一种解析和数值相结合的机器人逆解算法

刘 达 王田苗

(北京航空航天大学 机械工程及自动化学院, 北京 100083)

摘 要: 针对不存在解析逆运动学解的机器人结构, 提出一种解析与数值相结合的求解算法. 将传统的算法简化为寻找一个合适的目标函数, 对一维关节变量进行迭代求解, 其余关节变量可以用解析式求得, 从而将逆运动学的多变量迭代求解问题简化为一维变量迭代求解. 利用该算法对一种不存在解析逆解的5自由度机器人进行解算, 耗时不足0.3ms, 计算速度优于传统的算法. 这种算法的实时性和准确性满足了机器人实时控制的要求, 不需要进行正运动学计算, 可以解决大多数关节正交结构的机器人逆运动学问题.

关 键 词: 机器人; 逆运动学; 数值方法

中图分类号: TP 24

文献标识码: A 文章编号: 1001-5965(2007)06-0727-04

Algorithm combined analytical and numerical solutions for robot inverse kinematics

Liu Da Wang Tianmiao

(School of Mechanical Engineering and Automation, Beijing University of Aeronautics and Astronautics, Beijing 100083, China)

Abstract: Aiming at some robot configurations which can't be resolved by analytical solution, an inverse kinematics algorithm, in which analytical and numerical solutions are combined, was proposed. This algorithm simplified the previous method to search a suitable object function, and then iteratively solved the one-dimensional joint variable, by which the other joint variables could be derived. The algorithm developed here performed a kinematics inversion of a 5 degree of freedom robot with no closed-form solutions in less than 3 ms using a desktop computer, an order of magnitude faster than the times found in the literature. This method is simple and fast enough to be suitable for real-time manipulator control. This approach does not require computation of the forward kinematics and can be used with manipulators of any geometry, although it is most efficient when applied to orthogonal manipulators.

Key words: robot; inverse kinematics; numerical solution

机器人逆运动学求解是进行机器人控制的重要问题, 因为在完成路径规划后, 机器人必须沿一条工作空间内的连续路径进行逆运动学求解, 而且要求运算的实时性和准确性. 目前求解机器人逆运动学问题的方法主要有2种: 解析方法和数值方法. 解析方法简单直观, 运算速度快, 但是只有满足以下几何条件的机器人结构其逆运动学才有解析解: 即后3个关节轴相交于一点或有3个关节轴相互平行[1]. Paul等人和Ardayfio分别给出了一些工业机器人机构的解析解[2-3]; Doty分析了24类6自由度正交机器人结构, 指出其中5类有解析解, 其余各类结构是否有解析解则要进行进一步研究[4]. 对于不存在解析解的机器人结构, 只能采用数值方法进行逆解. Huang等人以雅可比矩阵为基础建立了求解关节变量的迭代算法, 该算法要进行多次矩阵求逆, 并且容易发散[5]. 于是一些学者对迭代算法进行了改进: Goldenberg等人根据非线性优化的思想, 采用一

收稿日期: 2006-07-17

基金项目: 国家自然科学基金资助项目(60525314); 国际科技合作计划资助项目(2006DFA1229); 教育部留学归国基金资助项目

作者简介: 刘 达(1972-), 男, 北京人, 讲师, drliuda@buaa.edu.cn.

种改进的牛顿法迭代算法[6]; Angeles则基于闭包方程旋转部分的不变性提出了一种迭代算法[7]; Lenaroic采用的是共轭梯度法[8]. 这些方法都是非线性优化中常用的算法, 实际应用中仍存在运算量大, 收敛速度慢的缺点. 以上研究发现, 数值方法基本是建立在多元变量迭代的基础上, 算法的快速性和准确性很难满足实际应用的需要. 为此, 针对不存在解析逆运动学解的机器人结构, 本文提出一种解析与数值计算相结合的方法, 解决了机器人逆运动学算法的实时性和准确性问题.

1 机器人运动学模型

把机器人简化为转动副和移动副组成的开环运动链, 含 4, 5, 6个自由度的机器人类型数目分别为 179, 1 185, 7 437种, 其中可以用解析法对逆运动学求解的数量很少. 通过 D-H 方法可以建立起机器人各个关节的坐标系, 相邻两杆件的坐标变换关系用变换矩阵表示为

$$ {}^{i-1}\boldsymbol{A}_i = \begin{bmatrix} c_i & -s_i\cos\alpha_i & s_i\sin\alpha_i & a_i c_i \\ s_i & c_i\cos\alpha_i & -c_i\sin\alpha_i & a_i s_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}_i & \boldsymbol{t}_i \\ \boldsymbol{0} & 1 \end{bmatrix} \tag{1} $$

其中, $c_i = \cos\theta_i$, $s_i = \sin\theta_i$, $i = 1, 2, \cdots, n$; $\boldsymbol{R}_i$ 为姿态转换的子矩阵; θ_i, α_i, a_i, d_i 统称为机器人的运动学 D-H 参数, θ_i 和 α_i 为角度参数, a_i 和 d_i 为长度参数. 一个 n 自由度的机器人具有 $4n$ 个 D-H 参数, 见图 1. 机器人末端在基础坐标系中的位姿为

$$ {}^{0}\boldsymbol{T}_n = \prod_{i=1}^{n} {}^{i-1}\boldsymbol{A}_i = \begin{bmatrix} \boldsymbol{E}_n & \boldsymbol{P}_n \\ \boldsymbol{0} & 1 \end{bmatrix} \tag{2} $$

其中, $\boldsymbol{E}_n = [\boldsymbol{n} \quad \boldsymbol{o} \quad \boldsymbol{a}]$表示机器人末端姿态, $\boldsymbol{n}$, $\boldsymbol{o}$, $\boldsymbol{a}$分别为法向、滑动和接近向量; $\boldsymbol{P}_n = [P_x \quad P_y \quad P_z]^{\mathrm{T}}$ 表示机器人末端位置.

将机器人分解为位置和姿态 2部分, 一般前 3个关节主要决定机器人位置, 称为机器人手臂, 以后的关节主要决定机器人的姿态, 称为机器人手腕. 考虑六关节机器人, 由式 (2)可得

$$ \boldsymbol{A}_1\boldsymbol{A}_2\boldsymbol{A}_3[P_x' \quad P_y' \quad P_z' \quad 1]^{\mathrm{T}} = [P_x \quad P_y \quad P_z \quad 1]^{\mathrm{T}} \tag{3} $$

$$ \boldsymbol{R}_4 \cdot \boldsymbol{R}_5 \cdot \boldsymbol{R}_6 = (\boldsymbol{R}_1 \cdot \boldsymbol{R}_2 \cdot \boldsymbol{R}_3)^{-1}\boldsymbol{E}_6 \tag{4} $$

其中, $[P_x' \quad P_y' \quad P_z']^{\mathrm{T}}$ 为机器人腕点的坐标位置, 腕点设在第 4个关节的坐标原点.

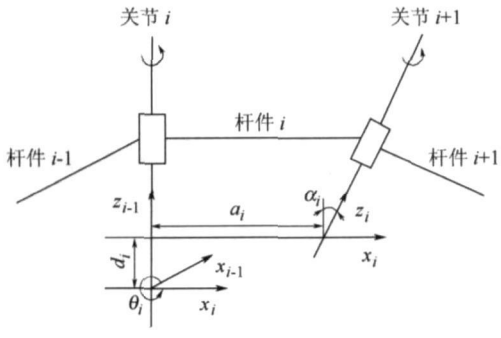

图 1 机器人杆件参数图

2 数值迭代算法

对于位置和姿态完全解耦的机器人, 可以由 ${}^{0}\boldsymbol{T}_6$ 直接求得 $[P_x' \quad P_y' \quad P_z']^{\mathrm{T}}$, 从而利用式 (3)求解前 3个关节变量, 再由式 (4)求解后 3个关节变量. 对于大多数 5自由度或 6自由度机器人, 位置和姿态并不完全解耦, 就只能根据式 (3)和式 (4)利用数值算法迭代求解机器人的关节变量, 算法的一般过程是:

1) 取初值 $\theta_4 = \theta_5 = \theta_6 = 0$;

2) 根据式 (5)以及由式 (3)推导出的机器人位置逆解求解出 θ_1, θ_2, θ_3;

$$ [P_x' \quad P_y' \quad P_z' \quad 1]^{\mathrm{T}} = \boldsymbol{A}_4\boldsymbol{A}_5\boldsymbol{A}_6[0 \quad 0 \quad 0 \quad 1]^{\mathrm{T}} \tag{5} $$

3) 根据式 (4)推导出的机器人姿态逆解求解出 θ_4, θ_5, θ_6;

4) 利用新解得的 θ_4, θ_5, θ_6, 重新计算 $[P_x' \quad P_y' \quad P_z']^{\mathrm{T}}$, 判断与式 (5)得到的初始位置的误差, 达到精度则已获得解, 否则返回步骤 2), 为 θ_4, θ_5, θ_6 重新赋值再进行整个过程.

3 数值与解析结合算法

第 2节的数值迭代算法对于所有构型的 5自由度或 6自由度机器人的运动学逆解具有通用性, 但是对于该算法中三维变量循环迭代, 收敛的速度和计算的精度无法满足机器人实时操作的要求, 在实际应用中必须对算法进行改进.

实际上, 为简化正逆运动学的计算, 一般使机器人机构相邻运动副的轴线相互平行或正交, 使得 2个轴线间的夹角 $\alpha_i = 0$或 $\pi/2$ 因此对于这种结构的机器人逆运动学可以表达为: 如果一个关节变量已知, 其它关节变量都可以用解析式的形式求得. 数值迭代算法可以简化为寻找一个合适的目标函数, 对一维关节变量进行迭代求解, 这一关节变量解得后其余关节变量可以用解析式求得. 因此把逆运动学的三维变量迭代求解问题简化为一维变量迭代求解, 大大提高了收敛速度和计算精度, 下面以图 2的机器人为例介绍该算法.

3.1　运动学正解

图 2所示的 5自由度机器人用 D-H 法可求得如表 1所示的杆系参数.

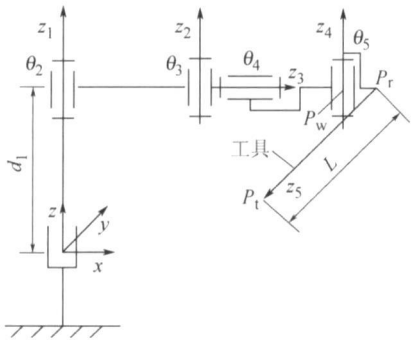

图 2　机器人杆件坐标系

表 1　机械臂的杆系参数

连杆 i	θ_i	α_i/(°)	a_i/mm	d_i/mm
1	0°	0	0	d_1
2	θ_2	0	300	0
3	θ_3	- 90	0	0
4	θ_4	90	0	300
5	θ_5	90	0	- 100

由式 (1)可以求得 $\boldsymbol{A}_1 \sim \boldsymbol{A}_5$, 再由式 (2)求得

$$ {}^0\boldsymbol{T}_5 = \begin{bmatrix} \boldsymbol{E}_5 & \boldsymbol{P}_5 \\ \boldsymbol{0} & 1 \end{bmatrix} \tag{6} $$

其中

$$ \boldsymbol{E}_5 = \begin{bmatrix} c_{23}c_4c_5 + s_{23}s_5 & c_{23}s_4 & c_{23}c_4s_5 - s_{23}c_5 \\ s_{23}c_4c_5 - c_{23}s_5 & s_{23}s_4 & s_{23}c_4s_5 + c_{23}c_5 \\ s_4c_5 & -c_4 & s_4s_5 \end{bmatrix} $$

$$ \boldsymbol{P}_5 = \begin{bmatrix} d_5c_{23}s_4 + d_4s_{23} + a_2c_2 \\ d_5s_{23}s_4 - d_4c_{23} + a_2s_2 \\ -d_5c_4 + d_1 \end{bmatrix} $$

其中, $c_{ij} = \cos(\theta_i + \theta_j)$, $s_{ij} = \sin(\theta_i + \theta_j)$, $i, j = 1, 2, 3$

3.2　运动学逆解

机器人末端工具用一个向量的始末坐标表示, 设为 $P_t(x_t, y_t, z_t)$和 $P_r(x_r, y_r, z_r)$, 长度为 L. 机器人进行手术路径规划时, 必须先根据 P_t和 P_r求得末端工具位姿的接近向量 $\boldsymbol{a}_6(a_{6x}, a_{6y}, a_{6z})$和位置坐标 $P_t(x_t, y_t, z_t)$, 然后结合正解结果求出机器人关节值. 用 $\boldsymbol{p}_r$表示工具末端坐标向量, 通过上述已知可得

$$ \left.\begin{aligned} \boldsymbol{P}_5 &= \boldsymbol{p}_r \\ a_x &= (x_t - x_r)/L \\ a_y &= (y_t - y_r)/L \\ a_z &= (z_t - z_r)/L \end{aligned}\right\} \tag{7} $$

根据正解结果, 由 ${}^0\boldsymbol{T}_5$可得

$$ P_{5x} = d_5c_{23}s_4 + d_4s_{23} + a_2c_2 \tag{8} $$

$$ P_{5y} = d_5s_{23}s_4 - d_4c_{23} + a_2s_2 \tag{9} $$

$$ P_{5z} = -d_5c_4 + d_1 \tag{10} $$

$$ a_x = c_{23}c_4s_5 - s_{23}c_5 \tag{11} $$

$$ a_y = s_{23}c_4s_5 + c_{23}c_5 \tag{12} $$

$$ a_z = s_4s_5 \tag{13} $$

从式 (7) ~式 (13)可以分析出, 如果 d_1, θ_4, θ_5这 3个关节变量中任何一个已知, 其它的关节变量都可以用解析式表达. 因此选择 θ_4为迭代变量, 首先给定 θ_4初值, 进行反解计算, 如计算出一组反解 $(d_1, \theta_2, \theta_3, \theta_4, \theta_5)$, 则代入正解验算. 如果满足收敛准则, 则该组解有效, 继续计算寻找下一组有效解, 直到找到所有的 4组解或 θ_4遍历完整个迭代区间; 如果当前 θ_4下没有有效解, 则 θ_4依照迭代步距更新, 进行反解, 直到 θ_4遍历完整个迭代区间.

此处分 $\theta_4 = 0$与 $\theta_4 \neq 0$两种情况, 具体步骤如下:

1) $\theta_4 \neq 0$

① 求 θ_5. 由式 (13)得

$$ \left.\begin{aligned} (\theta_5)_1 &= \arcsin\left(\frac{a_z}{\sin\theta_4}\right) \\ (\theta_5)_2 &= \pi - \arcsin\left(\frac{a_z}{\sin\theta_4}\right) \end{aligned}\right\} \tag{14} $$

② 求 $\theta_2 + \theta_3$. 由式 (8)和式 (9)得

$$ a_2\cos\theta_2 = P_{5x} - d_4\sin(\theta_2 + \theta_3) - d_5\sin\theta_4\cos(\theta_2 + \theta_3) $$

$$ a_2\sin\theta_2 = P_{5y} + d_4\cos(\theta_2 + \theta_3) - d_5\sin\theta_4\sin(\theta_2 + \theta_3) $$

即

$$ A\sin(\theta_2 + \theta_3) + B\cos(\theta_2 + \theta_3) + C = 0 \tag{15} $$

其中

$$ A = 2 \cdot P_{5x} \cdot d_4 + 2 \cdot P_{5y} \cdot d_5\sin\theta_4 $$

$$ B = 2 \cdot P_{5x} \cdot d_5\sin\theta_4 - 2 \cdot P_{5y} \cdot d_4 $$

$$ C = -(P_{5x}^2 + P_{5y}^2 + d_4^2 + d_5^2\sin^2\theta_4 - a_2^2) $$

令 $t = \tan\dfrac{\theta_2 + \theta_3}{2}$, 则

$$ \sin(\theta_2 + \theta_3) = \frac{2t}{1 + t^2} \qquad \cos(\theta_2 + \theta_3) = \frac{1 - t^2}{1 + t^2} $$

代入式 (15)得

$$ (C - B)t^2 + 2At + (C + B) = 0 $$

令 $\Delta = 4(A^2 + B^2 - C^2)$, 当 $\Delta > 0$时, 有

$$ t_{1,2} = \frac{-2A \pm \sqrt{\Delta}}{2(C - B)} $$

所以有 2个解:

$$\left.\begin{aligned}(\theta_2+\theta_3)_1 &= 2\arctan t_1\\ (\theta_2+\theta_3)_2 &= 2\arctan t_2\end{aligned}\right\} \quad (16)$$

③迭代收敛条件. 将式(14)、式(16)的结果代入式(11)和式(12)中, 令

$$\delta x = \cos(\theta_2+\theta_3)\cos\theta_4\sin\theta_5 - \sin(\theta_2+\theta_3)\cos\theta_5 - a_x$$

$$\delta y = \sin(\theta_2+\theta_3)\cos\theta_4\sin\theta_5 + \cos(\theta_2+\theta_3)\cos\theta_5 - a_y$$

令 ε为收敛指标, 如果 $\delta x \leqslant \varepsilon$, $\delta y \leqslant \varepsilon$, 则认为解出的 d_1, θ_5 和 $(\theta_2+\theta_3)$结果为有效解, 进入步骤④继续求解; 如果该条件不满足, 则改变 θ_4 的值, 转入步骤①继续迭代.

④求 d_1. 由式(10)得

$$d_1 = P_{5z} + d_5\cos\theta_4 \quad (17)$$

⑤求 θ_2. 将$(\theta_2+\theta_3)$代入式(8)和式(9)得

$$\cos\theta_2 = A/a_2 \qquad \sin\theta_2 = B/a_2$$

其中

$$A = P_{5x} - [d_4\sin(\theta_2+\theta_3) + d_5\sin\theta_4\cos(\theta_2+\theta_3)]$$

$$B = P_{5y} + [d_4\cos(\theta_2+\theta_3) - d_5\sin\theta_4\sin(\theta_2+\theta_3)]$$

则

$$\theta_2 = \arctan\left(\frac{B}{a_2}\Big\backslash\frac{A}{a_2}\right) \quad (18)$$

⑥求 θ_3. 将 θ_2 代入式(16)中可得 θ_3 的值.

2) $\theta_4 = 0$

由式(11)、式(12)可得

$$\theta_5 = \frac{1}{2}(\arcsin a_x + \arcsin a_y) \quad (19)$$

由式(8)、式(9)可得

$$\theta_3 = \arcsin\frac{p_{5x}^2 + p_{5y}^2 - d_4^2 - a_2^2}{2d_4a_2} \quad (20)$$

$$\theta_2 = \arcsin a_y - \theta_3 - \theta_5 \quad (21)$$

综上所述, 对于这种结构的机器人, 任意输入 $P_t(x_t, y_t, z_t)$和 $P_r(x_r, y_r, z_r)$, 机器人的运动学逆解有 4组可能解.

3.3 逆解的仿真验证

为验证上述算法是否满足机器人在路径拟合过程中对逆解实时性和准确性的要求, 本节用MATLAB编写了仿真程序. 命令图 2所示的机器人末端按固定姿态完成一个半径为 100mm 的圆周轨迹, 并在这个轨迹上设置 100个工作点, 对每个工作点分别进行逆解, 表 2给出了其中 10个位置逆解后得到的关节值. 图 3给出了工作点的理想位置和实际机器人到达位置之间的偏差, 最大位置偏差不超过 0.1mm, 运算结果表明逆解的准确性满足应用要求. 全部运行时间为 21ms, 平均每个位置计算 0.21ms, 可以实时进行轨迹插补.

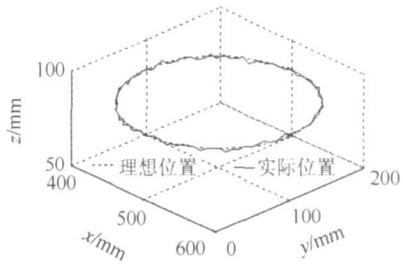

图 3 工作点理想位置和实际位置的偏差

表 2 机器人位置逆解后得到的关节值 (°)

d_1/mm	θ_2	θ_3	θ_4	θ_5
137.36	0.438	0.736	2.566	4.277
136.59	0.617	0.609	2.552	4.233
137.32	0.832	0.345	2.565	4.274
138.82	0.996	0.049	2.593	4.386
140.11	1.026	-0.182	2.618	4.559
140.41	0.886	-0.253	2.625	-1.571
140.26	0.667	-0.130	2.622	-1.456
140.41	0.487	0.122	2.624	-1.520
140.27	0.381	0.415	2.622	4.601
139.11	0.358	0.655	2.599	4.413

4 结束语

本文提出了一种解析与数值计算相结合的逆运动学求解新算法, 将逆运动学的多变量迭代求解问题简化为一维变量迭代求解. 仿真验证显示该算法在保证运算精度的前提下减少了运算时间, 完全满足机器人控制中路径拟合的实时性和准确性要求.

参考文献 (References)

[1] Paul R P. Robot manipulators: mathematics, programming, and control[M]. Cambridge: MIT Press, 1981

[2] Paul R P, Shimano B E. Kinematical control equations for simple manipulators[J]. IEEE Trans SMC, 1981, 11(6): 449-455

[3] Ardayfio D D. Micrasm microcomputer interactive codes for robot analysis and simulation[J]. MMT, 1985, 20(4): 271-284

[4] Doty K L. Machine intelligence lab report MIL[R]. 10 86 1, 1986

[5] Milenkovik V, Huang B. Kinematics of major robot linkage [C]//Proc 13th ISIR. Dearborn, Mich: SME, 1983, 2 16 31-16.47

[6] Goldenberg A A, Benhabib B, Fenton R G. A complete generalized solution to the inverse kinematics of robots[J]. IEEE Journal of Robotics and Automation, 1985, 1(1): 14-20

[7] Angeles J. On the numerical solution of the inverse kinematic problem[J]. Robotics Research, 1985, 4(2): 21-37

[8] Lenaroio J. An efficient numerical approach for calculating the inverse kinematics for robot manipulators[J]. Robotics, 1985, 3: 21-36

(责任编辑: 文丽芳)