CAPP 系统中基于孔特征信息约束最佳工艺路线的建模与求解

湖南怀化学院计算机系和数学研究所　乐光学

摘要： 通过分析分布于零件上孔的特征信息、实际制造环境和技术的约束条件，运用动态规划原理，提出了基于孔特征信息构造平面网络图的规则，建立了一个高效简洁的最佳工艺路线的求解模型和约束函数，并给出了具体的算法和效率分析。

关键词： 特征信息 ； 约束条件； 最佳工艺路线； 约束函数

1　问题的提出

在机械加工中，孔加工占有相当重要的比例。随着自动化加工技术的日益普及，各类自动化生成加工工艺路线的系统相继面市,但这类自动化加工路线生成系统,均是基于某种平台(即数控自动编程系统）上生成的，其生成的加工路线并非最优。通过实例分析发现，大多数此类系统均只考虑生成的 CNC 程序相对简单，而并非完全考虑效率和工艺路线最短。

2 最佳工艺路线的建模与求解[1][2][3]

分布于零件上的孔，其各种特征值一般是不同的，如孔径的大小，形位坐标，位置公差，不同机床的加工费用等问题。仅就同一种孔的加工而言，可描述为从原点出发，既不重复又不遗漏地加工完所有的孔，回到换刀点，再进行下一种孔径的加工，对数控编程这仅涉及到如何安排孔的加工（即走刀）路线，使空程移动时间最短，即所谓的最佳走刀路线问题。但实际加工中，在给定的制造环境和技术条件下，不仅要考虑走刀路径最佳，还要考虑效益、加工精度最好。

为了便于讨论，设分布于零件上孔的特征值相同，将孔视为节点，相互距离视为边，这样就将分布于零件上的孔转化为一个以孔为节点的平面网络图。从理论上讲，由任一节点（即孔）出发，可以不经过中间节点到达其余的 n-1 个节点，即每一节点有 n-1 条边。如按此法构造平面网络图，不但复杂冗余，且会导致搜索算法失效，于实际生产不相符。在此，我们引入相邻点的概念来构造平面网络图，其具体构图规则定义为 YGX0。如下：

1） 从分布于零件上的 n 个孔中，任取一节点 i（即第 i 个孔），连接与其相邻的孔，即边，并求其长度。
2） 取第 i+1 个节点，重复 1）的过程。当从点 i+1 到点 i 已存在一条边时，这条边不再生成。
3） 重复上述过程，直到第 n 个节点（即孔）。

那么就构成了一个节点为 n 的无向连通图，可表示为 G=(V,E,C,F),其中，V 为节点数，‖V‖=n，V={V_i }，V_i=(x_i，y_i)即V_i为第 i 个节点的相对坐标位置，E 为两点的边，

E={e_{ij}}, $e_{ij}=\sqrt{(x_i-x_j)^2-(y_i-y_j)^2}$,C 为代价函数，C= {C_i}，C_i为所选用加工机床集中的第 i 台机床的代价，为一常数，F 为机床坐标移动的速度，这就转化为在无向连通图 G 中求最短路径问题。运用最小代价生成树的方法来求解。

设某零件上分布有 7 个孔，分别表示为 V1—V7，并设其特征值相同。则由 YGX0 法则生成的平面网络图如图 1 所示：

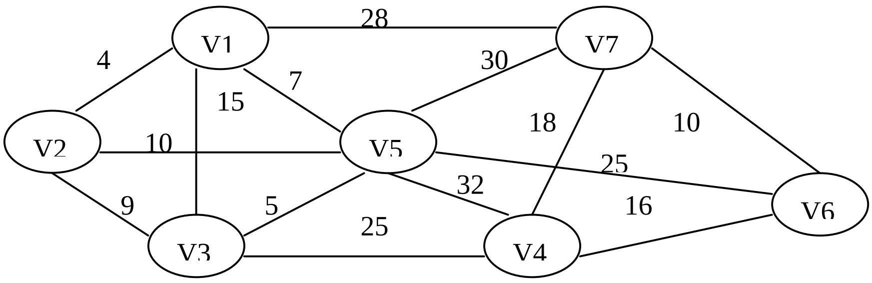


图 1

定义：设图 G=（V,E,C,F）是一个无向连通图，且‖V‖=n。G’=（V,E',C,F）,E'∈E,E'有且仅有 n 条边，如果 G’是 G 的连通子图，则称 G’是 G 的一棵生成树（spanning tree）。一棵生成树的代价定为这棵生成树各边的代价之和。在图 G 的一切生成树中，各边代价之和最小的一棵生成树称为图 G 的最小代价生成树（miaimum spanning tree）。

由定义，可构造如下约束函数：

$$\mathrm{COST}=\min\sum e_{ij}+C+F \qquad e_{ij}=\sqrt{(x_i-x_j)^2+(y_i-y_j)^2}$$

C=min {C_i} ,受制于加工孔位的精度特征值，即：孔位精度高，需选用高精度机床加工，费用越高，F=max {f_i}，受制于孔位精度和机床的驱动响应速度。

那么我们就可将图 G 转化为如下约束矩阵 E，设矩阵元素 e_{ij} 表示的第 i 个节点到第 j 个节点的距离，e_{ii} 无意义，由于求最小，因此，不能为零，设为无穷大，不相邻的孔位边值亦设为无穷大，那么约束矩阵 E 可表示为：

$$E=\begin{pmatrix} \infty & e_{12} & e_{13} & \cdots & e_{1m} \\ e_{21} & \infty & e_{23} & \cdots & e_{2m} \\ e_{31} & e_{32} & \infty & \cdots & e_{3m} \\ \vdots & \vdots & \vdots & & \vdots \\ e_{n1} & e_{n2} & \cdots & & \infty \end{pmatrix}$$

其中$e_{ij}=e_{ji}$。在算法描述中只对三角阵进行操作。

其最短路径求取算法可描述为：

1） 定义一个一维数组 T，它的初始状态为空，当算法结束时，T 中包含有最小代价生成树中的所有边。

2） VS 是一个不相交的顶点集的集合类。初始状态下，VS 中包含了‖V‖个顶点集，每个顶点集中只含有唯一的一个顶点。

3） 每次从边集 E 中选取一条未经考察的有最小代价的边（V，W）进行分析，如果 V 和 W 属于 VS 中的同一顶点集，则从 E 中删去这条边；如果 V 和 W 分别属于 VS 中的不同顶点集 W_1和 W_2，就将边（V，W）到 T 中，并将 W_1和 W_2 合并为一个集合。然后再从边集 E 中另选一条最小代价的边继续考察，直到找出一棵最小代价生成树为止。

其具体算法为：

```
procedure  YGX1
begin
   T ←φ；//置生成树边集 T 的初态//
   VS←φ; COST←O;
    将 E 中的边按代价从小到大排成递增序列 Q；
    for  每个顶点 V ∈ V  do 添 {V} 到 VS 中；
    while‖VS‖>1 do
    begin
       从 Q 中找出代价最小的边(V,W);
       从 Q 中删去边(V,W);
    if  V 和 W 在 VS 中的不同集合 W1 和 W2 中
      then begin
          将边 (V,W) 加入 T 中;
          合并集合 W1 和 W2;
          COST← COST + C(V,W)
          end
      end
    Write T 和 COST
```

end.

用 YGXI 算法考察图 1,其结果为：

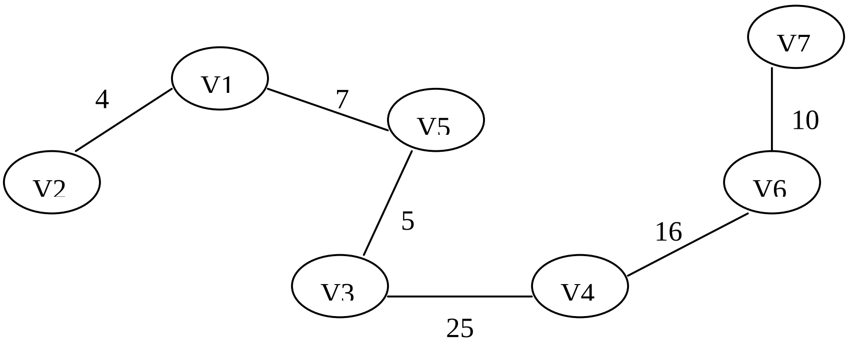
按代价函数从小到大给出的边表如下：

边	代价	边	代价
(V1,V2)	4	(V4,V6)	16
(V3,V5)	5	(V4,V7)	18
(V1,V5)	7	(V3,V4)	25
(V2,V3)	9	(V4,V7)	25
(V2,V5)	10	(V1,V7)	28
(V6,V7)	10	(V5,V7)	30
(V1,V3)	15	(V4,V5)	32

求出函数最小代价生成树的过程如下表所示

边	代价	操作	VS 中的集合状况	COST
(V1,V2)	4	加入 T	{V1,V2},{V3},{V4},{V5},{V6},{V7}	4
(V3,V5)	5	加入 T	{V1,V2},{V3,V5},{V4},{V6},{V7}	9
(V1,V5)	7	加入 T	{V1,V2,V3,V5},{V4},{V6},{V7}	16
(V2,V3)	9	放弃 F	{V1,V2,V3,V5},{V4},{V6},{V7}	16
(V2,V5)	10	放弃 F	{V1,V2,V3,V5},{V4},{V6},{V7}	16
(V6,V7)	10	加入 T	{V1,V2,V3,V5},{V6,V7},{V4}	26
(V1,V3)	15	放弃 F	{V1,V2,V3,V5},{V6,V7},{V4}	26
(V4,V6)	16	加入 T	{V1,V2,V3,V5},{V6,V7,V4}	42
(V4,V7)	18	放弃 F	{V1,V2,V3,V5},{V6,V7,V4}	42
(V3,V4)	25	加入 T	{V1,V2,V3,V5,V6,V7,V4}	67

生成的最小代价树为：



由此，我们可得出如下定理：

定理：设 G=（ V ,E ,C ,F）是一个连通无向图， S=（V, T ）是关于 G 的一棵生成树, 则有：

1） 对于 V 中的任何两个顶点 V 和 W ，在生成树 S 中， V 和 W 之间只有一条唯一的路径；

2） 如果树 E — T 中的任何一条边加入 T ，则 S 中将产生一条回路。

3） 算法时间开销为 O （‖E‖Log2‖E‖）。

3.实例分析与结论

图 2 是某制板厂的一台数控钻按一般 PCB—CAD 软件，使生成的钻孔走刀路线图（No，Citys=64 ），图 3 是利用本文算法生成的钻孔路线图，比较可知，其走刀空程缩短了 36.5%。

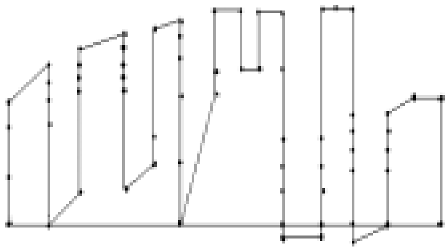


图 2 DT(tour.length)=173977.6 runtime=57.5S

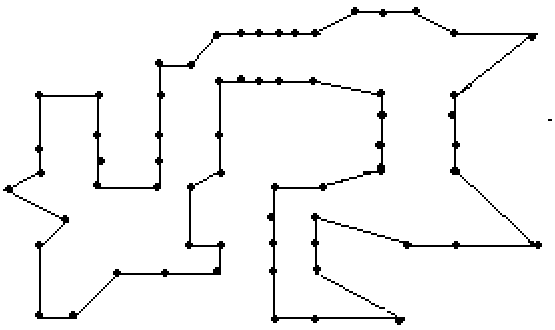


图 3 DT(tour.length)=110433.808. runtime=36.5S

通过大量的实例比较分析， YGX1 算法比一般的自动编程系统的生成路径缩短 25% 以上，效果显著，且该算法简单、实用、可靠，并能获得较好的解。该算法有效地解决了多孔

加工中，刀具路径冗长，空程太大导至加工效率低的问题，使孔加工数控工艺得以优化，并已应用于 JNCNC — CAPP 系统中。在实际应用中，代价函数 C 与零件的精度和加工工艺有关，制造环境和制造技术不同，有不同的代价开销；F 与选取的加工机床和零件的精度有关，F 越大，效率越高。因此，在具体操作时，需视具体情况而定。

参考资料

（1） 曹新谱编著 算法设计与分析 [M] 长沙湖南科学技术出版社 1984.11

（2） 卢开澄 组合数学算法与分析（下） 北京 清华大学出版社 1983.11

（3） 乐光学 数控加工程序编制中的工艺处理[J] 北京 《机械工艺师》 1993.6

***该项目为怀化市科委、怀化学院科研资助。编号：200104-02、200101-01**