

切削加工机器人与 CAD/CAM 系统集成化研究

肖文磊¹ 郇 极¹

(北京航空航天大学机械工程及自动化学院 北京 100191)

摘要: 使用 6R 工业机器人构架切削加工平台, 论述切削加工机器人与 CAD/CAM 系统连接和集成的数据流结构, 研究切削加工机器人的后置处理过程遇到的关键问题。阐述后置处理过程各个步骤的实现原理, 并以 REIS RV16 工业机器人为例, 建立切削加工机器人的原型系统, 对其后置处理过程的坐标系变换、运动学求解、冗余自由度和奇异点回避问题进行推导和论述。建立切削加工机器人的仿真和后置处理系统平台, 并完成 2D 和 3D 样件的加工。试验和仿真结果表明, 该系统能够完成切削加工机器人的后置处理过程及生成可执行的机器人加工指令。

关键词: 切削加工机器人 CAD/CAM 后置处理

中图分类号: TG659

Integration of a Cutting Robot with CAD/CAM System

XIAO Wenlei¹ HUAN Ji¹

(School of Mechanical Engineering and Automation, Beihang University, Beijing 100191)

Abstract: A cutting platform was established using a 6R industrial robot. The data flow for cutting robot to connect and integrate CAD/CAM system was studied. Some crucial problems which were encountered during the post-processing procedure were specially investigated. A prototype of cutting robot was established, on which some processing steps in the pos-processing process were studied and discussed. Those steps included transforming coordinate systems, solving kinematics, solving redundant degrees of freedom and avoiding singular points. A simulation and post-processing system was developed in order to perform the implementation of a cutting robot. Finally, some 2D and 3D workpieces were cut by the cutting robot, which verified the functionality and feasibility of the post-processor and the robot code generator inside the whole system.

Key words: Cutting Robot CAD/CAM Post-processing

0 前言

工业机器人在工厂自动化领域已经得到了广泛的应用。目前, 绝大多数的工业机器人是替代人的劳动, 作为搬运、焊接、喷漆等重复性等工作的执行器。在切削加工领域, 由于工业机器人的刚度和精度的不足, 其发展和应用一直受到限制。然而, 作为一种与数控机床相比有显著价格优势的替代方案, 其在一些大尺寸、低切削力和低精度要求的复杂曲面加工(如木材雕刻加工)时具有很大的优势和潜力。因此, 近些年来, 使用机器人与数控系统结合, 作为加工平台逐渐成为工业机器人的一个快速发展的应用方

向。

与工业机器人的其他可编程的重复性工作相比, 加工机器人需要实现更复杂的轨迹规划机制。在传统的数控领域, 主要是使用 CAD-CAM-CNC 的数据链完成复杂的零件模型输入和刀轨生成等任务。利用数控领域的相对较成熟的 CAD-CAM 技术实现加工机器人的轨迹规划功能将是一个减少开发工作量和缩短开发周期的最佳方案。因此, 加工机器人与传统的 CAD/CAM 的数据连接和系统集成将成为工业机器人数控加工的一个简便和必然的选择。目前, 国际上主要的加工机器人原型均采用这种数据链模式^[1-4]。

由于一般工业机器人控制器不接受传统 CAD/CAM 系统输出的 APT 或 G 代码等刀轨

* 20101125 收到初稿, 2011 收到修改稿

数据，因此开发数控代码到工业机器人语言的转换器和后置处理器成为数控加工机器人建立系统和连接CAD/CAM的核心任务之一。加工复杂三维曲面时，数控机床一般使用三坐标数控代码进行粗加工，使用五坐标数控代码进行精加工。对于三坐标代码，加工机器人后置处理器中主要应具有坐标系变换、逆运动学运算、运动学多解的唯一性确定等基本功能。对于五坐标数控加工代码，加工机器人还应具有去冗余自由度、奇异点回避、轴超限保护及碰撞检测等高级功能。

Albele等研究了基于笛卡尔坐标和五轴工业机器人的加工系统，并针对加工机器人切削铝材的切削力模型和误差校正进行了研究^[1]。Scheifele等使用KUKA机器人建立了加工机器人系统，并对使用ISG公司开发的虚拟制造系统virtuos对加工过程进行了建模和仿真^[2]。随着近几年工业机器人越来越广泛的应用，ABB和KUKA也相继推出了加工机

器人模块^[3-4]。已有的加工机器人研究和应用对结合NC技术的冗余问题很少讨论，而该问题的解决对加工机器人的后置处理器和代码的自动优化有重要意义。本文使用REIS RV16工业机器人作为研究实例，建立了一套切削加工机器人系统。使用SolidWorks和AlphaCAM作为CAD和CAM系统平台，生成G代码，并针对该机器人构建了加工机器人的后置处理器和虚拟制造仿真系统，以实现针对G代码的后置处理和转换为REIS机器人语言的功能。提出了针对后置处理过程的几种优化算法。最后，使用经过后置处理的机器人语言代码对加工机器人进行控制，实现示例零件的加工。

1 系统整体结构及数据链分析

1.1 CAD-CAM-Robot 数据链

借助于传统CAD-CAM-CNC的数据链模型，加工机器人系统数据流如图1所示。

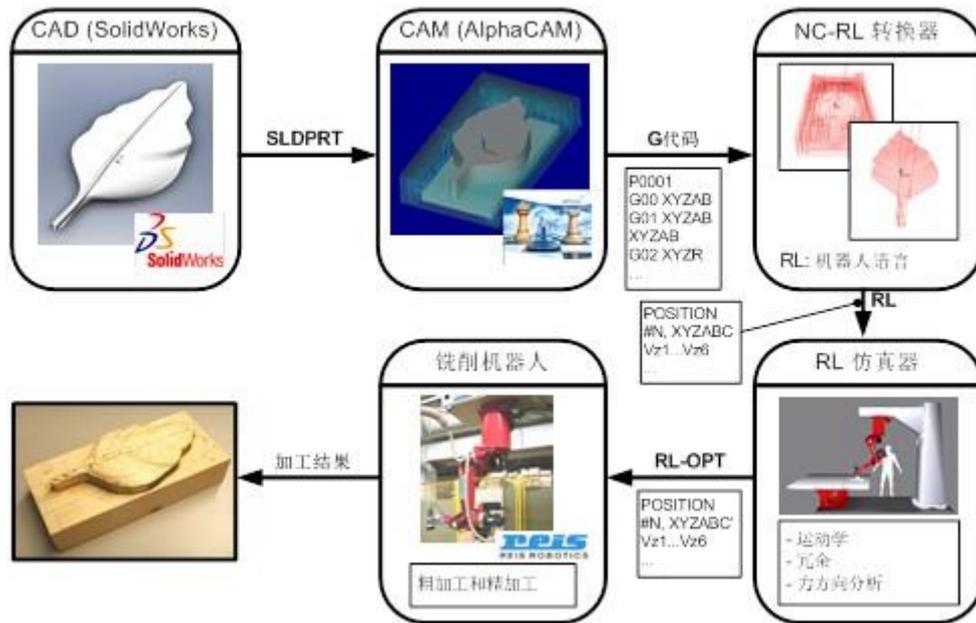


图1 CAD-CAM-Robot 数据链原理

在本文构建的切削加工机器人系统中,首先,使用 SolidWorks 进行系统建模,然后将 CAD 模型导入 AlphaCAM 软件中进行毛坯定义、加工策略选择和加工刀轨生成的工作。然后将输出的 G 代码进行坐标系变化并转换为机器人语言(Robot Language, RL)。之后,使用加工机器人后置处理和仿真器对输出的 RL 进行仿真验证、后置处理及轨迹优化,以输出机器人控制器可执行的优化机器人代码(RL-OPT)。最后,将 RL-OPT 代码下载到机器人控制器中,驱动加工机器人最终完成零件的加工。其中,由 G 代码(或刀位数据)转换到可执行机器人代码的过程统称为加工机器人后置处理过程。

1.2 加工机器人后置处理过程

与传统 CAD-CAM-CNC 数据链相比,切削加工机器人最大的不同之处为加入了加工机器人后置处理过程。加工机器人的后置处理过程及组成模块如图 2 所示。G 代码是针对传统数控机床设计的,本文重点研究五坐标机床的控制代码。

由于求解机器人的逆运动学算法时,需要使用欧拉角进行表示机器人末端执行器的坐标和方向,即 $(x, y, z, \alpha, \beta, \gamma)$ 。在由 G 代码向欧拉角的变换过程中,使用刀轴的矢量表达式作为中间表示将在很大程度上简化坐标变换过程。

由于五坐标 G 代码对欧拉角的 3 个分量 (α, β, γ) 中的 γ 没有指定(γ 为刀具轴线旋转方向),即一般六轴工业机器人对于五轴数控代码是冗余的。对于一般 6 自由度机器人的逆运动学过程,需要预先给定 γ ,即去冗余过程。这个值将由优化过程给出。优化过程将利用冗余的 γ 角(或冗余自由度),通过自运动完成回避奇异点、防止碰撞或轴超限等任务。

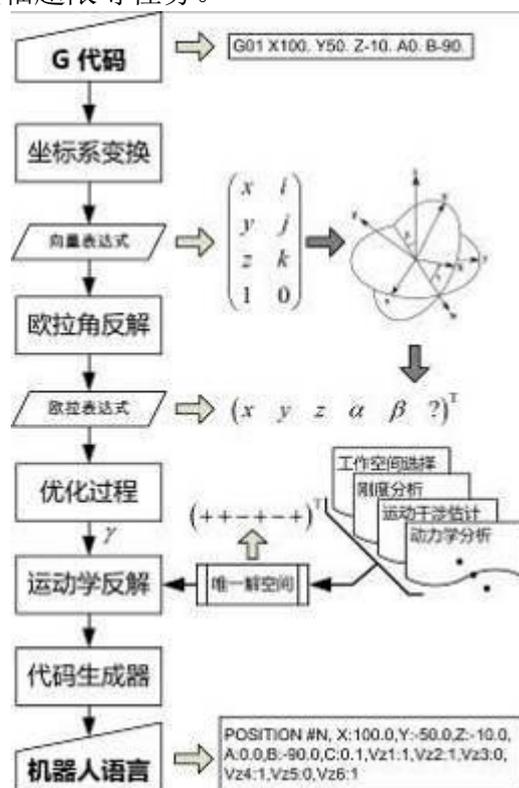


图 2 切削加工机器人的后置处理过程

在运动学反解过程之前,除了去冗余过程,还需要对解空间进行设定。对于指定的刀具点位,机器人存在的解往往不唯一。唯一解空间的确定一般由工件装夹位置、刚度分析、干涉估计以及动力学分析等方面决定。这些方面由于篇幅限

制, 在本文将不做论述。

在后置处理的最后阶段, 将针对机器人语言的定义输出机器人控制器可接受的控制代码。机器人控制代码分两部分组成。第一部分指定了点位的坐标值和方向角, 如: X:100.0, Y:-50.0, Z:-10.0, A:0.0, B:-90.0, C:0.1, 第二部分指定了唯一解空间, 如: Vz1:1, Vz2:1, Vz3:0, Vz4:1, Vz5:0, Vz6:1。每个 Vz 分量给出了对应轴的正负性。

2 坐标系定义及转换

2.1 G 代码坐标系定义及转换

G 代码使用专业标准 ISO 841-1974 给出的坐标系定义。在该标准中, 五坐标铣床和加工中心的坐标轴按以下规则给出^[5]。

(1) z 坐标垂直于机床工作台装夹面, 正方向是增大工件和刀具间距离的方向。

(2) x 坐标平行于工件装夹面, 其正方向为由主轴向主柱看时, 指向右方。

(3) y 坐标及其正方向根据 x 和 z 运动, 按右手笛卡尔坐标系确定。

(4) 旋转坐标 A、B、C 的轴线相应地平行于 x、y、z 的旋转运动, A、B、C 的正向, 相应地表示在正 x、y、z 方向上按右旋螺纹前进的方向。

(5) 标准坐标系的原点是任意的, A、B、C 的运动的原点可相应的平行于 y、z 和 x 轴。

根据上述定义和坐标变换原理, 五坐标机床的 G 代码坐标表示和刀轴矢量表示之间的关系如下所示。

定义角算符 $X(\theta)$ 为绕 x 轴旋转 θ 角的旋转矩阵, 则以下给出了刀轴矢量与 G 代码坐标之间的关系式

$$\begin{pmatrix} x & i \\ y & j \\ z & k \end{pmatrix} = \begin{pmatrix} x & 0 \\ y & 0 \\ z & 0 \end{pmatrix} + \mathbf{R}\mathbf{R}' \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & -1 \end{pmatrix} \quad (1)$$

$$\mathbf{R} = \mathbf{R}' = E_r \left\{ \begin{matrix} \mathbf{X}(A) \\ \mathbf{Y}(B) \\ \mathbf{Z}(C) \end{matrix} \right\} \quad (2)$$

$$\mathbf{X}(A) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c_A & -s_A \\ 0 & s_A & c_A \end{pmatrix} \quad (3)$$

$$\mathbf{Y}(B) = \begin{pmatrix} c_B & 0 & s_B \\ 0 & 0 & 0 \\ -s_B & 0 & c_B \end{pmatrix} \quad (4)$$

$$\mathbf{Z}(C) = \begin{pmatrix} c_C & -s_C & 0 \\ s_C & c_C & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (5)$$

式中, \mathbf{R} 为 G 代码中第一个旋转坐标对应的旋转矩阵, \mathbf{R}' 为 G 代码中第二个旋转坐标对应的旋转矩阵。 E_r 表示其右侧旋转矩阵集合内的枚举, 根据 G 代码中旋转坐标 A、B、C 的类型, 选择对应的旋转矩阵。式中运算符 sin 简写为 s, cos

简写为 c

以 AB 型五坐标数控代码为例, 其变换关系为:

$$\begin{pmatrix} x & i \\ y & j \\ z & k \end{pmatrix} = \begin{pmatrix} x & 0 \\ y & 0 \\ z & 0 \end{pmatrix} + X(A)Y(B) \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & -1 \end{pmatrix}$$

$$= \begin{pmatrix} x & -\sin B \\ y & \sin A \cos B \\ z & -\cos A \cos B \end{pmatrix} \quad (6)$$

如果按齐次坐标形式给出, 则为:

$$\begin{pmatrix} x & i \\ y & j \\ z & k \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} x & -s_B \\ y & s_A c_B \\ z & -c_A c_B \\ 1 & 0 \end{pmatrix} \quad (7)$$

公式(7)给出了刀尖和刀轴的矢量表达式。

2.2 机器人坐标系定义及转换

当今的机器人制造厂家和研究单位开发的机器人语言各不相同, 对点位的坐标系定义也不尽一致。本文针对 REIS 机器人语言的坐标系定义为对象进行讨论。REIS 机器人对末端执行器方位角的定义如图 3 所示。

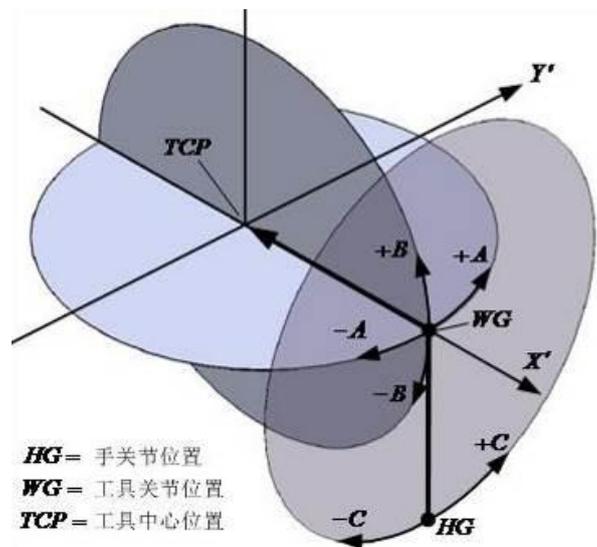


图 3 REIS 机器人末端执行器方向角定义

根据 REIS 机器人的坐标系定义, 下式给出了刀轴的矢量表达式与机器人旋转角 A 、 B 、 C 之间的关系 (这里的 A 、 B 、 C 不同于五坐标机床定义的旋转坐标, 因此记为 A' 、 B' 、 C'):

$$\mathbf{R} = \begin{pmatrix} c_{B'}c_{C'} & -c_{B'}s_{C'} & s_{B'} \\ c_{A'}s_{C'} + c_{C'}s_{A'}s_{B'} & c_{A'}c_{C'} - s_{A'}s_{B'}s_{C'} & -c_{B'}s_{A'} \\ s_{A'}s_{C'} - c_{A'}c_{C'}s_{B'} & c_{A'}s_{B'}s_{C'} + c_{C'}s_{A'} & c_{A'}c_{B'} \end{pmatrix}$$

$$= \begin{pmatrix} c_\alpha c_\gamma - c_\beta s_\alpha s_\gamma & c_\alpha s_\gamma & c_\beta s_\alpha c_\gamma \\ s_\alpha c_\gamma + c_\beta c_\alpha s_\gamma & s_\alpha s_\gamma & c_\beta c_\alpha c_\gamma \\ s_\beta s_\gamma & s_\beta c_\gamma & c_\beta \end{pmatrix} \begin{cases} A = \arctan\left(\frac{-s_\beta c_\gamma}{c_\beta}\right) \\ B = \arcsin(s_\beta s_\gamma) \\ C = \arctan\left(\frac{-s_\alpha c_\gamma - c_\beta c_\alpha s_\gamma}{c_\alpha c_\gamma - c_\beta s_\alpha s_\gamma}\right) \end{cases} \quad (8)$$

式中, α 、 β 、 γ 为 zxz 顺规的欧拉角。 s_i 表示角度的正弦, c_i 表示角度的余弦, $i=A'$ 、 B' 、 C' 、 α 、 β 、 γ 。若 B' 角为零, A' 、 C' 角的反解将出现奇异, 此时对 A' 、 C' 角赋值为 $A'=C'=0$ 。

2.3 矢量表达式与欧拉角表达式的转换

在求解机器人的优化过程前, 需要采用欧拉表达式, 使得冗余自由度 γ 与其他分量不相关。因此, 需要将式(7)的矢量表达式转换为欧拉表达式。将欧拉角旋转合成矩阵的第 3 列与式(7)中的方向矢量联立可得

$$\begin{pmatrix} i \\ j \\ k \end{pmatrix} = \begin{pmatrix} -s_\beta s_\alpha \\ s_\beta c_\alpha \\ -c_\beta \end{pmatrix} = \begin{pmatrix} -s_B \\ s_A c_B \\ -c_A c_B \end{pmatrix} \Rightarrow \begin{cases} \alpha = \arctan\left(\frac{s_B}{s_A c_B}\right) \\ \beta = \arccos(c_A c_B) \end{cases} \quad (9)$$

式中, 角 A 、 B 为 AB 型五坐标数控代码的两个旋转坐标。由公式(9)可以看出, γ 不能由坐标 A 、 B 导出, 因此 γ 是冗余的。

3 运动学模型



图 4 REIS 切削加工机器人

图 4 给出了本文使用的切削加工机器人的实体图。REIS RV16 是一个典型的 6R 机器人(具有 6 个旋转轴的串联机器人), 具有 6 可操作的自由度。其对于五坐标数控代码具有 1 冗余自由度。

在图 2 中的优化过程中, 需要结合机器人运动学算法进行讨论, 因此本文首先给出加工机器人的正、逆运动学算法。

3.1 正运动学算法

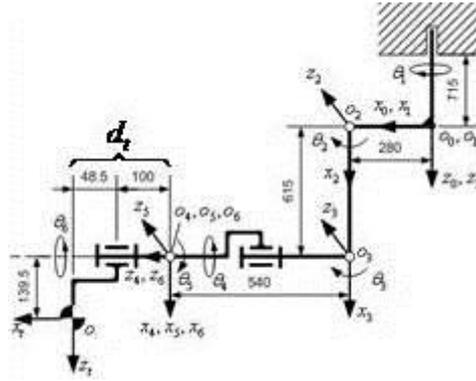


图5 使用 D-H 法构建的机器人坐标系

D-H 法 (Denavit-Hartenberg Convention, D-H Convention) 是机器人常用的建模方法, 可以很方便地建立串联机器人的连杆的坐标系和变换矩阵。图 5 给出了使用 D-H 法 REIS RV16 机器人的坐标系。表 1 为对应各连接副的 D-H 参数。

表 1 REIS RV16 的 D-H 参数表

i	a_{i-1}	$a_{i-1}/(\text{mm})$	$d_i/(\text{mm})$	θ_i
1	0	0	0	θ_1
2	$-\pi/2$	280	0	$\theta_2 - \pi/2$
3	0	615	0	θ_3
4	$-\pi/2$	0	540	θ_4
5	$\pi/2$	0	0	θ_5
6	$-\pi/2$	0	0	θ_6

将 D-H 参数代入式(10), 可求解出机器人的正运动学方程

$$\begin{pmatrix} x & i \\ y & j \\ z & k \\ 1 & 0 \end{pmatrix} = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_t^6 \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (10)$$

$$T_i^{i-1} = \begin{pmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & a_{i-1} \\ s_{\theta_i} c_{\alpha_{i-1}} & c_{\theta_i} c_{\alpha_{i-1}} & -s_{\alpha_{i-1}} & -d_i s_{\alpha_{i-1}} \\ s_{\theta_i} s_{\alpha_{i-1}} & c_{\theta_i} s_{\alpha_{i-1}} & c_{\alpha_{i-1}} & d_i c_{\alpha_{i-1}} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

3.2 逆运动学算法

机器人的逆运动学算法是对式(10)求反解的过程。一般, 3 自由度以内的机器人的逆运动学问题一定存在封闭解(或解析解), 而除了一些特例, 超过 5 自由度的机器人都不存在逆运动学方程的封闭解^[6]。而对于 6R 机器人, 当最后 3 个关节轴线相交于一点时(如 REIS RV16, 参考图 5), 也属于存在解析解的特例, 通常这种求解过程称为机器人的解耦过程。假定冗余自由度 γ 已经确定, REIS RV16 的逆运动学求反解由下面步骤给出:

首先, 对 D-H 模型的 O_4 点进行解耦, 将六个旋转轴划分为 $(\theta_1, \theta_2, \theta_3)$ 和 $(\theta_4, \theta_5, \theta_6)$ 两组, 分别进行求解。根据欧拉角变换原理, 求解 O_4 的坐标为

$$\begin{pmatrix} x_4 \\ y_4 \\ z_4 \end{pmatrix} = \begin{pmatrix} x - d_t (c_\alpha c_\gamma - c_\beta s_\alpha s_\gamma) - a_t s_\beta s_\alpha \\ y - d_t (s_\alpha c_\gamma + c_\beta c_\alpha s_\gamma) + a_t s_\beta c_\alpha \\ z - d_t s_\beta s_\gamma - a_t c_\beta \end{pmatrix} \quad (11)$$

式中, a_t 和 d_t 由图 5 中给出。

然后, 利用解耦点的坐标值, 求出前三个轴的解值。此处假设 $\theta_3 \in [-\pi/2, \pi/2]$, 如此逆运动学解可以唯一确定。

$$\theta_1 = \arctan\left(\frac{y_4}{x_4}\right) \quad (12)$$

$$\theta_2 = \arctan\left(\frac{x'_4 - a}{z'_4}\right) - \arccos\left(\frac{r^2 + a_2^2 - d_3^2}{2a_2 r}\right) \quad (13)$$

$$\theta_3 = \frac{\pi}{2} - \arccos\left(\frac{a_2^2 + d_3^2 - r^2}{2a_2 d_3}\right) \quad (14)$$

式中, x'_4 、 z'_4 表示 O_4 在 $O_1x_1y_1z_1$ 坐标系的坐标值, r 表示 O_2 和 O_4 之间的距离。

最后, 求解出后 3 个轴的解值。由前 3 个轴的解可以确定 $O_4x_4y_4z_4$ 坐标系在全局坐标系的方位, 进而可以推导出 $O_6x_6y_6z_6$ 与 $O_3x_3y_3z_3$ 的坐标变换矩阵 \mathbf{R}_6^3 。令

$$\mathbf{R}_6^3 = \begin{pmatrix} r_{00} & r_{01} & r_{02} \\ r_{10} & r_{11} & r_{12} \\ r_{20} & r_{21} & r_{22} \end{pmatrix}$$

则

$$\theta_4 = \arctan\left(\frac{r_{22}}{-r_{02}}\right) \quad (15)$$

$$\theta_5 = \arccos(r_{12}) \quad (16)$$

$$\theta_6 = \arctan\left(\frac{-r_{11}}{r_{10}}\right) \quad (17)$$

此处, 同样假定 $\theta_3 \in [0, \pi]$ 以确定唯一解。

4 去冗余及优化过程

4.1 机器人的冗余自由度

如前所述, 6R 机器人对于 G 代码是冗余的。要实现机器人最后的控制, 必须在生成最后机器人控制代码之间进行去冗余处理。目前, 使用 6R 工业机器人作为加工的去冗余处理方式主要有以下几种。

(1) 固定机器人的第 6 个旋转轴, 使得机器人的可控制自由度降为 5 个^[1]。

(2) 将切削主轴安装在其轴线与机器人末轴轴线重合的位置, 此时机器人的末轴与主轴耦合, 退化为 5 个可控制的自由度。

(3) 采用去冗余算法, 在轨迹规划和后置处理时求解冗余自由度的安全值。

前两种方式处理形式简单, 计算方便, 但损失了机器人的 1 可控自由度, 因此会降低加工机器人的灵活性。第(3)种方式需要加入去冗余算法, 计算相对复杂, 但能够利用机器人的冗余自由度进行避奇异点和避障等操作, 因此更具灵活性。本文采用第(3)种方式, 安装主轴轴线与末轴轴线正交。针对三坐标和五坐

标的 G 代码分别设计去冗余方案。

对于三坐标 G 代码, 由于欧拉角 α 和 β 恒定, 机器人在指定范围的工作空间内的运动不应造成主轴方向和自转角的变化, 可以简单的设定 $\gamma = 0$ 。然后进行逆运动学求解。

对于五坐标 G 代码, γ 不仅应跟据 α 和 β , 而且可以考虑其他优化指标进行调节。这些优化指标可以从不同的方面考虑。跟据机器人和加工的特性, 可以由以下几个方面考虑。

(1) 奇异位形或可操作度的最优化。在五坐标加工中, 奇异点对于数控机床和加工机器人是一个很关键的问题。机器人在奇异点的运动将造成某些关节的超速和超差, 从而对机器人造成危害及在加工零件上造成粗大误差。因此需要根据机器人结构对奇异点进行回避。

(2) 壁障和防止碰撞。在自由曲面加工时, 主轴需要在空间中连续摆动, 其运动过程将有可能造成机械臂与其他障碍物(如工作台、夹具)或零件发生碰撞。可以利用机器人冗余的自由度对机器人的姿态进行合理规划, 使得在运动过程中机械臂能够绕过或远离障碍物和加工的零件。

(3) 干涉检查。干涉检查与避障不同, 当输入的控制指令超过机器人的调节能力, 将与工作台或零件必然发生干涉。干涉检查通常在壁障处理之后, 首先检测出程序中存在的干涉点, 然后对干涉点的指令进行修改、添加或删除, 使其最终满足加工的安全性。

(4) 轴限位控制。由于机器人在结构上非线性, 从加工代码上很难对超过轴限位的点位作出判断。在考虑其他因素时, 需要将这个指标作为轴变量的限制。

由于机器人冗余优化是一个复杂的非线性最优化求解过程, 本文不对上述方面逐一论述, 仅以奇异点回避为例, 研究利用冗余自由度进行的指定目标优化的算法。

4.2 奇异点回避算法

奇异点的回避可以使用 3 种方法实现: 几何分析求解法、直接数值计算法和间接数值计算法。第 1 种方法对产生歧奇异的可能几何位形进行分析, 在奇异点处对欧拉角 γ 增加一个合适的补偿量, 使得机器人绕过可能的奇异点; 第 2 种方法是在逆运动学封闭解的基础上建立以 γ 为变量的雅各比矩阵函数, 以该函数为目标函数, 加入轴约束条件, 作为一个非线性最优化问题进行研究; 第 3 种方法以机器人的可操作度作为极大化的目标函数, 采用扩展雅各比矩阵的方法求机器人轨迹规划的可行解。

4.2.1 几何分析求解法

6R 串联机器人的奇异性主要分为 3 种: 肘部奇异、肩部奇异和腕部奇异^[7]。

(1) 肘部奇异发生在 $\theta_3 \approx -\pi/2$ 时, 这种情况往往是因为运动点位超出了机器人的工作空间, 因此又称为边界奇异。

(2) 肩部奇异发生在 O_4 点落在 z_0 轴上时, 这种情况使得 O_4 上沿 y_1 轴方向的微小位移将会造成 θ_1 上极大的转速。

(3) 腕部奇异发生在 z_4 和 z_6 共线时, 这时在末端执行器上的微小位移将会造成 θ_4 上极大的转速。

本文假定机器人在加工过程中一直处于工作空间范围之内, 且 O_4 点远离 z_0 轴(这些条件在安装夹具位置时能够保证)。因此, 不考虑肘部奇异和肩部奇异的情

况, 而只针对腕部奇异进行研究。

在奇异点回避算法之前, 需要对 γ 角进行去冗余预处理。

设 O_4 在 $O_0x_0y_0$ 平面上的投影为 O'_4 , O_i 在 $O_0x_0y_0$ 平面上的投影为 O'_i 。求解 γ 值, 使得 $O_0O'_4$ 与 $O_0O'_i$ 共线, 且 O'_4 尽可能靠近 O_0 。其几何求解关系如图6所示。

由图6中的几何关系可得

$$\gamma = \arcsin \left[\frac{a_i s_\beta (k s_\alpha + c_\alpha)}{d_i \Delta} \right] - \arctan \left(\frac{s_\alpha - k c_\alpha}{k s_\alpha c_\beta + c_\alpha c_\beta} \right) \quad (18)$$

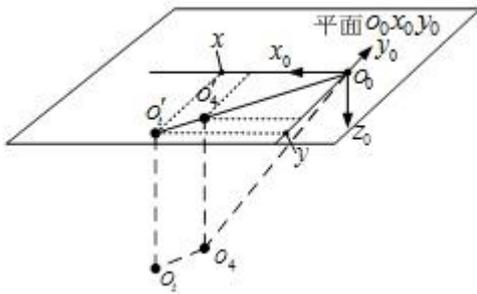
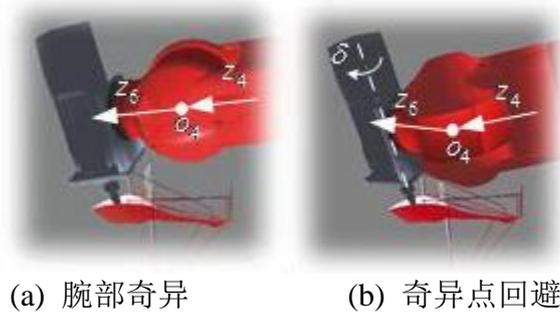


图6 γ 角去冗余预处理投影原理

经过式(18)的计算之后, 机器人基本处于合理姿态, 但没有检测和回避奇异点。腕部奇异一般发生的姿态如图7a所示。由图7a中可以看出, 由于 O_4 足够低, 奇异点发生的条件是存在一个明显不为零的 β 角。因此, 可以构建一个以 β 为自变量的 γ 角补偿因子

$$\delta = \lambda_1 s_\beta s_\alpha + \lambda_2 s_\beta c_\alpha \quad (19)$$

为保证运动轨迹的连续性, 引入作为 $\sin \alpha$ 和 $\cos \alpha$ 调整因子, 其调整权重 λ_1 和 λ_2 由经验值给出。加入补偿因子之后, 机器人在通过奇异点时能够自动产生一个对 γ 的补偿量, 从而达到奇异点回避的目的(图7b)。



(a) 腕部奇异

(b) 奇异点回避

图7 机器人腕部奇异及使用补偿因子回避奇异点

4.2.2 直接数值求解法

对于复杂结构, 尤其是不存在运动学解析解的机器人, 数值解法是一个行之有效的通用性方法。首先考虑非冗余情况, 式(20)给出了进给速度 $\dot{\mathbf{r}}$ 与关节空间速度 $\dot{\boldsymbol{\theta}}$ 之间的关系式。更一般化地, 机器人发生奇异是由于不可逆的雅可比矩阵造成的, 即式(20)中的线性齐次方程无解。因此, 回避奇异点可以通过最大化 $\det(\mathbf{J})$ 实现(\mathbf{J} 为 6×6 的方阵)。

$$\mathbf{J}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} = \dot{\mathbf{r}} \quad (20)$$

由于机器人的后 3 个关节的轴线相交于一点，其雅各比矩阵可以表示为以下的简单形式^[8]：

$$J = \begin{pmatrix} J_{11} & J_{12} \\ J_{21} & \mathbf{0} \end{pmatrix} \quad (21)$$

式中， $\mathbf{0}$ 为 3×3 的零矩阵，其余的是 3×3 的子块，具体计算过程可参考文献[8]。

如果只考虑腕部奇异的情况，可以只研究 J_{12} ，从而导出最优化目标

$$\det(J_{12}) = \mathbf{e}_4 \times \mathbf{e}_5 \cdot \mathbf{e}_6 \quad (22)$$

\mathbf{e}_4 、 \mathbf{e}_5 和 \mathbf{e}_6 分别是轴 z_4 、 z_5 和 z_6 方向上的单位矢量。应当指出，它们都是关于 $\{\theta_i\}$ 的函数。

现在构造最优化目标函数，设 $v = \{\theta_i, \gamma\} \in \mathbf{R}^7$ ， $i = 1, 2, \dots, 6$ ，则

$$f(v) \propto |\det(\mathbf{e}_4 \times \mathbf{e}_5 \cdot \mathbf{e}_6)|$$

约束函数定义为

$$h_j(v) \propto \Theta_j(\gamma) - \theta_j = 0, j = 1, 2, \dots, 6$$

式中， $\Theta_j(\gamma)$ 为以 γ 为自变量的逆运动学方程的解。

综上所述，可以将求解过程转换为一个典型的非线性最优化问题

$$\begin{aligned} \max \quad & f(v) \\ \text{s.t.} \quad & h_j(v) = 0, j = 1, 2, \dots, 6 \end{aligned} \quad (23)$$

使用数值解法求解上式即可得出 γ 的最优解^[9]。

4.2.3 间接数值求解法

直接法通过对 γ 直接求优得出。以下给出间接求解 γ 的原理和过程。冗余自由度机器人的控制常常会采用速度控制并对 $\dot{\theta}$ 积分的控制律(如图 8 所示)。此时雅各比矩阵 J 为 6×5 的矩阵， $J^\#$ 为 J 的伪逆。由于五坐标 G 代码往往由许多小直线段组成，可以近似认为小直线段的运动为速度控制的差分 δr 。根据这个原理，只需利用关节空间速度的通解并引入优化目标函数的梯度投影即可求得关节速度的差分 $\delta \theta$ ，然后对该过程进行迭代、积分至满足精度要求即可得出下一点的优化结果。关节空间速度通解为^[10]

$$\dot{\theta} = \dot{\theta}_s + \dot{\theta}_h = J^\# \dot{r} + k(E - J^\# J) \nabla H(\theta) \quad (24)$$

式中， $\dot{\theta}_s$ 为最小范数解， $\dot{\theta}_h$ 为通解。

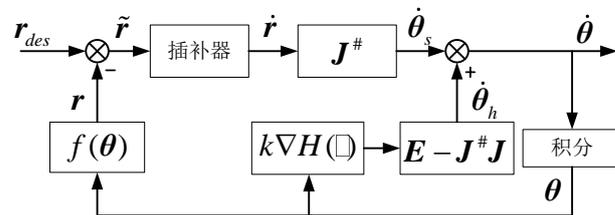


图 8 冗余自由度机器人的梯度投影法控制原理

由于基于广义逆方法的不可积性，本文采用不用计算伪逆且可积性好的扩展雅各比矩阵法对上述过程进行求解^[11]。使用可操作度作为极大化的性能指标

$$H(\theta) = \sqrt{\det(J^T J)} \quad (25)$$

定义函数 $G(\theta) = \nabla H^T N(J)$ ， $N(J)$ 为矩阵 J 的一维零空间。若 $G(\theta) = 0$ ， H 取得极值。将 $G(\theta)$ 与机器人运动学方程合并并对时间求导，可得扩展雅可比矩阵 J_e 。

$$J_e = \begin{pmatrix} J \\ \nabla G \end{pmatrix} \quad (26)$$

只要 J_e 不奇异, 就能求解关节速度差分 $\delta\theta$:

$$\delta\theta = J_e^{-1} \begin{pmatrix} \delta r \\ 0 \end{pmatrix} \quad (27)$$

令 $\theta_n = \theta_{n-1} + \delta\theta_{n-1}$, 再使用机器人的运动学方程求解出新的 r_n 值和 δr_{n-1} 。如上过程进行多次迭代, 直到 δr 足够小为止。这时的 r_n 和 θ_n 即为下一点位的位姿。

4.2.4 求解方法分析与小结

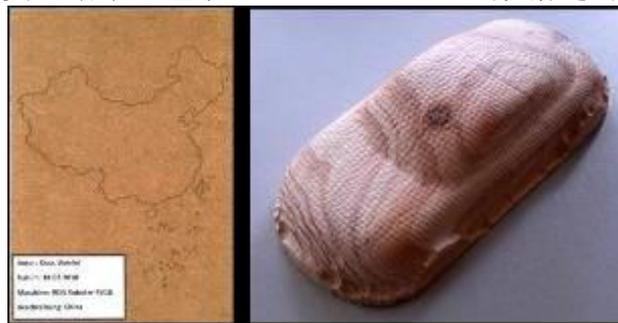
由求解过程可以看出, 几何分析求解法的算法比较简单, 其解的形式完全为封闭解, 没有数值运算的迭代过程。因此算法效率最高, 且稳定性好, 适合于实时运算。但该方法没有求解出最优解, 仅考虑奇异点情况的规避, 因此只能在基于特定的假设下使用(如算法中所述)。另外, 补偿权重为经验值, 不合理的取值可能会造成轴超限或碰撞等问题。

数值求解法更具一般性, 通常能够迭代出全局或局部的最优解, 且能够对多目标进行优化。其缺点在于计算复杂, 运算速度慢, 求解不稳定。因此不适合与实时运算功能。其中, 直接数值求解法适合于刀轨运动点位距离相差比较大或轮廓精度要求不高的点到点运动情况, 如下刀、退刀等动作, 而其初值问题可以使用几何分析求解法的结果。间接数值求解法适合于五坐标曲面加工中的小直线段运动。此时, 由于点位之间距离很小, 相当于速度的差分。因此只需要很少的迭代次数就可以完成下一点位的去冗余优化解。

5 仿真与试验

为了更为直观和有效地对切削加工机器人与 CAD/CAM 的数据接口功能和后置处理过程进行仿真和分析, 本文建立了一个针对切削加工机器人 REIS RV16 的 3D 仿真系统。该系统具有一个机器人的 3D 可视化虚拟环境, 能够接受 CAM 软件生成的 G 代码作为输入, 集成了机器人的后置处理器, 并能够将后置处理后的刀轨数据转换为机器人控制器能够执行的语言。其优化过程包含了几何分析求解算法。此外, 提供了调用 Matlab 引擎和 API 函数的接口功能, 因此可以利用 Matlab 强大的最优化工具箱和数值计算功能实现直接和间接数值求解算法, 并将结果返回给仿真环境。

仿真效果表明, 系统能够接受 G 代码输入, 从而与传统数控领域使用的 CAD/CAM 系统连接, 并实现切削加工机器人的后置处理过程。经过 CAD-CAM-Robot 数据链的加工指令最终传输到机器人的控制器中, 驱动机器人完成最后的加工。图 9 为基于 REIS RV16 平台的切削加工机器人加工的 2D 和 3D 的零件示例。实验结果验证了 CAD-CAM-Robot 数据链的可行性和可靠性。



(a) 2D 零件

(b) 3D 零件

图 9 机器人加工的示例零件

6 结论

本文介绍了将工业机器人应用于切削加工领域的实现过程。针对 6R 机器人分析了 CAD-CAM-Robot 数据链的构成和关键组成模块。着重研究了切削加工机器人的后置处理过程及在后置处理过程中遇到的坐标变换、运动学求解、优化过程和代码生成等问题。具有冗余自由度的机器人优化过程应该能够解决去冗余、回避奇异点、轴超限保护以及防止碰撞等问题。

本文重点针对 6R 切削加工机器人的奇异点回避的提出了三种算法:

(1) 几何分析求解法

能够得到机器人的封闭解, 具有较高的运算效率和稳定性。但是该算法只适合存在封闭解的机器人结构, 且很难求得最优解。

(2) 直接数值求解法

采用最优化算法的一维搜索法, 能够求得全局最优解。适合于下刀、退刀等大行程运动时的计算。

(3) 间接数值求解法

采用机器人速度控制和对轴速度积分控制的策略, 将五坐标曲面加工的小直线段运动等效为速度差分。适合于曲面小直线段走刀规划。

本文建立了针对切削加工机器人的 3D 加工仿真系统, 集成了 CAD/CAM 接口和后置处理过程。经过后置处理过程的加工代码可以直接驱动机器人的运动, 实现最终的零件加工。

参 考 文 献

- [1] ABELE E. Spanende Bearbeitung mit Industrierobotern: Forschungsprojekt ADVOCUT: Entwicklungen und Industriettransfer[R]. Deutschland: Institut für Produktionsmanagement, Technologie und Werkzeugmaschinen, Technische Universität Darmstadt. 2007.
ABELE E. Machining by industrial robots: research project ADVOCUT: development and industrial transfer [R]. Germany: Institute for production management, technology and machine tool. Technical University Darmstadt. 2007.
- [2] SCHEIFELE D. Robots as CNC machines[R], ISG-Industrielle Steuerungstechnik GmbH. 2009.
- [3] KUKA Milling 8kw[EB/OL]. [2006-01-01]. <http://www.kuka-robotics.com>.
- [4] IRBCAM GmbH[EB/OL]. [2006-01-01]. <http://www.irbcam.com/>
- [5] 李福生. 数控机床技术手册[M]. 北京: 北京出版社, 1996, 32-40.
LI Fusheng. Technical manual of NC machine tools[J]. Beijing Press, 1996, 32-40.
- [6] R.P. Paul, Robot manipulators[M]. Cambridge: MIT Press, 1981.
- [7] HAYES M, HUSTY M, ZSOMBOR-MURRAY P. Singular configurations of wrist-partitioned 6R serial robots: a geometric perspective for users[J]. Trans. Can. Soc. Mech. 1991, 26(1): 41-55.
- [8] 宋伟刚. 机器人学——运动学、动力学与控制[M]. 科学出版社, 2007.
SONG Weigang. Robotics——kinematics, dynamics and control[M]. Science Press. 2007, 107-117.
- [9] 刘惟信. 机械最优化设计[M]. 清华大学出版社, 1994.
LIU, Weixin. Machinery optimal design[M]. Tsinghua University Press, 1994.
- [10] NAKAMURA Y. Advanced robotics: Redundancy and optimization[M]. Addison-Wesley Publishing Company. 1991.
- [11] 陆震. 冗余自由度机器人原理及应用[M]. 机械工业出版社, 2007, 51-65.
LU, Zhen. Theory and application of robots with redundant degree of freedom[M]. China Machine Press, 2007, 51-65.

