

文章编号: 1673- 9469(2011) 02- 0071- 04

# 混合算法在轻钢结构优化设计中的应用

周书敬, 薄涛, 史三元

(河北工程大学 土木工程学院, 河北 邯郸 056038)

**摘要:** 结合粒子群优化(PSO)算法快速的全局收敛性和蚁群优化(ACO)算法较强的寻优能力,提出了一种融合 PSO 算法和 ACO 算法的混合算法。首先利用 PSO 算法较强的全局搜索能力,产生各粒子的最优位置值;然后对 ACO 算法的蚂蚁总个数进行调整,在保证算法全局搜索能力的同时,避免陷入局部最优;最后利用改进的 ACO 算法对最优位置值做进一步优化。将该混合算法应用于轻钢结构优化设计中,建立优化设计模型。以轻钢门式框架为例,利用该模型进行优化分析,并与文献[11]中改进模拟退火算法的优化结果进行对比。结果表明,混合算法经过 61 次迭代后能够求出较好的全局最优解,合理可行。

**关键词:** ACO 算法; PSO 算法; 混合算法; 轻钢结构; 离散变量优化

**中图分类号:** TU 391

**文献标识码:** A

## Application of hybrid algorithm in optimization design of light steel structure

ZHOU Shu-jing, BO Tao, SHI San-yuan

(School of Civil Engineering, Hebei University of Engineering, Hebei Handan 056038, China)

**Abstract:** Considering the rapid global convergence of the particle swarm optimization (PSO) algorithm and the strong optimization ability of the ant colony optimization (ACO) algorithm, the hybrid algorithm was put forward by combining with PSO algorithm and ACO algorithm. The optimal location values of the particles were generated by the strong global searching capability of the PSO algorithm. The total number of the ants was adjusted to ensure the global searching ability of the ACO algorithm and avoid trapping in local optimum. With the modified ACO algorithm, the optimal location values were further optimized. The hybrid algorithm is used to the optimization design of the light steel structure, and the design model was established. For an example, the wrap-round frame with light steel was optimized. By comparing to the optimization of the modified simulated annealing algorithm in literature [9], the results showed that the hybrid algorithm which solved the global optimal solution after 61 iterations, was feasible,

**Key words:** ACO algorithm; PSO algorithm; hybrid algorithm; light steel structure; discrete variable optimization

轻钢结构由于其具有自重轻,抗震性能好及施工速度快等优点而广受青睐,轻钢结构优化设计也成为近年来的一个研究热点<sup>[1-2]</sup>。蚁群优化(ant colony optimization, ACO)算法<sup>[3]</sup>是工程中一种常用的结构优化方法,具有并行性好、鲁棒性强等特点,如于永彪等<sup>[4]</sup>利用 ACO 算法对轻钢桁架进

行了优化设计;吴科等<sup>[5]</sup>利用基于 TSP 模型的 ACO 算法对刚架结构进行了优化研究。然而工程实践表明,ACO 算法存在着求解速度慢且易陷入局部最优的缺点,在结构优化中存在着一定的局限性。粒子群优化(particle swarm optimization, PSO)算法<sup>[6]</sup>是一种源于鸟群捕食等群体行为模拟的优

收稿日期: 2011- 04- 08

基金项目: 河北省自然科学基金项目(E2010001012) 特约专稿

作者简介: 周书敬(1956-),男,山东聊城人,教授,从事智能计算方法及其工程应用方面的研究。

化算法,具有全局搜索能力强,收敛速度快及与其他算法易结合的特点。本文将 PSO 算法和 ACO 算法相结合,在算法初期,利用 PSO 算法产生出各粒子的初始值,然后利用 ACO 算法对各粒子的初始值做进一步调整,以达到优势互补。

## 1 优化原理

### 1.1 PSO 算法

算法初始化为一群具有各自的速度和位置的随机粒子,通过追随当前最优粒子在解空间中搜索,迭代后找到最优解<sup>[7-8]</sup>。每次迭代中,粒子跟踪个体极值即个体历史最优解  $P_{id}$ ,和全局极值即整个种群目前找到的最优解  $P_{gd}$ ,并更新速度和位置。

$$v_{id}(t+1) = \omega \cdot v_{id}(t) + c_1 r_1 [P_{id}(t) - x_{id}(t)] + c_2 r_2 [P_{gd}(t) - x_{id}(t)]$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$

式中,  $v_{id}$ —第  $i$  个粒子在第  $d$  维上的速度;  $x_{id}$ —第  $i$  个粒子在  $d$  维上的位置;  $\omega$ —惯性权重;  $r_1, r_2$ —均匀分布于  $[0, 1]$  之间的随机数;  $c_1, c_2$ —加速因子。

### 1.2 ACO 算法

ACO 算法是由若干只蚂蚁共同构造解路径,通过在解路径上遗留并交换信息素来提高解的质量,进而达到优化的目的<sup>[9-10]</sup>。

设  $m$  为蚂蚁数,  $n$  为城市数,  $d_{ij}$  为城市  $i$  和城市  $j$  的距离,  $\tau_{ij}(t)$  为  $t$  时刻在路径  $(i, j)$  上的信息素量,蚂蚁根据各路径上的信息素量决定其移动方向。 $t$  时刻蚂蚁  $k$  由城市  $i$  转移至城市  $j$  的概率

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}(t)]^\beta}{\sum_{S \in allowed_k} [\tau_{is}(t)]^\alpha [\eta_{is}(t)]^\beta}, j \in allowed_k \\ P_{ij}^k(t) = 0, j \notin allowed_k \end{cases} \quad (1)$$

式中,  $allowed_k$ —蚂蚁  $k$  下一步允许选择的的城市,  $allowed_k = \{c - tabu_k\}$ ;  $tabu_k$ —蚂蚁  $k$  已访问的城市;  $\alpha$ —信息启发因子;  $\beta$ —期望启发因子;  $\eta_{ij}$ —由城市  $i$  转移到城市  $j$  的期望程度。

每只蚂蚁走完一步或走完所有的城市后,信息素按下式更新

$$\tau_{ij}(t+n) = (1-\rho) \tau_{ij}(t) + \Delta \tau_{ij}(t), \Delta \tau_{ij}(t) = \sum_{k=1}^m \Delta \tau_{ij}^k(t) \quad (2)$$

式中,  $\rho$ —信息挥发系数,  $\rho \in [0, 1]$ ;  $\Delta \tau_{ij}(t)$ —信息素增加量;  $\Delta \tau_{ij}^k(t)$ —第  $k$  只蚂蚁在本循环中留  $(i, j)$  在上的信息量。

若第  $k$  只蚂蚁在循环中经过  $(i, j)$ , 则  $\Delta \tau_{ij}^k(t) = Q/L_k$ , 其他情况下  $\Delta \tau_{ij}^k(t) = 0$ 。

式中,  $Q$ —体现蚂蚁所留轨迹数量的一个常数;  $L_k$ —第  $k$  只蚂蚁在本次循环中所走路程的长度。

## 2 算法混合

利用 PSO 算法的快速,全局收敛性作为前期搜索,得到各粒子的历史最优位置值;利用 ACO 算法的正反馈机制,将最优位置值作为后期 ACO 算法各个蚂蚁的位置,同时将信息素初始分布重新设置。

### 2.1 初始最优值的转化

蚂蚁  $i$  位置  $x_{(i)}$  对应于 PSO 算法所求解的各粒子历史最优位置  $P_{id}$ , 即

$$x_i = P_{id}$$

蚂蚁信息素初始分布利用各蚂蚁所在位置的评价函数值<sup>[11]</sup>, 如下所示:

$$\tau_{ij} = u \cdot v^{-f(x_i)}$$

式中,  $u, v$ —常数,  $u > 0, v \in [0, 1]$ , 根据实际问题定义  $u, v$  的大小;  $f(x_i)$ —目标函数值,  $f(x_i)$  越大,  $x_i$  所在位置留下的信息素越多。

### 2.2 蚂蚁总个数的调整

在 ACO 算法中,蚂蚁的总个数过大,则算法的收敛速度将减慢,过小则易陷入局部最优。综合考虑算法的全局搜索能力和收敛速度两项指标,将  $m$  只蚂蚁置于  $m$  个节点上,改随机放置为按一定规则作初始分布(均匀分布),算法中城市数  $n$  与蚂蚁总个数  $m$  有以下关系:  $n = 1.5m$ 。

### 2.3 全局最优值

结合 ACO 算法中蚂蚁总个数的调整,利用公式(1)求出蚂蚁的转移概率,利用公式(2)对信息素进行更新,通过 ACO 算法的算法模型对 PSO 算法产生的初始最优值做进一步优化,得到全局最优值。混合算法的运算流程如图 1 所示。

## 3 轻钢框架结构优化设计

(16) 设计变量 对于轻钢框架结构,以结构杆件

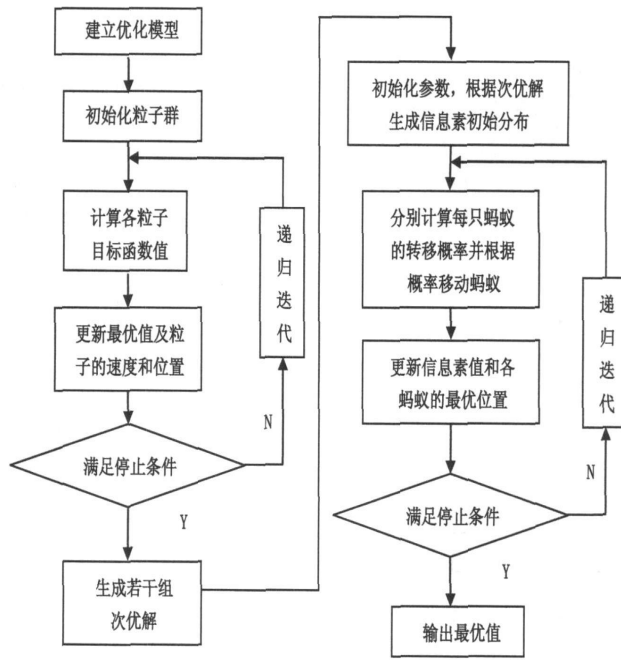


图1 混合算法的运算流程图

Fig.1 The operation flow chart of hybrid algorithm

的截面面积  $A_i$ 、截面抵抗矩  $W_i$  以及截面惯性矩  $I_i$  ( $i = 1, 2, \dots, n$ ) 为设计变量, 三者关系如下<sup>[12]</sup>:

梁:  $A_i = r_1 I_{xi}^{0.5}, W_i = r_1 I_{xi}^{0.75}, r_1 = 0.58;$

柱:  $A_i = r_2 I_{xi}^{0.5}, W_i = r_2 I_{xi}^{0.75}, I_{yi} = r_3 I_{xi}, r_2 =$

$0.8, r_3 = 0.36$

目标函数 采用轻钢框架总用钢量(总重量)为目标函数的公式, 即

$$\min W = \rho_m \sum_{i=1}^m (L_i A_i)$$

式中,  $W$ —结构杆件总重量;  $\rho_m$ —材料密度;  $A_i$ —一榀框架中第根构件的截面面积;  $L_i$ —一榀框架中第  $i$  根构件的长度。

20. 约束条件

a 应力约束:  $\sigma_i \leq [\sigma_i]$

b 位移约束:  $u_i \leq [u_i]$

c 界限约束:  $A_i \in \{E\}$

式中,  $\sigma_i, [\sigma_i]$ —构件的最不利应力和许用应力;  $u_i, [u_i]$ —特定节点在给定方向上的位移值和位

移允许值;  $\{E\}$ —构件截面尺寸变量的离散集合。

(8) 优化模型

上述优化设计数学模型可简单表述为  
设计变量  $x$

$$\min f(x)$$

$$s. t. g(x) \geq 0, x \in D$$

式中,  $g(x)$ —约束函数,  $D$ —有限个点组成的集合。

(5) 求解

将组成解的各分量视为一群随机粒子的初始位置(初始化粒子群), 将目标函数(总重量)作为适应度函数; 更新各粒子的位置并求解各粒子的历史最优位置, 作为各分量的次优解; 将蚂蚁的位置对应于所求得的次优解, 根据次优解生成信息素的初始分布(初始化蚁群), 将目标函数(总重量)看作为蚂蚁经过的总长度; 那么该结构优化问题, 可看作蚂蚁觅食过程中寻找最短路径的问题。本文运用 MATLAB 语言进行算法的程序编制, 对运行的结果, 利用 SAP2000 结构分析软件进行分析。

表 1 许用离散变量表

Tab. 1 Allowable discrete variable

变量	1	2	3	4	5	6
$A/\text{cm}^2$	99.0	105.83	112.26	134.92	140.01	145.00
$W/\text{cm}^3$	1 293.50	1 429.70	1 561.80	2 057.70	2 175.30	2 290.90
$I/\text{cm}^4$	29 136.00	33 299.00	37 461.00	54 110.00	58 272.00	62 435.00

## 4 算例分析

轻钢门式钢框架的结构及荷载如图 2 所示,算例中存在 3 种荷载工况,弹性模量  $E=206.88$  GPa,材料的容重  $\rho=76999.34\text{N/m}^3$ ,许用应力  $[\sigma]=163.86\text{MPa}$ ,节点的水平位移上限均为  $12.7\text{mm}$ ,许用离散变量集见表 1<sup>[13]</sup>。

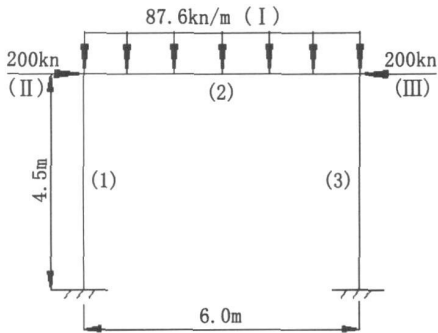


图2 轻钢门式框架的结构及荷载图

Fig.2 The structure of light steel portal frame and its load diagram

由于结构的对称性及荷载工况 II 和工况 III 的反对称性,所以只需考虑两种荷载工况(I, II)即可。本例将 3 个构件单元分为两组,其中构件单元 (1) (2) 为组 1,单元 (3) 为组 2。主要控制参数取值为

$c_1=c_2=2$ ,  $\omega=0.8$ ,  $\alpha=0.4$ ,  $\beta=0.7$ ,  $\rho=0.7$ ,  $Q=6000$

算法的迭代关系曲线如图 3 所示,算法曾 3 次跳出局部极小解,在经过 61 次迭代后收敛得到全局最优解。

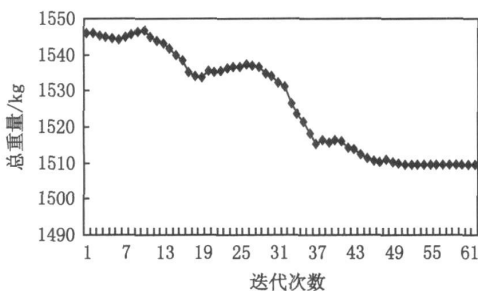


图3 迭代关系曲线图

Fig.3 Iterative curve diagram

文献[11]利用改进模拟退火算法对轻钢框架进行了优化分析,本文将混合算法的优化结果与改进模拟退火算法的优化结果进行了对比(表 2),可以看出,在迭代次数增加的情况下,应用混合算法能够求出更小的目标函数值(总重量),并且在满足约束条件的前提下,节省了用钢量,从而表明

该算法应用于轻钢结构优化是可行的。

表 2 离散变量优化结果

Tab.2 The results of discrete variable optimization

算法	截面优化结果/ $\text{cm}^4$		总重量 /kg	迭代次数
	组 1	组 2		
改进模拟退火算法	58 272.00	33 299.00	1 512.83	48
混合算法	58 266.01	33 285.00	1 509.51	61

注:表中迭代次数指结构重分析次数。

## 5 结论

1) 采用均匀分布策略对初始蚂蚁总个数进行调整,减少了因蚂蚁总个数过大或过小对 ACO 算法造成的不良影响。

2) 在迭代次数增加的情况下,本算法能够求出更小的目标函数值,达到用钢量最省的目的。

### 参考文献:

- [1] 史三元,邵莎莎,陈林.多层冷弯薄壁型钢结构住宅抗震性能分析[J].河北工程大学学报(自然科学版),2010,27(4):1-4.
- [2] 史三元,侯桂欣,苏晓亮.小高层钢结构住宅抗震性能研究[J].河北工程大学学报(自然科学版),2010,27(2):1-4.
- [3] LUCA M. Ant algorithms for discrete optimization[J]. Artificial Life, 1999, 5(2): 132-172.
- [4] 于永彪,鹿晓阳.蚁群算法在桁架结构离散变量优化设计中应用[J].科技信息,2007,9(1):100-101.
- [5] 吴科,李哲,等.基于蚁群算法的刚架结构优化设计[J].钢结构,2007,22(6):13-16.
- [6] YIN P.Y. Particle swarm optimization for point pattern matching[J]. Vis Commun Image R, 2006, 17: 143-162.
- [7] 雷德明,严新平.多目标智能优化算法及其应用[M].北京:科学出版社,2009.
- [8] 周书敬,吴超.房地产投资环境的模糊多群体准则评价方法[J].河北工程大学学报(自然科学版),2007,24(3):68-71.
- [9] 刘振.蚁群算法的性能分析及其应用[J].计算机软件与理论,2010,31(2):22-24.
- [10] 周书敬,李慧敏.一种变系数的自适应蚂蚁算法[J].数学的实践与认识,2008,38(12):66-71.
- [11] 支成秀,梁正友.融合粒子群算法和蚁群算法的随机搜索算法[J].广西科学院学报,2006,22(4):231-233.
- [12] 翁维素,张文山.轻型钢房屋结构的优化设计[J].河北建筑工程学院学报,2001,12(2):10-13.
- [13] 李国名.基于改进模拟退火算法的轻钢框架结构优化设计[D].济南:山东建筑大学,2007.

(责任编辑 马立)