

利用 ObjectARX 方法实现 NC 车削自动编程系统的图形输入

文 | 石鑫¹ 王忠建¹ 刘丽华¹ 石梦琳² 1. 中航工业江西洪都航空工业股份有限公司 2. 中南林业科技大学

一、引言

AutoCAD 软件在机械制造领域已经得到了广泛的应用, 能够完成各种机械零件设计及图形输入任务, 然而其所绘制的图形数据结构不能为 CAPP、CAM 等相关模块所利用, 为此需要对 AutoCAD 的图形数据进行读取并转化成数控编程系统所需要的数据结构。国内外学者对此作了很多研究工作, 如利用 AutoLISP 读取 DXF 文件等方法对 AutoCAD 的图形文件进行读取识别。然而这些方法仍然有很多的局限性, 不能进行信息的全面转化。针对上述问题, 本文利用 ObjectARX3.0 方法读取 AutoCAD2000 数据库的图形信息及 ID 值来识别图素的属性并进行相应的数据处理, 转换成 NC 加工所需要的基于特征的数据结构, 从而为 CAPP、CAM 等相关模块提供数据源, 最终实现 NC 数控编程自动化。

二、ObjectARX 简介

ObjectARX3.0 是 Autodesk 公司针对 AutoCAD 2000 平台的二次开发而推出的一个工具包, 它支持面向对象的编程方法 (OOP)。ObjectARX3.0 应用程序是一种 Windows 动态库。

ObjectARX 由以下类构成:

- (1) AcRx 用于绑定应用程序和运行时类的注册。
- (2) AcEd 用于注册内部命令和系统事件通知的类。
- (3) AcDb AutoCAD 数据库类。
- (4) AcGi 用于渲染 AutoCAD 实体的图形界面。

所有 ObjectARX 应用程序必须与 acad.Lib 和 raapi.Lib 连接, 根据使用的 ObjectARX 类库和函数的前缀不同, 还

可能需要其它的附加链接库如表 1 所列。

表 1 附加链接库

前缀	所需的库
AcRx	Acad.lib, rxapi.lib, acrx15.lib
AcEd	Acad.lib, rxapi.lib, acedapi.lib, acrx15.lib
AcDb	Acad.lib, rxapi.lib, acdb15.lib, acrx15.lib
AcGi	Acad.lib, rxapi.lib, acgiapi.lib, acrx15.lib
AcGe	Acad.lib, rxapi.lib, acge15.lib, acrx15.lib

三、总体结构

1、总体结构的设计

在 AutoCAD 环境下进行零件设计与零件工程图的绘制, 根据零件工程图信息进行加工特征的定义和零件特征模型的描述。图形输入系统的总体结构如图 1 所示。

2、几何数据结构

本程序的数据源来自 AutoCAD 数据库, 利用 ObjectARX 方法从数据库中读入数据需要存储以便进一步进行数据分析, 为此建立了图素的几何数据结构, 如下:

```
struct STusu
```

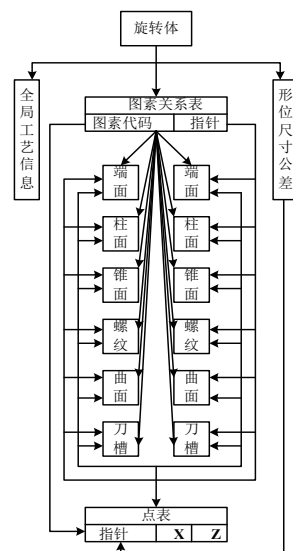


图 1 系统总体结构

```
{long x1,x2,y1,y2,c1,c2;
long R;//若为圆弧则表示圆弧的半径坐标,对其他图素
没有意义
double startAngle,endAngle;//若为圆弧则表示圆弧的起、
止角度值,对其他图素没有意义};
注: x1, y1——直线(或圆弧,或样条曲线)的始点坐
标。
x2, y2——直线(或圆弧,或样条曲线)的末点坐标。
c1, c2——若为圆弧则表示圆弧的中心点坐标,对其他
图素没有意义。
```

3、零件工艺信息

由于零件是以实体特征面构建的,所以各特征面的工艺信息可能是不同的,有的工艺信息是所有表面共有的,称作全局工艺信息。实体表面工艺信息是指不同的表面类型而具有不同的工艺参数。本文主要考虑了以下工艺信息:硬度、粗糙度、表面处理、轴向尺寸公差、直径尺寸公差、直线度、圆柱度、平行度、圆度、垂直度、同轴度、径向全跳动和径向圆跳动等。

四、程序设计

本程序分为数据读入模块、数据处理模块及工艺信息输入模块。

1、读入模块

读入模块是利用 ObjectARX 方法从 AutoCAD 数据库中读取零件的信息,并保存在前面确定的几何信息数据结构中,以便接下来的数据处理。读入模块源代码流程图如图 2 所示。

以下是读入模块遍历所

有实体的源代码:

```
void CNCDocData::ReadTxt()
{//打开当前块表
AcDbBlockTable*pBlkTbl;
acdbHostApplicationServices()-
>workingDatabase()
->getSymbolTable(pBlkTbl,AcDb:
:kForRead);
//打开模型空间块表记录
AcDbBlockTableRecord*pBlkT
```

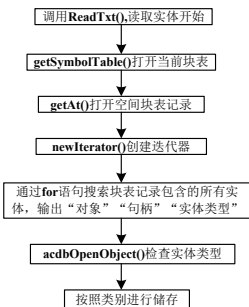


图 2 读取模块流程

```
blRcd;
pBlkTbl->getAt(ACDB_MODEL_SPACE,pBlkTblRcd,AcDb:
:kForRead);
pBlkTbl->close();
//创建迭代器
AcDbBlockTableRecordIterator*pBlkTblRcdItr;
pBlkTblRcd->newIterator(pBlkTblRcdItr);
AcDbEntity *pEnt;
//搜索块表记录包含的所有实体
int num=0;
ads_name Name;
for(pBlkTblRcdItr->start();!pBlkTblRcdItr->done();
pBlkTblRcdItr->step())
{pBlkTblRcdItr->getEntity(pEnt,AcDb::kForRead);
AcDbHandle objHandle;
pEnt->getAcDbHandle(objHandle);
acdbGetAdsName(Name,pEnt->objectId());
char handleStr[20];
objHandle.getIntoAsciiBuffer(handleStr);
const char*pCname=pEnt->isA()->name();
acutPrintf(“\n 对象ID:%lx,句柄: %s,
实体类名: %s\n”,pEnt->objectId(),handleStr,
pCname);
//检查实体类型
acdbOpenObject(pEnt,pEnt->objectId(),AcDb::kForRead);
int type=GetObjectType(Name);
检查实体的类型然后按照类别进行存储。
```

2、数据处理模块

数据处理模块共有三个函数:

Data process1 函数: 寻找圆点 $O(OX,OY)$, 删除中心线, 把坐标系转化为以 $O(OX,OY)$ 为圆点的坐标系。

Data process2 函数: 取图形的上半部分, 并将多余的图素删去。

Data process 3 函数: 通过对零件的几何信息进行分析, 提出了一种方便快捷的图形轮廓搜索法, 从而得到图形的外轮廓数据。它对从 AutoCAD 数据库读入的数据进行了进一步处理, 从而判断出各图素的工艺特征(柱面、端面 and 锥面等)。

数据处理的具体思路：搜索初始图素（有一端点坐标值为 (0,0) 的图素），将它的末点赋给 `endPoint0`，然后利用 `endPoint0` 的值继续搜索下一个图素（下一个图素与上一个图素首尾相接），每搜索到符合要求的点时就将其存储到 PPON 结构中。如果搜索到的图素的 X 坐标值等于 X_{max} （上个模块中得到的值，它是所有图素中 X 坐标的最大值），则图形的外轮廓搜索完成。接下来判断有没有内孔，通过将剩下的图素与外轮廓的最后一个图素进行求交，若求交成功则说明内孔存在，然后按照外轮廓搜索的规则继续进行搜索直到所得图素的 Y 坐标值为 0，再判断是否存在左、右盲孔，若存在就继续是使用上述方法进行搜索，直到所搜索到图素的 X 值为 0，全部搜索结束，得到图形的外轮廓。

3、主要图素的处理

（1）直线、圆弧的处理

直线程序：利用 ObjectARX 类库所提供的 `startPoint()` 及 `endPoint()` 函数得到直线的起点与终点坐标值。具体使用如下：

```
startdot=((AcDbLine*)pEnt)->startPoint();
```

```
enddot=((AcDbLine*)pEnt)->endPoint();
```

在进行外轮廓的搜索时，如果所搜索到的点是直线的终点，将直线的起点与终点进行调换。

圆弧程序：利用 ObjectARX 类库所提供的 `startAngle()`、`endAngle()` 及 `center()` 函数分别得到圆弧的起始角、终止角及圆弧的中心坐标值。具体实现如下：

```
startAngle=((AcDbArc*)pEnt)->startAngle();
```

```
endAngle=((AcDbArc*)pEnt)->endAngle();
```

```
centerdot=((AcDbArc*)pEnt)->center();
```

圆弧旋向的处理：在 AutoCAD 数据库中圆弧无论是顺时针绘制，还是逆时针绘制，其存储都是按照逆时针进行的，在进行图形外轮廓的搜索时，如果所搜索的点是圆弧的终止点，则将圆弧的旋向取反。

（2）样条曲线的处理

为了从 AutoCAD 数据库中读入样条曲线的型值点，需要利用 ObjectARX 所提供的函数。经过仔细的分析从 ObjectARX 所提供的类库中（`db spline.h`）找到了可以得到样条曲线型值点、始末端点及切线斜率等关键值的函数，其声明如下：

控制端点：

```
Acad::ErrorStatus getFitData(
AcGePoint3dArray& fitPoints,
int&degree, double&fitTolerance,
Adesk::Boolean&tangentsExist,
AcGeVector3d&startTangent,
AcGeVector3d&endTangent)const;
```

其中 `AcGePoint3dArray&fitPoints` 定义了一个动态的数组，用来存储搜索到的型值点坐标。

函数的使用如下：

```
((AcDbSpline*)pEnt)->getFitData(Points,Order,FitTolerance,TangentsExist,StartTangent,EndTangent)
```

从而得到了动态数组 `Points`，它存储样条曲线所有的型值点坐标。

得到了型值点的坐标后，接着利用追赶法得到绘制三次 B 样条曲线所需的控制点，经过插补得到 B 样条的拟合点。将这些点存储在点表 PPON 结构中，从而完成对样条曲线的处理。

4、工艺信息输入模块

根据零件整体及各特征形面的不同工艺要求，可以输入全局工艺信息或局部工艺信息。实验件的轴端面的工艺信息输入如图 3 所示。



图 3 轴端面工艺信息输入

五、实验结果与讨论

为了考察本系统程序运行的结果是否达到预期的要求，及其形成的文件是否满足 NC 数控加工所需的数据结构。首先在 AutoCAD 环境下绘制如图 4 所示的回转体实验件。

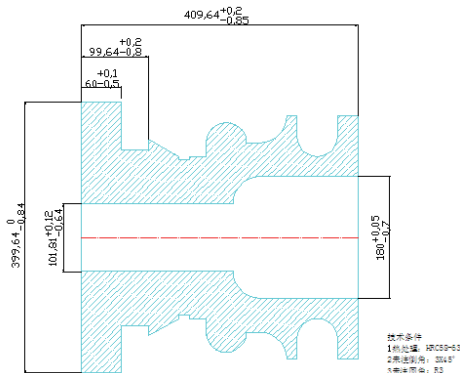


图 4 回转体实验件

加载应用程序 NCINTPUT.arx, 读取 AutoCAD 所绘制的图形并进行数据处理, 程序系统的主界面如图 5 所示。

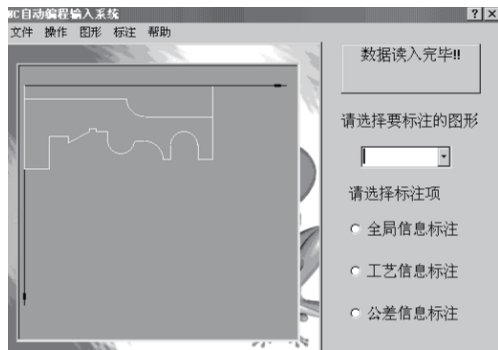


图 5 程序系统主界面

至此已经得到了实验件的几何图形信息, 这还不能满足数控加工的需要, 接下来需对零件的工艺信息进行输入, 全局工艺信息输入如图 6 所示, 其中, 淬火 C 表示渗碳淬火, 强度与韧性单位为 MPa。

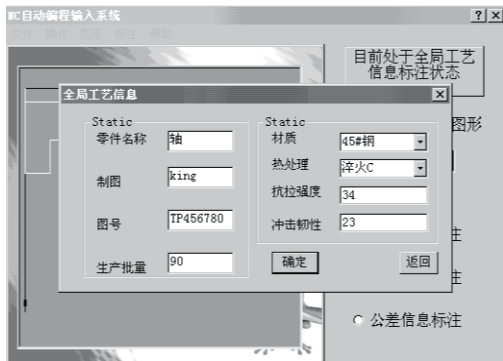


图 6 全局工艺信息输入

为了验证本程序所得数据的正确性, 对所形成的文件进行读取并重现了图形的轮廓如图 7 所示。

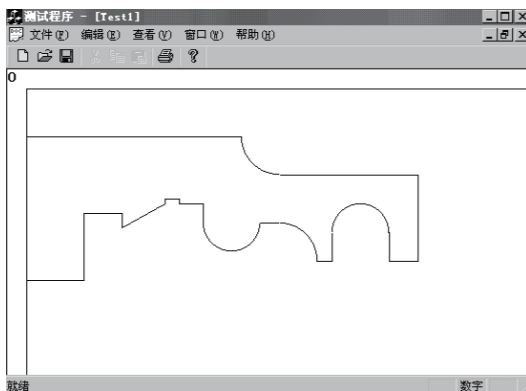


图 7 图形轮廓重现

六、结论

本文利用 VC++6.0 语言, 通过 ObjectARX3.0 类库读取 AutoCAD2000 数据库的图形信息, 从而实现 NC 自动编程系统的图形输入, 可以得到以下结论:

- ◎ 利用 ObjectARX 方法可对 AutoCAD 数据库进行操作;
- ◎ 搜索 AutoCAD 数据库中的图素时, 利用 ObjectARX 所提供的迭代器进行搜索;
- ◎ 进行工艺信息的标注时, 可随机选取图素进行工艺信息标注 (例如, 在点选外腔或内腔圆柱面后, 将弹出如图 8 所示的对话框提示用户输入圆柱面工艺信息)。

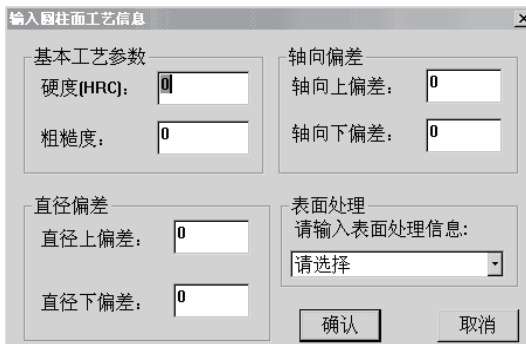


图 8 圆柱面工艺信息输入

经过测试, 利用 ObjectARX 方法及本系统程序所制定的数据转化规则所得到的数据符合 NC 自动编程所需要的数据结构形式, 实现了数控车削自动编程系统图形的快速高效输入, 提高了工训实践能力。