

DOI: 10.3969/j.issn.1001-4551.2019.06.019

基于改进蚁群算法的激光 切割加工路径优化研究*

侯普良, 刘建群*, 高伟强

(广东工业大学 机电工程学院, 广东 广州 510006)

摘要: 针对激光切割加工全局路径优化采用传统蚁群系统算法时, 存在收敛速度慢、易陷入局部最优的问题, 对蚁群系统算法进行了改进研究。利用激光加工图元的起点和终点信息, 建立了图元等价 TSP 问题的数学模型, 提出了通过最邻近插入算法对蚁群系统算法路径规划结果进行了再优化的改进算法; 详细阐述了改进蚁群系统算法的实现步骤, 分析了传统蚁群系统算法和改进蚁群系统算法的迭代次数和优化效果。研究结果表明: 该改进蚁群系统算法加快了收敛速度, 迭代次数减少了约 30%, 缩短了激光加工所走路径的总长度, 并成功应用到自主开发的高速激光切割加工系统中。

关键词: 激光切割; 路径规划; 改进蚁群算法; 最邻近插入算法

中图分类号: TP301.6; TG485

文献标志码: A

文章编号: 1001-4551(2019)06-0653-05

Optimization of laser cutting path based on improved ant colony algorithm

HOU Pu-liang, LIU Jian-qun, GAO Wei-qiang

(College of Mechanical and Electrical Engineering, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: Aiming at the global path optimization of laser cutting processing, the traditional ant colony system algorithm has a slow convergence rate and easy to fall into local optimum, the improved ant colony system algorithm was studied. Using starting point and end point information of the laser processing entity, the mathematical model of the equivalent TSP problem of the entity was established. An improved algorithm for re-optimizing the path planning results of ant colony system algorithm by nearest neighbor insertion algorithm was proposed, the implementation steps of the improved ant colony system algorithm was described in detail, the iteration times and optimization results of the traditional ant colony system algorithm and the improved ant colony system algorithm were analyzed. The results indicate that the improved ant colony system algorithm accelerates the convergence speed, reduces the number of iterations by about 30%, shortens the total length of the path taken by the laser processing and successfully applied to the self-developed laser cutting processing system.

Key words: laser cutting; route planning; improved ant colony algorithm; nearest neighbor insertion algorithm

0 引 言

在激光切割加工中, 激光器在每个加工轮廓之间移动需要耗费大量的时间, 因此, 激光加工路径优化是提高激光加工生产效率的关键。

学界对激光加工路径优化问题做了大量的研究,

提出了多种思路。Guk-ghan Han 和 Suck-jooNap^[1]提出了采用模拟退火算法求解激光板材的路径优化问题; 刘慧霞等^[2]提出了用分级规划的三步算法解决激光切割路径优化问题; 高伟增等^[3]提出了应用遗传算法求解激光加工的最优路径; CuneytOysu 和 Zafer Binguolto^[4]提出了一种混合算法(遗传算法和模拟退火算

收稿日期: 2018-11-19

基金项目: 广东省科技计划资助项目(2016B090911001); 广东省联合培养研究生示范基地项目(2017XJJD-06)

作者简介: 侯普良(1992-), 男, 山东菏泽人, 硕士研究生, 主要从事数控系统方面的研究。E-mail: 1689631025@qq.com

通信联系人: 刘建群, 男, 教授, 硕士生导师。E-mail: jqliu@sina.com

法结合的一种算法)以求解刀具空行程最短的路径优化问题;李妮妮等^[5]提出了将局部搜索算法与遗传算法相结合的算法求解激光切割路径优化问题;刘保业^[6]提出了采用改进遗传蚁群算法求解激光加工路径优化问题。

上述单一智能算法求解路径优化问题往往存在收敛速度慢、易陷入局部最优解、优化效果不理想的问题。而两种智能算法相结合的混合算法如遗传算法和模拟退火算法结合的算法、改进遗传蚁群算法,虽然改进了单一智能算法的缺陷,但是却使计算量增加很大,耗时较长。为此需要提出一种改进的蚁群系统算法,以减少迭代次数,更快得到较优的加工路径。蚁群系统算法是一种元启发式算法^[7],但单纯的蚁群系统算法存在收敛速度慢、求解质量差等缺点。

为了发挥蚁群系统算法的优良特性,并改进存在的问题,本研究引入最邻近插入算法对蚁群系统算法进行优化。

1 路径优化模型

1.1 路径优化的问题归类

加工路径的优化问题可以归类到 TSP 问题。将激光加工路径优化问题归类到 TSP 问题,就是将待加工图形中的图元等价于 TSP 问题中的城市^[8]。在 TSP 问题中城市是用一个坐标点表示的,如要使图元等价城市,现行的方法有用图元起点表示图元、图元终点表示图元、图元上的任意特征点表示图元等。但这些表示方法在涉及到激光加工工艺时都存在一些缺陷。因为在激光切割图元时,激光器并不是将图元上的某一个点作为切割起点,切割起点一般选择废料区,然后通过引入线将激光器过渡到图元上。

图元之间距离如图 1 所示。

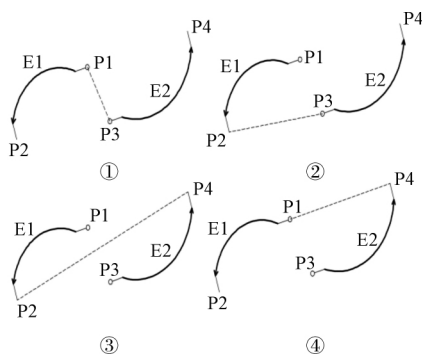


图 1 图元之间距离

如果引入线起点和引出线终点不是同一点,图元则不能用单一点表示,如图 1 中①③所示,在计算图元 E1 和 E2 间的距离时,图元引入线起点间的距离

|P1P3|和图元引出线终点间的距离|P2P4|差别很大,从而算法得出的最优路径可能达不到满意的效果。在激光实际加工中,激光器从图元 E1 到图元 E2 的距离和从图元 E2 到图元 E1 的距离是不同的。如图 1 中②④所示,E1 到 E2 的距离为 E1 的引出线终点 P2 到 E2 的引入线起点 P3 之间的距离,E2 到 E1 的距离为 E2 的引出线终点 P4 到 E1 引入线起点 P1 之间的距离。

为了能够合理计算图元之间的距离,本文提出图元等价城市的新方式,即每个图元本身等价于 TSP 中的每个城市,在计算两个图元之间的距离时,同时利用图元的引入线起点和引出线终点信息,即计算当前图元的终点到下一个图元的起点之间的距离。

1.2 加工图形排序的数学模型

设加工的图形有 n 个基本图元 P_i 和 P_{i+1} 是图元 i 的引入线起点和引出线终点, P_j 和 P_{j+1} 是图元 j 的引入线起点和引出线终点,若图元引入线起点和图元引出线终点是同一点,则 $P_i = P_{i+1}$, $P_j = P_{j+1}$ ($1 \leq i, j \leq n$)。

图元 i 到图元 j 距离为:

$$d_{ij} = \begin{cases} \infty & i = j \\ p_{i+1}p_j & i \neq j \end{cases} \quad (1)$$

一副有 n 个图元组成的加工图,经过优化算法生成的队列之中,任取队列中的一个图元 i ,若 $i! = n$,则队列中 i 的下一个图元就是 i 的后继。

设图元 i 的后继是图元 j ,则对于图元 i 到其后继图元的距离有:

$$U(i) = d_{ij} \quad (2)$$

激光行走的总路径是图元本身的长度加上图元之间的长度,图元本身长度是不变的,因此需要求得 n 个图元之间 $n-1$ 段距离尽可能短的路径,作为激光加工的最优路径,即:

$$\min D = \sum_{i=1}^{n-1} U(i) \quad (3)$$

2 蚁群系统(ACS)改进

蚁群系统算法具有较强的鲁棒性、优良的分布式计算等优点,但也存在一定的缺陷,如在解决大规模问题时,收敛速度慢、运算时间长、容易陷入局部最优解等^[9]。针对蚁群系统算法的缺陷,本研究通过最邻近插入算法对 ACS 算法进行优化,避免算法过早陷入停滞,同时减少了算法的迭代次数。

2.1 蚁群系统算法

针对 n 个城市的 TSP 问题的 ACS 算法^[10]:城市个

数为 n , 分别用 $\{1, 2, \dots, n\}$ 表示, 城市 i 到城市 j 的距离用 d_{ij} 表示, 任意时刻 t 在路径 (i, j) 上的信息素浓度为 $\tau_{ij}(t)$ 。初始时刻 $\tau_{ij}(0) = c$ (c —常数), 则 t 时刻位于城市 i 的蚂蚁 k ($k = 1, 2, \dots, m$) 根据伪随机比例规则从任一城市中选择城市 j 作为下一个访问城市, 规则为:

$$j = \begin{cases} \operatorname{argmax}\{\tau_{il}[n_{il}]^\beta\}, & q \leq q_0; \\ J, & q > q_0 \end{cases} \quad (4)$$

式(4)中的 J 是在基本蚁群算法中, 蚂蚁 k 利用伪随机概率选择下一个城市的方式。蚂蚁 k 由城市 i 到城市 j 的概率为:

$$p_{ij}^k = \begin{cases} \frac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{t \in N_i^k} [\tau_{it}]^\alpha [\eta_{it}]^\beta}, & j \in N_i^k \\ 0, & j \notin N_i^k \end{cases} \quad (5)$$

式中: η_{ij} —路径 (i, j) 上的能见度(代表预先给定的启发式信息) $\eta_{ij} = 1/d_{ij}$; β —启发信息重要程度; N_i^k —位于城市 i 的蚂蚁 k 还未走过的相邻城市集合; α —信息素在选择下一城市中所占的比重。

每只蚂蚁每经过一个边 (i, j) , 根据下式进行局部信息素更新:

$$\tau_{ij} = (1 - \xi) \tau_{ij} + \xi \tau_0 \quad (6)$$

当所有蚂蚁走完一遍路径之后, 从中找出至今最优的那条路径, 根据下式进行全局信息素更新:

$$\tau_{ij} = (1 - \rho) \tau_{ij} + \rho \Delta \tau_{ij}^{bs} \quad (7)$$

式中: ρ —信息素的挥发率; $\Delta \tau_{ij}^{bs} = 1/C^{bs}$, C^{bs} —至今的最优路径长度。

2.2 蚁群系统算法的改进

在一个蚁群中, 每个蚂蚁的运动是随机的, 虽然能够通过信息素沟通向着最优路径进化, 但是当城市数量较大时, 由于在初始情况下各个路径的信息素相差很小, 就很难在较短的时间内找出一条较好的路径^[11]。为了克服计算时间长、收敛速度慢的缺陷, 本研究利用最邻近插入算法对每次循环后的当前最优路径进行调整。经过优化, 可以极大地加速蚁群的收敛速度, 并且解的质量也会有所提高。

为了利用插入算法优化当前最优路径队列, 本研究采用随机值分割当前最优路径队列, 从而不容易陷入局部最优, 有利于发现较好解。改进算法是在所有的蚂蚁走完一遍给定的城市以后, 对得到的当前最优路径进行调整。调整步骤如下:

(1) 生成一个随机值 r ($0 \leq r \leq n/2$), 将当前最优队列分为 List1 和 List2。List1 包含的城市数目为 $n - r$, List2 中城市的数目为 r ;

(2) 计算 List1 中的所有城市和 List2 中所有城市之间的距离, 查找两者之间距离最短的城市, 设在 List2 中其城市为 k ;

(3) 计算城市 k 与 List1 中每两个相邻城市之间的距离之和减去这两个相邻城市之间的距离差值, 可以在 List1 中求得相邻的两个节点 m 和 n , 使得差值最小;

(4) 将城市 k 从(2)中删除并将其插入到 List1 中的 m 和 n 之间;

(5) 重复执行(2~4), 直到 List2 为空。

2.3 改进后蚁群系统算法的实现步骤

利用改进后的蚁群系统算法, 对激光加工的图元进行排序, 具体的算法实现步骤如下:

(1) 参数初始化。初始时间 $t = 0$ 和循环次数 $N_c = 0$, 设置最大循环次数 $N_{c_{\max}}$, 令蚂蚁个数。图元之间的距离按照式(1)计算, 令每条边 (i, j) 的初始化信息量。

(2) 将 m 只蚂蚁分别随机放置到 n 个图元中, 清空每只蚂蚁的禁忌表 $tabu_k$ ($tabu_k$ —已经走过的图元集合)。并将蚂蚁的当前所在图元加入到对应蚂蚁的禁忌表 $tabu_k$, 令当前蚂蚁 $k = 0$, 令 $N_c = N_c + 1$ 。

(3) 如果 $N_c \leq N_{c_{\max}}$, 转到(4), 否则结束循环, 得到当前最优路径。

(4) 判断所有蚂蚁是否遍历完一遍图元, 若还未遍历完时, 蚂蚁 m (最后一个蚂蚁) 的禁忌表为 $tabu_m$, 若 $Size(tabu_m) < n$, 令当前蚂蚁 $k = 0$, 转到(5); 否则求出当前最优路径, 转到(6)。

(5) 如果 $k > m$, 转到(4); 否则根据式(4~5)求得蚂蚁 k 的当前图元 i 的下一个图元 j , 并将图元 j 加入到禁忌表 $tabu_k$, 并令 $k = k + 1$ 。重复执行第(5)步。

(6) 当前最优路径队列设为 List, 产生随机数 r , 将 List 分为 List1 和 List2, 并且 List1 中的图元多于 List2 中的图元。将 List2 中的图元按照蚁群系统改进的规则全部插入到 List1 中, 比较 List1 和 List 路径长度, 选择较短路径作为当前最优路径, 根据公式(7)进行全局信息素更新, 并转到(2)。

3 算例验证

本文采用自主研发的高速数控激光切割加工机床进行实验。机床为直线电机龙门同步激光切割机, 型号为 GGX-1701。参数设置如下: 切割速度为 1 200, 加速度为 10 000, 减速度为 10 000。根据文献[12], 将参数设定为: $\alpha = 1$, $\beta = 3$, $\rho = 0.9$, $\xi = 0.1$, $q_0 = 0.9$, $m = 10$, $n_{\max} = 150$ 。

针对激光切割加工的图形, 笔者采用蚁群系统算

法和本文提出的改进的蚁群系统算法分别进行切割加工试验。为了分别验证算法对闭合图元和非闭合图元的优化效果,本研究设置了两个实例。

3.1 全封闭图元图形

实例一(全封闭图元组成的图形)如图2所示。

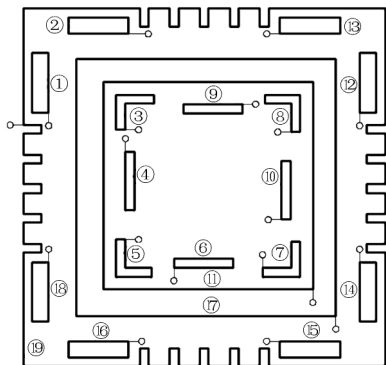


图2 实例一(全封闭图元组成的图形)

图2中共有19个闭合图元,每个图元都有一个标号,图元标号仅是图形文件读取时设定的一个图元记号,用来区分不同的图元,该图闭合图元的引出线和引入线为同一条。

图2中19个图元的坐标值分别为:1(81.17, 57.03); 2(99.51, 75.37); 3(98.80, 53.63); 4(98.37, 49.12); 5(98.80, 28.72); 6(109.20, 21.89); 7(129.09, 22.81); 8(135.00, 53.11); 9(125.11, 59.44); 10(132.86, 33.22); 11(140.36, 16.96); 12(150.85, 57.53); 13(132.51, 75.37); 14(150.35, 24.04); 15(132.51, 5.70); 16(99.51, 5.70); 17(87.36, 10.95); 18(81.17, 24.04); 19(74.99, 54.71)。

本研究对图2全封闭图元组成的图形,分别用传统蚁群系统算法和改进蚁群系统算法进行优化。两种算法的最优路径空行程进化曲线如图3所示。

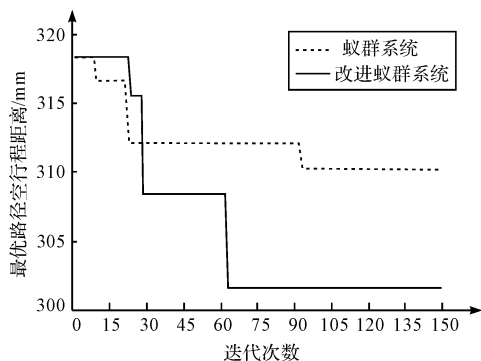


图3 两种算法的最优路径空行程进化曲线(实例一)

蚁群系统运算结果如图3中虚线所示,得到的最优解空行程长度为310.18 mm,其最优路径:19→1→4→3→2→9→13→12→8→10→7→11→14→15→6→5

→18→17→16。

改进的蚁群系统运算结果如图3中实线所示,得到的最优解空行程长度为301.63 mm,最优路径:12→8→9→13→2→3→4→1→19→18→17→16→5→6→7→10→14→11→15。

经多次实验,实例一改进的蚁群系统算法优化后平均加工耗时9.41 s,传统蚁群系统算法优化后平均加工耗时9.58 s。

3.2 非封闭图元图形

实例二(非封闭图元组成的图形)如图4所示。



图4 实例二(非封闭图元组成的图形)

实例二选取了一副具有103个图元组成的图形,区别于实例一的闭合图元,实例二中103个图元全部是非闭合的,该图不需要设置引入和引出线。

本研究对图4所示的非封闭图元组成的图形,分别用传统蚁群系统算法和改进蚁群系统算法进行优化。两种算法的最优路径空行程进化曲线如图5所示。

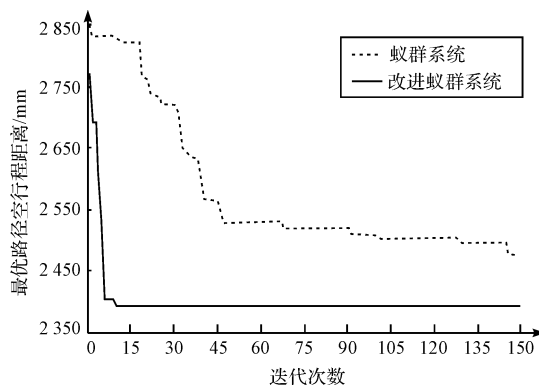


图5 两种算法的最优路径空行程进化曲线(实例二)

蚁群系统运算结果如图5虚线所示,得到的最优解空行程长度为2477.64 mm。改进蚁群系统运算结果如图5实线所示,得到的最优解空行程长度2394.46 mm。实例二经多次加工实验,改进的蚁群系统算法优化后平均加工耗时41.19 s,传统蚁群系统算法优化后平均加工耗时42.90 s。

对比改进前后的蚁群系统算法可以看出,改进后的蚁群系统得到的最优解的质量优于未优化的蚁群系统,改进后的蚁群系统收敛速度更快。在实例一中,改进的蚁群系统在第62次就得到迭代的最优解,但未改进的蚁群系统在第92次才得到最优解;在实例二中,改进的蚁群系统在第8次就已经得到迭代最优解,而未改进的蚁群系统在第145次才得到迭代最优解。

实验结果表明:改进后的蚁群系统算法所需的加工时间明显减少。经过大量实验,改进后的蚁群系统算法比未改进的算法空行程路径减少1.5%左右,迭代次数减少30%左右。本文由于篇幅所限,只列出了两个实例作为说明。

4 结束语

针对传统蚁群系统算法存在收敛速度慢、易陷入局部最优的缺陷,本文对蚁群系统算法进行了改进,使其在收敛速度和解的质量上都有所提升,并将改进算法应用于规划激光切割加工的路径,得到以下结论:

(1) 改进的蚁群系统算法,在每次迭代后会利用最邻近插入算法对其结果进行优化,既加快了蚁群寻优速度,也使蚂蚁寻优效果更好;

(2) 改进后的蚁群系统算法求解的结果更优,所需要的迭代次数减少约30%,验证了该算法是可行、有效的。

参考文献(References):

[1] HAN G, NA S. A study on torch path planning in laser cutting process part 2 cutting path optimization using simulated annealing[J]. *Journal of Manufacturing Process*, 1999, 1

(1): 62-70.

- [2] 刘慧霞,王霄. 共边排样件激光切割路径的规划[J]. *中国激光*, 2005, 31(10): 1269-1274.
- [3] 高伟增,张宝剑,陈付贵,等. 基于遗传算法的切割路径优化[J]. *西南交通大学学报*, 2006, 40(4): 457-461.
- [4] OYSU C, BINGUL Z. Application of heuristic and hybrid-GASA algorithms to tool-path optimization problem for minimizing airtime during machining[J]. *Engineering Applications of Artificial Intelligence*, 2009, 22(3): 389-396.
- [5] 李妮妮,陈章位,陈世泽. 基于局部搜索和遗传算法的激光切割路径优化[J]. *计算机工程与应用*, 2010, 46(2): 234-239.
- [6] 刘保业. 基于改进遗传蚁群算法的激光加工路径规划[D]. 青岛: 青岛理工大学机械工程学院, 2012.
- [7] 王剑文,戴光明,谢柏桥,等. 求解 TSP 问题算法综述[J]. *计算机工程与科学*, 2008, 30(2): 72-74.
- [8] 潘海鸿,刘晓琳,廖小平,等. 钣金激光切割加工 CAD/CAM 软件的孔群加工路径优化算法[J]. *组合机床与自动化加工技术*, 2013(11): 110-113.
- [9] 朱艳,游晓明,刘升,等. 基于改进蚁群算法的机器人路径规划问题研究[J]. *计算机工程与应用*, 2018, 54(19): 129-134.
- [10] DORIGO M, GAMBARDELLA L M. Ant colony system: a cooperative learning approach to the traveling salesman problem[J]. *IEEE Transactions on evolutionary computation*, 1997, 1(1): 53-66.
- [11] 章小兵,乔茹,赵光兴. 基于改进蚁群算法的移动机器人全局路径规划[J]. *机电工程*, 2009, 26(3): 77-80.
- [12] 刘利强,戴运桃,王丽华. 蚁群算法参数优化[J]. *计算机工程*, 2008, 34(11): 208-210.

[编辑: 李 辉]

本文引用格式:

侯普良,刘建群,高伟强. 基于改进蚁群算法的激光切割加工路径优化研究[J]. *机电工程*, 2019, 36(6): 653-657.

HOU Pu-liang, LIU Jian-qun, GAO Wei-qiang. Optimization of laser cutting path based on improved ant colony algorithm[J]. *Journal of Mechanical & Electrical Engineering*, 2019, 36(6): 653-657.

《机电工程》杂志: <http://www.meem.com.cn>