANG Lan. Quantum Behaved Particle Swarm Optimization Algorithm Merging Differential Evolution[J]. Computer Simulation, 2016, 33(2): 313-316.

[7] Huang Song, Tian Na, Ji Zhicheng. Study of Modified Particle Swarm Optimization Algorithm Based on Adaptive Mutation Probability[J]. Journal of System Simulation, 2016, 28(4): 874-879.

[8] LIU Ketong. Research on Novel Lattice Boltzmann Method for Fluid-Structure Interaction And Its Application in Bridge Wind Engineering[D]. Harbin, Harbin Institute of Technology, 2015.

[9] ZHANG ZHI-sheng. Quantum-behaved particle swarm optimization algorithm for economic load dispatch of power system[J]. Expert Systems with Applications, 2010, 37: 1800-1803.

[10] M W Dobbs, R B Nelson. Application of optimality criteria to automated structural design, AIAA Journal, 1976, 14 (10):1436-1443.

[11] LI LI-juan. A heuristic particle swarm optimizer for optimization of pin connected structures[J]. Computers and Structures, 2007, 85:340-349.

[12] S.O. Degertekin, An improved harmony search algorithms for sizing optimization of truss structures [J]. Computers and Structures, Computers and Structures, 2012,92-93: 229–241.

[13] R E Perez, K Behdinan. Particle swarm approach for structural optimization[J]. Computers and Structures, 2007,85 (19-20): 1579-1588.

[14] WU JIAN-shyue, CHOW PEI-tse. Steady state genetic algorithms for discrete optimization of trusses [J]. Computers and Structures, 1995, 56 (6): 979-991.

[15] A Kaveh, H Rahami. Analysis, design and optimization of structures using force method and genetic algorithm[J]. International Journal for Numerical Methods in Engineering, 2006, 65: 1570-1584.

[16] A Kaveh, H Rahami. A hybrid HS-CSS algorithm for simultaneous analysis, design and optimization of trusses via force method [J]. Periodica Polytechnica, 2012, 56(2): 197-212.