

6R 工业机器人运动学算法的改进^{*}

吴 挺, 吴国魁, 吴海彬^{*}

(福州大学 机械工程及自动化学院, 福建 福州 350108)

摘要:针对工业机器人实时控制中存在的运动学模块计算量大的问题,以垂直六关节工业机器人为研究对象,在建立其逆运动学逆解求解模型的基础上,根据机器人几何结构及解的特点,提出了一种逆解优化搜索算法。提出了在笛卡尔坐标空间对位置和姿态进行综合插补的算法;为了方便客户使用,给出了用户任意坐标系的建立方法及姿态的旋转的表示方法;最后,在实验室 6R 机械手实验平台上验证了该算法的有效性。实验结果表明,该算法可以使机器人在运动过程中,特别是在曲线插补过程中,使机器人各关节平稳变化,避免关节突变,有更高的精度和运行平缓性。

关键词:机器人运动学;逆解优化选取;插补算法;用户坐标系

中图分类号:TP242;TH112

文献标志码:A

文章编号:1001-4551(2013)07-0882-07

Improvement of kinematics algorithm of 6R industrial robot

WU Ting, WU Guo-kui, WU Hai-bin

(School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou 350108, China)

Abstract: Aiming at the intensive computation of kinematics module exist in the real-time control of the industrial robot, inverse kinematic solutions of vertical-six-joint industrial robot were calculated and analyzed. Then, an algorithm of searching the proper inverse kinematic solution was presented based on the characteristics of the solutions and the geometric structure of the robot. After that, the curve interpolation algorithm in cartesian coordinates was discussed. In order to make the robot easier to be used, the arbitrary user coordinates was built and the gesture rotation representation was presented. The algorithms were applied to a 6R manipulator in the laboratory to verify the effectiveness of the proposed algorithms. The results indicate that the algorithm can make the robotic joints move smoothly and avoid mutation when the robot is running, especially in the process of curve interpolation. The algorithms can make the motion of robot smoother and more accurate.

Key words: robot kinematics; inverse optimization; curve interpolation algorithm; user coordinates

0 引 言

机器人运动学在机器人运动控制中扮演着重要角色。机器人特别是工业机器人主要用于完成一定的作业任务,在完成任务过程中不可避免地要进行关节坐标空间与笛卡尔等坐标空间的相互转换,不可避免地要进行正运动学、逆运动学的求解以及插补算法

等运动学方法的实现。机器人运动学的作用不言而喻。

机器人运动学涵盖面较广,本研究只研究其中较为重要的运动学逆解及其优化选取、插补算法及一些常用坐标系的建立方式。目前,运动学逆解的方法主要有数值迭代法^[1]、几何法^[2-3]、神经网络法^[4]及遗传算法^[5]等方法。机器人运动学逆解算法复杂,数值迭

收稿日期:2013-02-05

基金项目:国家自然科学基金资助项目(51175074)

作者简介:吴 挺(1987-),男,福建莆田人,主要从事机器人技术方面的研究. E-mail:wuting04@yahoo.cn

通信联系人:吴海彬,男,博士,教授. E-mail:wuhb@fzu.edu.cn

代法依赖于初始点,收敛速度较慢。几何法依赖于机器人的机构模型。遗传算法涉及大量个体的计算,难以满足控制系统实时性的要求。在对逆解的选取时有学者提出一种基于“最短行程”的方法^[6],但该方法忽略了机械手当前状态,且要求出机器人的所有解,实时性不高。在插补算法方面也有不少学者提出一些插补算法^[7],但一般只介绍位置插补的实现,对于姿态插补却没有给出明确的算法。

本研究采用分离变量法求取逆解,提出一种基于机器人几何结构及解的特点的逆解优化方法,该方法无需求出机器人的所有逆解便可选择出较好的逆解,可节约机时,提高实时性,在插补算法中同时综合位置与姿态插补,以满足轨迹规划中对位置和姿态的要求。

1 工业机器人运动学逆解的求法

1.1 逆解求解过程

本研究研究的6自由度工业机器人如图1所示,根据D-H法^[8]对机器人建立坐标系。为了计算方便,这里在机器人末端三轴交汇处建立了坐标系 z_6-x_6 ,并在执行器末端建立了与 z_6-x_6 固连的坐标系 z_7-x_7 。因此,坐标系7并非按照D-H法建立。机器人D-H法关节连杆参数表,如表1所示。

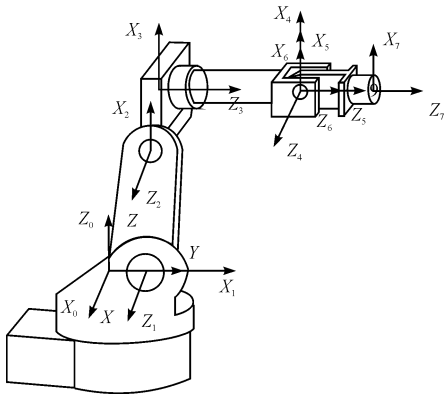


图1 机器人D-H表示法参考坐标

表1 机器人D-H参数表

关节	θ	d	a	α
1	θ_1	0	a_1	90°
2	θ_2	0	a_2	0
3	θ_3	0	a_3	90°
4	θ_4	d_4	0	-90°
5	θ_5	0	0	90°
6	θ_6	0	0	0

依照D-H表示法原理,可得出相邻坐标系间的变化矩阵,记为: A_1 、 A_2 、 A_3 、 A_4 、 A_5 、 A_6 。6个变换矩阵可按下式计算:

$$\begin{cases} A_1 = Rot(z, \theta_1) \times Trans(a_1, 0, 0) \times Rot(x, \pi/2) \\ A_2 = Rot(z, \theta_2) \times Trans(a_2, 0, 0) \\ A_3 = Rot(z, \theta_3) \times Trans(a_3, 0, 0) \times Rot(x, \pi/2) \\ A_4 = Rot(z, \theta_4) \times Trans(0, 0, d_4) \times Rot(x, -\pi/2) \\ A_5 = Rot(z, \theta_5) \times Rot(x, \pi/2) \\ A_6 = Rot(z, \theta_6) \end{cases} \quad (1)$$

由于执行器形状的不同,坐标系7的姿态和位置也会不同。但对于一定执行器,坐标系6和坐标系7的关系是固定的。坐标系7可由坐标系6经过平移变换 $\vec{d}(d_{7x}, d_{7y}, d_{7z})$,再依次绕自身坐标系的 x, y, z 轴旋转3个角度 ϕ_x, ϕ_y, ϕ_z 得到,这些参数可以由执行器的几何结构得到。故变换矩阵 A_7 可表示为:

$$A_7 = Trans(d_{7x}, d_{7y}, d_{7z}) Rot(x, \phi_x) Rot(y, \phi_y) Rot(z, \phi_z)$$

给定末端位姿矩阵 T :

$$T = \begin{bmatrix} nx & ox & ax & px \\ ny & oy & ay & py \\ nz & oz & az & pz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

解方程 $T = A_1A_2A_3A_4A_5A_6A_7$ 的过程即为求逆解过程。注意到 A_7 为常量矩阵,可在方程 $T = A_1A_2A_3A_4A_5A_6A_7$ 两边同时右乘 A_7 的逆矩阵,得到:

$$T_1 = TA_7^{-1} = \begin{bmatrix} nx & ox & ax & PX \\ ny & oy & ay & PY \\ nz & oz & az & PZ \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

式(3)中的参数均由 TA_7^{-1} 计算得来,这里为了书写方便,将其各用一个参数替代。对 A_7 处理后,相当于将图1中执行器末端坐标系原点等效到了机器人末端三轴交汇处。于是逆解的过程也转化为求 $T_1 = A_1A_2A_3A_4A_5A_6$ 的过程,不必再考虑坐标系 z_7-x_7 。

易知手腕3个转动关节轴线交于一点,满足Pieper给出的具有封闭解的条件^[9],故可用如下方法求逆解。由于运动方程中有许多角度的耦合,研究者不能从矩阵中提取足够的元素来求解单个的正弦和余弦来计算角度。为使角度解耦,可以用矩阵 T_1 左乘矩阵 A_n^{-1} ,进行变量分离,再对比左右矩阵求得相应角度。

可得如下方程:

$$A_1^{-1}T_1 = A_2A_3A_4A_5A_6 \quad (4)$$

比较方程左右两边,由两边矩阵中元素(3,4)相等得到:

$$\theta_1 = atan2(-PY, -PX) \text{ 或 } \theta_1 = atan2(PY, PX) \quad (5)$$

考虑式(4)左右两边矩阵中元素(1,4)和(2,4)相等,并计算化简可得:

$$\theta_3 = \text{atan2}(w, \pm \sqrt{1-w^2}) - \text{atan2}(a_3, d_4) \quad (6)$$

$$w = \frac{(PXC_1 - a_1 + PYS_1)^2 + PZ^2 - a_2^2 - a_3^3 - d_4^2}{2a_2 \sqrt{a_3^2 + d_3^2}}$$

考虑式 $A_2^{-1}A_1^{-1}T_1 = A_3A_4A_5A_6$ 左、右两边矩阵元素(1,4)和(2,4)相等,可得:

$$\theta_2 = \text{atan2} \left(\frac{a_3C_3 + d_4S_3 + a_2 - C_2(PXC_1 - a_1 + PYS_1)}{PZ}, \frac{(a_3S_3 - d_4C_3)PZ + (PXC_1 - a_1 + PYS_1)(a_3C_3 + d_4S_3 + a_2)}{PZ^2 + (PXC_1 - a_1 + PYS_1)^2} \right) \quad (7)$$

考虑式 $A_4^{-1}A_3^{-1}A_2^{-1}A_1^{-1}T_1 = A_5A_6$ 左右两边矩阵中元素(3,3)相等,可得:

$$\theta_4 = \text{atan2}(-ayC_1 + axS_1, C_{23}(axC_1 + ayS_1) + S_{23}az) \quad (8)$$

或者:

$$\theta_4 = \text{atan2}(-(-ayC_1 + axS_1), -(C_{23}(axC_1 + ayS_1) + S_{23}az)) \quad (9)$$

考虑式 $A_4^{-1}A_3^{-1}A_2^{-1}A_1^{-1}T_1 = A_5A_6$ 左右两边矩阵中元素(1,3)和(2,3)相等,可得:

$$\theta_5 = \text{atan2}(C_4(C_{23}(axC_1 + ayS_1) + azS_{23}) + S_4(-ayC_1 + axS_1), S_{23}(axC_1 + ayS_1) - azC_{23}) \quad (10)$$

考虑式 $A_4^{-1}A_3^{-1}A_2^{-1}A_1^{-1}T_1 = A_5A_6$ 左右两边矩阵中元素(3,1)和(3,2)相等可得:

$$\theta_6 = \text{atan2}(C_4(-nyC_1 + nxS_1) - S_4(C_{23}(nxC_1 + nyS_1) + nzS_{23}), C_4(-oyC_1 + oxS_1) - S_4(C_{23}(oxC_1 + oyS_1) + ozS_{23})) \quad (11)$$

1.2 逆解优化选取

从以上求取逆解的过程可以看出,对应于某一位置,可以有 8 组不同的封闭解。但实际的机器人运动控制只能选择其中一组解,并且关节角转角是有范围限制的,同时,在轨迹运动时,如何使各关节变化平缓,减少机器人工作过程中角度突变,也对解的选择提出了要求。

在进行轨迹跟踪时,为了减小关节运动的时间,使各关节变化平稳,应使目标关节角与当前关节角尽可能接近。可以采用下式来选取逆解:

$$K = \sum_{i=1}^6 |\theta_{i\text{goal}} - \theta_{i\text{now}}| \quad (12)$$

式中: $\theta_{i\text{goal}}$ —第 i 关节角的目标关节角, $\theta_{i\text{now}}$ —第 i 关节角的当前角度。

选取关节角度时,所要的关节角应满足使 K 值最小。但该法需要算出机器人的所有封闭解,从中选择

出使 K 值为最小的解,这样会浪费机时去求无用解,且算法较为复杂,效率不高。

可以从机器人结构及解的特性来简化算法。在运动中为使机器人有较好的轨迹跟踪精度,则必然要求每次转过的关节角较小。从上文求出的 θ_1 来看, θ_1 对应的两个解相差 180° ,从中可以得到一个与当前 θ_1 较接近的解,另一个解可以直接舍弃。选定了 θ_1 后, θ_3 只有两个解且 θ_3 的两个解相差的角度依赖于 θ_3 的大小, θ_2 完全依赖于 θ_3 和 θ_1 ,因此可以求出这两组解,采用下式来选取使 J 最小的解:

$$J = \sum_{i=2}^3 |\theta_{i\text{goal}} - \theta_{i\text{now}}| \quad (13)$$

θ_4 的两个解也相差 180° ,选择与当前角度相差较小的解,否则从图 1 中工业机器人的结构可以看出后面的两个关节角将产生突变,这是不允许的。 θ_5 和 θ_6 完全取决于前面的解。至此,已选出一组较好的解。该法可以避免去解不符合要求的解,提高效率,简化算法。具体流程如图 2 所示。

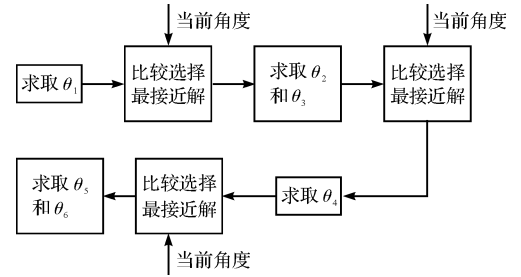


图 2 逆解选取流程

2 插补算法

轨迹规划在机器人运动控制中具有重要的作用,对机器人的作业有着重要的影响。本研究所述轨迹规划是在笛卡尔坐标系下进行的,必须运用插补算法并反复求解逆运动学方程来计算关节角。插补算法在轨迹规划中具有重要的地位,因此研究它具有重要的意义。在笛卡尔坐标空间中描述机器人末端执行器,包括位置描述和姿态描述,因此需要对位置进行插补,同时也要对姿态进行插补。

总体原则是:在进行插补时,将起点和终点间的变换分解为一个平移(位置插补)和一个绕空间中某根轴 k 的旋转(姿态插补)。平移实现末端执行器坐标系原点的移动,旋转实现末端执行器姿态的改变,两变换同时进行。

2.1 位置插补算法

空间曲线的位置插补可以近似地等效为由直线段

和圆弧段所组成,对应的位置插补就是直线插补和圆弧插补。

对直线插补,设直线的起点坐标为 $A(X_A, Y_A, Z_A)$, 终点坐标为 $B(X_B, Y_B, Z_B)$, 选定插补次数 N , 则有:

$$\begin{cases} \Delta x = (X_B - X_A)/N \\ \Delta y = (Y_B - Y_A)/N \\ \Delta z = (Z_B - Z_A)/N \end{cases} \quad (14)$$

于是直线的插补点 (x_i, y_i, z_i) 可以表示为:

$$\begin{cases} x_i = X_A + i\Delta x \\ y_i = Y_A + i\Delta y \\ z_i = Z_A + i\Delta z \end{cases} \quad (15)$$

其中: $0 \leq i \leq N$ 。

对圆弧插补,设该圆弧为由空间中任意不共线的3点 A, B, C 所组成的,示意图如图3所示。

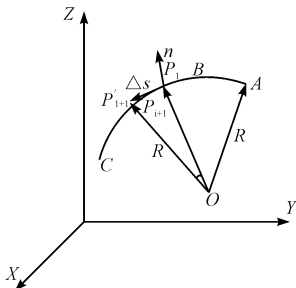


图3 圆弧插补示意图

易得线段 AB 的中垂面、线段 BC 的中垂面如下:

$$\begin{cases} (x - (x_A + x_B)/2)(x_B - x_A) + (y - (y_A + y_B)/2)(y_B - y_A) + (z - (z_A + z_B)/2)(z_B - z_A) = 0 \\ (x - (x_C + x_B)/2)(x_C - x_B) + (y - (y_C + y_B)/2)(y_C - y_B) + (z - (z_C + z_B)/2)(z_C - z_B) = 0 \end{cases} \quad (16)$$

以及点 ABC 所在平面的方程如下:

$$T(\theta_1) = \begin{bmatrix} u_1^2 + \cos\theta_1(1 - u_1^2) & u_1 u_2(1 - \cos\theta_1) - u_3 \sin\theta_1 & u_1 u_3(1 - \cos\theta_1) + u_2 \sin\theta_1 \\ u_1 u_2(1 - \cos\theta_1) + u_3 \sin\theta_1 & u_2^2 + \cos\theta_1(1 - u_2^2) & u_3 u_2(1 - \cos\theta_1) - u_1 \sin\theta_1 \\ u_1 u_3(1 - \cos\theta_1) - u_2 \sin\theta_1 & u_3 u_2(1 - \cos\theta_1) + u_1 \sin\theta_1 & u_3^2 + \cos\theta_1(1 - u_3^2) \end{bmatrix} \quad (21)$$

于是姿态的插补便可转化为转角 θ 的插补,插补次数为 N , 则有:

$$\Delta\theta = \theta/N \quad (22)$$

插补点的姿态可以表示为:

$$T_i = T_A T(i\Delta\theta) \quad (23)$$

式中: $0 \leq i \leq N$, T_A 一直线初始点的姿态。

将插补点的位置与姿态结合起来,便可得到插补点的位姿矩阵,再利用逆运动学方程便可解出各个关节角。

对有中间点约束的姿态插补,如图3中的圆弧插补,考虑中间点 B 的姿态,可按下述方法进行。由下式可得 AB 弧所对圆心角 θ_{AB} :

$$\begin{vmatrix} x - x_c & y - y_c & z - z_c \\ x_a - x_c & y_a - y_c & z_a - z_c \\ x_b - x_c & y_b - y_c & z_b - z_c \end{vmatrix} = 0 \quad (17)$$

联立以上3个方程,解方程便可得到圆弧的圆心坐标 (x_0, y_0, z_0) , 进而可得到圆弧半径 R 和圆心角 θ 。给定插补次数 N , 则可得步距角 $\delta = \theta/N$ 。为求插补点坐标可按图3所示过 P_i 点做切线 $P_i P'_{i+1}$, 进而得到圆弧上的点 P_{i+1} , 该递推公式已有学者推出^[6]。

2.2 姿态插补算法

根据曲线特点可将姿态插补分为有中间点约束的姿态插补(如圆弧插补)和无中间点约束的姿态插补(如直线插补)。

对无中间点约束的插补,设起点到终点的姿态变换矩阵为 T , 并将 T 扩充为一个齐次矩阵 R 。设 R 为:

$$R = \begin{bmatrix} n_x & o_x & a_x & 0 \\ n_y & o_y & a_y & 0 \\ n_z & o_z & a_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (18)$$

则可将该旋转变换(矩阵 R) 转化为绕空间中某个 k 轴旋转 θ 角得到,得到的 k 轴单位向量 (u_1, u_2, u_3) 和转角 θ 可表示如下^[10]:

$$\begin{cases} u_1 = (o_z - a_y)/2\sin\theta \\ u_2 = (a_x - n_z)/2\sin\theta \\ u_3 = (n_y - o_x)/2\sin\theta \end{cases} \quad (19)$$

$$\theta = \arctan\left(\frac{\sqrt{(o_z - a_y)^2 + (a_x - n_z)^2 + (n_y - o_x)^2}}{n_x + o_y + a_z - 1}\right) \quad (20)$$

此外,可以得到绕 k 轴转过某个角度 θ_1 得到的变换矩阵 $T(\theta_1)$:

$$\theta_{AB} = 2\arcsin\left(\left((x_B - x_A)^2 + (y_B - y_A)^2 + (z_B - z_A)^2\right)^{1/2}/(2R)\right) \quad (24)$$

对 $N_{AB} = N\theta_{AB}/\theta$ 取整,可得 AB 弧的插补次数 N_{AB} , BC 弧插补次数 $N_{BC} = N - N_{AB}$ 。由此便可分别对 AB 弧和 BC 弧进行姿态插补,方法同直线插补。

2.3 位置和姿态插补综合

轨迹插补需要有位置和姿态信息才能进行,需要将插补点的位置与姿态结合起来,得到插补点的位姿矩阵,再用逆运动学方程便可解出各个关节角。由上文可以得到插补点的位置 $P_i(x_i, y_i, z_i)^T$, 及姿态矩阵

T_i , 由此便可得到插补点的位姿矩阵 R_i :

$$R_i = \begin{bmatrix} T_i & P_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (25)$$

3 不同位姿表达方式之间的转换

工业机器人的运动学控制, 位姿状态的表述通常采用位姿矩阵, 但是位姿矩阵中姿态参数较多且不相独立, 在做速度或力矩控制时, 往往由于雅克比矩阵无法从位姿矩阵直接通过计算得到而带来麻烦。而如果采用相互独立的 6 参数表示机器人末端执行器的位姿状态, 并建立该 6 参数分别与 6 个关节角之间的函数关系式, 则可以较方便地得到雅克比矩阵。这 6 个参数分别是: dx, dy, dz 表示执行器坐标系原点坐标, 3 个角度 $\varphi_1, \varphi_2, \varphi_3$ 分别表示绕参考坐标系的 x, y, z 轴转过的角度。它可以较为形象地表达机器人的位姿。

3.1 由旋转参数转化为位姿矩阵

记 $Rot(d, \theta)$ 表示为绕 d 轴旋转 θ 角得到的位姿矩阵。则按顺序依次绕参考坐标系 x, y, z 轴旋转后所得的矩阵 R 可按下式计算:

$$R = Rot(z, \varphi_1) Rot(y, \varphi_2) Rot(x, \varphi_3) \quad (26)$$

3 个位置参数可以看成, 旋转后再相对于参考坐标系平移(d_x, d_y, d_z), 于是最终的位姿可按下式计算:

$$R = Trans(p_x, p_y, p_z) Rot(z, \varphi_1) Rot(y, \varphi_2) Rot(x, \varphi_3) \quad (27)$$

3.2 由位姿矩阵转化为旋转参数

设机器人位姿矩阵为 T :

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (28)$$

则有 $T = R$, 比较矩阵两边元素, 可得:

$$\begin{cases} d_x = p_x \\ d_y = p_y \\ d_z = p_z \\ \varphi_1 = \text{atan2}(T(3, 2), T(3, 3)) \\ \varphi_2 = \text{atan2}(-T(3, 1), (T(1, 1) \cos \varphi_3 + T(2, 1) \sin \varphi_3)) \\ \varphi_3 = \text{atan2}(T(2, 1), T(1, 1)) \end{cases} \quad (29)$$

式中: $T(a, b)$ —矩阵 T 中第 a 行第 b 列元素。

此外, 矩阵 T 可由机器人正运动学得到, 设 6 个关节角分别为 $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6$, 则矩阵 T 中元素均可表示为 $\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6$ 的函数。将其代入式

(25) 中, 可得:

$$\begin{cases} d_x = f_1(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \\ d_y = f_2(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \\ d_z = f_3(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \\ \varphi_1 = f_4(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \\ \varphi_2 = f_5(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \\ \varphi_3 = f_6(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \end{cases} \quad (30)$$

对式(26)求微分, 则可直接得到雅克比矩阵, 并可用于机器人的速度控制。

4 用户坐标系的建立

对用户来说, 工业机器人参考坐标系的建法未必很明确和实用, 参考坐标系往往不便于空间定位, 且随着环境改变, 描述操作对象的坐标系也应随之而变。为了能够方便地操作机器人, 更好地实现工作空间中的定位, 更好地规划机械手的运动, 有必要找到一个能够建立坐标系并实现其与机器人参考坐标系相互转化的方法。

为了建立这样一个坐标系, 用户需要给定 3 个适当的点, 设为 A, B, C, 其中点 A 为用户要建立的坐标系原点, B 为用户建立的坐标系 x 轴正向上一点, C 为用户建立坐标系的 xy 平面中的一点。用户坐标系的建立如图 4 所示。

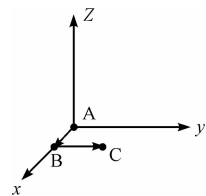


图 4 用户坐标系的建立

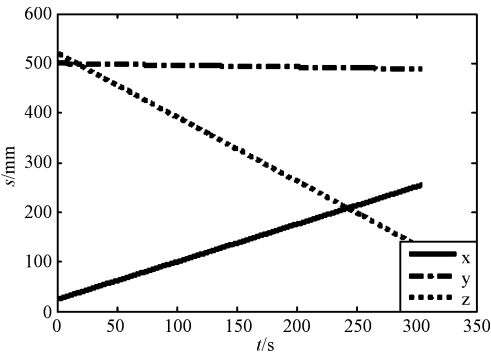
由此可得用户坐标系的原点为 A 点, x 轴为向量 $\overrightarrow{AB}$, 方向由 A 指向 B, $\vec{z}$ 可由 $\overrightarrow{AB} \times \overrightarrow{BC}$ 得到, 进而可以由 $\vec{z} \times \vec{x}$ 得到 $\vec{y}$ 。这样便可得到用户坐标系在参考坐标系中的位姿矩阵。

得到用户坐标系后, 就可以比较方便地确定空间中的点在该坐标系下的坐标。为了让机器人识别这些点, 需要研究用户坐标系与机器人参考坐标系的相互转化。设用户坐标系为 U , 机器人参考坐标系为 S , 用符号 ${}^S T_U$ 表示坐标系 U 相对于坐标系 S 的转化矩阵。已知用户坐标系下的点 ${}^U P$, 可由式 ${}^S P = {}^S T_U \times {}^U P$ 转化为机器人参考坐标系下的点的坐标。同理由 ${}^U P = {}^S T_U^{-1} \times {}^S P$ 可将机器人参考坐标系下的点转化为用户坐标系下。若已知用户坐标系下的位姿矩阵 ${}^U T_R$, 可由

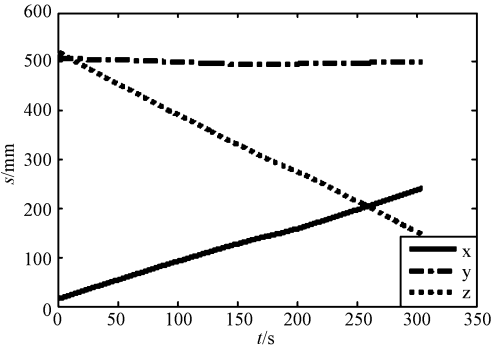
式 ${}^S T_R = {}^S T_U \times {}^U T_R$ 得到其在机器人参考坐标系下的位姿矩阵。由式 ${}^U T_R = {}^S T_U^{-1} \times {}^S T_R$ 可将机器人参考系下的位姿转化为用户坐标系下的位姿。

5 实验

实验借助一台工作半径为 650 mm 的垂直 6 关节工业机械手,将以上算法编成动态链接库,通过 VC++ 程序调用动态链接库来控制机械手的运动。在机械手做直线插补过程中,采集空间位置坐标和 6 个关节速度,并与原出厂程序控制的机械手的位置和速度进行对比。采集得到的位置坐标如图 5 所示。本研究算法得出的轨迹如图 5(a)所示,机械手的出厂程序得出的轨迹如图 5(b)所示。



(a) 本研究算法下的坐标轨迹



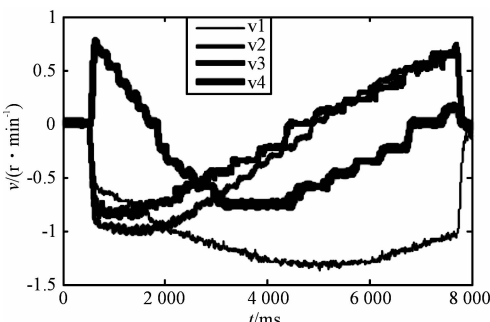
(b) 原算法下的坐标轨迹

图5 空间坐标轨迹对比图

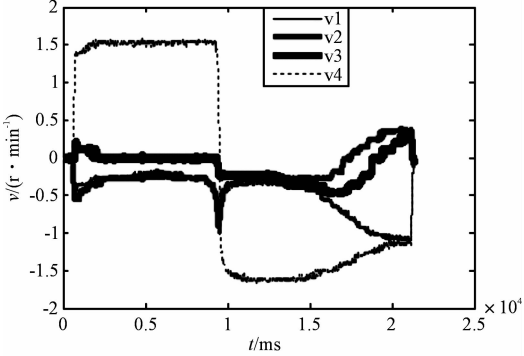
从图 5 可以看出,图 5(a) 的直线跟踪能力比图 5(b) 强。由此可得,改进后的逆运动学和插补算法是可行的,并且比原算法具有更强的直线轨迹跟踪能力。

机械手在做直线插补时前 4 个关节的速度曲线图如图 6 所示,采用本研究提出的方法得到的曲线如图 6(a) 所示,采用原出厂程序得到的曲线如图 6(b) 所示。

图 6(a)、6(b) 所对应的笛卡尔空间轨迹是完全一样的。从图 6(b) 可以看到,在一段较长时间内,第四轴的速度要比其他三轴速度大很多,因此,第四轴每个插补点所转过的角度比其他轴都大很多,有可能产



(a) 本算法下的各轴速度曲线



(b) 原算法下的各轴速度曲线

图6 各轴速度曲线对比图

生突变,导致插补的精度下降,同时插补时间也明显增加(约为图 6(a) 的 2 倍)。图 6(a) 中各轴速度较为均匀,不会出现某一轴速度长时间大于其他轴,由此可知各轴插补角度较为均匀,有效地减少了角度突变的可能。由于各插补角度较为均匀,可实现较小步距角的插补,有利于提高精度。

6 结束语

本研究针对 6 自由度工业机器人,采用分离变量法求其运动学逆解,并根据逆解的特点和机器人的几何结构从逆解中选取最优逆解,将曲线插补中的姿态插补转化为对绕空间中的某根轴旋转的角度的插补,并将以上逆解的改进算法运用到插补算法中。实验结果表明,机器人曲线插补更为精确,有效地避免了各关节角度突变。笔者给出了 6 参数位姿表示法与位姿矩阵之间的变换算法,由此可以得到雅克比矩阵,用于速度或力矩控制。用户坐标系的建立,使机器人运动空间的定位更加符合和方便用户需要,极大方便了用户的使用。

改进后的算法也存在一些缺点,当机械手的第四轴的旋转范围小于 $(-180^{\circ}, +180^{\circ})$ 且机械手插补过程中经过奇异点时,容易出现旋转角度超出范围,这些缺点有待于进一步研究解决。

(下转第 900 页)

5 结束语

本研究通过对超宽幅扫描仪在工业扫描过程中遇到的几何失真及织物疵点识别的介绍,引出对数字图像失真校正及图相配准的研究,利用数字图像的像素空间坐标变换和像素灰度内插法对失真图像进行校正处理,并通过基于灰度化的图相配准计算进行织物疵点检测。通过对织物进行检测实验,证实了该软件能够准确地得到织物疵点信息,且方便技术人员操作,对织造检测行业具有一定的意义。

扫描检测技术应用前景广阔,但是,目前国内超宽幅扫描仪市场尚未成熟,织物检测关键技术的研究状况仍处于初级阶段,尚需进一步提高自动检测技术水平。

参考文献 (References):

- [1] 李立轻,黄秀宝. 图像处理用于织物疵点检测的研究进展[J]. 东华大学学报:自然科学版,2002,28(4):118-112.

- [2] 孙文立,刘金珠,金宏健. 一种基于图像处理技术的坯布疵点检测及评估方法[J]. 纺织机械,2012(4):54-56.
- [3] 刘君杰,赵晓娣. 图像处理技术在纺织品评定中的应[J]. 毛纺科技,2005(11):50-54.
- [4] 崔洪州,孔 渊,周起勃,等. 基于畸变率的图像几何校正[J]. 应用光学,2006,27(3):183-185.
- [5] 杨大力,翁正新. 图像校正技术的研究及应用[J]. 控制工程,2008,15(5):168-171.
- [6] 张 金,成媛媛,李 洋,等. 一种基于虚拟键盘图像坐标变换的几何失真校正方法[J]. 计算机应用与软件,2012,29(6):84-88.
- [7] 曹亚君,邵玉兰. 基于双线性内插的图像处理算法及其优化[J]. 中州大学学报,2012,29(3):111-113.
- [8] 谢永胜,余正生. 图像快速配准算法的改进[J]. 机电工程,2010,27(2):36-38.
- [9] 唐雪莲,毕明德,孙志刚. 基于支持向量机的布匹图案匹配算法设计[J]. 机电工程,2011,28(12):1523-1526.
- [10] TSAI D M, LIN C T. Fast normalized cross correlation for defect detection[J]. **Pattern Recognition Letters**,2003,24(15):2625-2631.

[编辑:李 辉]

(上接第 887 页)

参考文献 (References):

- [1] MANOCHA D, CANNY J F. Efficient inverse kinematics for general 6R manipulator[J]. **IEEE Transactions on Robotics and Automation**,1994,10(5):648-687.
- [2] 李友虎,叶伯生,朱志红. 基于几何法的机器人运动学逆解[J]. 武汉船舶职业技术学院学报,2002(1):9-11.
- [3] NEPPALLI S, CSENCITS M A, JONEB B A. A geometrical approach to Inverse Kinematics for Continuum manipulators[C]// **Intelligent Robot and systems**, 2008: 3565-6570.
- [4] KARLIK B, AYDIN S. An improved approach to the solution of inverse kinematics problems for robot manipulators[J]. **Engineering Applications of Artificial Intelligence: The International Journal of Intelligent Real-Time Auto-**

mation,2000,13(2):159-164.

- [5] 刘永超,黄玉美,王效岳,等. 基于遗传算法的机器人运动学逆解[J]. 机器人,1998,20(6):421-426.
- [6] 王 伟,谢明红,周国义. 6-DOF 工业机器人逆解优化及其工作空间的研究[J]. 机械与电子,2011(1):57-60.
- [7] 卓扬娃,白晓灿,陈永明. 机器人的三种规则曲线插补算法[J]. 装备制造技术,2009(11):27-29.
- [8] NIKU S B. **Introduction to Robotics: Analysis, Systems, Applications**[M]. Beijing: Publishing House of Electronics Industry,2004.
- [9] PIEPER D L. The kinematics of manipulators under computer control[D]. Stanford: Stanford University college of Engineer,1968:151-157.
- [10] 殷际英,何广平. 关节型机器人[M]. 北京:化学工业出版社,2003.

[编辑:李 辉]