

文章编号:1000-582X(2004)12-0017-03

TSP 改进算法及在 PCB 数控加工刀具轨迹中的应用*

王英章,李 坚,徐宗俊

(重庆大学 机械工程学院,重庆 400030)

摘 要:电子信息产业的迅猛发展给电子制造技术带来严峻挑战和剧烈的市场竞争。针对电子元器件最基础部件——印制电路板的孔位加工所急需解决的效率问题,提出了一种印制电路板焊盘(孔位)加工刀具轨迹优化技术,即一种面向工程应用的综合应用贪婪算法和蚁群算法进行问题求解的方法,以解决由于孔多、空行程现象严重而带来的加工时间浪费问题。实际应用效果较好,提高加工效率 23.9%,取得了较好的经济效益。

关键词:数控加工;TSP;贪婪算法;蚁群算法;PCB

中图分类号:TP391

文献标识码:A

印刷电路板(PCB)数控钻铣床是电子信息产业必不可少的数控加工工艺装备,主要用于印制电路板精密孔位的高速加工。在加工过程中,刀具的移动顺序对加工效率的影响很大。因此,在将 CAD 数据传递给机床之前,需要对其进行预处理,以确定最佳加工顺序。

针对 PCB 数控钻铣床进行钻铣加工时后置处理软件的实际需要,作者重点研究了面向工程应用的海量焊盘钻孔轨迹的优化处理技术。

1 PCB 焊盘加工的数学模型

PCB 板作为电子元器件的安装母板,在焊装元件时,需要在 PCB 板的对应位置上钻出焊装用的孔(又称焊盘)。一般来说,根据安装元件的数量和种类的不同,PCB 板上的焊盘数量和焊盘大小差别较大,从数量上讲,一块复杂的 PCB 板上,同一尺寸的焊盘可能有成千上万个。当利用数控钻铣床进行焊盘的加工时,理想的加工过程如下:对某一类给定尺寸的焊盘,换上对应的刀具后,从下刀点开始,沿着使该刀具总的空行程最短的轨迹,从一个焊盘移动到另一焊盘,直到该类焊盘中的所有对象都被加工完毕,再进行下一尺寸焊盘的加工,如此循环,如图 1 所示。

该问题可描述如下:

设有 1 个换刀点和 n 个焊盘, $H = \{H_0, H_1, H_2, \dots,$

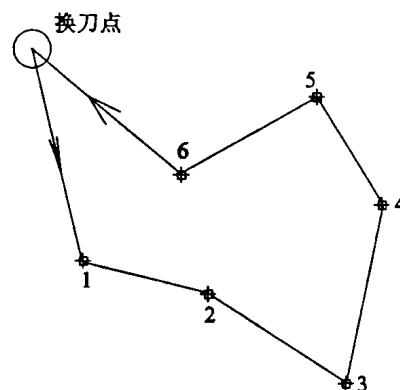


图 1 焊盘加工过程

$H_n\}$; 已知其中任意两焊盘间的距离 $d(H_i, H_j) = d(i, j)$, $0 \leq i, j \leq N$ 且 $i \neq j$; 现要求得到一个焊盘的排列 $R = \{R_0, R_1, R_2, \dots, R_n\}$; 使得 $\min \sum_{i,j \geq 0 \text{ 且 } i \neq j}^N d(R_i, R_j)$, s. t $R_0 = H_0$ 为换刀点的位置。

2 TSP 问题及改进的贪婪算法

从该问题的处理过程来看,这是一个典型的 TSP 问题,在这里,钻头充当了旅行商的角色,同孔径的焊盘充当着城市的角色,目标为加工过程中的空行程最短^[1]。

TSP 问题是实际应用中出现的复杂问题的集中概括和简化形式,是一个概念简单但求解复杂的问题,求

* 收稿日期:2004-08-12

基金项目:重庆大学基础及应用基础研究基金(2003-27)

作者简介:王英章(1969-),男,重庆大学机械工程学院博士研究生,主要研究方向为数控机械加工装备的设计与制造。

解的任务就是安排一次完整的旅行,使得总费用最小,求解的过程也是一个搜索的过程, n 个城市的 TSP 问题的搜索空间是 n 个城市的所有排列的集合,大小为 $n!$ 。TSP 问题是一个 NP 完备问题,一个 10 个城市的 TSP 有大约 18 000 个可能的解,因此,要得到 10 城市 TSP 问题的最短路径,最坏的可能是要做 18 000 次搜索;而 20 城市 TSP 问题,最坏可能要做 10^{16} 次搜索^[2];50 个城市 TSP 问题的搜索次数更是一个天文数字。在 PCB 板焊盘数控加工过程中,要处理的焊盘可能有成千上万个,因此,要以有限的时空复杂度得到焊盘加工的最优轨迹实际上不可能实现,在工程应用中,大多采用近似的方式进行处理。TSP 可用的近似解法主要有:近邻法、贪婪法、最近/最远插入法、启发式方法、神经网络法等。

求解 TSP 问题的最直观的贪婪算法是基于最近领域的启发式方法:随机从某个城市出发,前往最近的未被访问的城市,一直到所有的城市都被访问过并仅被访问一次,最后返回初始城市,这样的路径很难做到完美,通常要为在起始阶段的贪婪的选择付出很大的代价。图 2 为一焊盘加工的直接贪婪算法的运行结果,从图中可以看出,直接贪婪算法导致的结果是焊盘加工过程中的长的空行程的出现。



图 2 焊盘加工的直接贪婪算法

为此,提出了一种改进的贪婪算法:

STEP1: 建立 4 个数组 (CArray) U, V, W, Z ;

STEP2: $U = \{1, 2, \dots, n\}, V = \{0\}, W = \Phi, Z = \Phi$;

STEP3: $\text{len} = \text{LENGTH}(V)$; $\text{dist} = \text{minDIST}(i, j)$
where: $i \in U$ 且 $i \notin W, j = V(\text{len} - 1)$;

$\text{ADD}(I, W), \text{DELETE}(I, U)$;

STEP4: $\text{size} = \text{LENGTH}(W)$, IF ($\text{size} < 6$), 转 STEP3;

STEP5: 调用蚁群算法得 $W + V(\text{len} - 1)$ 的最优解,将顶点序列存入数组 Z 中;

STEP6: $\text{length} = \text{LENGTH}(Z)$; $k = 0$, while ($k < \text{length}$) { $\text{ADD}(Z(k), V)$; $\text{DELETE}(Z(k), U)$; $k + 1$; } $W = \Phi, Z = \Phi$;

STEP7: IF ($U = \Phi$), 结束, 否则, 转 STEP3;

最后得到的数组 V 中的焊盘序列即为数控钻铣床进行焊盘加工的近似解。

3 改进算法的程序实现

3.1 蚁群算法原理

蚁群算法是一种新型的模拟进化算法。该算法由意大利学者 M. Dorigo、V. Maniezzo、A. Colorini 等首先提出,称之为蚁群系统 (Ant Colony System, 简称 ACS), 并应用该算法求解 TSP 问题、分配问题、job-shop 调度问题,取得了较好的结果^[5]。

人工蚁群算法是受到对真实蚁群行为研究的启发而提出的。蚂蚁在运动过程中,能够在它所经过的路径上留下一一种称之为外激素 (pheromone) 的物质进行信息传递,而且蚂蚁在运动过程中能够感知这种物质,并以此指导自己的运动方向,因此由大量蚂蚁组成的蚁群集体行为便表现出一种信息正反馈现象:某一路径上走过的蚂蚁越多,则后来者选择该路径的概率就越大。

3.2 蚁群算法

1) 状态转移规则

$p_{ij}(k)$ 表示蚂蚁 k 由城市 i 转移到城市 j 的转移概率^[6]:

$$p_{ij}(k) = \left\{ \frac{\tau_{ij}^{\alpha}(t) \eta_{ij}^{\beta}(t)}{\sum_{0, \text{else}} \tau_{is}^{\alpha}(t) \eta_{is}^{\beta}(t)}, j, s \notin \text{tabu}(k) \right\}$$

$\tau_{ij}(t)$ 为 t 时刻在路径 ij 上的信息素量, η_{ij} 为城市 ij 间距离的倒数, $\text{tabu}(k)$ 用以记录蚂蚁 k 当前所走过的城市, s 为下一转移城市, α, β 为系数;

2) 信息素修正规则

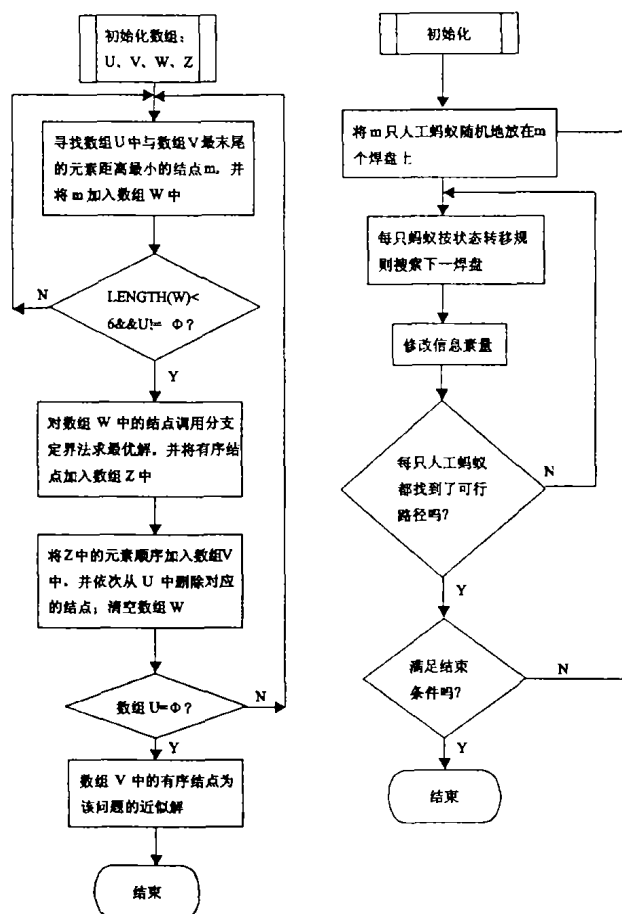
$$\tau_{ij}(t+n) = \rho \tau_{ij}(t) + \Delta \tau_{ij}$$

$$\Delta \tau_{ij} = \sum_{k=1}^n \Delta \tau_{ij}(k)$$

$$\Delta \tau_{ij}(k) = \left\{ \begin{array}{l} \frac{Q}{L_k}, \text{若蚂蚁 } k \text{ 走过路径 } k, \\ 0, \text{else} \end{array} \right\}$$

Q 为常数, L_k 为第 k 只蚂蚁在本次循环中所走过的路径的长度。

算法的实现流程如图 3 所示。



(a) 总的流程; (b) 蚁群算法流程

图 3 算法实现流程

4 应用实例

某公司是一家长期致力于印制电路板数控工艺装备研发的企业,改变了我国在印制板加工设备领域中长期依赖进口的局面。随着世界和我国电子信息制造业的蓬勃发展,市场给印制电路板加工工艺装备提出了更高的要求,其加工的速度和精度已成为数控钻铣床的生存和发展的主题。数控加工刀具轨迹的优化处理已成为制约印制电路板海量孔位加工生产效率的瓶颈环节。

实例平台: windows 2 000 研祥工控系统;

自主开发的 MT 全闭环控制系统;

日本松下 A 系列驱动系统;

台达变频器;

德国上银导轨及日本 THK 丝杠传动系;

WESTWIND 主轴系统(转速 100 kprpm/min, 可调);

X、Y、Z 向运动速度 40 m/min(可调);

定位精度 0.005 mm;

下钻频率 300 次/min;

加工对象:材料:覆铜板

孔数:11 768 个;

孔径数:9 种($\Phi 0.25$ —4 mm);

表 1 实验数据分析

孔径/mm	孔数	主轴转速 /kprpm	进给速度 /m/min	未处理路程 /mm	优化处理路 程/mm	缩短路程 /mm	
$\Phi 0.25$	300	100	8	1 672.5	1 451.75	220.75	结论: 未经过处理的路径长 55 385.9 mm, 经优化处理的路径长 42 139.35 mm, 共缩短 13 246.55 mm, 相比提高实际加工效率 23.9%
$\Phi 0.40$	1 110	100	8	2 460.8	2 160.2	300.6	
$\Phi 0.50$	4 668	100	8	38 648.6	28 031.7	10 616.9	
$\Phi 0.65$	1 400	80	12	1 806.4	1 435.6	370.8	
$\Phi 0.70$	2 008	80	18	3 468.6	2 678.8	789.8	
$\Phi 0.80$	1 152	70	25	2 605.6	2 149.3	456.3	
$\Phi 1.00$	1 050	60	30	1 865.4	1 542	323.4	
$\Phi 1.80$	60	40	30	1 860	1 790	70	
$\Phi 4.00$	20	40	30	1 000	900	100	

从实验数据(见表 1)可以看出,优化处理后的路径明显得到改善,提高实际加工效率 23.9%。若是在大批量、单件孔数特别多的情况下,改进效果更加明显。

5 结 论

作者面向工程应用提出了一种算法,用以解决 PCB 数控焊盘加工时的轨迹优化问题,该算法综合应用贪婪算法和蚁群算法,实际应用效果较好,加工效率提高 23.9%;但由于考虑到生产的实际问题,这一算

法所得到的结果只是近似的,尚需进一步提高算法的效率和性能,用以解决海量焊盘加工的轨迹优化问题。

参考文献:

- [1] ZBIGNIEW MICHALEWICA, DAVID B FOGEL 著. 如何求解问题—现代启发方法[M]. 北京:中国水利水电出版社,2003.
- [2] 梁吉元. CAM 系统中孔加工路径的优化处理[J]. 计算机集成制造系统, 2000, 6(2): 3-5.

(下转第 23 页)