

机器人三维空间无碰撞路径规划

徐毓良 杨起帆

(浙江大学计算机系, 杭州)

摘要 本文介绍了机器人操作手在三维空间中的一个路径规划系统. 它通过对三维环境障碍物进行两次投影, 从而把复杂的三维规划问题简化为在二维投影面上进行规划. 并根据机器人的几种典型姿态, 设计了一种路径搜索算法, 使得系统以局域搜索的代价, 得到全局较优的规划路径. 系统在 MicroVAX II 上运行, 达到了实时速度.

关键词: 机器人, 规划, 投影, 无碰撞路径, 搜索.

1 引言

机器人无碰撞路径规划问题至今未能很好解决, 特别是对三维空间的无碰撞路径规划困难更大. 原因之一是对机器人三维工作空间至今还没有一个好的描述方法, 以致对其规划十分困难, 复杂度高.

针对这些情况, 本文在采用一般方法描述机器人工作空间的基础上, 通过对三维障碍物进行两次投影, 把三维规划问题简化为二维规划, 从而大大减少了规划的复杂度.

2 环境及机器人数学模型

本文把环境障碍物定义为任意多面体, 尽管许多研究者避免对凹多面体进行讨论, 但针对实际情况中常常存在凹物体. 本着对障碍物几何形状尽力少作限制的出发点, 系统考虑了对凹多面体的处理. 对于实际中一些如圆柱、圆球等物体, 则采取多面体逼近, 根据实际需要确定逼近的精度.

本系统研究的机器人是类似 PUMA560 多关节机器人(图 1), 它共有 6 个关节, 其中

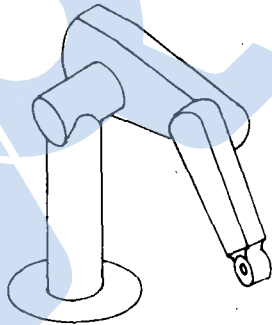


图 1 本系统研究的机器人

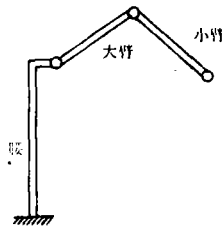


图 2 PUMA 机器人的表示

3 个是手爪的关节, 另 3 个是腰、臂关节. 而手爪的 3 个关节与抓取规划密切相关, 腰臂关节与路径规划密切相关. 要完成机器人完整的操作, 需要对路径规划和抓取规划一起考虑, 但抓取规划本身是个十分复杂的研究课题. 因此, 本文着重研究路径规划, 考虑机器

人的腰臂3个关节,而忽略手爪3个关节.把手爪部连到小臂上,作为一个整体与小臂一起讨论.

当机器人运动时,要计算其各臂扫过空间的精确值是十分困难的,多数规划系统都是对机器人作近似描述.本文把PUMA机器人的大臂和小臂看作是2个圆柱体.由于腰转动时其扫过的空间不变,不会与障碍物发生碰撞,只需以大小臂两圆柱中较大的半径作为参数,对环境障碍物进行外形扩展,从而把机器人简化成没有外形的线段⁽¹⁾(图2).这样在规划过程中,可用图2表示PUMA机器人.

3 障碍物投影计算

针对机器人无碰撞绕过三维障碍物的特点,我们对障碍物进行两次投影,一次是底面投影,另一次是侧面投影.对规划来说,它们能比较准确地刻划出原三维障碍物.

第一次投影是从障碍物顶部垂直往下作直线投影(即正投影),这样得到障碍物的底部投影.对多面体来说,其投影图是多边形.将多边形做外形扩展后,可以得到机器人与该障碍物可能发生碰撞的角度范围,一旦腰将转到该角度范围时,可预先规划出绕过该障碍物的无碰路径.对整个工作空间,可依次计算所有障碍物的底部投影,确定各个可能发生碰撞的角度范围.

至于如何绕过障碍,还需对障碍物作侧面投影,所得的投影面就是机器人对该障碍的碰撞区域.我们定义在绕过障碍之前机器人先规划出绕过该障碍的无碰撞最优姿态,然后只转动腰绕过障碍物.因此在机器人绕过某个障碍过程中,机器人手臂上任一点都是作绕腰轴的圆周运动.显然对障碍物作侧面投影时不能是直线投影,而可根据机器人臂上每一点都是沿圆弧运动的特点,对障碍物从侧面沿圆弧线进行投影,我们称之为圆弧投影.其数学定义如下:把多面体上每一点的坐标值 (x, y, z) 映射到投影面空间的点 (r, z) ,其中 $r = \sqrt{x^2 + y^2}$.该映射过程称多面体的圆弧投影.

三维空间中的直线经直线投影,其结果是二维空间直线.因此多面体通过直线投影得到的是多边形.但三维空间中的直线经圆弧投影,其结果就不一定是直线了.以图3(a)为例,直线段 ab 的二端点均在半径为 R 的圆柱面上,而整个线段都在圆柱面内,所以其上必存在一个切点 j ,使得以 r 为半径圆柱正好与线段 ab 相切在 j 点.这样线段 ab 经圆弧投影后,得到的投影图是一条曲线,如图3(b)所示.如果把由曲线边围成的区域作为圆弧投影的结果进行规划处理,显然运算量极大,不利于快速规划出机器人无碰路径.所以我们用图3(b)中的直线段 $a''b''$ 近似 $a'b'$,这样虽然要牺牲一些自由空间,但换取了规划过程的大大简化.

具体地如何把多面体进行投影,至今还没有一个成熟的算法.本文基于计算机图形学中凹多边形裁剪Weiler-Atherton算法⁽⁶⁾,设计了一种计算多面体投影的算法.算法思想为:对任一多面体的各个多边形面分开投影,当其二个多边形面通过对各顶点投影(直线投影或圆弧投影)处理后,可分别得到二个投影多边形,再求该两个多边形的和,即求出两个多边形的最大外轮廓.这样依次对多面体各面分别投影,再与上一次得到的和多边形进一步求和,最后得到多面体的投影区域——投影多边形.

求两多边形和的过程为:设图4中多边形 $s_1s_2s_3$ 称主多边形, $c_1c_2c_3$ 称副多边形,搜索它们的最大外轮廓从主多边形顶点 s_1 开始,沿顺时针方向,找到交点 I_1 后,沿副多边形从 I_1 按顺时针方向前进,经过 c_1 ,再找到交点 I_2 ,回到主多边形上继续前进,经过 s_2 ,

到达交点 I_3 ，再沿副多边形边前进，如此每碰到一个交点就换一个多边形，不断前进。其经过的顶点都是和多边形的顶点，最后可找出两个多边形的最大外轮廓顶点： $s_1 I_1 c_1 I_2 s_2 I_3 c_2 c_3 I_4 s_3$ ，由这些顶点围成的多边形就是和多边形。

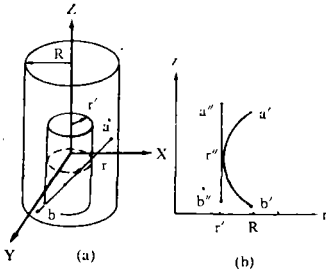


图 3 投影例

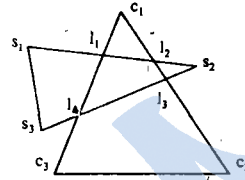


图 4 主多边形、副多边形及多边形

4 多边形外形扩展

多面体经投影后，得到二维的投影多边形。为了把简化成圆柱的机器人手臂用其轴线表示，需要对障碍物的投影多边形进行外形扩展。扩展参数 d 取大臂、小臂圆柱半径中较大的一个。

外形扩展过程为：对多边形各边作向外平移 d 的扩展边，再对多边形所有顶角分成凸角和凹角分别处理。对于凹角，取相应二边的扩展边的交点作为扩展多边形的顶点；对于凸角，先仅作该角的角平分线并向角外延伸，在距角点 d 处作角平分线的垂线，它与相应两扩展边的交点均是扩展多边形的顶点。对多边形的所有顶角均按上述处理，最后可得到扩展多边形的所有顶点。

5 关节空间转换

我们通过对障碍物作侧面投影，得到了绕过该障碍时的无碰撞区域。但对于用线段表示的机器人，要表示相对障碍物的自由空间，十分不方便。为此，我们把投影面空间上的

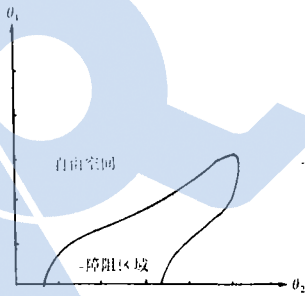


图 5 长方形障碍及绕过的三种典型姿态

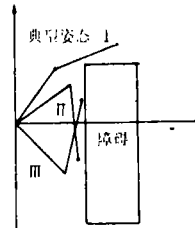


图 6 将图 5 的障碍转换到关节空间所得的区域

障碍多边形转换到机器人关节空间上，使得在关节空间上，机器人能作为一个点来考虑。这样关节空间上障碍区域之外的均是自由空间。由于我们考虑机器人绕过障碍时，只需调整其大、小臂姿态，使它处在与障碍物投影区域无相交的姿态，转动腰关节就一定能安全地绕过障碍。所以考虑的关节空间只需针对大、小臂关节的二维关节空间。多边形的障碍投影图在关节空间上是不规则图，其边界是曲线。例如把图 5 的长方形障碍转换到关节空间上，所得区域如图 6 所示。该区域与机器人的大、小臂几何长度及关节变化范围均有关。

6 路径搜索

机器人的任一姿态, 在关节空间上都有一个点相对应, 也就是说在机器人关节空间上, 机器人可当作一个点来看待。下面如不作说明, 都在机器人关节空间上进行讨论。如果机器人的当前位置在关节空间的障碍区域之外, 则说明机器人的当前姿态不会与障碍物发生碰撞。

机器人在绕过障碍之前, 如果其姿态已处于无碰撞姿态, 则让机器人直接绕过该障碍, 而无需调整其大、小臂关节; 如果机器人的当前姿态在关节空间的障碍区域内, 则需要调整位置, 使它处于障碍区域外的自由空间上。但自由空间很大, 机器人取哪一位置最合适呢? 这就涉及到路径搜索问题。

在机器人路径规划中, 常用的搜索方法有爬山法和 A^* 算法。爬山法是一种局部搜索方法, 它具有快速、简洁的优点, 但一旦掉入“陷井”就无法自拔。而 A^* 算法可从全局作搜索, 它具有启发式信息, 只要从起始点到目标点之间存在路径, A^* 算法总能找到其中最佳的一条路径。 A^* 算法是目前比较成熟的搜索算法, 被广泛使用。但它比爬山法运算量要大, 一旦被搜索结点很多且被算法扩展的结点数目较大时, A^* 算法搜索量会很大。针对二维关节空间障碍物的具体情况, 我们集爬山法和 A^* 算法的优点, 设计一种特殊算法。

在机器人路径规划环境中, 若用 A^* 算法按全局角度从所有障碍物中搜索出一条最佳的路径, 难度比较大, 因为同时考虑多个二维搜索空间, 其搜索面极大, 会使规划系统难以有效实现。因此要保证规划的高效率, 只能选择局部搜索, 针对爬山法的缺点, 我们引入 A^* 算法的启发式估值函数, 定义估值函数 $f(n)$ 为:

$$f(n) = g(n) + h(n) * d(n)$$

其中 n 是规划路径上当前位置的下一步搜索的位置, $g(n)$ 是路径起始点到点 n 的代价值, 该值是确定的, $h(n)$ 是启发式因子, $d(n)$ 是点 n 到目标点的直线距离, $h(n) * d(n)$ 是点 n 到目标点的确切路径值的代价估计。取 $h(n) < 1$, 则 $h(n) * d(n)$ 是 $d(n)$ 对当前位置所起作用的程度, 越靠近目标, $h(n)$ 越大; 越远离目标点, 因受后面将要遇到的障碍物的影响, $d(n)$ 的准确程度低, $h(n)$ 取值越小, $h(n)$ 的具体取值是根据经验确定的。

虽然在启发式估值函数中引入了 $h(n) * d(n)$, 但搜索仍是局域搜索, 它主要根据当前障碍物得出。由于各局部最优之和不一定是整体最优, 因此为避免单一的局域最优的搜索路径可能不是真正较优的路径, 根据 PUMA 机器人绕障碍时可能出现的三种典型姿态, 如图 5 所示, 当机器人绕过某障碍物时, 依次搜索出三个较优的路径点。这样路径起始点只有一个, 绕过第一个障碍物可搜索出三个路径点, 再绕过第二个障碍物时三个源点共可搜索出 9 个较优路径点。但这 9 个点分属三种典型姿态, 因此只需从中选出三个各代表一种典型姿态的路径点。依次对各障碍物做同样的搜索, 最后可得到三条较优的路径, 它们都是根据局域最优原则得到的, 再从中选出一条最佳的路径作为全局最优的规划路径。

上述搜索过程, 可看作一个三叉树搜索。每绕过一个障碍物, 对 9 个树结点进行剪枝, 最后剩三个结点再生成子树。因此对该三叉树的搜索非常快, 但仍是非完全搜索。事实上, 我们的搜索从根本讲是局域搜索, 得到的结果可能不是全局最优的, 仅仅是全局较优。引入三条搜索路径的目的主要是避免单一路径其优化程度可能不理想的情况, 因此没有必要对三叉树作完全搜索。

许多路径规划算法都试图根据欧几里德空间距离, 搜索出一条最优路径, 但对于近似

处理的世界模型和机器人实际存在的定位误差, 通过大量计算及详尽搜索得到的规划路径, 是否最优值仍难以确定. 退一步说, 即使找到了一条最优路径, 使机器人在运动中少走了一些路, 但这节省下来的时间远抵不上在规划过程中所耗费的时间. 到目前为止, 还没有一个很有效的算法能在三维空间中规划出一条最优的路径. 我们通过适当简化, 在很短的时间里规划出一条近似最优的路径. 这比花费大量时间去找一条最佳路径更易被人接受.

7 实验结果

对上述设计思想, 我们已在 Micro VAXII 上用 C 语言实现, 系统运行所占内存很少, 规划速度快, 基本能达到实时速度. 对图 7 例子, 规划一条从起始点到目标点的路径, 总共只花费约 4.6 秒时间.

例子中, 机器人从起始点 start 到目标点 end 之间规划路径, 它首先碰到障碍 1, 由于它的姿态调整到从障碍中间凹槽中通过路径最短, 所以手臂从 start 调整到 s_1 , 然后通过凹槽, 绕过障碍物 1. 又因 s_1 姿态与障碍物 2 会发生碰撞, 再调整姿态 2 从 s_1 到 s_2 , 绕过障碍物 2, 最后到达目标点 end. 从而规划出一条较佳的路径, 具体运算结果见图 8.

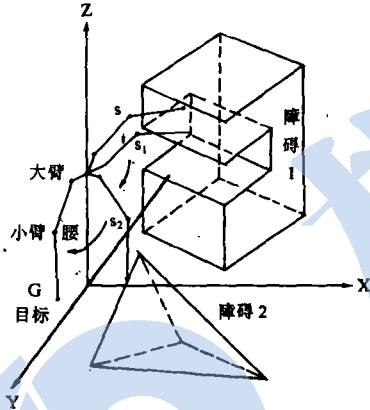


图 7 通过凹槽的路径

The least cost of path from start point to end point is searched in joint space :

start:(40,51)--->S1:(42,51)--->S2:(44,59)--->end:(44,53) .

The configuration of PUMA robot are :

(large arm , small arm , waist)

start state : (-52.767590 , 173.299301 , 108.435043)
 S1(before pass obstacle 1) : (-47.812500 , 170.156250 , 79.267960)
 S1(after pass obstacle 1) : (-47.812500 , 170.156250 , 10.732117)
 S2(before pass obstacle 2) : (-39.375000 , 203.906250 , -14.986911)
 S2(after pass obstacle 2) : (-39.375000 , 203.906250 , -80.535355)
 end state : (-37.304096 , 178.888794 , -90.000000)

图 8 计算结果

针对国内其它一些类似的规划系统, 它们大多只适用于凸多面体的障碍物环境, 无法规划出通过图 7 例子凹槽的路径, 因此本文系统所能适应的环境更大. 另一方面, 本文采取把三维障碍物经侧面投影, 简化成二维图形后, 再转换到关节空间, 这二部分的运算量比直接把三维障碍物转换到关节空间上要少得多, 并使得搜索空间简单, 最优路径点搜索的运算量也大为减小.

8 结束语

本文设计的路径规划方法, 对类似 PUMA 560 等多转动关节的机器人十分有效. 通过两次投影, 使三维空间的规划问题简化成二维空间规划, 大大简化了问题的复杂度. 本文提出圆弧投影方法, 进行三维障碍物投影, 对规划无碰路径来说, 能较准确地刻画三维障碍物的侧面信息. 虽然在障碍物的投影面上进行规划, 但对机器人自由空间的影响很小, 从而使机器人手臂能找到一条从障碍物侧面绕过的优化路径. 并使用一种特殊的路径搜索方法, 使规划系统以局域搜索的代价, 得到相对全局较优的优化路径. 使得规划系统能以实时速度进行三维空间的无碰撞路径规划.

当然,本文只是从路径规划角度进行讨论,所考虑的机器人关节只包括腰、臂关节,而对机器人手爪未作考虑.虽然当今对路径规划的研究都是如此,但要机器人完成一个实际操作路径规划与抓取规划必须同时考虑,也就是说在设计路径规划中要考虑抓取规划的要求以及手爪关节的姿态,这有待于今后进一步研究.本规划系统今后的发展方向是作为机器人任务级语言的一个实用子系统.

参 考 文 献

- 1 Lozano-perez T, Wesley M A. An algorithm for planning collision-free paths among polyhedral obstacles. *Commun Ass Comput Mech*, 1979; 22: 560-571
- 2 Lozano-perez T. Spatial planning: an configuration space approach. *IEEE Trans on Computers*, 1983; 32(2): 108-120
- 3 Brooks R A. Planning collision-free motion for pick and place operations. *Internat J Robotics Research*, 1983; 2(4)
- 4 Wong E K, Fu K S. A hierarchical-orthogonals pace approach to collision-free path planning. *IEEE Int Conf on Robotics and Automation*, 1985; 506-511
- 5 姜 松, 夏 凯, 吕 林. 机器人路径规划. *机器人*, 1989; 3(1):6-11
- 6 Rogers D F. 计算机图形学的算法基础. 北京, 科学出版社, 1987: 204-210

ROBOT COLLISION-FREE PATH PLANNING IN 3-D SPACE

XU Yuliang YANG Qifan

(Zhejiang University, Hangzhou)

Abstract

A path planning system of manipulators in 3-D space is described. After projecting each 3-D obstacle twice, the problem of 3-D planning is converted to a 2-D one in projection planes. According to several typical poses of the manipulator, a special search algorithm is designed, which can find a better overall path using a local search approach. This planning system is realized in Micro VAX II in real time.

Keywords: robot, plan, projection, collision-free path, search.

“山彦”获世界首届机器人奥林匹克大会的金奖

1990年9月27、28日在英国斯过托·古拉斯过的斯古拉伊大学举行的世界首届机器人奥林匹克大会的比赛中,日本筑波大学的油田信一教授领导的智能机器人研究室研制的名为“山彦”的智能移动机器人获得了大会的最优秀奖金牌。

以美国,苏联为首的11个国家、大约50台机器人参加了设有障碍物回避、投掷、跳跃等多种项目的首届机器人奥林匹克大会.机器人“山彦”参加的是、一边回避障碍物,一边在复杂的迷路中高速行走的比赛.机器人“山彦”在决赛中,以高超的功能、准确、迅速的动作,绕过了所有的障碍物,得到了高度的评价,以压倒世界强豪的优势,被称为“世界第一”,获得了世界最优秀机器人的桂冠。

日本筑波大学智能机器人研究室留学生:毛 峡