

文章编号 1000-5013(2001)01-105-06

ADS-C 在 CAM 激光迭层成型加工系统中的应用

王孔碧^① 王依新^②

(① 福建省中心检验所, 福州 350002; ② 福州大学中心实验室, 福州 350002)

摘要 研究传统的激光迭层成型加工系统, 应用 AutoCAD 二次开发系统 ADS-C 设计三维实体的网格剖切分层、求交截断和接口转换等程序。将三维待加工实体预处理、实体处理和后续处理, 获得标准的三维网格截面轮廓数据, 为激光迭层成型加工提供输入数据面模型。

关键词 ADS-C, 三维实体, 激光迭层成型

中图分类号 TG 485 : TP 391.7

文献标识码 A

激光迭层成型加工(LOM)是一种快速成型技术, 也称为激光切割胶合成型。它采用 CAD 方法生成产品的三维实体模型, 通过软件切片程序, 使其沿高度方向离散化。然后, 用形成的分层信息控制激光束对每一薄层材料加工, 生成截面轮廓, 对于非零件部分则全部切碎, 以便于去除。当本层加工完成后, 再铺上一层材料, 接着使用热压滚筒将薄层材料滚压粘合。由于薄层材料单面涂敷了一层很薄的热敏感材料, 在材料表面达到一定温度后, 薄层材料之间就会很快胶熔在一起。如此反复, 直至加工完毕。最后去除切碎部分, 最终得到三维立体零件。激光迭层加工技术打破传统的零件加工界限, 大大缩短加工周期, 提高了生产的效率^[1,2]。该技术具有广阔的前景, 必将成为 RP 领域的重要组成部分^[3]。本文应用 AutoCAD 二次开发系统 ADS-C (AutoCAD Development System-C) 设计三维实体的网格剖切分层、求交截断和接口转换等程序, 为激光迭层成型加工提供输入数据面模型。

1 ADS-C 与 AutoCAD 和 Autolisp 的接口原理

低版本的 AutoCAD 编程语言一直是 Autolisp 语言, 但 Autolisp 存在严重的不足。ADS-C 的引入完全克服了 Autolisp 的缺点。首先, ADS 应用程序能扩展 Autolisp 的接口。其次, ADS 应用程序允许执行由 C 和 C++ 开发, 各种标准编译实现的二进制文件。在程序执行期间可与绘图环境通讯。ADS-C 已成为职业编程人员开发高级应用程序的强有力工具。

AutoCAD 通过专用的通讯缓冲器和 Autolisp 进行信息通讯。ADS-C 程序从 Autolisp 装

收稿日期 2000-08-03

作者简介 王孔碧(1974-), 男, 检验员

基金项目 福建省自然科学基金资助项目

入执行, 程序中包含有 ADS 与 AutoCAD 通讯初始化函数的调用. 完成初始化后, 它的所有外部函数(用户可定义的函数)都处于非激活状态, 直到 Autolisp 向它发出一个请求消息. 在 ADS 程序对 Autolisp 的请求消息作出响应过程中, Autolisp 和 AutoCAD 又都处于非激活状态, 等待从 ADS 程序返回结果. 这就是 ADS 程序无限循环地由消息进行驱动的机制^[6].

2 三维加工实体的网格剖切分层

用 ADS-C 编写的这套实体剖切分层程序, 与一般的 C 源程序相比, 有许多不同之处, 先从结果缓冲区开始叙述.

2.1 Result Buffer 的构造及实现

Result Buffer(结果缓冲区)是 ADS 支撑环境的核心概念之一. 一个结果缓冲区, 被用来代表 AutoCAD 实体和各种表的数据信息^[6].

本程序中共构造了 7 个 Result Buffer(结果缓冲区), 在如下的 Result Buffer 中定义.

```
struct resbuf
{
    struct resbuf * rbnext;    定义了指到下一个 "Result Buffer" 的指针变量
    short restype;            标识了该 "Result Buffer" 节点上同位字段
                                resval 值的类型标识变量
    union ads_u_val resval;    定义了存放该 "Result Buffer" 节点值的
                                resval 变量
}
```

其中 ads_u_val 是个综合同位类型变量, 在如下定义中说明

```
union ads_u_val
{
    ads_real rreal;
    ads_REAL rpoint[3];
    short rint;
    char * rsring;
    long rlname[2];
    long rlong;
    struct ads_binary rbinary;
}
```

2.2 实体剖切程序中外部函数的定义

本程序在实现三维实体的网格剖切分层之前, 先加入到 ACAD.ADS 中, 随 ACAD 的启动而预先装载, 然后通过编写的下拉菜单以及交互式的人机界面, 输入加工参数, 最终在屏幕上模拟三维实体分层剖切.

本程序的外部可执行函数, 由 ads_defun() 定义模块为

```
static int loadfunes()
{
    int return_code;
    if (ads_defun("C:cutting", 0) == RTNORM)
    {
        return_code = RTNORM;
    }
    else {
        return_code = RTERROR;
    }
    return return_code;    定义和调用模块段结束
}
```

2.3 三维实体剖切分层中实体的选择

在三维实体剖切加工中所选择的实体有两种,一种是单一实体,另一种为实体组集(选择集)。在 ADS 环境中,它们属于 `ads_name` 类型。这两种实体要通过“结果类型码”(result type code)来区别。对于单一实体,其“结果类型码”为“RTENAME”,选择集则为“RTPICKS”。在实际的 ADS-C 编程中,要特别注意这一点,以免程序运行时出错,给调试带来极大困难。

ADS 程序对某个单一实体或选择集进行操作前,先要调用一个库函数获得其名称(选择集和实体名是一对长整型数,由 AutoCAD 定义和支持),这就是建立实体选择集。建立实体选择集方法很多,而 `ads_ssget()` 函数提供了建立选择集最一般的方法。它有提示用户选择实体,直接指定要选择的实体(包括使用 PICKFIRST Set, Crossing Polygon, Fence, Last, Previous, Window 及 Window Polygon 等方法),列出一些条件或属性来过滤式选取当前图形数据库等。在三维加工实体剖切中,主要使用列表条件过滤式、关系条件过滤式及布尔条件过滤式来选取实体集。建立了选择集之后,就可以利用函数 `ads_ssadd()` 和 `ads_ssdel()` 将实体加进选择集或把实体从选择集中删掉。同时,`ads_sslength()` 函数提供了选择集中实体的数目。这样,我们就完全可以对图形数据库中的实体进行单一或组合的操作,以完成三维实体自动而非交互的剖切分层加工^[8]。

本程序中,三维加工实体剖切后形成的“Region”(区域)图元,是通过过滤条件列表方式来选择。对于单一的图元(如直线、圆、圆弧、椭圆、样条曲线……),则要通过布尔条件过滤方式来选取。

2.4 三维实体网格剖切分层程序流程分析

在程序总体流程图中,最重要的是实体网格剖切分层模块的设计。该模块设计符合 ADS-C 编程规则,作为总程序中实体分层模块,提供了由三维实体的 DWG 文件(或由其它三维实体文件转换而来)输入。经过剖切、分层、网格处理化等步骤,最终形成具有实际意义的分层数据,为后续的插补等运算提供了必要的输入数据面模型。

程序先初始化有关变量,为后续代码的执行做好准备。接着程序执行函数 `ads_entnext()`,从图形数据库中得到没有被删除实体的实体名,赋给 `ads_name` 型变量 `ent_laser`。执行 `for()` 循环语句,判断截面数目是否大于零。如果等于或小于零,则调用 `ads_ssfree()` 函数释放 `ent_laser` 选择集,退出本模块。若大于零,执行剖切分层程序代码。得到选择集后,按预定的剖切方式,根据事先确定好的剖切面的位置,进行实体的剖切工序,在图形数据库中生成一个“REGION”实体,并使存放截面数参数的变量值减一。

通过过滤条件列表方式,在图形数据库中获取“REGION”实体,接着程序通过 `ads_command()` 函数执行移动和剖切所选“REGION”实体的 AutoCAD 命令。完成上述操作后,程序再次判断图形数据库中是否还有“REGION”实体,使用一开关变量 `tag`,循环进行“REGION”区域的分解,直到图形数据库中查找不到“REGION”实体。部分程序代码为

```
rg.restype= 0;
strcpy(sbuf1,"REGION"); 指定实体类型为“REGION”(区域)
rg.resval.rstring= sbuf1;
rg.rbnxt= NULL; 接下去指定其它性质
if(ads_ssget("X",NULL,NULL,&RG,rg_allname)!= RTNORM)
```

```
{ads _fail("rg _allname error!");  
ads _exit(1);} 取得所有 'REGION' 实体
```

在实体网格剖切加工程序调试的过程中, 开始先选用简单的三维实体(如半球、空心圆柱体、空心立方体), 程序调试比较顺利, 生成几百个数据面模型。模型中的信息, 符合预定的标准。后来增加了三维待加工实体的复杂性, 选用常见的零件三维模型(如茶壶、复杂零件铸造模型等), 结果程序调试出现错误。几经研究, 找不到解决方案。然后在程序中设断点运行, 终于发现出错在 'REGION' 实体选择集的构造上。原来, 一般的 'REGION' 实体在 'EXPLODE' 之后, 总会形成单一图元, 如 'LINE', 'CIRCLE', 'ARC' 等。这样实体分解一次后, 形成的图元全都是不能分解的单一图元, 后续程序得以顺利运行。但是, 对于较复杂的 'REGION' 实体, 经过一次 'EXPLODE' 之后, 不会形成单一图元, 而是分解成多个 'REGION' 实体。这种实体若不经过多次分解, 后续程序根本无法处理, 导致程序异常出错。所以, 分层模块中采取 '循环搜索' 的办法, 顺利解决了这个问题。

经过上述步骤处理的分层实体还不能成为最终的数据面模型, 它还要经过网格处理这一工序。分层图元网格化后, 网格部分的实体类型(DXF 码值为 0) 是 'INSERT' 型。同样地, 通过过滤条件列表取得该实体, 并判断存在性, 然后分解该实体以形成最终可用的单一图元。

最后是输出到 DXF 文件模块。首先通过布尔条件过滤方式获得层面上所有单一图元, 并判断存在性。若不存在, 程序异常退出; 若存在单一图元, 先变换其层属性, 将其从原始 0 层变换到 LASER 层。接着修改颜色属性, 然后利用 AutoCAD 输出模块, 将分层上的所有图元输出到 rgxxxx.dxf 文件中。最后把该层层面图元从图形数据库中删除, 返回循环语句开头, 开始实体第二层层面的剖切加工……这样层层加工, 最终程序生成上百至千个(视需要而定)模型文件。

3 三维加工实体生成网格截面后的求交截断处理

三维加工实体剖切分层后, 生成网格截面, 可以进一步对其进行求交截断处理。即在两两图元处, 将图元求交并打断, 形成新的图元, 以此类似, 遍历网格面上的每一个交点, 将所有图元由交点处截断, 形成一个新的图形数据库, 最终输出成 DXF 文件。这样就为以后的插补运算及后续处理提供了精确的输入数据面模型。在本文设计的求交截断子程序中, 核心部分是进行二维图形的布尔运算。

4 三维加工实体网格剖切面的预处理及后续处理

4.1 三维加工实体的预处理

利用 AutoCAD 生成的三维加工实体, 它的预处理包括两个部分, 首先是加工实体各种属性的修改, 如层属性的修改等, 使之适合本文设计的软件。其次是将三维待加工实体定位在指定的边界之内, 使之底边界的 X, Y, Z 坐标值均为零。这样, 使每一个待加工的三维实体均有统一的基准点(原点), 也便于以后对每个截面进行网格剖分。更重要的是, 保证了与实际的激光切割加工有公共的加工基点。

4.2 实体选择集的释放处理

对于一个 ADS 应用程序, 它不能同时打开多于 128 个的选择集, 这一限制由多种因素决定. 因此, 应使用 `ads_essfree()` 函数来优化实体选择集. 这就是说, 当包含在选择集中的数据不再需要的时候, 应及时将其释放掉. 本程序是在使用完选择集后, 马上释放该选择集. 这样, 在 ADS 程序运行期间, 不会因为内存中残留有旧的信息而使内存空间变得过于混乱, 最终导致程序非法出错.

4.3 DXF 文件图元信息处理

三维加工实体经过网格剖切分层、求交截断形成上千个 DXF 文件后, 包括了图形数据库的所有信息, 如层属性、线型、颜色号等. 这此信息对于激光分层加工无任何实际意义. 因此, 优化处理 DXF 文件图元信息, 去除一些不必要的信息, 尽可能减少图形存储量, 是三维实体网格剖切生成 DXF 文件后续处理中不可缺少的一步.

DXF 文件经过本文编写的接口软件模块转换后, 生成了自定义的格式文件, 本软件自定义的格式文件也遵循 DXF 文件的格式规范, 符合国际标准化的图形交换格式, 便于插补运算等后续处理. 图元信息处理的效果是显而易见的, 对一个圆锥体进行三维剖切分层生成 DXF 文件, 每个 DXF 文件大约占了 17.4 KB 的磁盘空间. 经过本接口模块转换后, 生成的自定义格式文件大约占了 7.7 KB 的磁盘空间, 比原来的 DXF 文件减少了 56%. 按一千个文件计, 则共少占用 9.7 MB 的磁盘空间.

4.4 DXF 文件图元筛选排列处理

AutoCAD 是一种完整的图形编辑系统,它具有十分紧凑的图形数据库,其中存储着有关当前的全部图形信息和管理信息.由它生成的 DXF 文件能够完全表达图形的所有信息,且被其它应用程序或软件所接受,成为国际上通用的标准^[6].DXF 文件中各图元也是按照一定的次序来排列的,利用本软件对三维实体网格剖切处理,对其产生的任意一个 DXF 文件进行定性分析,发现它的排列顺序如图 1 所示.对于激光加工系统的 LOM 工艺来讲,激光的扫描方式也是极其重要的,而图 1 所示的图元排序则不符合传统的激光连贯扫描方式(图 2).为

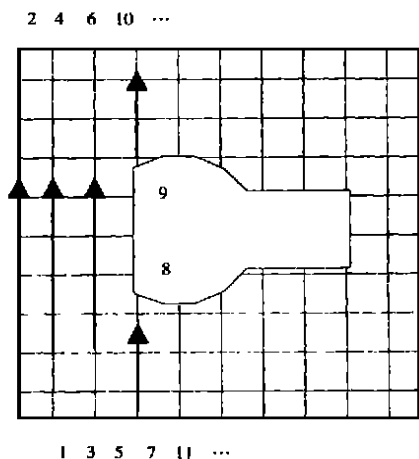


图 1 原始 DXF 文件排序规则

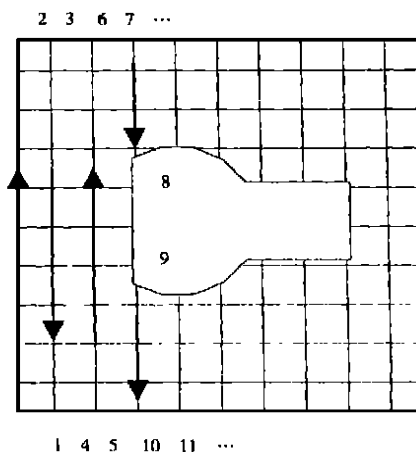


图 2 激光加工连贯扫描方式

此,必需先对 DXF 文件进行图元的筛选排列,使得它的次序符合传统的激光扫描方式,以优化

激光加工路线, 尽可能减少激光加工带来的能量损耗.

5 结束语

本文设计的三维实体网格剖切分层、求交截断、接口转换等模块化程序, 是基于 AutoCAD R13C4 版, 利用 VC 语言编程在 VC++ 5.0 环境下编译链接形成的. 软件的操作平台使用 Delphi 4.0 编程语言设计, 具有图象三维立体性能和图、声、文并茂的效果. 同时提供了在线式的帮助信息, 使用户操作更为方便、直观. 使用由 AutoCAD 提供的 DCL-对话框语言编程, 设计了交互式输入参数界面, 几组参数输入只需用鼠标点击就可实现, 使三维实体剖切的实现更为迅速.

参 考 文 献

- 1 颜永年, 张 伟, 卢清萍. 基于离散/堆积成型概念的 RPM 原理和发展[J]. 中国机械工程, 1994, 5(4): 64~66
- 2 尹希猛, 王运赣, 黄树槐. 快速成型技术-90 年代新的造型工具[J]. 中国机械工程, 1993, 4(6): 25~27
- 3 Bautenbach G C, McMillin D. Desktop manufacturing LOM vs pressing[J]. American Ceramic Society Bulletin, 1994, 73(8): 80~113
- 4 方 铁. 语言高级编程[M]. 北京: 清华大学出版社, 1997. 87~103
- 5 Autotools 工作群编. AutoCAD ADS-C 程序设计实务[M]. 北京: 清华大学出版社, 1995. 92~99
- 6 Jamieson R, Hacker H. Direct slicing of CAD models for rapid prototyping[J]. Rapid Prototyping Journal, 1995, 1(2): 4~12

Application of ADS-C to the CAM System of Laser Laminated Moulding

Wang Kongbi^① Wang Yixin^②

(^① Fujian Central Institute of Inspection, 350002, Fuzhou; ^② Central Lab., Fuzhou Univ., 350001, Fuzhou)

Abstract Starting from a study of conventional laser laminated moulding system, an AutoCAD secondary development system or ADS-C for short is applied to the programming for gridded slicing lamination of three dimensional object, truncation at intersectional point and change of interface. By pretreating the three dimensional object to be moulded and treating the object and follow-up treatment, the data of standard three dimensional gridded sectional profile can be obtained so as to provide the laser laminated moulding with a model of input data face.

Keywords ADS-C, three-dimensional object, laser laminated moulding