

# 数控弯管的空间转角数据转换算法

罗志猛, 柴苍修, 冯晓娟, 赵永辉

(武汉理工大学机电工程学院, 湖北 武汉 430070)

**摘 要:** 应用矢量弯管原理, 推导和比较了数控弯管的空间转角数据转换的三种算法; 并基于空间三角形算法, 采用可视化编程软件 VC++6.0 开发了三维弯管数控系统软件, 实现自动弯管。

**关键词:** 矢量弯管; 空间转角; 数据转换

**中图分类号:** TG659

**文献标识码:** A

**文章编号:** 1002-6673 (2005) 05-124-03

## 0 引言

弯管作为气体、液体的输送管路和结构件, 广泛应用于汽车工业、航空航天、石油化工等领域<sup>[1]</sup>。

对于弯管加工, 已从传统的手工操作生产方式, 在半自动弯管机上完成管件的成型, 逐渐过渡到采用数字化产品定义和用数控弯管机成型。而数控弯管机是基于矢量弯管原理<sup>[2]</sup>, 根据零件图上订制的尺寸 (或是实物测量尺寸), 通过输入专用的管件加工数据实现弯管自动控制过程。通常弯管的设计信息是以图纸上的连接点坐标来表示管件的图纸数据<sup>[3]</sup>; 由于数控系统所需的数据不是连接点坐标数据信息, 而是数控弯管机所专用的管件加工数据, 因此实现三维弯管从图纸数据向加工数据转换。其中重点是选择合理的算法完成空间转角数据转换, 是一个相当关键的设计内容。本文推导出了空间转角 POB 数据转换三种不同算法。并基于空间三角形算法, 开发了三维弯管数控系统软件。

## 1 矢量弯管原理

在矢量弯管中, 管形都看成是由直线段和圆弧组成。如果将管件放在迪卡尔坐标系中, 并将管件直线段中心线以一系列的空间矢量来表示, 求出各直线段的交点坐标, 即弯管图纸上连接点的坐标值; 然后利用矢量的基本概念和运算法则, 可以计算出增量管形数据, 即管件加工数据; 最后将管形加工数据输入给数控弯管机, 由弯管机自动弯制, 即可完成所需的三维成形弯管。数控弯管机所需要的增量管形数据包括以下三个方面内容: ①直线送进距离 P1B1; ②弯曲角度 DOB1、DOB2;

③前后两弯管平面之间的空间转角 POB。

三维弯管中管形的每一个弯都有此三类数据, 前一个弯的数据是后一个弯的数据的基础, 并依次逐步递增, 因此称为增量管形数据。

基于矢量弯管技术, 由图纸数据向加工数据的转换计算的三类参数中, 对于直线送进距离和弯管平面弯曲角度这两类参数的计算方法相对简单, 最关键的是前后两弯管平面之间的空间转角 POB 数据转换。

## 2 数据信息转换算法

假设有一段空间管件的管形, 如图 1 所示。给定管件图纸数据连接点  $P_1$ 、 $P_2$ 、 $P_3$ 、 $P_4$  坐标值, 如图 2 所示, 即  $P_1(x_1, y_1, z_1)$ ,  $P_2(x_2, y_2, z_2)$ ,  $P_3(x_3, y_3, z_3)$ ,  $P_4(x_4, y_4, z_4)$ 。根据直线段交点坐标计算出相邻两交点间距离, 即矢量  $P_1P_2$ 、 $P_2P_3$ 、 $P_3P_4$ 、 $P_1P_4$  的长度:  $|P_iP_j| = \sqrt{(x_i-x_j)^2 + (y_i-y_j)^2 + (z_i-z_j)^2}$ , (式中  $i, j=1, 2, 3, 4; j>i$ ); 矢量  $P_1P_2$ 、 $P_2P_3$ 、 $P_3P_4$  的方向余弦:  $\cos\alpha_i = \frac{x_i+1-x_i}{|p_i p_{i+1}|}$ ;  $\cos\beta_i = \frac{y_i+1-y_i}{|p_i p_{i+1}|}$ ;  $\cos\gamma_i = \frac{z_i+1-z_i}{|p_i p_{i+1}|}$ , (式中  $i=1, 2, 3$ )。

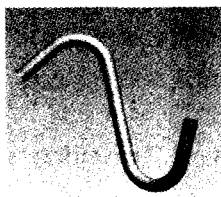


图 1 某空间弯管形状  
Fig.1 Shape of tube bending

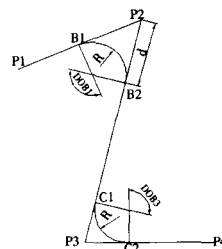


图 2 弯管管形数据  
Fig.2 Data information of tube bending

### 2.1 直线送进距离的计算

$$|P_iB_i| = |P_iP_{i+1}| - R \tan \frac{DOB_i}{2}, \quad |C_iP_{i+1}| = |P_iP_{i+1}| - R \tan \frac{DOB_{i+1}}{2},$$

收稿日期: 2005-06-23

作者简介: 罗志猛 (1979-), 男, 湖北孝感人, 硕士研究生。研究方向: 机电一体化。

$$|B_2C_1|=|P_2P_3|-R\tan\frac{DOB_1}{2}-R\tan\frac{DOB_2}{2} \quad (R \text{ 为弯曲半径})。$$

## 2.2 弯曲角度 DOB 的计算

矢量  $P_1P_2$  与矢量  $P_2P_3$  夹角余弦为:  $\cos DOB_1 = \cos\alpha_1\cos\alpha_2 + \cos\beta_1\cos\beta_2 + \cos\gamma_1\cos\gamma_2$ ; 弯曲角度:  $DOB_1 = \arccos(\cos\alpha_1\cos\alpha_2 + \cos\beta_1\cos\beta_2 + \cos\gamma_1\cos\gamma_2)$ ; 同理可得弯曲角度  $DOB_2$  为:  $DOB_2 = \arccos(\cos\alpha^2\cos\alpha^3 + \cos\beta_2\cos\beta_3 + \cos\gamma_2\cos\gamma_3)$ 。

## 2.3 空间转角 POB 的计算方法比较

(1) 采用方向余弦算法。采用空间两矢量的矢量积, 求出两相邻直线构成的平面的法线, 然后再用两空间矢量的数量积, 求出两法线的夹角即为空间转角 POB<sup>[4]</sup>。矢量  $P_1P_2$  与矢量  $P_2P_3$  的方向余弦:

$$\cos\alpha_{ix} = \cos\beta_i\cos\gamma_{i+1} - \cos\beta_{i+1}\cos\gamma_i$$

$$\cos\beta_{ix} = \cos\alpha_{i+1}\cos\gamma_i - \cos\alpha_i\cos\gamma_{i+1}$$

$$\cos\gamma_{ix} = \cos\alpha_i\cos\beta_i + 1 - \cos\alpha_{i+1}\cos\beta_{i+1} \quad (\text{式中 } i=1,2)$$

采用规格化办法, 求出矢量  $P_1P_2$  与矢量  $P_2P_3$  构成平面的法线  $P_2V_1$  方向余弦:  $\cos\alpha_{1d}$ 、 $\cos\beta_{1d}$ 、 $\cos\gamma_{1d}$ 。同理矢量  $P_2P_3$  与矢量  $P_3P_4$  构成平面的法线  $P_3V_2$  方向余弦:  $\cos\alpha_{2d}$ 、 $\cos\beta_{2d}$ 、 $\cos\gamma_{2d}$ 。进而求得空间转角 POB 大小为:  $|POB1| = \arccos(\cos\alpha_{1d}\cos\alpha_{2d} + \cos\beta_{1d}\cos\beta_{2d} + \cos\gamma_{1d}\cos\gamma_{2d})$ 。此外, 还应判断 POB 符号的正负, 即确定空间转角的旋转方向, 此时需求矢量  $P_2V_1$  和矢量  $P_3V_2$  的矢量积  $V_{xd}$  的方向余弦, 并规格化矢量  $(\cos\alpha_{xd}, \cos\beta_{xd}, \cos\gamma_{xd})$ , 最后判断该矢量与矢量  $P_2P_3$  的数量积  $d_0 = \cos\alpha_{xd}\cos\alpha_2 + \cos\beta_{xd}\cos\beta_2 + \cos\gamma_{xd}\cos\gamma_2$ , 若  $d_0 \geq 0$ , 空间转角为正向旋转; 反之则负向转动。

(2) 采用平面方程算法。由  $P_1$ 、 $P_2$ 、 $P_3$  点坐标, 确定  $P_1P_2P_3$  平面方程为:  $a_1(x-x_1)+b_1(y-y_1)+c_1(z-z_1)=0$ 。

$$\text{式中: } a_1 = \begin{vmatrix} y_2-y_1 & z_2-z_1 \\ y_3-y_1 & z_3-z_1 \end{vmatrix}, \quad b_1 = \begin{vmatrix} x_2-x_1 & z_2-z_1 \\ x_3-x_1 & z_3-z_1 \end{vmatrix}, \quad c_1 = \begin{vmatrix} x_2-x_1 & y_2-y_1 \\ x_3-x_1 & y_3-y_1 \end{vmatrix},$$

$$\text{同理 } P_2P_3P_4 \text{ 平面方程可表示为: } a_2(x-x_2)+b_2(y-y_2)+c_2(z-z_2)=0。 \text{ 其中 } a_2 = \begin{vmatrix} y_3-y_2 & z_3-z_2 \\ y_4-y_2 & z_4-z_2 \end{vmatrix}, \quad b_2 = -\begin{vmatrix} x_3-x_2 & z_3-z_2 \\ x_4-x_2 & z_4-z_2 \end{vmatrix}, \quad c_2 = \begin{vmatrix} x_3-x_2 & y_3-y_2 \\ x_4-x_2 & y_4-y_2 \end{vmatrix}。 \text{ 由此可以确定 } P_1P_2P_3$$

$$\text{平面与 } P_2P_3P_4 \text{ 平面的空间夹角为:}$$

$$\theta = \arccos \frac{a_1a_2+b_1b_2+c_1c_2}{\sqrt{a_1^2+b_1^2+c_1^2}\sqrt{a_2^2+b_2^2+c_2^2}}$$

$$\text{最后判断旋向因子 } d_1 = \begin{vmatrix} x_4-x_1 & y_4-y_1 & z_4-z_1 \\ x_2-x_1 & y_2-y_1 & z_2-z_1 \\ x_3-x_1 & y_3-y_1 & z_3-z_1 \end{vmatrix} \text{ 的正负}$$

$$\text{值, 若 } d_1 \geq 0, \text{ 则 } P_2P_3P_4 \text{ 平面在 } P_1P_2P_3 \text{ 平面正向, 空间转角为正向旋转; 反之则负向转动。}$$

(3) 空间三角形算法。①直线送进距离; ②求出平

面弯曲角度; ③空间转角。

$$\text{直线送进距离: } |P_1B_1| = a - \frac{R}{\tan(\angle p_1p_2p_3/2)} \quad |B_2C_1| = b - \frac{R}{\tan(\angle p_1p_2p_3/2)} - \frac{R}{\tan(\angle p_2p_3p_4/2)} \quad |C_1P_4| = c - \frac{R}{\tan(\angle p_2p_3p_4/2)}。 \text{ 式中: } a=|P_1P_2| \quad b=|P_2P_3| \quad c=|P_3P_4|。$$

求出平面弯曲角度  $\angle P_1P_2P_3$  与  $\angle P_2P_3P_4$ : 在  $\triangle P_1P_2P_3$  中, 利用余弦定理:  $\angle P_1P_2P_3 = \arccos \frac{a^2+b^2-e^2}{2ab}$ ,

$$\text{同理在 } \triangle P_2P_3P_4 \text{ 中 } \angle P_2P_3P_4 = \arccos \frac{b^2+c^2-f^2}{2bc}。$$

式中  $d=|P_1P_4|$ ;  $e=|P_2P_4|$ ;  $f=|P_1P_3|$ 。

空间转角:  $\triangle P_1P_2P_3$  与  $\triangle P_2P_3P_4$  所在平面之间的夹角:  $\alpha = \arccos \frac{l_1^2+l_3^2+l_5^2-d_2^2}{2l_1l_3}$ 。式中:  $l_1=a \times \sin \angle P_1P_2P_3$ ,  $l_2=a \times$

$\cos \angle P_1P_2P_3$ ,  $l_3=a \times \sin \angle P_2P_3P_4$ ,  $l_4=a \times \cos \angle P_2P_3P_4$ ,  $l_5=b-l_2-l_4$ 。判断弯管空间转角为正负旋向方法同 (2) 中  $d_1$  算法。

以上所导出的三种算法中, 方向余弦算法主要运用矢量的方向余弦来求解, 优点是判断空间转角的旋向方便简洁, 但整个过程推导冗长, 算法相当繁琐; 平面方程算法利用行列式求解, 并推导出判断空间转角的旋向因子  $d_1$ , 使得判断空间转角的旋向更为清晰, 但推导比较复杂; 空间三角形算法方法简单, 计算量小, 数据循环处理便捷, 逻辑性强, 但判断空间转角旋向时, 仍要借用旋向因子。因而以上三种方法各有优缺点, 相对而言, 空间三角形算法比较适合用于计算机编程。

## 3 软件实现过程

在自行开发的三维数控弯管仿真系统软件中, 采用 VC++6.0 作为开发工具, 运用空间三角形算法, 生

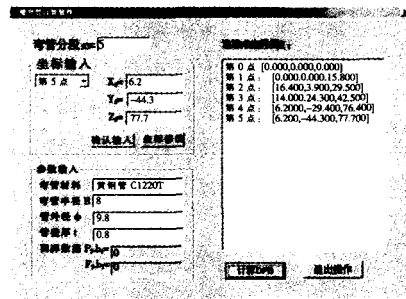


图3 坐标输入界面

Fig.3 Coordinate input window

成坐标输入界面如图3所示。首先根据图纸设计要求, 将弯管分段并输入分段数, 再输入管形数据关键点的  $x$ 、 $y$ 、 $z$  坐标值及弯管半径等相关参数。数控系统自动计算出弯管机所需的增量管 (下转第128页)

光和漫射光均采用 RGBA (181, 181, 181, 1), 同时设置毛坯材质为 RGB (181, 181, 181)。模型表面用一系列小三角形予以逼近和表示, 本文采用光滑明暗处理模式, 在这种模式下, 使用了双线性插值法, 使由三角形表示的曲面光强连续, 而且计算量很小。

可见面的确定在 OpenGL 中自动完成。为了减少计算量, 毛坯表面小三角形采用同一方向 (逆时针) 绘制, 并指定逆时针方向为前面 (FRONT-FACE)。

## 5 结论

本系统可以验证数控代码的正确性, 采用离散矢量

求交法对正确的数控代码进行三维模拟加工。在求交计算中利用加工的特殊性避免了求解曲面方程, 减少计算量, 而且计算出加工面的近似法向量, 使光照效果更接近实际加工环境。

### 参考文献:

- [1] 王晓斌, 王太勇, 等. 基于 VR 技术的三维数控仿真系统研究[J]. 精密制造与自动化, 2003, (增刊).
- [2] 马良, 张谓源. 服装衣片三角形网格自动剖分[J]. 东华大学学报 (自然科学版), 2002, 8.

## Research of NC Milling Simulation Based on OpenGL

WANG Yao-Guo, MENG Wen, HUANG Can-Quan

(Aptitude Mechatronic Technology Institute of Southwest Jiaotong University, Chengdu Sichuan 610031, China)

**Abstract:** Through examining the exactness of NC code, the system makes the orbit of the cutter. For the NC milling machining, building the triangular meshes according to Delaunay rule on the top-surface of the roughcast, cutting the roughcast step by step with the method that calculate the intersection between the discrete vector and cutter swept volume, and carry out the machining process simulation with OpenGL.

**Key words:** OpenGL; NC milling machine; Simulation

(上接第 125 页)

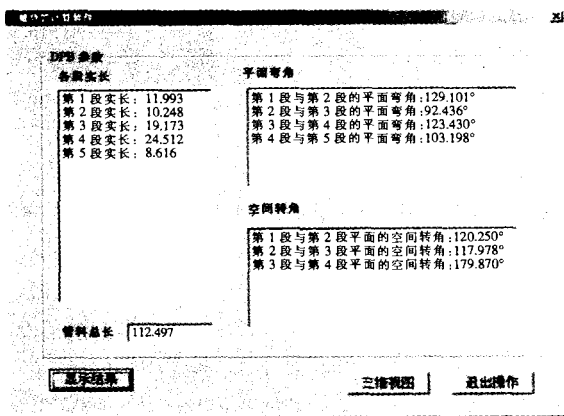


图 4 增量管形数据输出界面

Fig.4 Increment data output window

形数据, 生成界面如图 4 所示。

## 4 结束语

基于空间三角形算法原理编制的三维弯管数控系统软件, 自动计算生成弯管增量管形数据, 应用于三维数控弯管机上, 实际效果良好。

### 参考文献:

- [1] 赵臻淞, 杨合, 林艳, 等. 数控弯管仿真系统的设计与开发[J]. 锻压技术, 2003, 1.
- [2] 王立新. 矢量弯管[M]. 北京: 国防工业出版社, 1984.
- [3] 崔振勇, 印建平. 平面弯管的计算机辅助设计与绘图[J]. 现代制造工程, 2002, 12.
- [4] 倪霖, 张湘伟. 弯管机工程 CAD 中的转换计算及图像模拟[J]. 重庆大学学报 (自然科学版), 1997, 20(5): 34~38.

## The Data Conversion Algorithm about Three Dimensional Angle of Numerical Control Tube Bending

LUO Zhi-Meng, CHAI Cang-Xiu, FENG Xiao-Juan, ZHAO Yong-Hui

(School of Mechanical and Electronic Engineering Wuhan University of Technology Wuhan Hubei 430070, China)

**Abstract:** According to the method of vector tube bending, this paper deduces and compares the three algorithms about three dimensional angle of NC tube bending. And based on the space angle algorithm, which also belongs to the three algorithms. By using VC++6.0, the Three Dimensional NC Tube Bending System Software is designed and developed, and the automatic tube bending can be achieved.

**Key words:** vector tube bending; three dimensional angle; data conversion