Co-evolutionary design of discrete structures based on the ant colony optimization*

Tao Zhenwu** and Xiao Renbin

(CAD Center, Huazhong University of Science and Technology, Wuhan 430074, China)

Accepted on February 9, 2007

Abstract In order to optimize the sizing and topology of discrete structures together and resist the combinatorial explosion of the solution space, a co-evolutionary design (CED) method based on ant colony optimization (ACO) for discrete structures is proposed. The tailored ant colony optimization for the sizing of structures (TACO-SS) and the tailored ant colony optimization for the topology of structures (TACO-TS) are implemented respectively. Theoretical analysis shows that the computation complexity of each sub-process in CED based on ACO above is polynomial and it guarantees the efficiency of this method. After the parameter settings in TACO-SS and TACO-TS are discussed, the convergence history of a sub-process is studied through an instance in detail. Finally, according to the design examples of the 10-bar and 15-bar trusses under different cases, the effectiveness of the CED based on ACO is validated and the effects of the loads and the deflection constraints on the co-evolutionary design are discussed.

Keywords: evolutionary design, ant colony optimization, sizing optimization, topology optimization.

The concept "evolution" in biological nature can be used to describe the complex design process[1], in which the design solutions are assumed to be evolvable while the design problems are changeless. In fact, this assumption should be deliberated, because the design problems are also evolvable in the design process, especially for the ill-defined designs. Therefore, Maher et al. proposed the co-evolutionary design (CED) in which both the design solutions and the design problems can be evolved in the complex design processes, and they also developed the CED method and the CED system[2]. The CED is more fit for the complex design problems satisfying the two conditions as follows: ① The design problems change along with the design processes. ② The problems in some stages cannot be solved by traditional mathematical programming methods, but the evolutionary computation methods can deal with them effectively. The optimum design of discrete structures belongs to this kind of problem, in which the sizing, the shape, and the topology of structure need be optimized at the same time.

Optimum design of discrete structures has shown great function in the fields of mechanism, aeronautics, astronautics, ship, and vehicle[3]. The elementary characteristic of the optimum design of discrete structures is the discreteness of design variables, which leads to the discontinuity of objective and constraint functions. The difficulties of the optimum design of discrete structures are that they belong to NP-hard problems whose solution spaces have the characteristics of combinatorial explosion.

The methods of optimum design of discrete structures can be divided into three classes: the accurate algorithms, the approximate algorithms, and the heuristic algorithms. The accurate algorithms include the enumeration algorithm, the implicit enumeration algorithm, the cutting-plane algorithm, the branch and bound algorithm, and the dynamic programming algorithm. The common advantages of these algorithms are that they can find the global optimal solution of the problems whose constraint functions are just the design variables. The common disadvantages are that they can only solve the small-scale problems, in which the number of design variables is no more than 30. Templeman and Yates proposed a method by converting the discrete optimization problem to continuous optimization problem[4]. The methods like this are approximate algorithms. The advantage of the approximate algorithms is that it can estimate the maximum deviation between the feasible solutions and global optimal solution. The simulated annealing (SA) and genetic algorithms (GAs) developed in re-

* Supported by National Natural Science Foundation of China (Grant No. 60474077) and the Excellent Young Teachers Program of Ministry of Education of China

** To whom correspondence should be addressed. E-mail: taozw@163.com

cent years belong to the heuristic algorithms[5]. There are some disadvantages of SA, e.g. low efficiency, no determinate rules to choose the algorithmic parameters, etc. When the GAs are applied to the optimum design of discrete structures, they can approach the global optimal solution effectively though they cannot be mathematically proved to find it certainly. The fatal weakness of GAs is that the times of structural reanalysis are too many when they are applied to the optimum design of discrete structures. Recently, a newly developed heuristic algorithm, ant colony optimization (ACO)[6], is booming. When applied to some standard NP-hard problems, such as traveling salesman problem (TSP), quadric assignment problem (QAP), scheduling problem, and multiple knapsack problem (MKP), ACO has shown excellent performance, e.g. global convergence in given conditions, distributed parallelism, robustness, etc. ACO has been successfully used to solve some complex engineering problems, e.g. the isomorphism identification of mechanisms[7], the layout of satellites[8], and the sizing optimization of trusses[9], etc., which brings new hope to the optimum design of discrete structure.

Optimum design of discrete structures includes four aspects: the sizing optimization, the shape optimization, the topology optimization, and the layout optimization. When the traditional methods are used to optimize the structure, only one aspect above is considered. If a new method can be developed to optimize the sizing and topology together, the solutions may be better than any individual aspect. Optimizing the sizing and topology of discrete structure at one time will meet the ideas and conditions of the co-evolution. Therefore, in this paper, we propose a CED method for the sizing and topology of discrete structures based on ACO by borrowing the Maher's idea of CED and taking the tailored ACO as the evolutionary search algorithm of solution space and problem space.

1 Models of the sizing and topology optimization of discrete structures

1.1 The model of the sizing optimization of discrete structures

For the sizing optimization of discrete structures, the objective function is to minimize the weight. The constraint conditions are stress and deflection, and the design variables are the cross-sectional areas. The mathematical model of the sizing optimization is as follows.

$$
\begin{aligned}
\min \quad & W = \sum_{i=1}^{M}\left(\gamma_i A_i \sum_{j\in G_i} L_j\right) \\
\text{s.t.} \quad & \underline{\sigma}_j \leqslant \sigma_{jl} \leqslant \bar{\sigma}_j; \quad \underline{\delta}_k \leqslant \delta_{kl} \leqslant \bar{\delta}_k \\
& A_i \in \{S_1, S_2, \cdots, S_{NS}\} \\
& i = 1, 2, \cdots, M; \quad j \in G_i \\
& l = 1, 2, \cdots, NL; \quad k = 1, 2, \cdots, NN
\end{aligned} \tag{1}
$$

where $\boldsymbol{A}=[A_1, A_2, \cdots, A_M]^{\mathrm{T}}$ is a set of design variables, M is the number of the member groups in structures, and A_i is the cross-sectional area of the member groups i; A_i belongs to the discrete set $S=\{S_1, S_2, \cdots, S_{NS}\}$, NS is the number of the elements in the set, and $S_1 < S_2 < \cdots < S_{NS}$; γ_i is the material density of the member group i, G_i is the set of serial numbers of members belonging to the member group i, L_j is the length of the member j, σ_{jl} is the stress of the member j under the operating condition l and δ_{kl} is the deflection of the constrained node k under the operating condition l; $\underline{\sigma}_j$ and $\bar{\sigma}_j$ are the lower and upper bounds of stress; $\underline{\delta}_k$ and $\bar{\delta}_k$ are the lower and upper bounds of deflection; NL is the number of the operating condition, and NN is the number of the deflection constraints.

1.2 The model of the topology optimization of discrete structures

The topology optimization of discrete structures is to find the optimal connecting relation between the nodes under the given node set, the bearings, and the external loads. The mathematical model of the topology optimization is

$$
\begin{aligned}
\min \quad & W = \sum_{i=1}^{M}\left(\gamma_i A_i \sum_{j\in G_i} Z_{ij} \cdot L_j\right) \\
\text{s.t.} \quad & f = 1 \\
& \underline{\sigma}_j \leqslant \sigma_{jl} \leqslant \bar{\sigma}_j; \quad \underline{\delta}_k \leqslant \delta_{kl} \leqslant \bar{\delta}_k \\
& A_i \in \{S_1, S_2, \cdots, S_{NS}\} \\
& i = 1, 2, \cdots, M; \quad j \in G_i \\
& Z_{ij} \in \{0, 1\}
\end{aligned} \tag{2}
$$

where $\boldsymbol{Z}=[Z_{11}, Z_{12}, \cdots, Z_{1j}, \cdots, Z_{ij}, \cdots, Z_{Mj}, \cdots]^{\mathrm{T}}$ is the topology design vector of the members and Z_{ij} is the logical variable (0 or 1) denoting whether the member j in the member group i is in the structure. If $Z_{ij}=1$, it means the member is in the structure. If

$Z_{ij}=0$, it means the member is not in the structure. In the topology optimization, the generated structure needs to be stable and cannot be a movable mechanism. A logical variable f is used to describe the movability constraint in the model of topology optimization. If $f=1$, it means the topology of the structure is stable. If $f=0$, it means the topology of the structure is movable. The meanings of other expressions are the same as (1).

Compared with the sizing optimization of structures, the topology optimization needs to determine not only the sizes of the cross-sectional areas, but also the connecting relations of members between the nodes. Strictly, the logical variable or integer variable should be adopted to describe whether there is a member connecting two nodes. Generally, the strategy of member deleting is often used in the existing methods for the topology optimization of structures and then the topology optimization can be transformed to the sizing optimization. However, this kind of transformation will lead to the singular solution or the case that some members cannot be deleted. When the values of the cross-sectional areas in the base structure are discrete, the differential-based methods cannot work. In view of these difficulties, the ACO, a method that is good at solving the problems of discrete optimization, is used to deal with the sizing and topology optimization in this paper. Furthermore, the connecting relations between the members are just described by logical variables, which keep the original character of the topology optimization. Considering the mutual influence between the sizing and the topology of the structures, the CED method based on ACO for discrete structures is proposed, which can drive the co-evolution of the sizing and topology of discrete structures.

2 The framework of the co-evolutionary design for discrete structures

2.1 The fundamental principle of the co-evolution

Denote the problem space of design as P and solution space as S. The process of the co-evolutionary design is described in Fig. 1. There are some basic concepts in the process of the CED.

(1) There are two different search spaces in the search process: problem space and solution space.

(2) The two spaces influence each other in the search process.

(3) The process of the CED is along the horizontal axis in Fig. 1.

① The evolution of the problem space: Evolving from P_1 to P_2, …, P_i, …, and so on.

② The evolution of the solution space: Evolving form S_1 to S_2, …, S_i, …, and so so.

Here, i can be taken as the process of the evolution along with time, the variety of design focus, or the variety of the granularity of the design problem.

(4) The arrows on the diagonal lines denote the processes of solution driven by objectives, where the downward arrows of the diagonal lines represent from design problems to design solutions and the upward arrows of the diagonal lines represent from the design adjustments to redefinitions of the design problems. These exploration processes describe such an evolutionary design mechanism that there are two evolving spaces: the problem space and solution space, and the evolution in one space is performed under the influence of the other space, which is the so-called CED model.

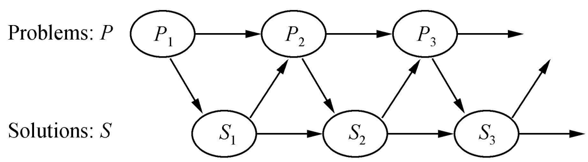

Fig. 1. The co-evolutionary design model.

2.2 The procedures of the co-evolutionary design method for the sizing and topology of discrete structures

It can be known from the mathematical models (1) and (2) that the sizing optimization of discrete structures is carried out under the given topology and the model of the topology optimization is varied along with the sizing variables of the structures. Therefore, different topologies determine different results of the sizing optimization, and the results of the sizing optimization influence the process of the topology optimization in reverse. The sizing and topology optimization are both the independent and interacted processes. According to the model of co-evolutionary design, the evolution of the topology of structures can be looked upon as the exploration process in the problem space and the evolution of the sizing of structures as the exploration in the solution space. The procedures of the CED for the discrete structures can be described in Fig. 2.

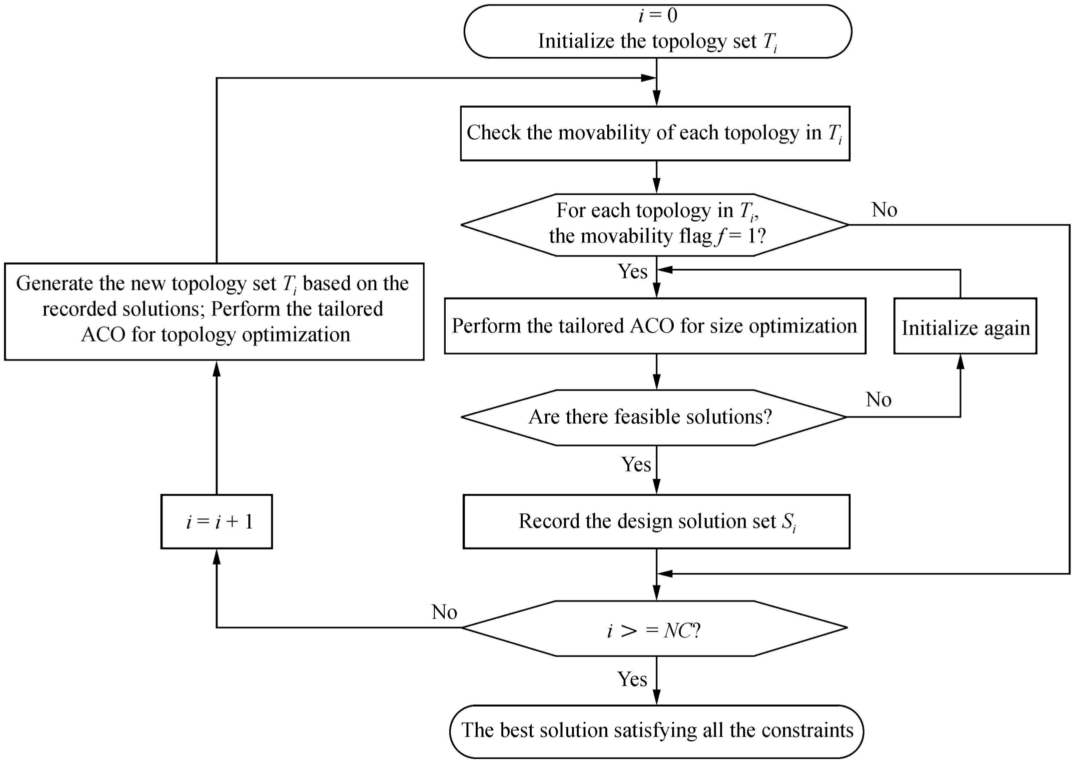

Fig. 2. The flow chart of the co-evolutionary design method for the sizing and topology of discrete structure.

3 The co-evolutionary design method for discrete structure based on ant colony optimization

3.1 Tailored ant colony optimization for the sizing of structures

3.1.1 The model transformation of the sizing optimization of structures

When ACO is applied to the optimum design of discrete structures, it has the similar advantages of GAs, e.g. the discrete design variables, the open formats to express the constraint, and suitable for the situation of multiple loads. However, ACO does not need the explicit relations between the objective functions and the constraints. It penalizes the objective functions corresponding to a set of design variables to reflect any violation of the constraints.

When a discrete structure is optimized, the objective is usually to minimize the cost of the structure satisfying the constraints of the stresses and node deflections. The cost of a structure is a function of the volume of the materials needed. When the topology of a structure is determined, the cost of the structure is directly related to the cross-sectional areas between the members.

In order to apply the ACO to the sizing optimization of discrete structures, the model of this problem is transformed to a kindred-TSP, in which the conception of "path" used in traditional TSP is redefined. This kind of transformation can be realized according to the following steps.

① In the sizing optimization, there are multiple choices from member i to cross-sectional area j. But in the traditional TSP, there is only one edge between cities i and j.

② In the sizing optimization, the sequence of the members in the structure is less important. But in the traditional TSP, the sequence of the cities visited makes up the solution.

③ In the sizing optimization, the design solutions obtained need not be feasible. But in traditional TSP, the tabu lists are needed to guarantee the feasibility of the solutions.

Fig. 3 explains the meaning of the virtual edge in the structure. The "length" of each virtual edge is determined by the volume of the member. Edge (i, j) means that the cross-sectional area j is chosen for the member i. If the number of all the possible cross-sectional areas is NS, there are NS different volumes from V_1 (corresponding to the minimum area) to V_{NS} (corresponding to the maximum area). When an ant travels from i to j, it will choose one of the virtual edges. The shortest edge is just the minimum volume.

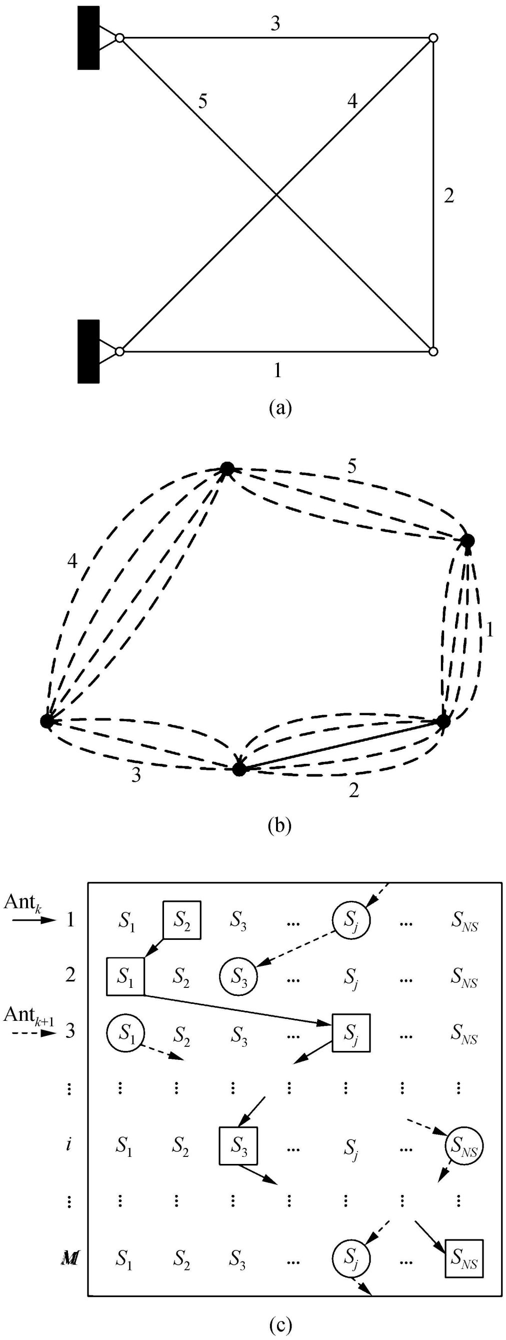

Fig. 3. The choosing paths of ants in a discrete structure. (a) A structure with 5 members and 4 nodes; (b) possible virtual paths of the structure; (c) the choosing cross sectional areas of ants for every member.

The topology of the virtual path in discrete structures is different from that in the traditional TSP. In the latter, each ant visits a city only once in one circle. However, when the ACO is applied to the sizing optimization of discrete structures, the ant does not complete a path until it has chosen the cross-sectional area for every member in a structure. Fig. 3(a) gives a structure with 5 members and 4 nodes. Fig. 3(b) describes that there are NS virtual paths representing different volumes for every member. Fig. 3(c) gives the abridged general view in which the ants choose the cross-sectional areas for every member in the structure. The sequence that ant k visits the members is $\{1, 2, 3, \cdots, i, \cdots, NS\}$ and the corresponding virtual path is $\{S_2, S_1, S_j, \cdots, S_3, \cdots, S_{NS}\}$. In the large-scale structures, the members are usually grouped and the members in a group share a same cross-sectional area.

When the ACO is applied to solve TSP, the feasibility of the solution is controlled by the tabu list of the ant. So the feasibility of a solution obtained by an ant in one circle need not be determined. When the ACO is applied to the sizing optimization of the discrete structure, a set of the cross-sectional areas chosen by an ant may define such a structure that its stresses and/or node deflections violate the allowable limitations and lead to an unfeasible design. Therefore, after each ant obtains a complete solution, the feasibility of it should be checked. In order to deal with the unfeasibility of the solution, a penalty function can be added to the weight of the structure. The penalized weight of the structure will help the ACO focus the search on the structure that has less weight and satisfies all the design constraints. This is the key difference from the ACO for the traditional TSP. The ACO for the sizing optimization of discrete structures is designed in Section 3.1.2, which is mainly based on Ref. [9].

3.1.2 Algorithm design

Step 1. The initialization of the algorithm.

The first step of TACO-SS is to compute the "length" (the volume of the member) of each virtual edge and set an initial value to the pheromone on the edge that the ants will possibly visit.

The "length" of the virtual edge from the member i to the cross-sectional area j is defined as

$$d_{ij} = L_i \circ A_j \tag{3}$$

Since the objective is to be minimized, the ants prefer the shorter edges. This kind of preference is represented by the visibility η_{ij}.

$$\eta_{ij} = 1/d_{ij} \tag{4}$$

Before the ants choose the cross-sectional areas for each member, the pheromone on every edge is initialized by

$$\tau_0 = 1/W_{\min} \tag{5}$$

where $W_{\min}$ is the weight of the structure, in which each member is assigned the minimum nonzero cross-sectional area. $W_{\min}$ can be computed by

$$W_{\min} = \sum_{i=1}^{M} \gamma_i L_i A^l \tag{6}$$

where $A^l = \min\{S_1, S_2, \cdots, S_{NS}\} = S_1$, and the constraints of stress σ and the deflection δ need not be satisfied.

The members in the structure are numbered from 1 to M and each ant in the colony is assigned a number i, being the start point of its journey, where $i = 1, 2, \cdots, M$.

Step 2. Ants choose the cross-sectional areas for the members and the pheromone is locally updated.

Then the first ant chooses the cross-sectional area for the member standing on in the structure according to the decision process as follows. Firstly, the decision table $a_{ij}(t)$ is computed as

$$a_{ij}(t) = \frac{[\tau_{ij}(t)] \cdot [\eta_{ij}]^{\beta}}{\sum_{u=1}^{NS} [\tau_{iu}(t)] \cdot [\eta_{iu}]^{\beta}} \tag{7}$$

where β is the parameter controlling the relative importance between the density of pheromone and the visibility of the edge. On time t, the possibility $p_{ij}^k(t)$, according to which the ant chooses the cross-sectional area j for the member i, is computed by

$$p_{ij}^k(t) = \frac{a_{ij}(t)}{\sum_{u=1}^{NS} a_{iu}(t)} \tag{8}$$

After the ant in the colony chooses the cross-sectional area j for the member i, the pheromone on the edge (i, j) is decreased according to the local updating rule of the pheromone to promote the further exploration of the search process and avoid the premature of the algorithm:

$$\tau_{ij}(t+1) = \xi \cdot \tau_{ij}(t) \tag{9}$$

where ξ is an adjustable parameter between 0 and 1 representing the maintenance of the pheromone.

Thereafter, the second ant chooses the cross-sectional area for each member in the structure and uses the local updating rule of pheromone again. This process continues until each ant in the colony has chosen an area for the starting member of it, and the first iteration of every path is completed.

Then each ant continues to deal with the next member in the structure. The numbering scheme of the members in the structure defines the sequence according to how each ant chooses the cross-sectional area. If an ant is standing on the member M, the next member to be visited will be the member 1. Each ant chooses the cross-sectional area for the member $(i+1)$ in the structure, the local updating rule of the pheromone is applied to the edges they passed, and then the next member is dealt with like the above. This decision process of the ants continues until M iterations are completed, when each ant in the colony has chosen a cross-sectional area for each member in the structure. Since the ants travel according to the sequence of the members in the structure, it is unnecessary to set a tabu list to prevent an ant from visiting a member more than once as the ACO solving the traditional TSP does.

Step 3. The feasibility analysis and the penalty of the solutions and the global updating rule of pheromone.

After each ant chooses a cross-sectional area for each member in a structure, a set of solutions is generated. Then the stress of each member and the deflection of each node in the structure can be analyzed by computation. The results are compared with the design constraints to judge whether the solution is feasible. Denote σ_i as the stress of the member i in the structure, $\sigma^{\pm}$ as the maximum allowable stress where "$-$" represents the compressive stress limit and "$+$" represents the tension stress limit, δ_c as the deflection of the node c, and δ as the allowable deflection. The stress penalty to the member i can be computed as

$$\Phi_{\sigma}^i = \begin{cases} 0 & \sigma^- \leqslant \sigma_i \leqslant \sigma^+ \\ \dfrac{\sigma_i - \sigma^{\pm}}{\sigma^{\pm}} & \sigma_i > \sigma^+ \text{ or } \sigma_i < \sigma^- \end{cases} \tag{10}$$

Then the total stress penalty to the solution generated by the ant k is

$$\Phi_\sigma^k = \sum_{i=1}^{M} \Phi_\sigma^i \tag{11}$$

If the deflections of the node c in the directions of x, y and z are denoted as Φ_δ^{cx}, Φ_δ^{cy} and Φ_δ^{cz}, respectively, the deflection penalty to the node c is

$$\Phi_\delta^{cx,cy,cz} = \begin{cases} 0 & \delta_{cx,cy,cz} \leqslant \delta \\ \dfrac{\delta_{cx,cy,cz} - \delta}{\delta} & \delta_{cx,cy,cz} > \delta \end{cases} \tag{12}$$

Then the total deflection penalty to the structure generated by the ant k is

$$\Phi_\delta^k = \sum_{c=1}^{N} (\Phi_\delta^{cx} + \Phi_\delta^{cy} + \Phi_\delta^{cz}) \tag{13}$$

where N is the number of nodes in the current structure.

The total penalty Φ^k to the structure generated by the ant k is the function of the sum of the stress penalty and deflection penalty, which can be defined as

$$\Phi^k = (1 + \Phi_\sigma^k + \Phi_\delta^k)^\varepsilon \tag{14}$$

where ε is the penalty exponential and $\varepsilon > 0$.

The penalized total weight W^k of the structure generated by the ant k can be computed as

$$W^k = \Phi^k w^k \tag{15}$$

where w^k is the actual weight of the structure generated by the ant k.

The penalized weight will be used in the global updating rule of the pheromone. The rule adopted in this paper is similar to that in AS_{rank}[10]. Then, according to the values of the penalized weights of the structures, the levels of the corresponding ants are arranged. The ant generating the structure with the minimum weight in all the circles of the algorithm is called the elitist ant. The minimum penalized weight W^+ is adopted to compute the total pheromone added to the path that the elitist ant traveled through:

$$\Delta\tau_{ij}^+ = \frac{1}{W^+} \tag{16}$$

Then the pheromone on the paths traveled by the $\lambda (1 \leqslant \lambda < NA)$ ants with higher levels are globally updated according to the process as follows. Supposing that the level of an ant is $\mu (1 \leqslant \mu \leqslant \lambda)$, the total pheromone added to the edge (i, j) is

$$\Delta\tau_{ij}^\mu = (\lambda - \mu)\frac{1}{W^\mu} \tag{17}$$

where W^μ is the penalized weight of the structure generated by this ant.

Since there are possibly the same edges traveled by different ants (the same "member-area" combination in different structures), the pheromone added to the edge (i, j) by all these ants should be summed up, that is

$$\Delta\tau_{ij}^r = \sum_{\mu=1}^{\lambda} \Delta\tau_{ij}^\mu \tag{18}$$

The global updating rule of the pheromone in this algorithm is different from that in the AS_{rank}. When this global updating rule based on levels in Eq. (19) is used, the term representing the variety of the pheromone needs to be multiplied by the factor ρ, where $(1-\rho)$ means the evaporating rate of the pheromone

$$\tau_{ij}(t+n) = (1-\rho)\tau_{ij}(t) + \rho(\lambda\Delta\tau_{ij}^+ + \Delta\tau_{ij}^r) \tag{19}$$

Till now, a circle of all the ants is completed. If the best solution found does not change any more in some continuous circles or the time of circles reaches the maximum number NC, the algorithm is considered to have converged to a solution and goes to Step 4 for local search. Otherwise, the algorithm goes to Step 2 and continues to search globally.

Step 4. The process of local search in the later stage of the algorithm

In order to improve the quality of the solution, the local search needs to be performed. Firstly, the optimal solution $[O_1, O_2, \cdots, O_M]$ obtained by the elitist ant is taken as the benchmark. For every member i in the structure, the global search space $\{S_1, S_2, \cdots, S_{NS}\}$ is reduced to a local search space that consists of the areas a little bigger and smaller than O_i. Then the initial values of the pheromone in the local search space are set to be τ_0 while the others are set to be 0. The size of the local search space around the elitist ant varies with the dimension of the problem. Generally, 10% of the global search space is enough to obtain the improved solutions and decrease the computation time in all.

Step 5. Output the computation results

The final solution found by the elitist ant is just the optimal solution of the problem. The feasibility of the solution then is checked by structure analysis. If it is feasible, output the computation results (the total weight, the cross-sectional area of each member, the stress of each member, and the deflection of each node). Otherwise, the algorithm fails this time and needs to be rerun.

3.2 Tailored ant colony optimization for the topology of structures

3.2.1 The analysis and transformation of the problem of the topology optimization of discrete structures

In order to induce more topologies from the base structure, it is generally realized by deleting some members, that is, let the cross-sectional areas of the deleted members be 0. However, this kind of operation will not only lead to the singularity of the stiffness matrix of the structure and cause the trouble of computation, but also make it difficult to restore the deleted members. In order to avoid these difficulties, we have also tailored a kind of ACO for the topology of structures (TACO-TS).

The topology optimization of structures in the problem space can be described as the weighted graph in Fig. 4. Let the vertex 0 represent the base structure, the vertexes 1, 2, ⋯, M' represent the members in the structure, and the weight on the edge from vertex 0 to the arbitrary vertex j ($j \neq 0$) is d_j. If ant chooses the jth edge, it means that the member j is chosen into the current topology. The structure (vertex 0) can be seen as the start point of the ants foraging, every vertex j ($j \neq 0$) can be seen as the food source provided for the ants, and d_j can be seen as the distance from the start point of the ants foraging to the food sources. When the problem of the topology optimization of the discrete structure is transformed by the operations above, we can tailor an ACO algorithm to solve it.

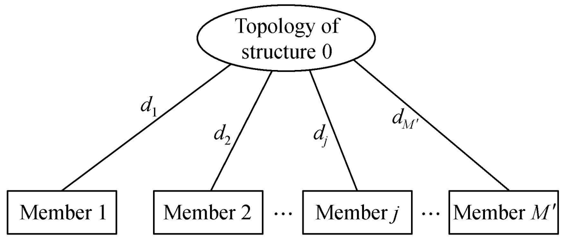

Fig. 4. The weighted graph description of the topology optimization of structures.

3.2.2 Algorithm design

Step 1. The initialization of the algorithm

In the ACO for the traditional TSP, the visibility of the ant is defined as $\eta_j = 1/d_j$. In the TACO-TS, we directly define the level of the stress of each member as the visibility. The visibility from the node 0 to the node j ($j=1, 2, \cdots, M'$) is

$$\eta_j = \frac{\sum_{l=1}^{NL}\sum_{k\in G_i} |F_{kl}|}{\sum_{j=1}^{M'}\sum_{l=1}^{NL}\sum_{n\in G_i} |F_{nl}|} \tag{20}$$

The level of the stress of each member can be computed by analyzing the structure obtained by TACO-SS.

Firstly, the pheromone on each edge is initialized. The initialization rule is

$$\tau_0 = 1/W'_{\min} \tag{21}$$

where $W'_{\min}$ is the minimum weight of the structure obtained by the sizing optimization in the previous generation of the CED.

Each member in the structure is numbered from 1 to M'. Assign stochastically an edge j ($j=1, 2, \cdots, M'$) to the NA ants in the colony as their start points.

Step 2. Choose the members for the current topology and update the pheromone locally

The first ant chooses the member for the current topology according to the decision process as follows. Firstly, the decision table $a_{ij}(t)$ is computed by

$$a_{ij}(t) = \frac{[\tau_{ij}(t)] \circ [\eta_{ij}]^{\beta}}{\sum_{u=1}^{M'}[\tau_{iu}(t)] \circ [\eta_{iu}]^{\beta}} \quad (i=0, j=1, 2, \cdots, M') \tag{22}$$

where β is also the parameter controlling the relative importance between the density of pheromone and visibility of the edge. On time t, the possibility $p_{ij}^k(t)$, according to which the ant chooses the member j for the current topology i, is computed by

$$p_{ij}^k(t) = \frac{a_{ij}(t)}{\sum_{u=1}^{M'} a_{iu}(t)} \quad (i=0, j=1, 2, \cdots, M') \tag{23}$$

Because of the particularity of the problem of the topology optimization, each ant needs only one step from the start point to an arbitrary food source between 1 and M'. Let r be a random value and $r \in [0, 1]$. Whether the member j is chosen for the current topology can be judged by the following rule

$$x_j^k = \begin{cases} 1 & \text{if } r \leqslant p_{ij}^k(t) \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

When the ants choose each member for the current topology, the process of TACO-TS runs continuously. After an ant makes a choice, in order to promote the further exploration and avoid the premature of the algorithm, the local updating rule of the pheromone is applied to decrease the density of the pheromone on the corresponding edge.

$$\tau_{ij}(t+1) = \begin{cases} \xi \circ \tau_{ij}(t) & \text{if } x_j^k = 1 \\ \tau_{ij}(t) & \text{if } x_j^k = 0 \end{cases} \tag{25}$$

where ξ is an adjustable parameter between 0 and 1 representing the maintenance of the pheromone, which is similar to that in Section 3.1.2.

Then the second ant chooses the first member for its current topology and the local updating rule of the pheromone is used again. This process continues until each ant in the colony has judged its first member and the first iteration of the algorithm is completed.

Thereafter, each ant continues to judge the next member in the structure. The numbering scheme of the member in the base structure defines the sequence according to how the ants judge the members. If the numbering of a member that an ant is judging now is M', the numbering of the member that the ant will judge next is 1. Then each ant judges the member $(j+1)$ and the local updating rule of the pheromone is used. This decision process of the ants continues until M' iterations are completed and each ant in the colony has chosen the members for its corresponding topology. Since the ants always travel according to the numberings of the members in the base structure, it is unnecessary to set a tabu list to prevent an ant from judging the same member more than one time.

Step 3. The feasibility analysis and the penalty of the solutions and the global updating rule of pheromone

When NA ants have chosen the members for their corresponding topologies, NA solutions are generated and the feasibility of them needs to be checked. Firstly, the movability condition of each solution is checked to judge whether it becomes a mechanism, which can be realized by checking the singularity of the stiffness matrix of the topology.

If the topology generated by ant k violates the movability condition, the penalty of the pheromone to each edge of this topology can be computed as

$$\Delta\tau_{ij}^k = \begin{cases} -\dfrac{1}{W_k^m} & \text{if } x_j^k = 1 \\ 0 & \text{if } x_j^k = 0 \end{cases} \tag{26}$$

where W_k^m is the weight of the structure corresponding to this topology.

Then the total penalty of the pheromone to each member on the base structure caused by the unfeasibility of the topology is

$$\Delta\tau_{ij}^m = \sum_{k \in \text{unfeasible set}} \Delta\tau_{ij}^k \tag{27}$$

If the constraints cannot be satisfied, the objective function will be penalized based on the method in Section 3.1.2 and this will also reflect on the global updating of the pheromone. At first, compute the penalized weight W^k according to Eqs. (10)—(15). Then compute the total pheromone $\Delta\tau_{ij}^+$ released by the elitist ant on its path and the total pheromone $\Delta\tau_{ij}^r$ released by the r leveled ant on their paths.

The pheromone on the base structure is globally updated as

$$\tau_{ij}(t+n) = (1-\rho)\tau_{ij}(t) + \rho(\lambda\Delta\tau_{ij}^+ + \Delta\tau_{ij}^r) + \Delta\tau_{ij}^m \tag{28}$$

Till now, a circle of all the ants is completed. If the best solution found in some circles does not change any more or the time of circles reaches the maximum number NC', then the TACO-TS is considered to have converged to a solution and the process of the algorithm goes to Step 4. Otherwise, the process of the algorithm goes to Step 2 for further exploration.

Step 4. Output the computation results

Analyze the feasibility of the solution found by the topology optimization. If the solution is feasible and the weight of its corresponding structure is less than that in the previous stage of the CED, output the computation results (the total weight of the structure, the topology of the structure, the cross-sectional area of each member, the stress of each member, and the deflection of each node). Otherwise, the topology optimization of this time fails and needs to be rerun again.

3.3 The analysis of computation complexity

For the convenience of analysis, only one operating condition ($NL=1$) and only one member in each member group (only one element in set G_i) is considered, which obviously has little effect on the computation complexity of the algorithm in theory.

For the TACO-SS, if the number of the members of a structure is M and the number of the cross-sectional areas is NS, the computation complexity of the initialization in Step 1 is $O(M \cdot NS)$. If the number of the ants is NA, the computation complexity of Step 2 is $O(NA \cdot M \cdot NS)$ when Step 2 is in the global search stage and it is $O(NA \cdot M \cdot NS')$ when Step 2 is in the local search stage. Suppose there are N nodes and NR constrained nodes in the structure, then the computation complexity of Step 3 is $O(NA \cdot (NR \cdot N + N^3 + M \cdot NR))$. Since $NR \leqslant N$ and $NR \cdot N \leqslant N^3$, the computation complexity of Step 3 can be represented as $O(NA \cdot (N^3 + M \cdot NR))$. Step 4 and Step 3 are two mutual replaceable stages and the computation complexity of Step 4 can be also computed as that of Step 3. The computation complexity of Step 5 is $O(N^3 + M \cdot NR)$. Thus, the computation complexity of one circle in TACO-SS is $O(NA \cdot (M \cdot NS + N^3 + M \cdot NR))$.

For the TACO-TS, if the number of the members in the base structure is M and that in the new topology evolved from base structure is M', the computation complexity of Step 1 is $O(M' + NA)$ and that of Step 2 is $O(NA \cdot (M' + NS))$. In Step 3, the computation complexity of the pheromone updating is $O(NA \cdot (M' + N^2))$ and that of the analysis and penalty of the solutions is $O(NA \cdot (NR \cdot N + N^3 + M' \cdot NR))$. The computation complexity of Step 4 is $O(N^3 + M' \cdot NR)$. Since $M' \leqslant M$ and $NR \leqslant N$, the computation complexity of one circle in TACO-TS is $O(NA \cdot (NS + N^3 + M \cdot NR))$.

In summary, the computation complexity of the two tailored ACOs is polynomial and it guarantees the efficiency of the CED method for discrete structures based on ACO.

4 Numerical experiments and design examples

4.1 The parameter setting and performance analysis of the algorithms

For the TACO-SS, the parameter β controlling the visibility is set to be 0.2 since it influences the choices of the cross-sectional areas of the structure[9]. Because ξ has effect on the balance between the "exploitation" and the "exploration" of the algorithm, it is set to be 0.67 in the first stage of the algorithm to enhance the "exploration" and be 0.33 in the second stage of the algorithm to enhance the "exploitation". The penalty exponent ε influences not only the balance between the exploitation and the exploration, but also the feasibility of the solutions. So ε is set to be 1 in the first stage of the algorithm to enhance the "exploitation" and be 4 in the second stage to increase the penalty to prevent the algorithm from converging to an unfeasible solution. The setting of the other parameters is that ρ is equal to 0.5, the number of the ants NA is equal to 100, and the number of the leveled ants λ is generally 10% of NA[9]. In order to maintain enough space for local search, if $10\% \cdot NA > 3$, $\lambda = \lfloor 10\% \cdot NA \rfloor$; otherwise, $\lambda = 3$.

For the TACO-TS, the parameter β is set to be 0.1 to keep the variety of the topologies. When β decreases, the probability to choose the shorter edge (member with the bigger stress) decreases and there may be more kinds of topology emerging. The values of ξ, ε, ρ and NA are the same as those in TACO-SS.

According to many computation experiments, we find that 100 iterations of the TACO-SS and TACO-TS are enough to obtain satisfactory solutions.

We take a structure of 10-bar truss with 6 nodes and 10 members[11] as an example to discuss the converge process of the TACO-SS. The structure is sketched in Fig. 5. The modulus of elasticity of every member is $E = 6.897 \times 10^6$ N/cm^2, the mass density of the material is $\gamma = 2.7 \times 10^{-3}$ kg/cm^3, the allowable stresses of all the members are ± 17240 N/cm^2, and the allowable deflection of every movable node in direction y is 5.08 cm. The set of cross-sectional areas is $S = \{0.645, 1.29, 1.935, \cdots, 219.3\}$ cm^2, which has 340 discrete values in total. The length of

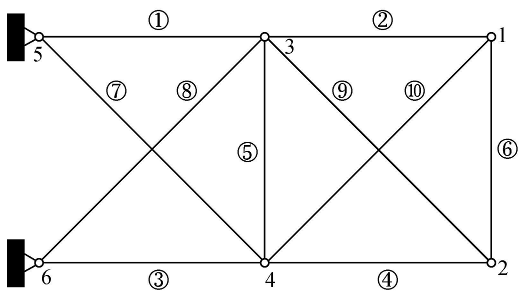

Fig. 5. The structure of 10-bar truss.

each member in the structure is either 914.4 cm or 1292.7 cm.

Fig. 6 describes the convergence history of the TACO-SS when the structure in Fig. 5 is optimized. The continuous line represents the average value of the penalized weights in each iteration. The penalized weight varies violently in the first phase, which reflects the exploration of the algorithm and prevents it from premature. But in the second phase, since the search space is reduced, the exploitation of the ants is performed, which is helpful to find the global feasible solution. The symbol "×" represents the best feasible solution found in each iteration. Because of the penalty to the weight of the structure in the first phase, there are no feasible solutions found in some iterations, which is different from the ACO for the traditional TSP. But in the second phase, the feasibility and quality of the solutions are improved. So we can conclude that the TACO-SS can achieve a good balance between the exploration and the exploitation.

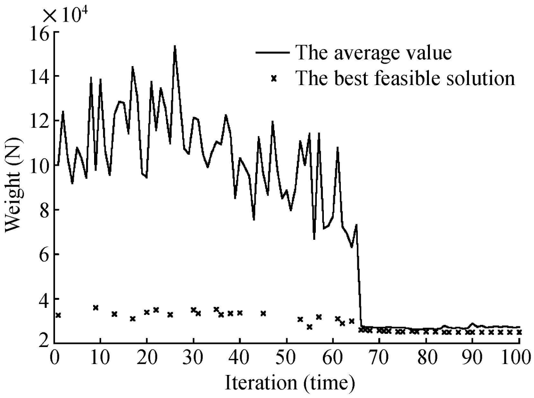

Fig. 6. The convergence history of the TACO-SS.

4.2 The effects of loads on the co-evolutionary design of discrete structures

In order to discuss the effects of loads on the co-evolutionary design of discrete structures, a 10-bar truss is taken as an example to find the topology and cross-sectional areas that make the weight of the structure minimized under different loads. Though the co-evolutionary design method proposed in this paper is suitable for multiple operating conditions, the problems with only one operating condition is considered here to simplify the computation. If multiple operating conditions are considered, only the stress constraints of the members need to be added. The computation comparison of two cases with different loads is given below.

The base structure of a 10-bar truss with 6 nodes and 10 members[12] is sketched in Fig. 5. The modulus of elasticity of every member is $E = 2.2\times10^7$ N/cm^2, the mass density of the material is $\gamma = 7.85\times10^{-3}$ kg/cm^3, the allowable stresses of all the members are ±30000 N/cm^2, and the allowable deflection of every movable node in direction y is 0.4 cm. The set of cross-sectional areas is $S = \{20, 50, 80, 100, 150, 200, 250, 300, 350, 400, 500, 600\}$ cm^2, which has 12 discrete values in total. The length of each member in the structure is either 900 cm or 1272.8 cm.

Scheme ① Apply the downward 5.0×10^5 N concentrated force to nodes 2 and 4.

The optimum topology is described in Fig. 7 (a). The weight of the structure corresponding to this topology is $W^* = 34319$ kg and the combination of cross-sectional areas is $A^* = [600, 150, 600, 0, 600, 150, 500, 600, 600, 250]^T$, where $A_4 = 0$ means that the node 4 is deleted in the process of co-evolutionary design.

Scheme ② Apply the downward 5.0×10^5 N concentrated force only to node 2.

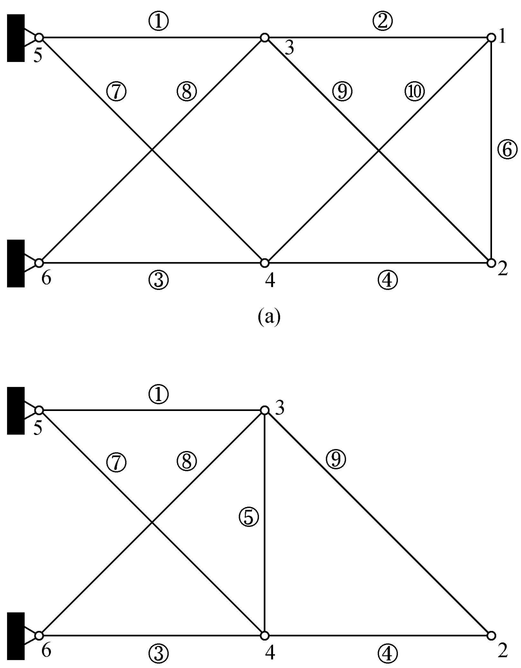

Fig. 7. The optimum structure obtained in Scheme ① and Scheme ②. (a) The optimum topology of structure in load case ①; (b) the optimum topology of structure in load case ②.

The optimum topology is described in Fig. 7

(b). The weight of the structure corresponding to this topology is $W^{*}=24446$ kg and the optimal combination of cross-sectional areas is $\boldsymbol{A}^{*}=[600, 0, 500, 500, 50, 0, 80, 600, 600, 0]^{\mathrm{T}}$, where $A_{2,6,10}=0$ means the nodes 2, 6, and 10 are deleted in the process of co-evolutionary design.

From the computation results above, it can be seen that the optimum topology and cross-sectional areas of each member are different in the different cases of loads.

4.3 The effects of deflection constraints on the co-evolutionary design of structures

In order to discuss the effects of deflection constraints on the co-evolutionary design of structures, a 15-bar truss[11] is taken as an example to find the topology and cross-sectional areas that make the weight of the structure minimized under different deflection constraints. The base structure of a 15-bar truss with 8 nodes and 15 members is sketched in Fig. 8(a) and the group relations of the members in the 15-bar truss are listed in Table 1. The modulus of elasticity of every member is $E=6.897\times10^{6}$ N/cm^{2}, the mass density of the material is $\gamma=2.7\times10^{-3}$ kg/cm^{3}, the allowable stresses of all the members are ±17240 N/cm^{2}, and the allowable deflection of nodes 2 and 4 in direction y is ±5.08 cm. There are two operating conditions: $P_{3y}=P_{5y}=P_{7y}=-4.45\times10^{5}$ N, $P_{4y}=P_{6y}=P_{8y}=-4.45\times10^{5}$ N. The set of cross-sectional areas is $S=\{6.45, 9.68, 22.58, 32.26, 45.16, 70.97, 83.87, 103.23, 129.03, 161.29, 193.55\}$ cm^{2}.

Table 1. The connection relations of the cross-sectional areas in the 15-bar truss

Group No.	1	2	3	4	5	6	7	8
Member No.	1, 2	3, 4	5, 6	7, 8	9, 10	11, 12	13, 14	15

(i) There is no deflection constraint. The topology sketched in Fig. 8(b) is just the optimum one. The design solution is $W^{*}=355.84$ kg and $\boldsymbol{A}^{*}=[70.97, 32.26, 22.58, 6.45, 6.45, 6.45, 6.45, 22.58]^{\mathrm{T}}$.

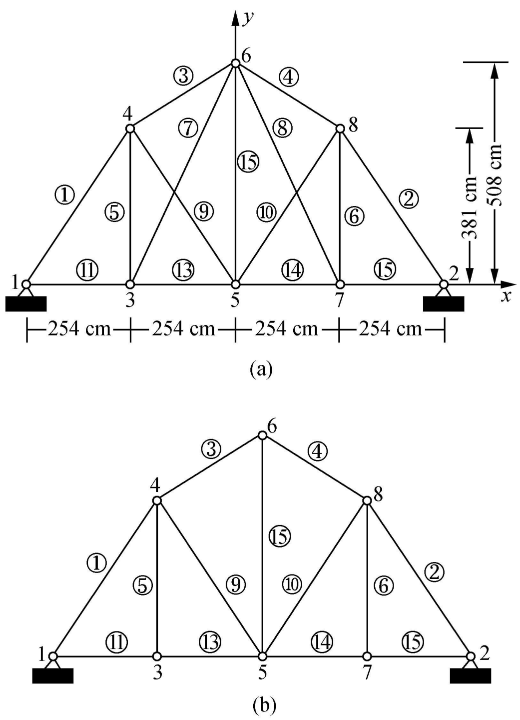

Fig. 8. The base structure of 15-bar truss. (a) The base structure of the 15-bar truss; (b) the optimum topology of the 15-bar truss under the deflection constraint (ii).

(ii) The deflection constraint of node 5 in direction y is 1.524 cm. The optimum topology is sketched in Fig. 8(b). The design solution is $W^{*}=474$ kg and $\boldsymbol{A}^{*}=[83.87, 45.16, 32.26, 0, 6.45, 6.45, 6.45, 70.97]^{\mathrm{T}}$, where $A_4=0$ means that the member group 4 (member 7 and 8) is deleted in the process of the co-evolutionary design.

When the optimum topologies obtained in (i) and (ii) are compared, it can be seen that the deflection constraints directly control the optimum topology. Therefore, the deflection constraints should be remained when design, otherwise the optimum ones may be lost.

5 Conclusion

The sizing and topology optimization design is very important for engineering applications. In order to deal with the difficulties of these problems, e.g. the discreteness of the design variables, the discontinuousness and non-vertex of the solution space, the combinatorial explosion in the design process, and the local optimality of the solutions, the CED method based on ACO for discrete structures is proposed. This new method can make the sizing and topology of the discrete structures optimized together. The tailored ant colony optimization for the sizing of structures (TACO-SS) and the tailored ant colony optimization for the topology of structures (TACO-TS) are designed respectively.

The characteristics of the CED method based on ACO can be concluded as follows: ① The solution space and the problem space co-evolve, which is different from the traditional method for structural optimization in essence. ② Inherent parallelism. Similar to the GAs, ACO is also a colony-based search method, so it can be performed in parallel, which is good for large-scale problems. ③ Effectiveness. Theoretical analysis shows that the computation complexity of each sub-process in CED based on ACO is polynomial and it guarantees the effectiveness of this method. ④ Robustness. In the ACO, the failure of some individual ants will not influence the search process and the final solutions, which is very important for the CED to be implemented in engineering practice.

According to the design examples of the 10-bar and 15-bar trusses under different conditions, the effectiveness of the method in this paper is validated and the effects of the loads and the deflection constraints on the co-evolutionary design are discussed.

References

1 Bentley PJ. An introduction to evolutionary design by computers. In: Evolutionary Design by Computers. San Francisco: Morgan Kaufmann, 1999, 1—74

2 Maher ML. A model of co-evolutionary design. Engineering With Computers, 2000, 16(3, 4): 195—208

3 Luo Y. Study on dynamic optimization of large deployed antenna and structural systematic optimization. Dissertation for the Doctoral Degree, Xi'an: Xidian University, 2004, 15—16

4 Camp C, Pezeshk S and Cao G. Optimized design of two-dimensional structures using genetic algorithm. Journal of Structure Engineering, 1998, 124(5): 551—559

5 Xiao RB, Tao ZW and Liu Y. The Principles and Techniques of Intelligent Design (in Chinese). Beijing: Science Press, 2006, 16—22

6 Bonabeau E, Dorigo M and Theraulaz G. Inspiration for optimization from social insect behavior. Nature, 2000, 406(6791): 39—42

7 Xiao RB, Tao ZW and Liu Y. Isomorphism identification of kinematic chains using novel evolutionary approaches. ASME Journal of Computing and Information Science in Engineering, 2005, 5(1): 18—24

8 Sun ZG and Teng HF. An ant colony optimization based layout optimization algorithm. In: Proceedings of 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering, Beijing, China, Oct 28—31, 2002, 675—678

9 Camp CV and Bichon BJ. Design of space trusses using ant colony optimization. Journal of Structural Engineering, 2004, 130(5): 741—751

10 Bullnheimer B, Hartl RF and Strauss C. A new rank based version of the ant system: A computational study. Central European Journal for Operations Research and Economics, 1999, 7(1): 25—38

11 Sun HC, Chai S and Wang YF. Discrete Optimum Design of Structures (in Chinese). Dalian: Dalian University of Technology Press, 1995, 105—106, 235—236

12 Liu GH and Wei RY. A co-evolution algorithm for the topology and size optimization of trusses (in Chinese). Guangxi Sciences, 2000, 7(2): 81—84