

摘 要

避障路径规划问题是在具有障碍物的环境中，按照某个评价标准（如最短路径长度、最短行进时间、最小能量消耗等），规划一条从起始点位置到达目标点位置最优（或次优）的无碰（避障）路径。

机械臂的运动学是避障路径规划研究的基础，本文首先建立了多关节检修机械臂的运动学模型。运动学逆问题的求解较为复杂，因为它实质上是一个非线性方程的问题，目前尚没有通用的解法。根据该机械臂结构的冗余度低，工作环境受限的实际情况，给出了一种简单易行的几何图解法，并在VC++仿真平台下对其运动学进行了仿真验证。

在机械臂进行检修工作时，如何判断和避免机械臂碰撞是实现自动作业的前提。针对该检修机械臂的具体结构及特殊的工作环境，提出了一种基于包围盒和等效空间圆柱体相结合的网络包围法，将空间局部域进行有效分解，通过简单有效的几何等效，把机械臂的碰撞检测问题转化为空间解析几何问题，大大简化了问题的求解。

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。由于随机路标法具有易实现、通用性好的特点，特别是在解决多自由度机器人路径规划上所表现的有效性，近几年引起了研究人员的重视。本文采用一般随机路标法得到一个表示机械臂位形空间结构信息的路标，提前检验各个路标之间的连通关系得到一个路标连通图，当实际路径规划时，分别找到距出发点和目标点最近的两个路标点，然后利用启发式搜索方法搜索路标连通图得到连接两个路标的路径，从而得到需要的路径。

在综合分析路径规划研究的基础上，为了提高路径规划的实时性，采用基于贪婪碰撞检测的启发式双向路径搜索对机械臂运动路径进行实时规划。其具有以下特点：不需要预计算路标图，这样节省了大量的预计算时间；进行双向采样，逐步增加以输入查询位形为根的两棵路标树来建立路标图，两棵树的交替采样能保持两棵树的平衡；采用贪婪碰撞检测，延迟碰撞检测操作直到完全需要检测，可以节省规划时间。

最后针对具体的路径规划问题分别进行了仿真分析，结果表明运用的规

划方法可以很好解决机械臂在工作过程中的平稳性，以及控制时的实时性问题。

关键词：检修机械臂；避障路径规划；包围盒；两阶段法；双向搜索

ABSTRACT

Avoidance obstacle path planning problem is able to be stated as follows: in obstacles environment, based on a certain evaluation criterion (such as the shortest length of path, the shortest time of moving, the minimal consuming of energy, and so on), from start point to destination point, plans a optimal (or sub-optimal) collision free (obstacle avoidance) path.

Kinematics of manipulator is the foundation of path planning to avoid obstacle, firstly, kinematics model was established in the paper, inverse kinematic problem-solving is relatively complicated, because in essence it is a settling nonlinearity equation problem, there is no universal method yet. According to the low redundancy, limited working environment of the manipulator, a simple and easy to apply method of a kind of geometry diagram method which is simple and easy to apply is given, and the simulation of the kinematics of the manipulator is verified in VC++ simulation platform.

When the manipulator is repairing, how to judge and avoid collision is the prime of automatic operating. A method named network bounding based on bounding box and equivalent cylinder is proposed, and the areas that might collision has been systematic analysed, by decomposing spatial local field effectively, simple and effective geometry equivalent, we transformed the manipulator's collision detecting to the solution of space analytic geometry, greatly simplifies the problem.

Robot collision-free path planning has not be well resolved until now, especially for the robot collision-free path planning in 3D space is more difficult. For probabilistic roadmap method has the characteristic that is easy to realize, has good generality, especially the effectiveness in solving multi-degree of freedom robot path planning, it causes researchers' attention in recent years. This paper

adopts normal probabilistic roadmap method to get a roadmap which expresses structure information in configuration space, tests connective relation between each milestone in advance to get a connected roadmap. When in actual path planning, at first to find out the nearest two milestones to the starting point and the target point, and then by the use of heuristic searching method to search the connected roadmap to find out a path connected the two milestones, that is the wanted path.

Based on comprehensive analysis of path planning, to improve the real-time performance, we adopt bi-directional search based on greedy collision detecting to plan the manipulator's motion in real time. This method has the following characteristics: do not need to compute the roadmap in advance, thus saved a lot of precomputation time; bi-directional sampling, to establish the roadmap by gradually increases the two landmark trees which root are the input query configurations, the two trees alternate sampling can keep the two tree balance; greedy collision detecting, delay the collision detection until definitely needed, which can save the planning time.

Finally, we carried out simulated operation separately aiming at the concrete path planning problem, the result show that the method we used can well resolve the running stability and controlling real-time quality when the manipulator works.

Key words: Repairing Manipulator, Obstacle Avoidance Path Planning, Bounding Box, Two-phase Method, Bi-directional Search

哈尔滨工程大学 学位论文原创性声明

本人郑重声明：本论文的所有工作，是在导师的指导下，由作者本人独立完成的。有关观点、方法、数据和文献的引用已在文中指出，并与参考文献相对应。除文中已注明引用的内容外，本论文不包含任何其他个人或集体已经公开发表的作品成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

作者（签字）：包卫卫

日期：2009年3月4日

哈尔滨工程大学 学位论文授权使用声明

本人完全了解学校保护知识产权的有关规定，即研究生在校攻读学位期间论文工作的知识产权属于哈尔滨工程大学。哈尔滨工程大学有权保留并向国家有关部门或机构送交论文的复印件。本人允许哈尔滨工程大学将论文的部分或全部内容编入有关数据库进行检索，可采用影印、缩印或扫描等复制手段保存和汇编本学位论文，可以公布论文的全部内容。同时本人保证毕业后结合学位论文研究课题再撰写的论文一律注明作者第一署名单位为哈尔滨工程大学。涉密学位论文待解密后适用本声明。

本论文（在授予学位后即可 在授予学位12个月后 解密后）由哈尔滨工程大学送交有关部门进行保存、汇编等。

作者（签字）：包卫卫

导师（签字）：陈军

日期：2009年3月4日

2009年3月4日

第 1 章 绪论

1.1 课题研究的背景和意义

核电站利用核燃料发生的受控自持链式裂变反应所释放的能量作为热源发电，而常规电厂则利用化石燃料燃烧所释放的能量作为热源发电。反应堆堆芯就是放置核燃料，实现持续的受控链式反应，从而成为不断释放出大量能量的场所。它相当于常规电厂中释放大量热量的锅炉，因此又可称核反应堆为核锅炉。

反应堆冷却剂系统又称为一回路系统，其主要功能是把堆芯正常运行时产生的热量传输给蒸汽发生器（简称 SG），将 SG 二回路侧的给水加热并转化为驱动汽轮发电机组的饱和蒸汽。

蒸汽发生器(简称 SG)是核动力装置中一、二回路之间的连接枢纽，也是核动力装置运行中发生故障最多的设备之一。蒸汽发生器的故障大多是由于各种腐蚀使传热管或管与管板接头处发生泄漏，从而影响动力装置的安全运行。蒸汽发生器的工作性能及安全可靠性对于整个系统的安全稳定运行是至关重要的。由于蒸汽发生器里面存在放射性物质，当发生泄漏时，人不能进去进行维修，这时就需要机械臂代替人完成维修工作。

蒸汽发生器是一种热交换器，其蒸发段主要由数千根倒置的 U 形传热管组成。一回路水在传热管内流动，二回路给水的加热和蒸发在传热管外侧进行。U 形管在管板孔内全胀，并与一回路侧管板表面相焊接。一回路水侧封头是一个半球形封头（下封头），与管板相焊接，并由一块隔板分隔为两个水室，每个水室有一个与反应堆冷却剂系统相连接的接管和一个用于进入检修的人孔。

通常，在核电站大修时，需对蒸汽发生器进行必要的检修活动。在 SG 一回路侧，一般有传热管的涡流检查（ET）、传热管超声检查（UT）、传热管堵管、水室表面状况的电视检查，等等。大多数的检修活动是远距离操控进行的，例如，一个机械臂通过人孔安装于 SG 水室内，机械臂的前端携挂不同的检修工具即可实施不同的检修活动。目前一些国家已经研发出一些用于 SG 检修的机械手产品，并且在实际的 SG 检修工作中投入使用，但国内目前

此项仍然空白。该课题是在我们与某研究所合作，研制蒸汽发生器检修机械臂的科研背景下进行的。附图 1.1~1.4 列出了韩国、印度、法国、美国 SG 检修机械臂的图片，美国 Westinghouse Electric Corporation 的 ROSAIII 是一个用于一回路侧检修活动的机器人，有六个关节，六轴联动，形成三维运动轨迹，机械臂的末端携挂工具完成一回路的各种检修工作，其主要完成 SG 水室内的管板检修与堵管，外壁检修和探伤等工作。

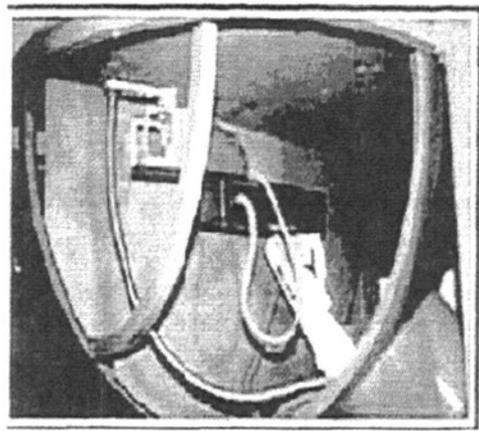
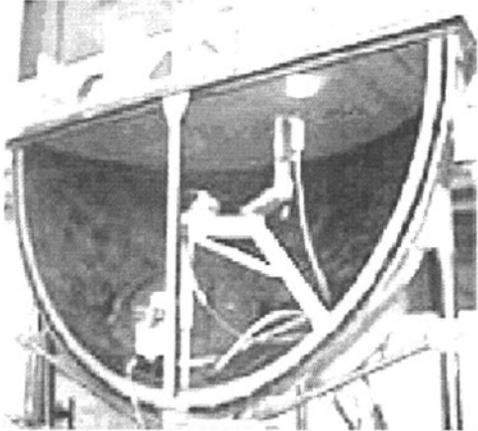


图 1.1 韩国原子能研究院检修机械臂 图 1.2 印度核动力公司检修机械臂

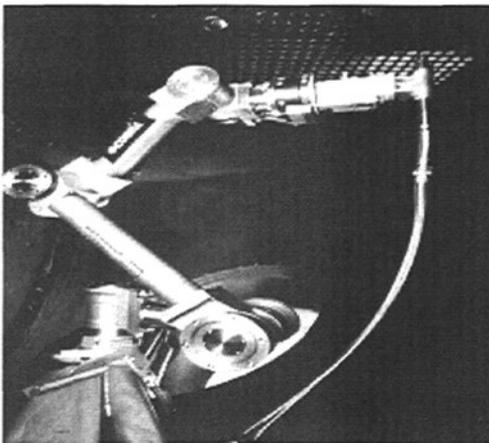


图 1.3 法国 Framatome 公司检修机械臂 图 1.4 美国西屋公司检修机械臂

避障路径规划是机器人应用中的一项重要技术^[1]。例如：在执行装配、焊接及抢险救灾等任务时，采用良好的机器人路径规划技术可以节省大量机器人作业的时间、减少机器人磨损，同时也可以节约人力资源，减小资金投入，为机器人在多种行业的应用奠定理论基础。蒸汽发生器检修机械臂的路径规划成为其经济、安全、高效检修不可或缺的技术。

1.2 机器人避障路径规划概述

移动机械手的研究涉及到很多不同的科学领域，如：移动机械手的路径规划、移动机械手的自主定位与导航技术及运动与操作之间的协调控制的理论和方法等，这些方面已经成为近年来国内外学术界的热点。在我国移动机械手目前还处于研究、跟踪和试验阶段。从目前情况看，其主要原因是：在非结构环境下，移动载体和其上的多自由度机械手的协调控制技术一直未能很好地解决，移动机械手的协调与控制机理和耦合机制还有待揭示，这也是移动机械手应用所面临的困难和挑战所在。

1.2.1 避障路径规划的定义

蒋新松为规划路径作出了这样定义：路径规划是自治式移动机器人的一个重要组成部分，它的任务就是在具有障碍物的环境内，按照一定的评价标准，寻找一条从起始状态，包括位置和姿态，到达目标状态位置和姿态的无碰路径。障碍物在环境中的不同分布情况当然直接影响到规划的路径，而目标位置的确定则是有更高级的任务分解模块提供的。

避障路径规划典型的描述是：避障路径规划是在具有障碍物的环境中，按照某个评价标准（如最短路径长度、最短行进时间、最小能量消耗等），规划一条从起始点位置到达目标点位置最优（或次优）的无碰（避障）路径^{[2][3]}。在此，“规划”的含义实际上是直观的求解带约束的几何问题，而不是操作序列或行为步骤。另一方面，如果把运动物体看作是所研究问题的某种状态，把障碍物看作是问题的约束条件，而无碰撞路径则为满足约束条件的解，那么，空间路径规划就是一种多约束的问题求解过程。

从数学角度看，避障路径规划问题可以表达为求解某个目标函数极值问题：目标函数就是所规划路径的成本；约束条件是避免与障碍物相碰撞，其数学模型可以表达为如下形式：

$$\min f(x) \quad x \in R_n \quad g_i(x) \leq b_i, \quad i = 1, 2, \dots, p \quad (1-1)$$

其中： $f(x)$ 为目标函数， $g_i(x)$ 是（非线性）约束条件， p 代表约束不等式个数^[4]。

在这里需要指出路径规划(Path Planning)、轨迹规划(Trajectory Planning)和轨迹跟踪(Trajectory Tracking)是有区别的。

路径只是指空间的曲线，而不包含时间的概念，对于机器人来说，定义为机器人位形的一个特定序列，而不考虑机器人位形的时间因素。一个点在空间移动，它所通过的全部路径叫做这个点的轨迹。

路径规划没有考虑系统的动力学，它作为一个几何问题，主要是对算法的研究；轨迹规划是给定机器人末端执行器的跟踪路径及描述机器人动力学和描述力矩约束的微分方程，规划出机器人运动过程中的位置、速度及加速度，以使得某个目标最优。动力学问题是轨迹规划主要考虑的问题，轨迹规划的结果作为轨迹跟踪的输入，机器人控制器必须保证机器人精确跟踪已规划出的轨迹^[9]。

1.2.2 避障路径规划问题的分类

据对环境的了解情况，路径规划可以划分为全局路径规划和局部路径规划。其中，全局路径规划需要知道关于环境的所有信息，根据环境地图进行大粒度的路径规划，并产生一系列关键点作为子目标点下达给局部路径规划系统。而局部路径规划则只需要知道距离机器人最近的障碍物信息。机器人行走过程中，根据传感器的信息来不断地更新其内部的环境表示，从而确定出机器人在地图中的当前位置及周围局部范围内的障碍物分布情况，并在此基础上，规划出一条从起点或某一子目标点的优选路径。但局部路径规划只适合于起终点相距较近的情况，可以看作是全局路径规划在机器人实时行走过程中的一个补充。如果从静态或动态地获取障碍物信息角度看，全局路径规划属于静态规划，而局部路径规划则是动态规划^{[6][7]}。

另外，根据完整性还可以把路径规划算法分为确定型和启发型两类。确定型算法可以找到一个问题的答案，除非它证明问题无解。启发型算法致力于在较短的时间内找出问题的答案，它可能无法解决较困难的问题或无法给出一个问题的最优解。

1.2.3 避障路径规划问题的环节及特点

路径规划的主要环节包括环境建模、路径搜索及生成、路径优化或平滑。在机器人规划前，即研究如何将环境信息有效地表达出来，它首先要做的就是，将机器人活动空间的描述由外部的原始形式通过一系列处理转化为适合

规划的内部模型，这个过程我们称为环境建模，其中主要是障碍物的表示方法。简单地说，就是要将物理上的境描述为计算机可以识别的形式。合理的环境表示有利于减少规划中的搜索量及时空开销。不同的规划方法正是基于不同的环境建模基础之上的。路径搜索算法负责从环境模型中搜索出路径的可行空间，而且，一般都与建模方法有关。路径生成则是从搜索到的路径可行空间中生成一条可行路径。路径优化是在考虑智能机器人自身动力学特性的基础上，为了让路径更有利于机器人的执行而对路径进行的平滑。

路径规划属于组合优化的范畴。在一定的环境表达和规划方法的基础上，仅从路径搜索策略上来分析，可将搜索策略区分为三大类。第一类是局部优化算法，它的做法是牺牲解的搜索空间来获取解的搜索时间，典型的方法有：最速下降法、部分贪婪算法（如凸点法^[8]）等；第二类是全局优化算法，它以获取全局最优解为目的，主要的算法有^[9]：单源最短路径的 Dijkstra 算法、任意两点间最短路径的 Floyd 动态规划算法、A*搜索算法等；最后一类是计算智能方法^{[10][11]}。

避障路径规划问题具有如下特点：

- 1、复杂性。在复杂环境尤其是动态时变环境中，避障路径规划问题非常复杂，且需要很大的计算量。
- 2、随机性。复杂环境的变化往往存在很多随机性和不确定因素。动态障碍物的出现也带有随机性。
- 3、多约束。运动对象存在几何约束和物理约束。几何约束指的是运动对象的形状制约，而物理约束是指运动对象的速度和加速度。
- 4、多目标。运动对象在运动过程中路径性能要求存在多种目标，如路径最短、时间最优、安全性能最好以及能源消耗最小等。但诸目标之间往往存在冲突。

因此避障路径规划问题是动态多目标优化问题。

1.2.4 避障路径规划的方法

狭窄空间内对机械臂进行避障前提下的路径规划是相当困难的。目前，机器人运动规划可以分为两类：全局路径规划和局部路径规划。

全局路径规划是指在一个特定的工作区域中自动生成一个从起始作业点

开始,经过一系列作业点,到达目标点的最优工作序列。规划路径的精确程度取决于获取环境信息的准确程度。全局路径规划方法通常可以寻找最优解,但需要预先知道准确的全局环境信息。通常该方法计算量大,实时性差,不能较好地适应动态非确定环境。基于环境建模的全局路径规划的方法主要有自由空间法^[12]、构型空间法、拓扑法^{[13][14]}和栅格法^[15]等。

局部路径规划指在相邻序列点之间通过一定的算法搜索一条无碰撞的机器人运动路径。路径规划是机器人的核心技术之一,规划功能的强弱反映了机器人的智能水平。全局路径规划根据环境地图进行,通过全局规划产生一系列关键点作为子目标点下达给局部路径规划系统。局部路径规划是以全局路径规划为指导,把全局路径规划得到的一系列路径关键点作为系统子目标,根据传感器信息确定机器人在环境中的当前位置及周围局部范围内的障碍物分布情况,然后实时规划出一条从起点或某一子目标点到下一子目标点或终点的最优路径,并实时避开出现的未知的障碍物。

与全局规划方法相比,局部规划更具实时性和实用性,对动态环境具有较强适应能力其缺点是由于仅依靠局部信息,有时会产生局部极值点或振荡,无法保证机器人能顺利地到达目标点。这类方法包括人工势场法^[6]、遗传算法^{[16]—[20]}、模糊逻辑算法^[21]、神经网络方法^[22]、蚁群算法、粒子群算法、路标法等。

高维位形空间中的完整路径规划促使研究人员采用启发式方法来弱化其完整性,或者部分采用特殊的方法来提升在这些方面的性能。

近年来,利用人工势场法得到了一些重要的结果。这些方法主要缺点是容易陷入局部最小值。很难摆脱这些最小值。局部最小值的势函数在^{[23][24][25]}定义了。在高维空间中计算这些势函数很耗时,并且很难应用到多自由度的情况。

文献[26]中讲述了单点随机规划器,在[27]中扩展为一种学习方法。这些文章的重点放在自由度较少的情况上。相似的技术应用在了可以前后移动和只可以向前移动的车辆机器人上^{[28][29]}。在[30]中该技术被扩展为多个机器人的运动协调。文献[31]中对学习阶段也独自进行了相似的处理。也在自由空间中建立了概率路标图,但主要针对多自由度问题。强调了在自由空间中“难点”区域扩展的必要。文献[32]中介绍了更好的扩展技术。该方法在[33]

中进行了详细的描述,文献[34]对使用该规划器的时间消耗进行了理论分析。本文对之前的方法进行了有效的整合,提出了两种效能不同的路径规划方法。

1.3 本论文所研究的目标和内容

1.3.1 本论文研究的目标

狭窄空间内多关节检修机械臂的避障路径规划。SG 机械臂的工作空间一般较为狭窄,但是其较大的关节力矩要求限制了机械臂尺寸无法进一步缩小,而且机械臂结构形式复杂,导致机械臂的碰撞约束非常复杂,从而使得机械臂在 SG 内进行无碰撞自动路径规划变得相当困难,既要考虑机械臂与周围环境的避碰,还要考虑其自身关节及检修工具之间的碰撞,美国 ROSA III 机械臂的控制系统目前该功能尚不完善。

本课题以六自由度蒸汽发生器检修机械臂为例,提出在狭小空间内进行工作并且避免碰撞的路径规划方法。并通过仿真比较,选出一种适合于蒸汽发生器检修机械臂路径规划方法。本课题的路径规划主要是在关节空间中,根据检修机械臂运动周围环境和自身约束及可实现性的要求来进行规划,预解决在机器人操作空间路径规划时,运动的平稳性以及实时性和精确性的矛盾,并要完成避碰检测。

1.3.2 本论文主要内容

路径规划问题涉及环境表达、规划方法和路径搜索策略。环境表达研究如何将环境信息有效地表达出来;规划方法关心的是在环境表达的基础上而进行的数学模型的抽象;路径搜索策略指的是求解路径函数的方法技术。

(1)六自由度机械臂运动学模型的建立

由于路径规划是建立在机器人运动学和动力学基础上的,所以有必要先对该检修机械臂进行数学建模。文中首先给出了机械臂的 D-H 描述,在此基础上建立了机械臂的正逆运动学方程,并在 VC++ 仿真平台下对运动学求解进行了仿真。

(2)狭窄空间机械臂动态碰撞检测的完成

由于机械臂的造价比较昂贵,为保障机械臂检修工作的安全进行,针对其具体结构及工作环境设计碰撞检测算法。

(3)路径规划的实现方法

路径规划是属于机器人规划的底层规划，是指移动机器人按照某一性能指标(如距离、时间、能量等)搜索一条从起始状态到目标状态的最优或次优路径。

由于本机械臂路径规划的约束条件较多，要避免其与周围水室、管板面、挡板面、自身各关节及检修工具的碰撞，另外不是水室各个位置均可以达到，所以还涉及到其自身六个关节的可实现性问题，以上两点成为路径规划的难点。

面对一个具体的路径规划，首先考虑存在哪些约束，论文把它们划分成四大类：有障碍物和有路径约束；有障碍物和无路径约束；无障碍物和有路径约束；无障碍物和无路径约束。由于实际中，机械臂工作环境的复杂性，本文重点放在有障碍约束和无路径约束上，即考虑有障碍约束的路径规划问题。其次，要考虑路径规划的可实现性。由于该机械臂由六个关节构成，涉及到规划路径的可实现性问题，即要判断规划的路径能否通过正反解算法转化成机械臂各关节的转角，达到预期的位置。参考相关机械臂或机器人路径规划方法，并针对本机械臂在考虑以上两个基本要素情况下完成对路径的规划。

(4)路径规划的模拟实现平台

用 C 语言编写程序，在 VC++ 仿真平台下模拟路径规划的执行过程，通过建立的三维运动模型实时仿真，观察并分析路径规划的可行性与正确性。

第 2 章 狭窄空间机械臂运动学建模

机械臂的路径规划属于机械臂低层规划，是在机械臂运动学的基础上，研究在关节空间和笛卡儿空间中机械臂运动的路径规划和路径生成及优化方法。由于它的基础是机械臂的运动学，所以有必要首先建立六关节检修机械臂的运动学模型。

2.1 机械臂的 D-H 描述

本文在这里，将要对于蒸汽发生器检修的机械臂进行数学分析建模，并把它作为后面仿真分析的模型。图 2.1 中四幅图给出了机械臂检修过程中不同位置的实物图，图 2.1 (a) 为机械臂进行检修工作前抓顶位置的图片，图 2.1 (b) 为安装检修工具后的检修图，图 2.1 (c) 为机械臂堵管前六关节及工具头位置图，图 2.1 (d) 为完整的（携挂工具的）机械臂机械结构图。

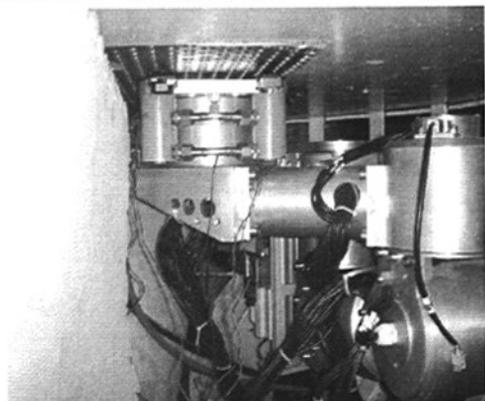


图 2.1 (a) 机械臂抓顶图

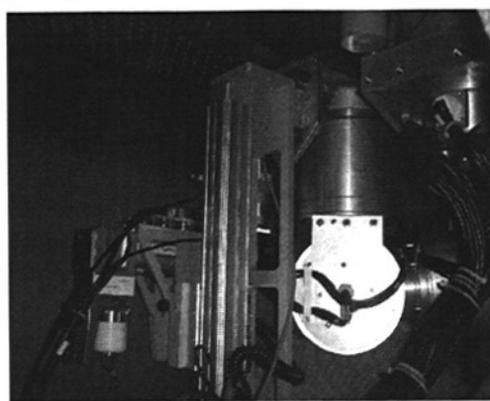


图 2.1 (b) 携挂工具检修图

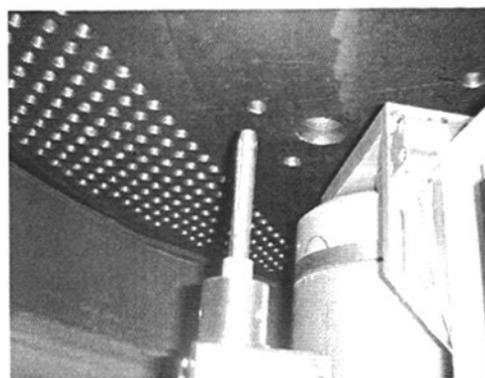


图 2.1 (c) 机械臂堵管图

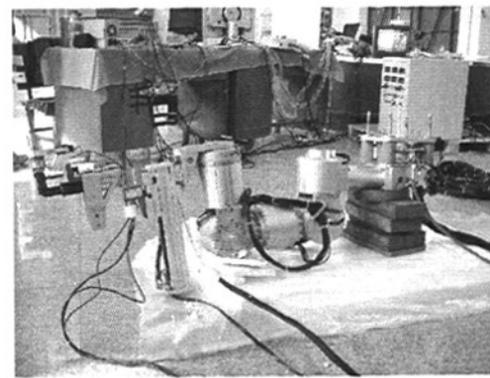


图 2.1 (d) 携挂工具的机械臂本体图

该机械臂共包含六个关节，全部为旋转关节。图 2.2 为与图 2.1 (d) 相对应的检修机械臂的三维效果图，也是机械臂工作过程的零位图。工作过程一至六关节的初始转角为 $(0^\circ, 90^\circ, 0^\circ, 0^\circ, 90^\circ, \theta_B)$ ，其中 θ_B 视安装工具时工具的偏心角而定。

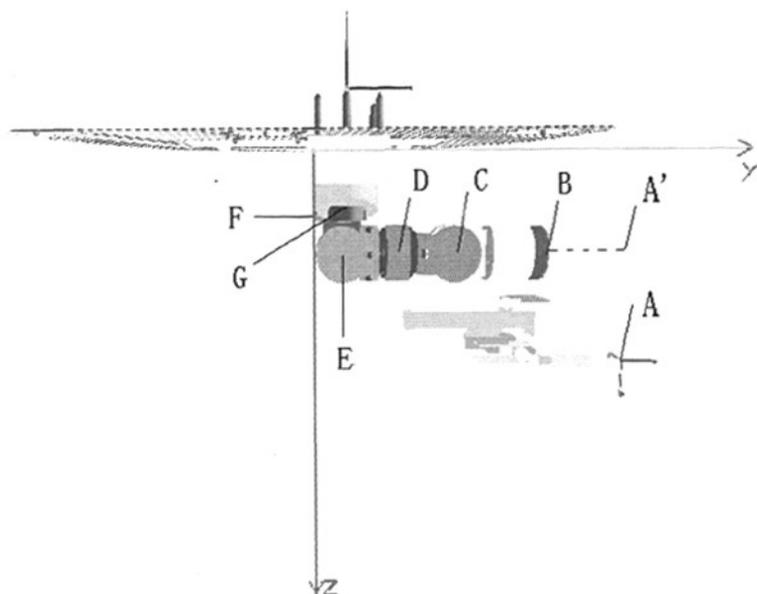
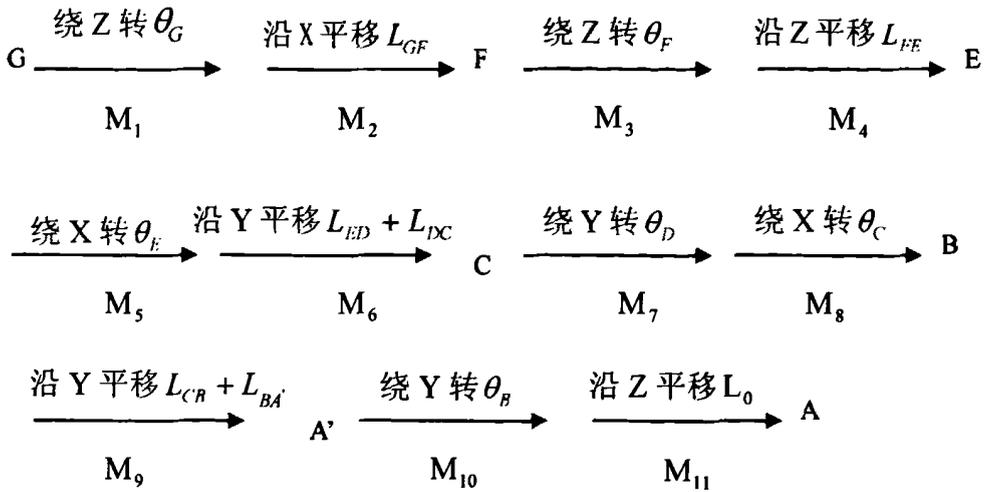


图 2.2 机械臂三维效果图

设 G 位于基坐标系原点， A 位于末端坐标系（工具坐标系）原点，六个关节中心分别位于 B, C, D, E, F, G 处，由于工作需要，该机械臂使用偏心工具，使得工具末端位置 A 点偏离原工具中心 A' 点，为方便分析问题，我们将最后一个关节（ B ）坐标系的原点置于 A' 点处，先解算到 A' 点再向 A 点进行旋转平移变换，设机械臂各关节长为 $L_{GF}, L_{FE}, L_{ED}, L_{DC}, L_{CB}$ ，工具偏心尺寸为 L_0 ，变换关系如下：



由于机械臂的具体工作需要，使得机械臂工具的末端安装偏心工具，而且工作时，末端姿态三个旋转关节中有一个关节（工具绕自身轴的旋转）可任意，A' 点为不考虑工具偏心的机械臂末端点，由于 A' 点的位置与最后一个关节的旋转角度 θ_B 无关，所以如果我们先不考虑工具偏心，此时机械臂只有五个待求关节角度，且工作空间有五个限定自由度，成为非冗余机械臂问题。

表 2.1 机械臂连杆参数

J_i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0	0	0	θ_G
2	0	L_{GF}	L_{FE}	$\theta_F - 90^\circ$
3	-90°	0	0	$\theta_E - 90^\circ$
4	90°	0	$L_{ED} + L_{DC}$	θ_D
5	-90°	0	0	θ_C
6	90°	0	$L_{CB} + L_{BA}$	θ_B

当各连杆参数确定以后，我们便可以直接写出相邻两坐标系间的变换矩阵^[35]：

$${}^{i-1}T_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & a_{i-1} \\ \sin \theta_i \cos \alpha_{i-1} & \cos \theta_i \cos \alpha_{i-1} & -\sin \alpha_{i-1} & -\sin \alpha_{i-1} d_i \\ \sin \theta_i \sin \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & \cos \alpha_{i-1} & \cos \alpha_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-1)$$

我们有

$${}^0T_1 = \begin{bmatrix} \cos(\theta_G) & -\sin(\theta_G) & 0 & 0 \\ \sin(\theta_G) & \cos(\theta_G) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad {}^1T_2 = \begin{bmatrix} \sin(\theta_F) & \cos(\theta_F) & 0 & L_{GF} \\ -\cos(\theta_F) & \sin(\theta_F) & 0 & 0 \\ 0 & 0 & 1 & L_{FE} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T_3 = \begin{bmatrix} \sin(\theta_E) & \cos(\theta_E) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \cos(\theta_E) & -\sin(\theta_E) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad {}^3T_4 = \begin{bmatrix} \cos(\theta_D) & -\sin(\theta_D) & 0 & 0 \\ 0 & 0 & -1 & -L_{ED} - L_{DC} \\ \sin(\theta_D) & \cos(\theta_D) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4T_5 = \begin{bmatrix} \cos(\theta_C) & -\sin(\theta_C) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\sin(\theta_C) & -\cos(\theta_C) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad {}^5T_6 = \begin{bmatrix} \cos(\theta_B) & -\sin(\theta_B) & 0 & 0 \\ 0 & 0 & -1 & -L_{CB} - L_{BA} \\ -\sin(\theta_B) & \cos(\theta_B) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

其中 $M_1 = {}^0T_1, M_3 = {}^2T_3, M_5 = {}^3T_3, M_7 = {}^3T_4, M_8 = {}^4T_4, M_{10} = {}^5T_5$,

$$M_2 = \begin{bmatrix} 1 & 0 & 0 & L_{GF} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad M_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_{FE} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad M_6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L_{ED} + L_{DC} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$M_9 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & L_{CB} + L_{BA} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad M_{11} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & L_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

依次连乘各个相邻坐标系间的变换阵，便可以得到机械臂末端工具坐标系到基坐标系间的变换关系，设 G 点齐次坐标为 P，则可以得到 A 点坐标 $M_1 \cdot M_2 \cdot M_3 \cdot M_4 \cdot M_5 \cdot M_6 \cdot M_7 \cdot M_8 \cdot M_9 \cdot M_{10} \cdot M_{11} \cdot P$ ，由此可见，若已知各关节转角即可以很方便地求解该机械臂的运动学正解问题。

2.2 检修机械臂正运动学方程的建立

我们已经推导机械臂各连杆坐标系间的变换矩阵，下面将在此基础上给出机械臂运动学方程：

$$T = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2-2)$$

在此， T 代表机械臂的末端连杆位姿矩阵，该矩阵可由指定的 16 个元素的数值决定。在这 16 个元素中，只有 12 个元素有实际含义。底行由 3 个 0 和一个 1 组成。左列矢量 n 是第二列矢量 o 和第三列矢量 p 的差乘。其中各个坐标系间的变换阵及杆长参数在前面已经给出，直接带入(2-2)有

$$n_x = (((C6 * S5 + S6 * C5) * S4 * C3 + (C6 * C5 - S6 * S5) * S3) * C2 + (C6 * S5 + S6 * C5) * C4 * S2) * C1 + (-(C6 * S5 + S6 * C5) * S4 * S3 + (C6 * C5 - S6 * S5) * C3) * S1 \quad (2-3)$$

$$o_x = -(((C6 * S5 + S6 * C5) * S4 * C3 + (C6 * C5 - S6 * S5) * S3) * C2 + (C6 * S5 + S6 * C5) * C4 * S2) * S1 + (-(C6 * S5 + S6 * C5) * S4 * S3 + (C6 * C5 - S6 * S5) * C3) * C1 \quad (2-4)$$

$$a_x = ((C6 * S5 + S6 * C5) * S4 * C3 + (C6 * C5 - S6 * S5) * S3) * S2 - (C6 * S5 + S6 * C5) * C4 * C2 \quad (2-5)$$

$$p_x = L0 * (((C6 * S5 + S6 * C5) * S4 * C3 + (C6 * C5 - S6 * S5) * S3) * C2 + (C6 * S5 + S6 * C5) * C4 * S2 + 3/20 * (-(C6 * S5 + S6 * C5) * S4 * S3 + (C6 * C5 - S6 * S5) * C3 + 151/500 * ((C6 * S5 + S6 * C5) * S4 * C3 + (C6 * C5 - S6 * S5) * S3 + (C6 * S5 + S6 * C5) * C4 * C2 * (-L1 - L2) - (C6 * S5 + S6 * C5) * C4 * L3 - (C6 * S5 + S6 * C5) * C4 * L4 + C6 * L6) \quad (2-6)$$

$$n_y = (((S6 * S5 - C6 * C5) * S4 * C3 + (C6 * S5 + S6 * C5) * S3) * C2 + (S6 * S5 - C6 * C5) * C4 * S2) * C1 + (-(S6 * S5 - C6 * C5) * S4 * S3 + (C6 * S5 + S6 * C5) * C3) * S1 \quad (2-7)$$

$$o_y = -((S6 * S5 - C6 * C5) * S4 * C3 + (C6 * S5 + S6 * C5) * S3) * C2 + (S6 * S5 - C6 * C5) * C4 * S2 * S1 + (-S6 * S5 - C6 * C5) * S4 * S3 + (2-8) (C6 * S5 + S6 * C5) * C3) * C1,$$

$$a_y = ((S6 * S5 - C6 * C5) * S4 * C3 + (C6 * S5 + S6 * C5) * S3) * S2 - (S6 * S5 - C6 * C5) * C4 * C2 \quad (2-9)$$

$$p_y = L0 * (((S6 * S5 - C6 * C5) * S4 * C3 + (C6 * S5 + S6 * C5) * S3) * C2 + (S6 * S5 - C6 * C5) * C4 * S2) * C1 + 3/20 * (-S6 * S5 - C6 * C5) * S4 * S3 + (C6 * S5 + S6 * C5) * C3) * S1 + 151/500 * ((S6 * S5 - C6 * C5) * S4 * C3 + (2-10) (C6 * S5 + S6 * C5) * S3) * S2 + (S6 * S5 - C6 * C5) * C4 * C2) * (-L1 - L2) - (S6 * S5 - C6 * C5) * C4 * L3 - (S6 * S5 - C6 * C5) * C4 * L4 + S6 * L6$$

$$n_z = (C4 * C3 * C2 - S4 * S2) * C1 - C4 * S3 * S1 \quad (2-11)$$

$$o_z = -(C4 * C3 * C2 - S4 * S2) * S1 - C4 * S3 * C1 \quad (2-12)$$

$$a_z = C4 * C3 * S2 + S4 * C2 \quad (2-13)$$

$$p_z = L0 * (C4 * C3 * C2 - S4 * S2) * C1 - 3/20 * C4 * S3 * S1 + (-C4 * C3 * S2 - S4 * C2) * (-L1 - L2) + S4 * L3 + S4 * L4 + L5 \quad (2-14)$$

式中 C_i, S_i 分别代表 $\cos(\theta_i), \sin(\theta_i)$, 其中 $n = [n_x, n_y, n_z]^T$ 为法向矢量, 为机械臂末端坐标系的 OX 轴在其坐标系下的表示; $o = [o_x, o_y, o_z]^T$ 为法向矢量, 为机械臂末端坐标系的 OY 轴在其坐标系下的表示; $a = [a_x, a_y, a_z]^T$ 为法向矢量, 为机械臂末端坐标系的 OZ 轴在其坐标系下的表示。
 $p = [p_x, p_y, p_z]^T$ 为机械臂工具末端在其坐标系下的位置矢量。

2.3 检修机械臂逆运动学方程的建立

机械臂求取运动学逆解的方法很多, 其中主要有坐标变换法、图解法、数值解析法等。根据该机械臂结构的冗余度低, 工作环境受限的实际情况, 给出了一种简单易行的几何图解法。为了表述方便, 设 $\alpha_6 \sim \alpha_1$ 分别为第一关节至第六关节的转角。由 A 与 C 的空间坐标可直接确定机械臂的末端位置和姿态, 为此整个求解过程分为两步: 根据 C 点的坐标解算电机角度 $\alpha_4, \alpha_5, \alpha_6$; 根据 A 点的坐标解算电机角度 $\alpha_1, \alpha_2, \alpha_3$ 。

1. 根据 C 点的坐标解算电机角度 $\alpha_4, \alpha_5, \alpha_6$

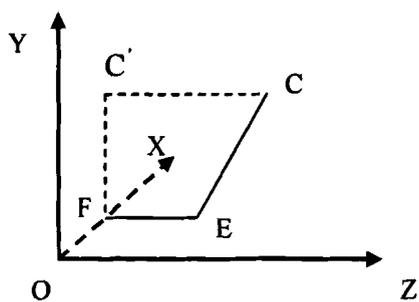


图 2.3

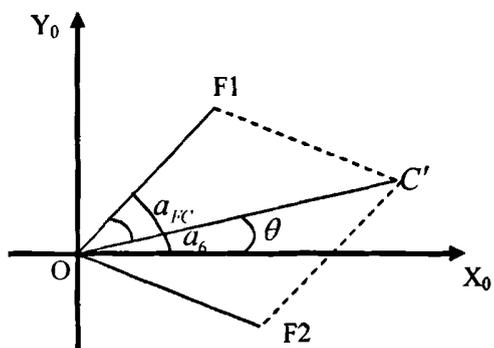


图 2.4

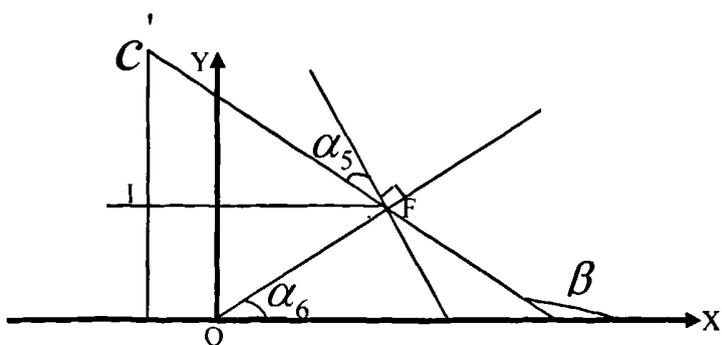


图 2.5

如图 2.3 所示将臂 CE 向 XY 面投影得 $C'F$ (F 点不一定在 X 轴上), 并有:

$$d_{CF} = d_{CE} \cdot \cos \alpha_4 \quad (2-15)$$

$$Z_c = d_{CE} \sin \alpha_4 + d_{EF} \quad (2-16)$$

由(2-15)式和(2-16)式可解得角度 α_4 与 $C'F$ 的长度 d_{CF}

$$\alpha_4 = \arcsin((Z_c - d_{EF}) / d_{CE}) \quad (2-17)$$

$$d_{CF} = \sqrt{d_{CE}^2 - (Z_c - d_{EF})^2} \quad (2-18)$$

如图 2.4 所示, 由

$$d_{oc'} = \sqrt{x_c^2 + y_c^2} \quad (2-19)$$

$$d_{C'F}^2 = d_{OF}^2 + d_{OC'}^2 - 2d_{OF} \cdot d_{OC'} \cdot \cos \alpha_{FC'} \quad (2-20)$$

联立可以求得 $\alpha_{FC'}$,

$$\alpha_{FC'} = \arccos((d_{OF}^2 + d_{OC'}^2 - d_{FC'}^2) / (2d_{OF} \cdot d_{OC'})) \quad (2-21)$$

若 $\angle FOX > \angle C'OX$, 如图 2.4 所示, 可以求得:

$$\alpha_6 = \theta + \alpha_{FC'} = \arctan(y_c / x_c) + \arccos((d_{OF}^2 + d_{OC'}^2 - d_{FC'}^2) / (2d_{OF} \cdot d_{OC'})) \quad (2-22)$$

若 $\angle FOX < \angle C'OX$, 同理可以求得:

$$\alpha_6 = \theta - \alpha_{FC'} = \arctan(y_c / x_c) - \arccos((d_{OF}^2 + d_{OC'}^2 - d_{FC'}^2) / (2d_{OF} \cdot d_{OC'})) \quad (2-23)$$

要求 α_6 在 $0^\circ \sim 90^\circ$ 和 $270^\circ \sim 360^\circ$ 之间, 实际计算中利用限定条件来选择较优的一组解, 机械臂必须始终位于挡板右侧, 故要求 F 点的 X 坐标满足 $X_F > 0$, 当两组解同时满足条件时可选择距上次 α_6 较近的解。另外, 若求出的结果在 $90^\circ \sim 180^\circ$ 之间, 应进行角度转换, 保证求得的角度在 $0^\circ \sim 360^\circ$ 之间, 即计算 $360^\circ - (\alpha_6 - 90^\circ)$ 作为最后的结果。

若 $\angle FOX < \angle C'OX$, 由图 2.5, 有:

$$\beta = \arctan\{(y_c - L_{GF} \cdot \sin \alpha_6) / [x_c - L_{GF} \cdot \cos \alpha_6]\} \quad (2-24)$$

$$\alpha_5 = \beta - 90^\circ - \alpha_6 \quad (2-25)$$

若 $\angle FOX > \angle C'OX$ 同理可以求得如上结果。

2. 根据 A 点的坐标解算 $\alpha_1, \alpha_2, \alpha_3$ 。

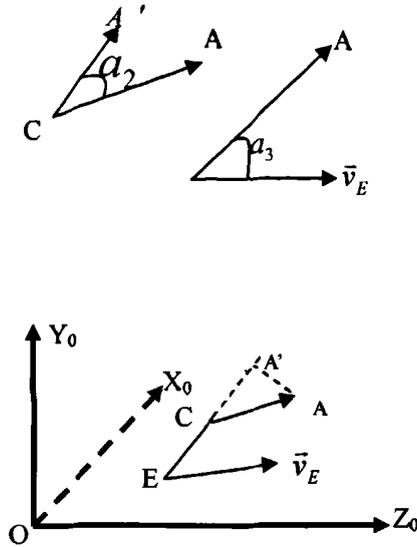


图 2.6

图 2.6 中只画出机械臂 E、C、A 段， A' 为 A 在 EC 轴上的投影， \vec{v}_E 为 E 点对应的电机轴向向量。

$$\vec{v}_{CA'} = (\vec{v}_{CA} \cdot \frac{\vec{v}_{EC}}{\|\vec{v}_{EC}\|}) * \frac{\vec{v}_{EC}}{\|\vec{v}_{EC}\|} \quad (2-26)$$

\vec{v}_{EC} 也就是 D 点电机轴向向量。

$$\vec{v}_{A'A} = \vec{v}_{CA} - \vec{v}_{CA'} \quad (2-27)$$

$$\alpha_2 = \arccos\left(\frac{\vec{v}_{CA} \cdot \vec{v}_{CA'}}{\|\vec{v}_{CA}\| \cdot \|\vec{v}_{CA'}\|}\right) \quad (2-28)$$

$$\alpha_3 = \arccos\left(\frac{\vec{v}_E \cdot \vec{v}_{A'A}}{\|\vec{v}_E\| \cdot \|\vec{v}_{A'A}\|}\right) \quad (2-29)$$

上式中 E 点坐标与 \vec{v}_E 矢量经已知的 $\alpha_4, \alpha_5, \alpha_6$ 利用坐标变换阵推出。另外由于臂的前端结构，导致对于任一 A 点， α_2, α_3 有两组解。系统中为得到唯一解，限定 $\alpha_2 > 0$ 。

由于使用偏心工具，要考虑第六关节的转角 α_1 。通过调用工作过程的正解，求取向量 \vec{BP} (B 是六关节中心点，P 为工具偏心点)，其与起始工具偏

心的位置向量的夹角即为 α_1 。考虑到转动方向的不同， α_1 有正负两种可能，通过判断 \vec{BP} 与起始工具偏心的位置向量的差乘得到向量与第六关节轴向向量方向异同取负正号。

2.4 检修机械臂运动学求解仿真

在VC++开发平台下利用3D MAX和OpenGL构建的检修机械臂工作环境的仿真模型，编制了上述正逆解程序，得到实验结果如下：

表 2.2 仿真数据及结果

采样点	1	2	3	4
x	0.086	0.155	0.236	0.326
y	-0.288	-0.288	-0.288	-0.288
z	0.043	0.043	0.043	0.043
α	179.044	179.044	179.044	179.044
β	90.927	90.927	90.927	90.927
γ	61.937	61.937	61.937	61.937
α_1	6.660	10.779	11.970	9.468
α_2	57.898	67.142	80.118	97.952
α_3	161.675	161.675	161.675	161.675
α_4	1.325	1.1839	0.965	0.663
α_5	197.894	197.614	197.361	197.153
α_6	48.038	61.355	75.4534	90.691
x'	0.086	0.155	0.236	0.326
y'	-0.288	-0.288	-0.288	-0.288
z'	0.043	0.043	0.043	0.043
α'	179.043	179.043	179.043	179.043
β'	90.926	90.926	90.926	90.926
γ'	61.937	61.937	61.937	61.937

表 2.2 中 $x, y, z, \alpha, \beta, \gamma$ 为设定的已知量， $\alpha_1 \sim \alpha_6$ 为经逆解解算的结果， $x', y', z', \alpha', \beta', \gamma'$ 为根据逆解结果求得的正解，正反解算法解得的结果基本

一致，从而验证了该求解方法的精度和有效性。

2.5 本章小结

本章分析了机械臂的运动学方程的建立方法，介绍了在连杆坐标系中，把基于基坐标系或是变化坐标系任何的齐次坐标变换进行分解的原理；重点讲述了机械臂的 D-H 描述，机械臂正、逆运动学方程的建立方法及过程。特别地，根据该机械臂结构的冗余度低，工作环境受限的实际情况，针对机械臂逆解问题，给出了一种简单易行的几何图解法。并对位于狭窄空间中机械臂的运动学正逆解解算方法进行了仿真实验，结果验证了该求解方法的精度和有效性。

第 3 章 狭窄空间机械臂的碰撞检测约束

对于多关节机器人的碰撞检测问题国内外学者进行了大量的工作,研究出了许多模型和算法^{[36][37][38]}。有的学者将机器人关节系统简化为球类实体模型。例如文献[39]用一组球和球台来逼近机器人系统。显然,某些部位的逼近精度有待提高。有的学者将机器人关节简化为圆柱体两端带半球的实体模型^{[40][41][42]}。其中文献[40]把机器人的杆件(关节和手腕)简化为圆柱和球体,利用最小距离函数规划避障路径。文献[42]将机械臂的各关节均视为由若干圆柱体两端带半球的实体所组成,计算出两个实体的圆柱体轴线段的最小距离,然后判断出机器人的关节实体之间是否碰撞。

蒸汽发生器是核动力装置中一、二回路之间的连接枢纽,由于蒸汽发生器里面存在放射性物质,当其内部发生故障时,需要机械臂代替人完成故障检修任务。由于机械臂的造价比较昂贵,为实现机械臂检修工作的安全进行,针对其具体结构及工作环境设计了该碰撞检测算法。为实现对机械臂的远程控制以完成检修任务,设计了上位机监控软件对机械臂进行实时控制,实时模拟机械臂在蒸汽发生器内的姿态^{[43][44]},显示出机械臂的运行参数并在必要时给出警告信息,在要发生碰撞的位置能够实时显示出碰撞点。

在 VC++ 开发平台下,完成了蒸汽发生器检修机械臂的碰撞检测,实现了机械臂的快速有效避碰,为机械臂的安全工作提供了保障。

3.1 工作环境及工作过程介绍

利用 3D MAX 和 OpenGL^[45]构建了检修机械臂工作的仿真模型,包括机械臂本体、检修工具、推进装置、水室等几个部分,图 3.1 为机械臂进行检修工作的三维模型图,各个组成部分位置如图 3.1 所示。

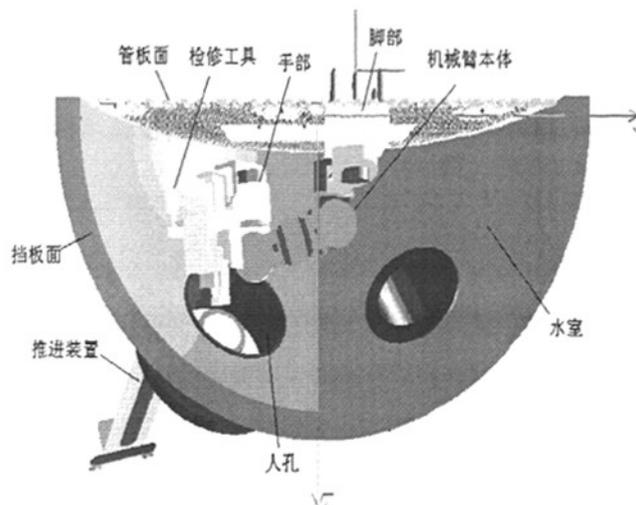


图 3.1 机械臂进行检修工作的三维模型图

根据蒸汽发生器实际检修任务的需要，机械臂设计成六个关节的串联机械臂。机械臂的脚部有五个脚趾，工作时五个脚趾分别插入蒸汽发生器管板面上的五个管孔内，靠脚趾内的液压装置涨紧，机械臂的手部装有检修用的各类工具，并且工具偏心安装，以适应蒸汽发生器内狭窄的工作空间。

首先将机械臂的手部安装在人孔处的推进装置（套筒）上，通过推进装置将调整好关节位置的机械臂送入蒸汽发生器内，之后控制五个脚趾插入管孔并自动涨紧，然后控制机械臂手部脱离推进装置；机械臂安装抓顶完毕之后在手部通过适配器安装上检修工具，控制机械臂收入蒸汽发生器内部进行各种检修工作，检修完毕将机械臂从人孔退出水室。

3.2 动态碰撞检测的基本原理

由于实时性的要求，在虚拟场景中遍历所有基本几何元素的两个凸多面体的碰撞检测很难应用。一般是用相对简单的包围盒包裹虚拟对象，并用包围盒代替虚拟对象进行碰撞检测^[46]。包围盒的简单性和它包裹虚拟对象的紧密性是一对矛盾^{[47][52]}，根据对碰撞检测精度要求的不同采用不同的检测深度，以提高整个系统的检测效率。碰撞检测需要检测在连续路径上所有组成部分是否无碰撞。

该机械臂的工作空间比较狭窄，可能发生的碰撞部位很多，如何能够快速有效地实现检修工作的避碰成为碰撞检测的难点。文中采用类似于 AABB

盒的网络包围法，在建立有效的信息存储结构前提下进行检测。机械臂本体用线段包络等效后的模型体如图 3.2 所示。

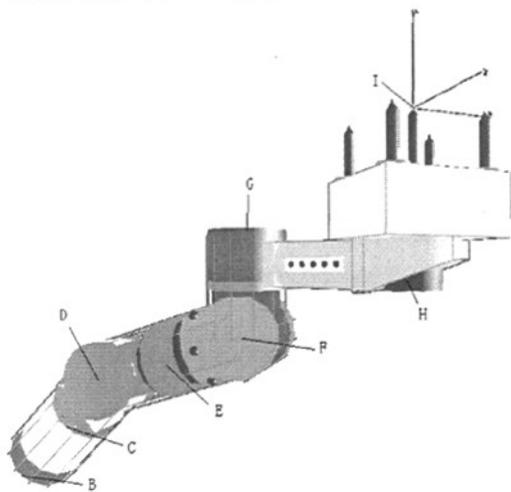


图 3.2 机械臂本体等效后模型体

其中，C、D、E、F、G、H 分别表示第六至第一关节点，B 为首部末端点，I 为脚部末端点。兼顾到包围盒的简单性和紧密性，各个臂用一定粗度的线段包络起来，即建立对应的包围盒树。这样，问题归结为遍历两棵树之间的 AABB 子盒之间各条线段的检测。对于该机械臂的工作任务，该法能够很好地解决碰撞检测的实时性和精确性的矛盾。

将机械臂的碰撞分为三种情况：机械臂各关节与环境内壁的碰撞，机械臂自身各关节间的碰撞，检修工具与环境内壁以及与机械臂各关节间的碰撞。

多关节机械臂关节间的碰撞检测可以简化为多个两圆柱状物体间的最短距离求解^[18]。当空间两圆柱状物体简化为两线段时，两圆柱状物体间的碰撞检测就转化为求两空间线段相互关系的空间解析几何问题。我们只要对可能出现碰撞的两个臂间进行检测，将两个臂抽象成连接两个关节点的线段，最后求取两线段间的最短距离，只要该距离大于两个圆柱半径的和说明无碰撞，否则认为出现碰撞。

检修工具包络后的等效图如图 3.3 所示。

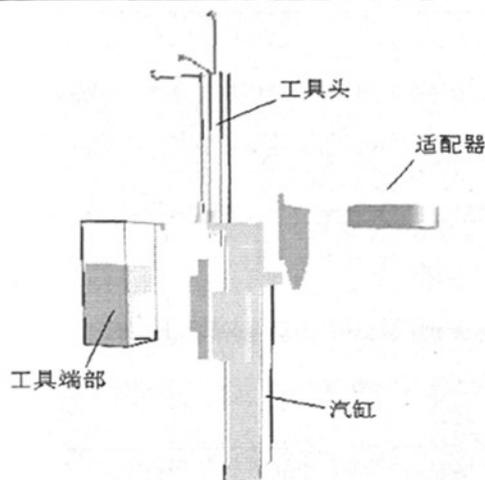


图 3.3 检修工具包络后的等效图

机械臂各关节及检修工具与脚部的碰撞检测可以通过求取线面交点，判断交点位置，若交点在线段外则认为无碰撞，否则认为发生碰撞。

由于各圆柱杆在该球形空间中运动时，只要保证圆柱两个端点不与内壁碰撞，则整个圆柱便不会与其碰撞，这样只要检测各个圆柱杆件端点即可。因此，各关节及检修工具与水室的碰撞检测可以通过求取圆柱两端点与球心距离，比较其与球半径大小，若小于则无碰撞，否则发生碰撞。

而与管板面和挡板面的碰撞检测直接可以通过判断包络各圆柱杆件线段端点的 z 、 x 方向深度值来确定。

3.3 动态碰撞检测的具体实现

碰撞检测算法优化的一个方面体现在搜索效率上，本文将机械臂进行简单有效的几何等效，针对检修工作中实际工作段的不同建立不同的信息存储结构体，直接排除了不需要判断碰撞的部位。对于单关节控制时，只需要实时更新运动关节的位姿信息，而不必对整个包围体进行重新构建。

需要判断机械臂各关节与环境内壁的碰撞，机械臂自身各关节间的碰撞，检修工具与环境内壁以及机械臂各关节间的碰撞。

机械臂检修工作等效图如图 3.4 所示。各个可能发生碰撞的部位用不同粗度的线段包络起来。

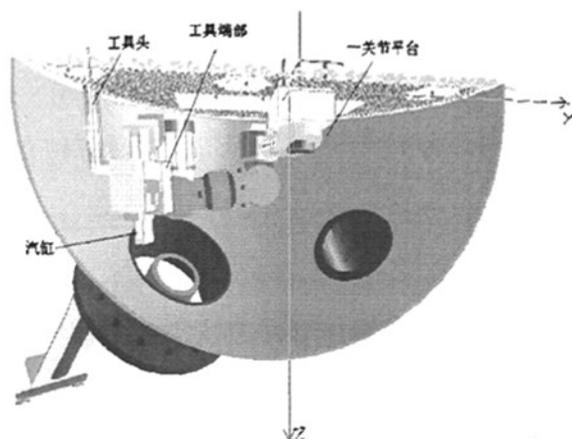


图 3.4 机械臂检修工作等效图

3.3.1 检修机械臂与环境内壁的碰撞检测

机械臂各关节可视为由圆柱杆件所组成，可将杆件简化为两端部分具有半径 r 的球，其中间部分简化为半径为 R 的圆柱体。由于各圆柱杆在该球形空间中运动时，只要保证圆柱两个端点不与内壁碰撞，则整个圆柱便不会与其碰撞，因此只要检测各个圆柱杆件端点即可。

(1) 关节及检修工具与挡板面、管板面的碰撞

机械臂在检修过程中，2 关节、3 关节、5 关节、6 关节及检修工具可能会与挡板面碰撞，5 关节、6 关节及检修工具可能会和管板面碰撞。对于相应关节、检修工具与挡板面、管板面的碰撞检测，只需判断他们外轮廓小圆柱圆心坐标与挡板面、管板面的距离即可。其中，将检修工具分为工具头、汽缸和工具端部三个部分。具体检测过程如下所述：

根据关节角度，利用机械臂的运动学正解，可以方便地求出关节及检修工具外轮廓小圆柱端部的圆心坐标，由于参考坐标系的 X 轴垂直于挡板面（向里），Z 轴垂直于管板面（正方向向下），求出 X、Z 方向的深度值，所有圆心坐标 X 和 Z 的最小值，当最小值小于一定值时，判断已经碰撞，否则无碰撞，如式 (3-1) 所示，其中 k 取 0.005 米（所留余量）。

$$\begin{cases} x - k > 0 \\ z - k > 0 \end{cases} \quad (3-1)$$

(2) 关节及检修工具与水室的碰撞

机械臂在检修过程中, 3 关节、5 关节、6 关节及检修工具可能会与水室碰撞, 对于其余关节由于并非机械臂端点, 无需检测。对于相应关节、检修工具与水室的碰撞, 判断它们外轮廓小圆柱两端圆心坐标与参考坐标系的圆心的距离即可。具体检测过程如下所述:

根据关节角度, 利用机械臂的运动学正解, 求出关节及检修工具外轮廓小圆柱端点的圆心坐标, 遍历搜索求出所有圆心坐标与参考坐标系的圆心距离的最小值, 当最小值大于一定值时, 判断已经碰撞, 否则无碰撞, 如式(3-2)所示。

$$\begin{cases} x_L^2 + y_L^2 + z_L^2 > \varepsilon^2 \\ x_N^2 + y_N^2 + z_N^2 > \varepsilon^2 \end{cases} \quad (3-2)$$

式(3-2)中 x_L 、 y_L 、 z_L 和 x_N 、 y_N 、 z_N 为各个小圆柱两端圆的中心坐标。 ε 的选择是困难的, 尤其是在关节型机械臂或多机器人情况下。如果 ε 取得很小, 碰撞检测效率不高, 如果 ε 取得很大, 就可能丢失碰撞。经可靠性测试, ε 取 0.919 米, 水室内壁的半径为 0.924 米, 保留 0.005 米的余量。

3.3.2 关节间、关节与检修工具间的碰撞

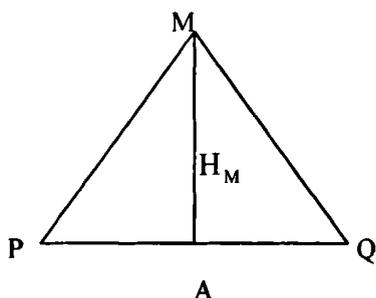
关节间, 关节及与检修工具间的碰撞检测可以简化为多个两圆柱状物体间的最短距离求解。对可能出现碰撞的两个臂间进行检测, 首先由运动学正解求出每个臂两端关节的中心点坐标, 然后将两个臂抽象成连接两点的线段, 最后求取两线段间的最短距离, 只要该距离大于两个圆柱半径之和就说明无碰撞, 否则认为出现碰撞。

线段间可能发生碰撞是: HG 和 FD, HG 和 DB, 以及检修工具外轮廓(工具头、汽缸、端部)与 HG、FD 间的碰撞。两线段之间的最短距离求解如下:

两线段的位置关系有四种情况: 平行、相交(含共线)、异面、不相交也不异面。我们把这四种位置关系归结为三种情况处理: 平行与非平行, 共面不平行也不相交。

①平行:

假设 MN, PQ 为待判定的两线段, 令 H_M 为 M 到 PQ 的高, L_{PQ} 、 L_{MP} 、 L_{MQ} 分别 PQ、MP、MQ 间的线段长, H_M 的求解如下:



先求 P 角度，再求出 PQ 长，得到向量 PQ，把它归一化，之后乘以 PA 长，得出 PA 向量，再由 P 点坐标和 PA 向量，求出 A 点坐标，最后求得 $H_M = PM \cdot \sin P$ 。若存在 (3-3) 式

$$H_M^2 + L_{PA}^2 < (\max(L_{MP}, L_{MQ}))^2 \quad (3-3)$$

则说明 M 在 PQ 上的投影在 PQ 延长线上，M 到 PQ 的最短距离 $L_M = \min(L_{MP}, L_{MQ})$ ，否则 $L_M = H_M$ 。对 N、P、Q 作相同的处理，最后可以求解出 MN、PQ 之间的最短距离 $L_{MNPQ} = \min(L_M, L_N, L_P, L_Q)$ 。

②非平行：

首先求出直线 MN、PQ 间公垂线的垂足坐标，方法如下：

- 1、MN 和 PQ 差乘，求出垂直 MN 和 PQ 的向量 V1；
- 2、求出直线 MN 和 PQ 的距离 L，如果 L=0 则转步骤 8；
- 3、将向量 V1 归一化；
- 4、用 L 乘以 1 式中的向量得出新的向量 V2；
- 5、用 V2 与 MN 向量差乘得到垂直 MN 和 V2 的向量，即平面法向量 T；
- 6、求出 PQ 与 V2 和 MN 组成的平面的交点（即公垂线与 PQ 的交点）；
- 7、通过 V2 与 6 式中的交点求出另一个交点（公垂线与 MN 的交点）；
- 8、若 L=0，说明两直线共面。若两线段相交说明最短距离为 0，否则转

③。

最短距离的求取分为三种情况：

(1)两个垂足都位于两线段内，最短距离为公垂线长；(2)两个垂足都在两线段外，最短距离为连接线段四个端点的线段中最小者；(3)一垂足在线段内、另一个垂足在另一线段外，若令 A 为在线段 MN 上的垂足，则最短距离

即 ΔAPQ 中 PQ 边上的高的长，若垂足在 PQ 上计算方法相同。

③ 共面不平行也不相交

把垂足不在线段上的两点向另一条线段引垂线，如果两垂足都不在线段内，则最短距离为两线段四个端点分别连线所得四个线段的最小者为其最短距离；否则最短距离为垂足在线段上对应高的距离；如果两垂足都在直线上，则最短距离为对应两个高的较小者。

线段间最短距离求取流程图如下：

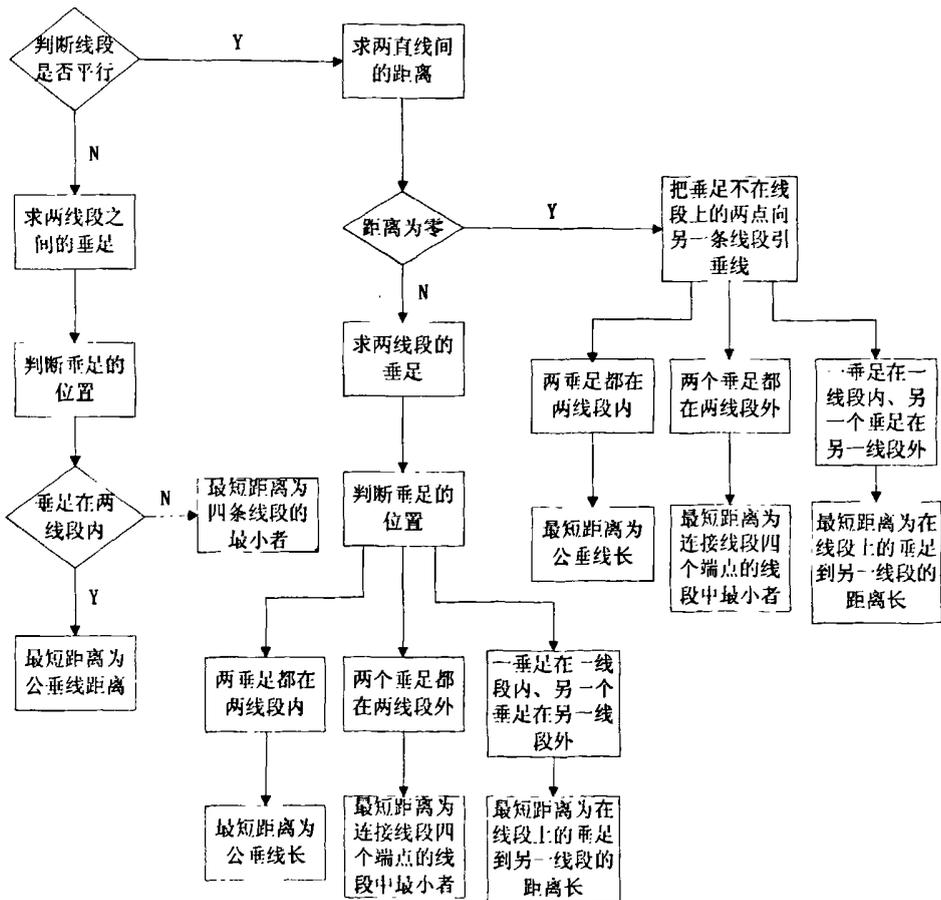


图 3.5 线段间最短距离求取流程图

FD 和 DB 碰撞检测用 5 关节的角度作限制，即将角度限位，限位角为 $0^\circ < \theta_5 < 90^\circ$ 且 $340^\circ < \theta_5 < 360^\circ$ (θ_5 表示第五关节的转角)，当第五关节的角度在该范围内时说明无碰撞，否则认为出现碰撞。

机械臂进行检修时还要考虑检修工具与脚部平台和第一关节平台（脚部平台下部分）的碰撞。该部分的碰撞检测可以通过检验检修工具外轮廓各条

线段所在直线与平台各面的交点是否在线段上并且在脚部棱线所包络的平面内来判断，若不在线段上则无碰撞，否则发生碰撞。

3.4 动态碰撞检测仿真及分析

本仿真实验在 256MB 内存 Intel(R) Celeron(R) 2.4GHz 计算机上用 VC++ 6.0 编程实现。在实模式下启动碰撞检测，在机械臂单关节控制和联动控制下进行机械臂的检修工作，将单关节角速度设定在 $4^\circ/s$ ，联动线速度设定在 $6.67mm/s$ 范围内。对于每种碰撞情况分别进行碰撞检测，在实验中，快速有效地实现了机械臂的避碰，并给出了碰撞点信息。 $\theta_1 \sim \theta_6$ 表示机械臂各个关节的角度， x, y, z 表示手末端点在世界坐标系下的坐标。表 3.1 为碰撞检测实验采样的数据，图 3.6 为与表 3.1 第 2 至 6 列相对应的机械臂位姿和碰撞点显示。

表3.1 碰撞检测实验算例

	(1)	(2)	(3)	(4)	(5)
θ_1	0.8610	326.7325	336.0925	9.3556	339.7325
θ_2	236.4960	228.7481	146.5881	222.3889	135.1481
θ_3	36.37742	18.8851	18.8851	18.7040	17.8451
θ_4	4.8460	359.2889	359.288	176.444	359.2889
θ_5	343.2676	346.2115	346.2115	346.1129	345.6915
θ_6	175.8435	69.1782	144.5782	194.9229	116.4982
x	0.2246	0.3339	0.0018	0.5916	0.0163
y	-0.5387	-0.3735	0.3602	0.13418	0.2829
z	0.2040	0.0310	0.0667	0.0482	0.0049
碰撞情况	5 关节与球壁碰撞	6 关节与挡板面碰撞	1、2 关节与 5、6 关节碰撞	工具头与挡板面碰撞	工具头与管板面碰撞

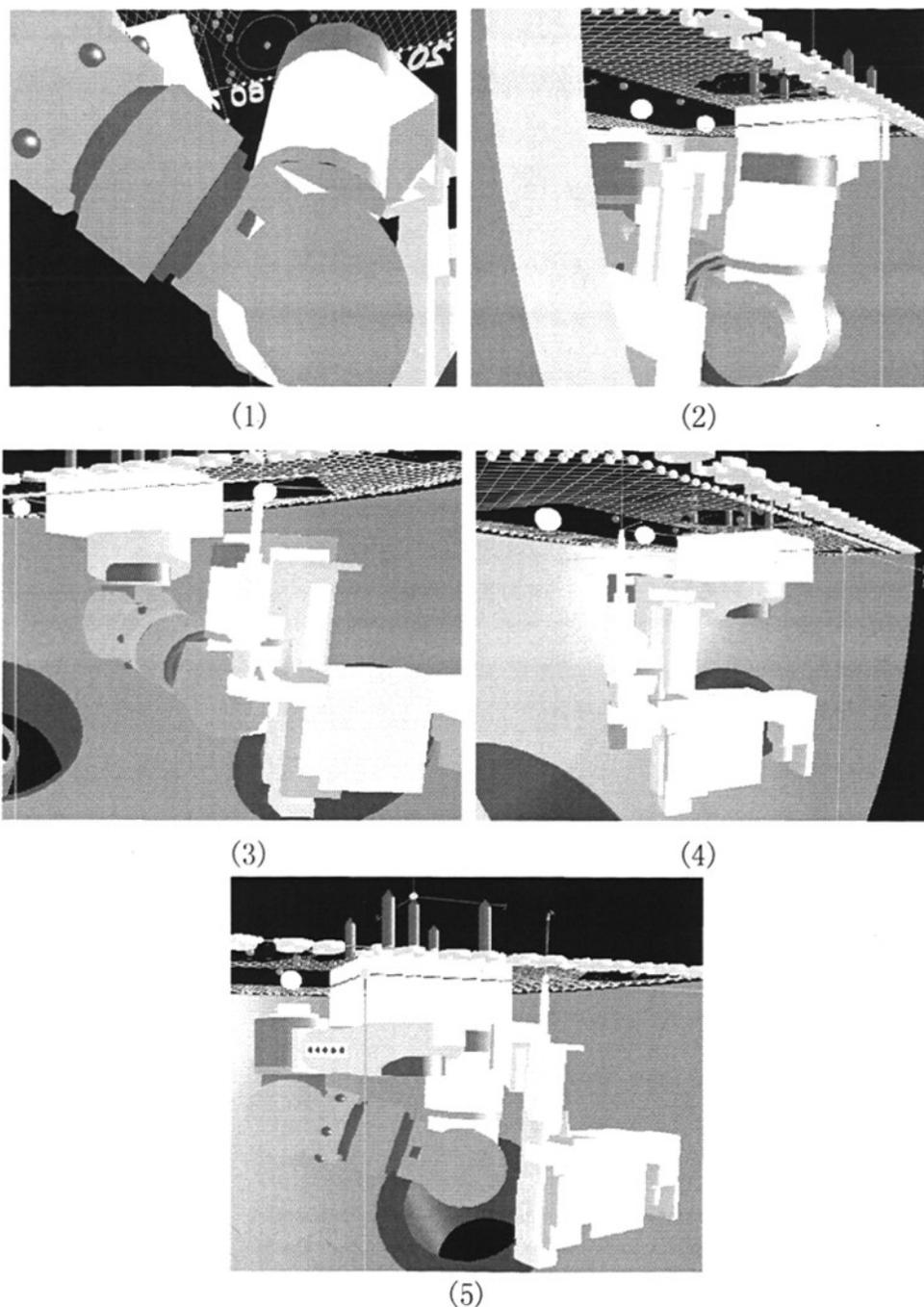


图 3.6 仿真实验实例图

3.5 本章小结

采用基于包围盒和等效空间圆柱体相结合的网络包围法，将空间局部域

进行有效分解，通过简单有效的几何等效，把机械臂的碰撞检测问题转化为空间解析几何问题，大大简化了问题的求解，由机械臂运动学的正解可以实时快速地解出各个碰撞检测点的空间坐标，建立有效的信息存储结构体，调用该碰撞检测算法即可以保证检修工作的无碰撞。该算法复杂度较低、实用性强，利于在线求解。实际工作验证了该算法的有效性、快速性，很好地解决了实时性和精确性的矛盾，为机械臂的安全检修提供了切实可靠的保障。

第 4 章 基于两阶段法的避障路径规划

为了解决超冗余度一类的多自由度机械臂路径规划问题^[53], Kavraki 等提出了路径规划的随机路标法^[54]。由于随机路标法具有易实现、通用性好的特点,特别是在解决多自由度机器人路径规划上所表现的有效性,近几年引起了研究人员的重视。在此基础上,一些学者又进一步完善和发展了随机路标法^[55],从而使该方法解决复杂环境下机器人路径规划的能力得到进一步增强。

对于规模较大的全局规划问题,为了避免搜索时间过长,一些学者想出各种办法来提高算法速度,一种比较适用的方法是通过将全局路径规划问题分解成为一系列局部路径规划问题来解决的,对于该机器人,因障碍复杂,当始末姿态相距较远时,无碰撞路径规划问题是一个较大规模的问题,采用路标法将使得该情况下规划的实时应用成为可能。

采用一般随机路标法得到一个表示机械臂位形空间结构信息的路标,计算机械臂在各个位形下与障碍物之间的最小距离,提前检验各个路标之间的连通关系得到一个路标连通图,并把此信息加到随机路标法所求的路标上,从而把路标转化为网格结构,当实际路径规划时,分别找到距出发点和目标点最近的两个路标点,然后利用启发搜索方法搜索路标连通图得到连接两个路标的路径,从而得到需要的路径。

随机路标法的基本原理与路标法相同,两者的关键区别在于,随机路标法采用随机采样方式在位形空间(configuration space) C 中构建路标,一般路标法中则采用显示计算的方法(这种方法要花费大量时间在位形空间中构建路标),因而可有效地解决多自由度机器人的路径规划问题。随机路标法的实施分为学习和查询两个阶段。

随机路标法的两个阶段:在学习阶段,对于给定的场景以概率的方式构建称为路标图的数据结构。该路标图是一个无向图 $R=(N, E)$,其中, N 表示无向图 R 的节点集合, N 中的节点对应处于自由位形空间区域中的机械臂位形; E 表示无向图 R 的边集合, E 中的边 (a,b) 对应一条连接位形 a 和 b 的可行路径。这些路径我们称之为局部路径,通过局部规划器来计算。局部路径在路标图中并不明确存储起来,因为重新计算是很廉价的。这节省了相当

多的空间，但是要求局部规划器的成败是确定的。此处假定学习阶段严格在查询阶段之前执行。然而，我们已经注意到学习阶段和查询阶段是可以交叉进行的。

在查询阶段，路标图用于解决在输入场景中的个别路径规划问题。给定一个初始位形 s 和目标位形 g ，首先尝试在 N 中将 s 和 g 分别与 \tilde{s} 和 \tilde{g} 相连，如果成功，之后查询 R ，在 E 中寻找连接 \tilde{s} 和 \tilde{g} 的序列。最后，通过重新计算相应的局部路径并将之相连，把该序列转换为对于机器人的可行路径。

下面，令 C 代表机器人的位形空间， C_f 代表机器人的自由空间。

4.1 学习阶段

学习阶段包括两个连续的步骤，我们称为构建阶段和扩展阶段。第一步的目的是想得到一个合理的连接图，用足够的点来覆盖相对一致的自由 C 空间，保证在该空间中的大多数“难点”区域至少要包含几个点。第二步的目的是根据某些启发式求值程序，尽可能提升图的连通性，选择位于 C 空间的难点区域的 R 的节点，通过产生额外的邻节点扩展图。因此，由最终路标图覆盖的 C_f 是不均匀的，取决于 C 空间的复杂性。

1. 构建阶段

通过构建路标以表示位形空间自由区域 C 的连接性和复杂性。以下简要说明构建阶段的步骤：

第一步： 随机方式在机器人位形空间采样。

第二步： 在此位形下对该点进行合理性检测。

此处所说的合理性检测主要包括机械臂的碰撞检测和路标点的距离检测。碰撞检测是为了保证所选的路标点在机械臂的可行位形空间内，可直接采用第三章介绍的动态碰撞检测方法。路标点的距离检测是为了保证选择的路标点比较均匀地分布在机械臂的可行区域内。本文采用如下方法判断：设定当前选择的路标点为第 N 个点，如果该点碰撞，则直接舍弃，继续选点，当该点位于可行位形空间内时，对该点与前 $N-1$ 个点作距离检测，如果所有的距离都小于某一个设定的阈值 D_{\max} ，则采用该点，进入下一个点的选择，否则舍弃该点继续循环选点。

第三步： 若合理，则将此位形保留起来作为路标中的节点。

重复上述过程预定次数后，便可获得一定数量的节点，即产生了机器人的位形空间 C 。

第四步：建立节点之间的连通关系，即判断两节点之间是否存在局部路径，使机器人从一个位形安全的移动到另一个位形空间，该步通过局部规划器来完成。

到此为止：一个隐式表示机械臂位形空间结构信息的路标构建完成，该路标以无向图的形式保存在文件中，供查询阶段使用。

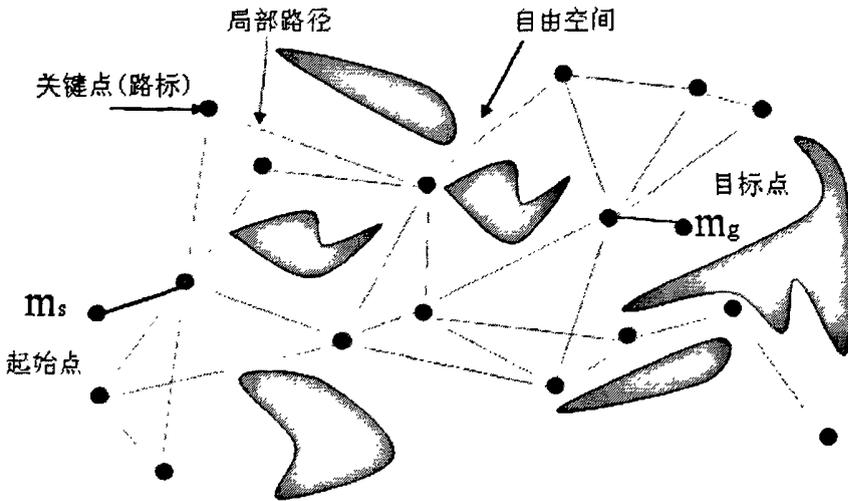


图 4.1 学习阶段生成的随机路标示意图

以下详细讲述构建阶段的过程。

初始时，图 $R = (N, E)$ 是空的。之后，重复随机产生自由位形并加入到 N 中。对于每一个这种新的节点 c ，从当前的 N 中选择一些节点，用局部规划器尝试将 c 与每一个选中的节点相连。一旦成功计算到连接 c 与所选节点 n 的可行路径就将边 (c,n) 加到 E 中。真实的路径并不需要存储到 E 中去。

下面讲述如何从 N 中选取一些节点与新节点 c 相连，以确保两节点之间一定存在可行路径。

①从 N 中选取候选的邻节点集合 N_c ，这些节点要求与 c 之间的距离在一定范围 D_{max} 内。②按照与 c 点之间的距离逐渐增加的顺序从 N_c 中选择节点。如果候选的邻节点与 c 不是图连通的，就试着将 c 与候选的邻节点相连。这样就不会建立循环，结果图为一森林，例如树的集合。对循环边而言，查询始终会失败，实际上不会花费很多的时间和空间来计算和存储这样的边。

然而，在一些情况下，循环的缺失将会导致在查询阶段构建出一些无用的长路径。这种弊端可通过在学习阶段针对路标图使用平滑处理技术，或者在查询阶段构造出特殊路径来避免。

一旦局部规划器在两个节点之间成功找到了路径，图 R 就会动态的更新。这样，就不需要用图搜索来确定从 N_C 中选择的节点是否与 c 相连。

为了使描述更为准确，令：

1) Δ 是 $C_f \times C_f \rightarrow (0,1)$ ，返回局部规划器是否能在两个位形点处创建出一条局部路径；2) D 是一个函数 $C \times C \rightarrow R^+ \cup \{0\}$ 距离函数，作为 C 空间中的虚拟距离函数。

构建阶段涉及到以下几个问题：(1)随机位形如何构建；(2)局部规划器如何创建；(3)候选邻接节点如何确定；(4)如何选择距离函数 D。

(1)随机位形的构建

R 应该包含较为一致的随机 C_f 采样。每一个这样的位形通过从在相应自由度值上使用均匀概率分布绘制每一个坐标来得到。得到的位形要进行合理性检测检测是否碰撞。若合理则加进 N 中，否则舍弃。

(2)局部规划器的创建

如果采用不确定的局部规划器，局部路径必须存储在路标图中。此时路标图需要更大的存储空间，但对于现在的计算机内存来说并不是主要问题。

如果选择一个搜索能力很强的局部搜索算法，则在机械手的可行空间将需要较少的路标点，这样形成的路标网络规模比较小，在查询阶段会大大减小搜索量，但这样的搜索算法将会比较耗时，如果在不同的路标之间需要两两进行联通性判断，将需要 $N*(N-1)/2$ 次局部搜索，这样会消耗非常多的时间。另外，对于搜索方法不具有确定性的情况，即不能保证每次都具有确定性成败结果的搜索方法，必须在学习阶段保留每次局部搜索得到的联通路径，才能保证查询阶段正常运行。如果路径具有较多的关键点，将会占用较大的存储空间。如果选择一个搜索能力较弱的算法，一般可以大大提高局部搜索速度，而且该类搜索方法一般具有确定性，可以不需要保留学习阶段搜索到的连通路径，只保留连通性的结果信息，查询阶段需要路径时可以通过该局部搜索方法重新计算，但该方法将需要路标点的密度较高，会增加查询阶段的搜索时间。实际应用一般根据具体情况，在两类方法中作出折中考虑。

局部规划器的选择也会影响到查询阶段。进行学习阶段的目的是能够立刻回答路径规划的查询阶段的路径搜索。将任一给定的初始和目标位形连接到路标图中，或者快速检测到不存在这样的连接是重要的。这要求路标图足够密，以使至少包含几个节点（至少一个），这些节点很容易与初始和目标位形相连。看上去更应该采用快速的局部规划器，建立广泛分布于自由 C 空间的大的路标图。另外如果局部规划器非常快，我们可以使用相同的规划器在查询阶段将初始位形和目标位形连接到路标图中。局部路径不需要存储，因为在查询阶段重新计算是廉价的。

文献[56]中提出了一种比较实用的局部路径规划方法，如图 4.2 所示，即直接选择一个中间点，并分别判断起始点与目标点到途经点的连线，如果两条连线都无碰撞则认为两点联通，并记下途经点坐标，供查询阶段使用。文献[56]中选择途经点的办法是在起始点与目标点间连一条直线，并以直线为长轴作椭圆，在椭圆线上寻找途径点，文献中通过不断增加椭圆短轴的办法来搜索，这样当短轴为零时，变成上述的直线检测，不断增长短轴时，经过途经点的折线路径也是不断增加，这样找到的第一个可行点一般能够保证路径尽可能短。该方法能够保证节省搜索时间，同时具有较强的搜索能力，而且搜索的路径较优，但是仅限于二维空间的情况，向高维空间推广比较复杂。

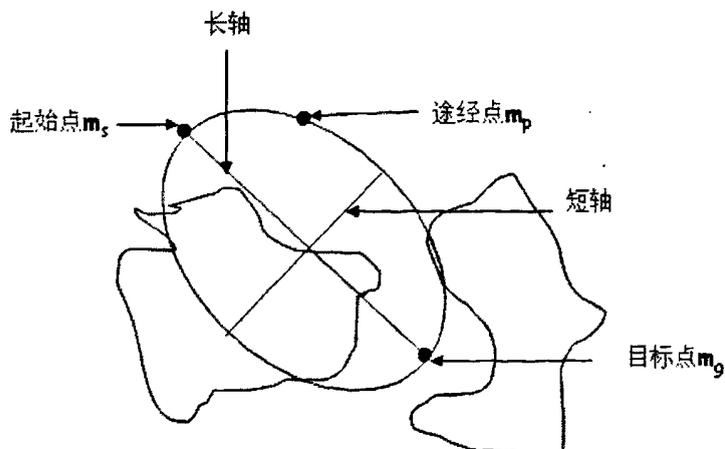


图 4.2 局部路径规划示意图

下面介绍一种适用于所有通用机器人的局部规划器。在位形空间中用直线段将两个给定的位形连接起来，检测这些线段是否碰撞，是否满足关节限制（若有）。实验证明保留在关节限制中的直线段是简捷的。另外，碰撞检测

可以按照下面的步骤完成。首先,当机器人位于位形 c_i 时,将线段离散化为一系列的位形 c_1, c_2, \dots, c_m (为了具有一般性每一个由局部规划器产生的直线段都进行这样的处理),这样对于每一个连续的位形 (c_i, c_{i+1}) 没有在机器人上的点,当机器人在位形 c_{i+1} 处时,位于离其位置远于某一步长 eps 的地方(该步长 eps 为预先指定的正常数)。之后,对于每一个位形 c_i ,在机器人位于 c_i 处且以设定的步长 eps 增大时,用碰撞检测器测试其是否发生碰撞。如果这 m 个位形都没有发生碰撞,就认为该路径是无碰撞的。由于 eps 是一常量,机器人本体按照步长 eps 增长的计算只做一次。

下面以平面关节机器人为例讲述局部规划器的使用

令 a 和 b 为任意给定的两个位形,我们想用局部规划器把它们连接起来。按照下面的方法构建了局部规划器。立刻将所有关节的相对速度进行奇此幂转换,例如进行 $J_{2,i+1}$ 次,沿着工作空间的直线段,把它们在位形 a 和位形 b 处的位置连接起来。在运动中,规划器用该关节相对于关节 $J_{2,i-1}$ 和 $J_{2,i+1}$ 简单的逆运动学方程调整每一个其它 $J_{2,i}$ 个关节的位置。这样关节 $J_{2,i}$ 随着 $J_{2,i+1}$ 运动。如果 q 是偶数, J_q 的位置不由上面的规律决定,通过立即旋转关节 J_q 相对于自身线速度的角速度来计算。当使用我们的特殊技术的时候,必须保证每一个这样关节的坐标要在关节的限制范围内。这样,如果发生碰撞或者关节移动到其范围之外,或者某一关节 $J_{2,i}$ 跟踪不到关节 $J_{2,i+1}$ 的运动,该运动就停止。我们注意到,当上面的运动不能将 a 和 b 连接起来的时候,然而机器人可以运动到离 b 非常近的 b' 处。在 C 空间中尝试将 b 和 b' 用直线段连接起来获得很好的效果,仅当该步失败的时候说明局部规划器在连接 a 和 b 失败。我们将上面的规划器称之为特定局部规划器。

机器人沿着特定局部规划器计算出的位形 a, b 之间的路径运动扫过的工作空间区域明显比用一般规划器的小。这样,由特定规划器产生的局部路径比用一般规划器产生的局部路径更趋向于无碰撞。而且,对于给定步长,碰撞检测更容易实现,局部路径的离散化产生更少的位形。另外,特定规划器虽然非常快,但是要比一般规划器慢。事实上,要求运动学逆运算沿着路径的位形坐标。然而,对于多关节的平面关节机器人,运用特定规划器的效果要更好。

(3) 临节点的选取

另外一个要完成的任务是 N_c 序列和候选临节点 c 的选取。调用局部规划器将 c 与 N_c 中的节点相连，这些调用的累积代价将影响学习阶段的执行时间。

我们通过仅提交相对距离（参看距离函数 D ）比某一常数门限 D_{\max} 小的位形对来避免调用可能失败的局部规划器。这样，我们定义如下：

$$N_c = \{\tilde{C} \in N \mid D(C, \tilde{C}) \leq D_{\max}\} \quad (4-2)$$

另外，根据上面给出的方法，我们尝试按照与 c 的距离逐渐增加的顺序来连接 c 与 N_c 中的所有节点，但是忽略那些与此时 c 相连的相同连接成分中的节点。按这样的顺序考虑此时的 N_c ，我们期望将快速连接 c 与其它位形的可能性最大化，这样就可以减少局部规划器的调用次数（因为每一次成功的连接都将导致把两个连接的成分融合为一个）。

我们发现通过某一常数 maxneighbors （特定为 30）限制 N_c 序列的大小很有用。额外的标准保证在最坏的情况下，每一次构建阶段算法主要循环的每一次迭代时间与当前路标图的大小是无关的。这样，调用局部算法的次数与构建的图的大小成线性关系。

(4) 距离函数 D 的定义

函数 D 用于针对每一个新节点 c 的候选临节点的 N_c 序列集合的构建和分类。对于每一个位形对 (c, n) ， $D(c, n)$ 反映局部规划器计算这些位形间可行路径失败的几率。一种可能性是定义 $D(c, n)$ 为当机器人沿着由局部规划器计算的在无障碍区域的 c 和 n 之间运动时，机器人扫过的工作空间区域的一种量度（面积或体积）。这样每一个局部规划器会自动推导出它自有的距离函数。由于准确扫描面积或者体积比较费时，严格但廉价的扫描区域量度给出较好的实际结果。例如，当上面描述的一般的局部规划器用于连接 c 和 n ， $D(c, n)$ 描述如下：

$$D(c, n) = \max_{x \in C} \|x(n) - x(c)\| \quad (4-3)$$

x 表示机器人上的点， $x(c)$ 为当机器人在位形 c 时在工作空间中的位置， $\|x(n) - x(c)\|$ 为 $x(c)$ 和 $x(n)$ 之间的欧几里德距离。

2. 扩展阶段

如果在构建阶段产生的节点数目足够多， N 的集合就给出自由 C 空间的

较为均匀的覆盖。在障碍少的环境中， R 的连接较好。但是障碍较多的环境中，自由 C 空间实际上是连通的，因此不会得到有效的 C_f 连通。扩展阶段的目的是提升由构建阶段产生的图 R 的连通性。特别地，若在 C_f 不存在的区域图是非连通的，这些区域与一些窄道相对应，这样就产生了自由 C 空间的难点区域。基于扩展阶段的想法是从很可能位于这样区域的 N 中选择一些节点，对其进行扩展。通过扩展位形 c ，我们预在 c 的临节点中选择一个新的位形添加到 N 中，按照构建步骤中的相同方法，把它与其它 N 的节点相连。这样，扩展阶段增加了被认为是难点区域的 C_f 中路标图的位形。由于图 R 的成分间隙经常位于这样的区域中， R 的连通性就可能增加。

针对扩展阶段我们提出了下面的概率方案。对于 N 中的每一个节点 c ，我们辅以一个正的位权 $w(c)$ 作为 c 附近难点区域启发式度量。这样，一旦 c 被认为位于难点区域里的时候， $w(c)$ 会很大。我们将 w 规范化，对于 N 中的所有节点全部权重加在一起。之后，依照概率从 N 中重复选择节点 c ： $w(c)=c$ 被选中的概率。扩展这个节点。

有几种方法来定义启发式的权重 $w(c)$ 。一种是计算 N 中距离 c 在给定距离范围内节点的数目。如果这个数目很小，障碍区域可能占据了 c 的临节点的大部分区域。这说明 $w(c)$ 可以定义为与离 c 在某一距离范围内的节点的数目成反比。另一种方法是观察从 c 到最近的不包含 c 的连接成分的距离 d_c 。如果该距离很小，那么 c 位于两个成分连接失败的区域当中，这说明该区域很可能是个难点区域（也可能已经发生碰撞）。该法可以定义 $w(c)$ 与 d_c 成反比。

若不使用 R 来定义难点区域，我们可以根据局部规划器的行为来定义 $w(c)$ 。例如，如果局部规划器在连接 c 与其它节点的时候经常失败，这也说明 c 位于难点区域当中。该使用哪一个启发式函数在某种程度上取决于周围环境。

(1)在构建阶段结束的时候，对于每一个节点 c ，计算失败的几率 $r_f(c)$ ：

$$r_f(c) = \frac{f(c)}{n(c) + 1} \quad (4.4)$$

$n(c)$ 为局部规划器尝试将 c 与其它节点相连的总的次数， $f(c)$ 为失败次数。

(2)在扩展阶段对于每一个 N 中的节点 c ，计算与失败率成正比的 $w(c)$ ，并进行适当的标度，这样把所有的权重加到一起。

$$w(c) = \frac{r_f(c)}{\sum r_f(a)}, a \in N \quad (4-5)$$

为了扩展一个节点 c ，计算一个始于 c 的短的随机回弹运动。对于完整的机器人，随机回弹运动包含重复随机选择 C 空间的运动方向，并沿着该方向运动直到与障碍发生碰撞。当碰撞发生的时候，又产生了一个新的随机方向。由随机弹跳运动得到的最终位形 n 和边 (c,n) 插入到 R 中。再有，计算的 c 和 n 之间的路径显示存储，因为该数据是使用不确定的技术得到的。我们也记录 n 属于像 c 一样的相同连接成分这个事实。之后我们尝试用与构建阶段使用相同的方式将 n 与网络中的其它的连接成分相连。这样，扩展阶段不会在 R 中建立新的成分。

在扩展阶段开始时，权重 $w(c)$ 只计算一次，当有新的节点添加到 R 中的时候不用进行修正。一旦扩展阶段结束，所有保留的 R 的小元素都被舍弃掉。此处若一个成分的节点数目比 N 中节点总数的某一百分比少的时候(0.01%)，就认为是小的成分。在图 R 舍弃了小的成分之后表示路标图可以用于查询阶段了。它可能包含一个或者几个成分。

令 T_L 为分配给学习阶段的时间。适当的 T_L 值的范围取决于环境，该值要对于每一个环境通过试验来确定。若 T_C 为花在构建阶段的时间， T_E 为花在扩展阶段的时间，我们发现这两个时间的比率为 2:1，如 $T_C = \frac{2}{3}T_L, T_E = \frac{1}{3}T_L$ ，在大范围问题上可以得到较好的结果。

4.2 查询阶段

在开始位形 s 和目标位形 g 的情况下，用学习阶段构建的路标图寻找一条连接 s 和 g 的两个节点的路径。假定当前的自由 C 空间是连通的，路标图包含单一的连接成分 R 。

对应图 4.1 所示的路标，采用一定算法搜索到的一条位形空间路径如图 4.3 所示。由图可见：① 从起始位形节点 s 到目标位形节点 g 之间存在若干不同的路径，当采用不同的搜索方法时，就可能搜索到不同的路径；② 搜索到的路径通常不是起始位形节点 s 到目标位形节点 g 的捷径(图中由实线连接

的路径), 一般可经过平滑处理而获得捷径(图中由虚线连接的路径)。

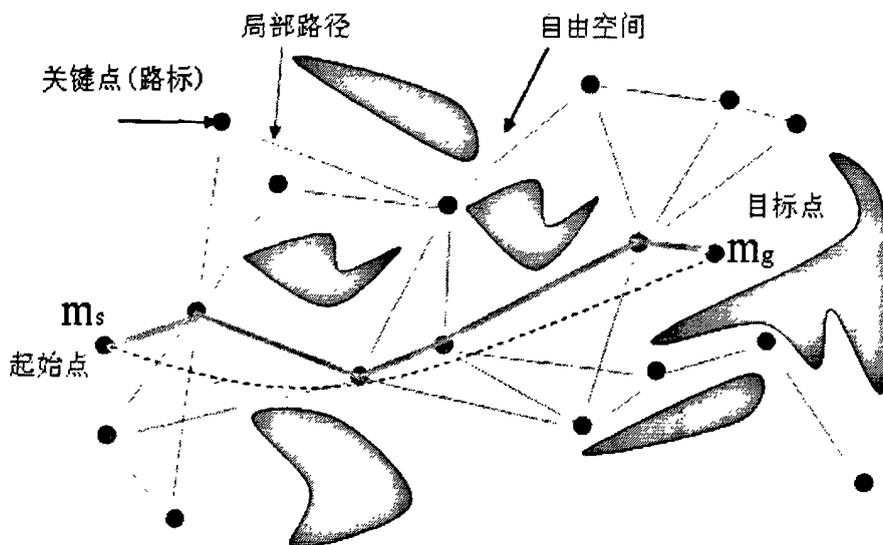


图 4.3 查询阶段搜索到的路径示意图

第一步: 采用位形空间直线规划器, 把 s 和 g 分别连接到 N 中离其距离最近的两个节点 \tilde{s} 和 \tilde{g} , 形成连接 s 和 \tilde{s} , 以及 g 和 \tilde{g} 之间的可行路径 P_s 和 P_g 。若该步失败则转到步骤三。

第二步: 若第一步成功, 在 C_f 空间中采用 A^* 搜索方法 (或者深度优先搜索方法等) 搜索出一条连接 \tilde{s} 和 \tilde{g} 的路径 \tilde{P} 。若搜索成功则将上述获得的所有局部路径相连得到一条连接起始位形 s 和目标位形 g 的路径 P 。该路径可以通过使用路径平滑技术来加以改进。若该步失败, 则转步骤三。

第三步: 若前两步有一者失败, 则说明局部路径搜索失败。返回第一步, 将 s 和 g 与 N 中除 \tilde{s} 和 \tilde{g} 外离其距离次最近的两个节点 \tilde{s}' 和 \tilde{g}' 相连, 继续搜索。若与 s 和 g 的路标点不在 D_{\max} 范围之内, 则说明查询失败。

查询阶段的主要问题是计算 P_s 和 P_g 。期望查询能立即应答, 所以该处没有采用耗时的算法。将 s 连接到 R 的策略是按增加与 s 距离的顺序 (根据 D) 来考虑 R 中的节点, 用局部规划器把 s 与其它节点相连, 直到有连接成功为止。我们忽略了离 s 的距离大于 D_{\max} 的节点, 因为此时局部规划器成功的可能性非常低。若所有的连接都失败了, 我们执行一个或更多的 4.1 扩展阶段中描述的随机回弹运动。我们现在尝试用局部规划器把它连接到 R 中, 而不是在每一随机回弹运动结束时把节点添加到路标图中。只要 s 成功连接

到 R 中，我们就应用相同的程序将 g 连接到 R 中去。

从 P 中节点序列产生的机器人路径减少了 P 的临节点的级联。这些路径中的一部分已经在学习阶段通过随机回弹运动产生了，并存储在 R 的相关边中。与在学习阶段通过局部规划器找到的连接相对应的路径重新进行了计算。局部规划器是确定的，每次调用输入相同的位形时产生相同的路径。当局部规划器具有当检测到碰撞就舍弃的特点的时候，不必沿着重新计算的局部路径进行碰撞检测。如果当检测到碰撞的时候，局部规划器执行某一特定的动作，碰撞检测就需要沿着重新计算的路径进行，以保证在检测到碰撞之后可以重复相同的动作。

然而，一般来说，路标图可能包含几个连接了的成分 $R_i, i=1 \cdots n$ 。这通常是当自由空间本身不是连通的时候。也可能发生在自由空间是连通的时候，如路标图不是足够的密。如果路标图包含几个成分，我们尝试将 s 和 g 与相同成分中的两个节点相连，从离 s 和 g 最近的成分开始。若 s 和 g 与某一成分 R_i 成功连接，在单成分情况下就构建了一条路径。如果不能将起始点和目标点与相同的路标图成分成功连接，该方法就失败了。由于在大多数例子中，路标图包含较少的成分，失败很快会探测到。

如果路径查询频繁失败，说明路标图没有充分实现自由 C 空间的连通。这样，要在学习阶段花费更多的时间，例如 T_L 应该增加。然而，从开始构建新的路标图是不必要的。我们可以通过重用构建阶段或者扩展阶段的算法简单地扩展当前的路标图，从当前的路标图开始，交叉地进行学习和查询。

4.3 路径平滑技术

Bezier 曲线具有良好的几何性质，能简洁、完美地描述和表达自由曲线曲面，在 CAD/CAM 技术中得到广泛的应用。Bezier 曲线的最大优点是：控制点如果构成凸多边形，即特征多边形是凸时，Bezier 曲线也是凸的。所以要将曲线升高或降低，只要将一个控制点升高或降低即可，计算非常方便。

Bezier 曲线是由一组折线集或称之为 Bezier 特征多边形来定义的。曲线的起点和终点分别与该多边形的起点、终点重合，且多边形的第一条边和最后一条边表示了曲线在起点和终点处的切矢量方向。曲线的形状趋于特征多边形的形状。当给定空间 $n+1$ 个点的位置矢量 P_i ，则 Bezier 曲线上各点的

坐标插值公式是：

$$P(t) = \sum_{i=0}^1 P_i J_{i,n}(t), 0 \leq t \leq 1 \quad (4-6)$$

P_i 构成该曲线的特征多边形, $J_{i,n}(t)$ 是 Bernstein 基函数, 也是曲线上各点位置矢量的调和函数。

一般称折线 $P_0P_1P_2 \dots P_n$ 为曲线 $P(t)$ 的控制多边形, 称点 $P_0, P_1, P_2, \dots, P_n$ 为 $P(t)$ 的控制顶点。在空间曲线的情况下, 曲线 $P(t) = (x(t), y(t), z(t))$ 和控制顶点 $P_i = (X_i, Y_i, Z_i)$ 的关系分别用分量写出即为：

$$\begin{cases} X(t) = \sum_{i=0}^1 X_i J_{i,n}(t) \\ Y(t) = \sum_{i=0}^1 Y_i J_{i,n}(t) \\ Z(t) = \sum_{i=0}^1 Z_i J_{i,n}(t) \end{cases} \quad (4-7)$$

当 t 在区间 $[0, 1]$ 上变动时, 就产生了 Bezier 曲线。给定空间四个矢点 P_0, P_1, P_2, P_3 , 可导出平面四阶三次 Bezier 曲线为：

$$P(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}, 0 \leq t \leq 1 \quad (4-8)$$

文献[57]中采用 de Casteljau 算法得到 Bezier 曲线的递推计算公式：

$$P_i^k = \begin{cases} P_i & k = 0 \\ (1-t)P_i^{k-1} + tP_{i+1}^{k-1} & k = 1, 2, \dots, n, i = 0, 1, \dots, n-k \end{cases} \quad (4-9)$$

(4-9)式中 $P_i^0 = P_i$ 是定义 Bezier 曲线的控制点, P_i^n 为曲线 $P(t)$ 上具有参数 t 的点。

当 $t=1/2$ 时, 由式(4-9)可知, 求取 P_i^k 只需要进行加法和除 2 运算, 计算方便且速度较快。所以通常情况下取 $t=1/2$ 最方便, 即每次求得 Bezier 曲线的中点。

这一算法可通过简单的几何作图来实现。如图 4.5 所示, 有 0, 1, 2, 3 四个控制点, 计算每条线的中点 01, 12, 23, 再得中点 012, 123, 最后

得到的中点 0123 即在曲线上。四个控制点 0, 01, 012, 0123 定义了左边的曲线,四个控制点 0123, 123, 23, 3 定义了右边的曲线,重复对左右的两条曲线再进行定比分割,多次进行定比分割后,就可以用直线段代替曲线段。

从(4-9)式可知,计算 Bezier 曲线上的点,可用 Bezier 曲线方程。已知控制点的坐标,只要 t 取 $0\sim 1$ 之间不同的值,就可以求出 Bezier 曲线上的很多点,然后将这些点用小直线段、折线相连,Bezier 曲线也就绘制出来了。

进行定比分割到什么程度可以用直线段代替曲线。如果曲线上的点到其两端点的连线的距离小于指定的很小正数 ε 时。则在绘制时可用直线段代替曲线段,否则对其控制多边形再进行定比分割。在适当次数的分割后,分得的每一段曲线都能由其两端点的连线所代替。

计算 Bezier 曲线 $P(t)$ 到其两端点 P_0P_3 的距离 $d(P(t), P_0P_3)$ 很麻烦,但是由凸包性可知:

$$d(P(t), P_0P_3) \leq \max(d(P_1, P_0P_3), d(P_2, P_0P_3)) \quad (4-10)$$

其中, $d(P_i, P_0P_3)$ 表示线段 P_i 到 P_0P_3 的距离,计算点到直线的距离要相对容易,故在一定的误差范围内,可用右端代替左端^[57]。

如图 4.4,只要让 d_1 和 d_2 中最大值小于一个指定的很小正数 ε 时,就可以用两端点的直线代替曲线段。这种用分割 Bezier 曲线控制多边形的方法为绘制离散化的 Bezier 曲线提供了方便。

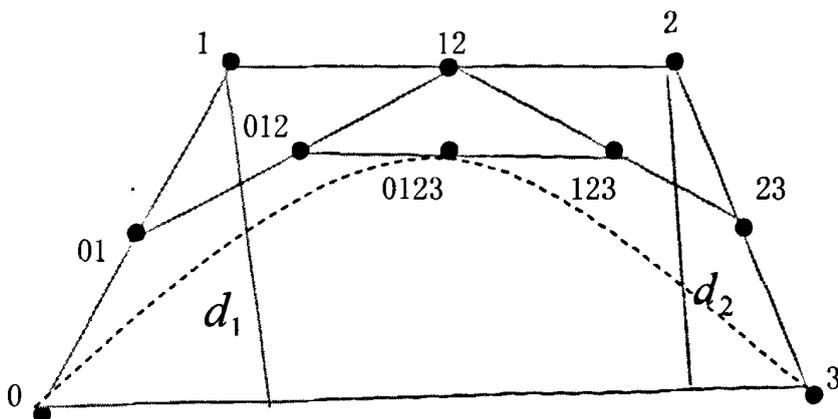


图 4.4 de Casteljau 算法几何示意图

4.4 软件设计及实现

4.4.1 软件设计流程

基于两阶段法的避障路径规划方法相对比较复杂，根据前面给出的规划方法在 VC++ 平台下，编制了相应的程序，下面给出软件设计中的几个简单流程。

以下为采用两阶段法避障路径规划方法中学习阶段构建步骤算法描述流程图，如图 4.5 所示：

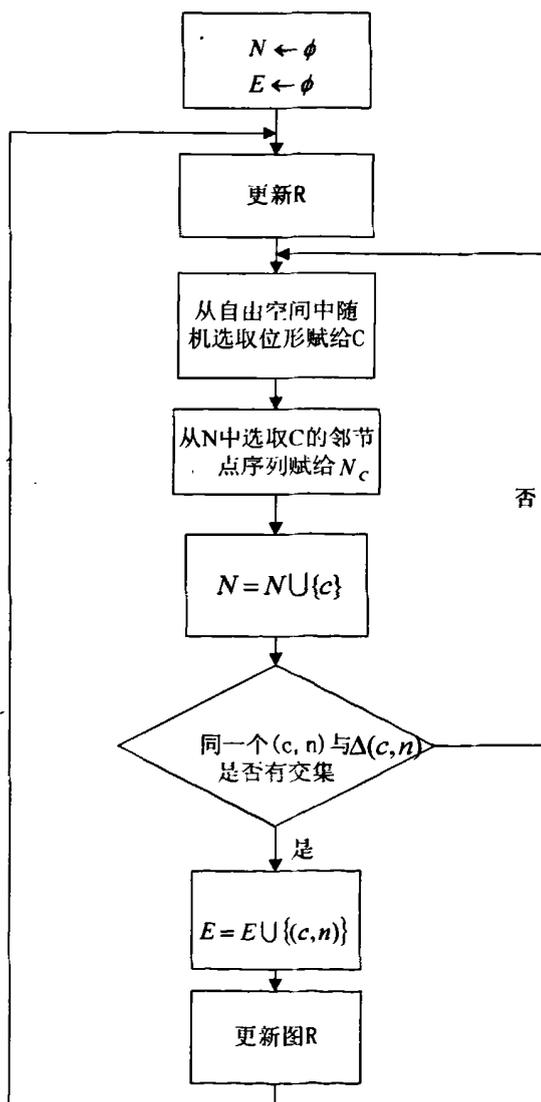


图 4.5 构建阶段算法流程图

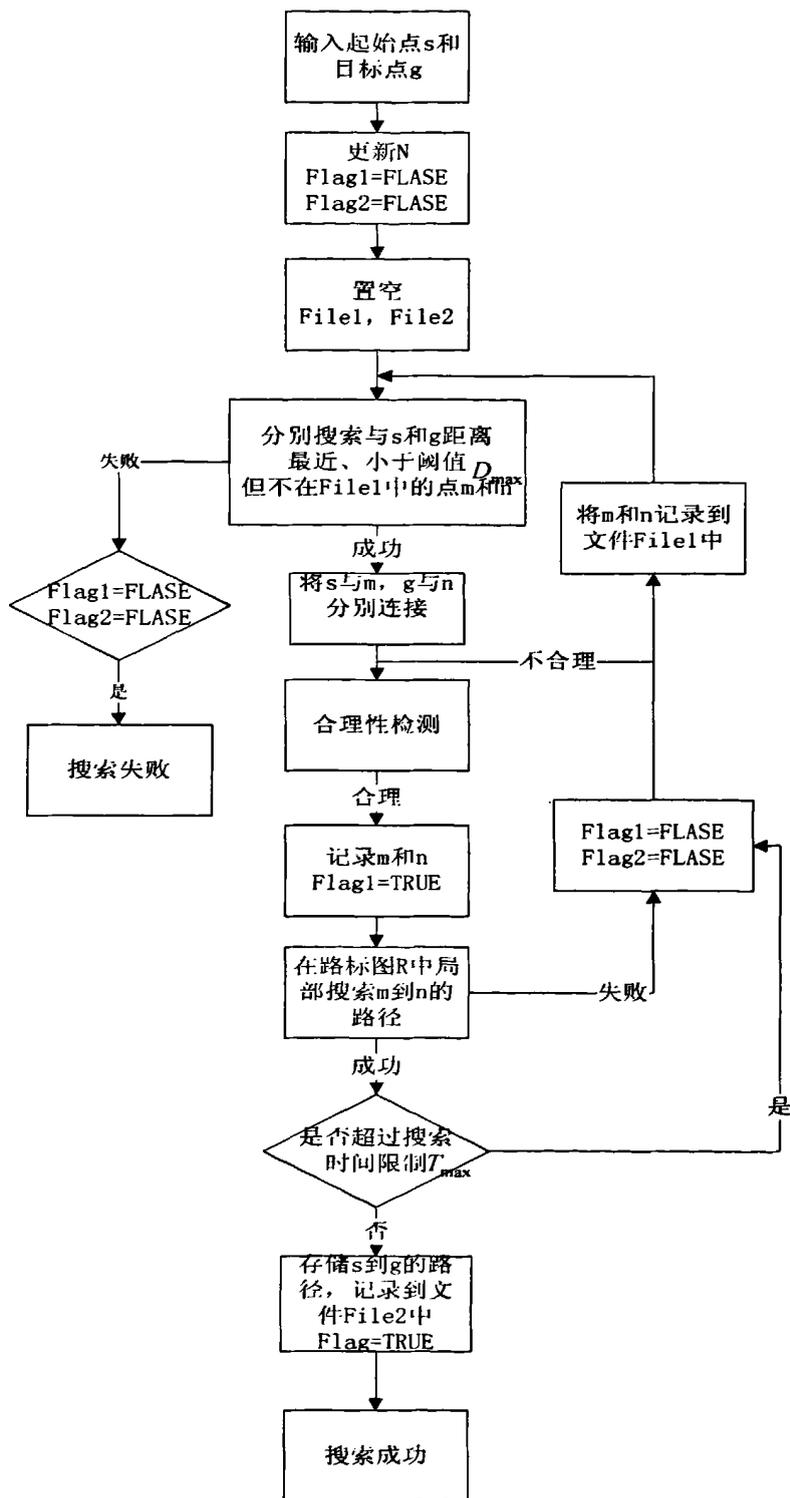


图 4.6 查询阶段算法流程图

图 4.6 为采用两阶段避障路径规划算法中查询阶段流程图，算法中引入了查询时间限制 T_{\max} ，对于实时性要求不高的场合可以放宽或者去除该限制。为了便于表示，流程图中的 m 、 n 分别为 \tilde{s} 和 \tilde{g} 。

4.4.2 仿真结果及分析

该仿真用到的数学模型是在第二章中所建立的机械臂的运动模型，碰撞检测方法为第三章所述的动态碰撞检测算法，采用基于两阶段法的避障路径规划方法实施路径规划。本章及第 5 章所做仿真分析，机械臂一至六关节长度分别为 $L_1=0.2295m$ ， $L_2=0.2440m$ ， $L_3=0.2370m$ ， $L_4=0.1130m$ ， $L_5=0.1450m$ ， $L_6=0.7500m$ 。

本仿真实验在 256MB 内存 Intel(R) Celeron(R) 2.4GHz 计算机上用 VC++ 6.0 编程实现。在虚模式下启动碰撞检测，在机械臂单关节控制和联动控制下控制机械臂抓顶进入检修状态，调整机械臂的位姿，使横滚角接近 180° ，俯仰角接近 90° ，即将机械臂工具端（手端）与管板面接近平行。运动时要将单关节角速度设定在 $4^\circ/s$ 内，联动线速度设定在 $6.67mm/s$ 内，路径规划运动角速度限定在 $1.2^\circ/s$ 范围内。

由于机械臂抓顶位置的不同，导致可达的检修区域有所不同，因此有必要确定其抓顶范围。现对机械臂抓顶的范围进行仿真分析，分别对管孔面的每个孔进行分析（以中间脚趾为准），推进装置的空间姿态角为 $(38.69452^\circ、50.16447^\circ)$ 。

抓顶完成后俯视图如图 4.7 所示。图中抓顶位置为第 9 行第 9 列。



图 4.7 抓顶完毕俯视图

表 4.1 列出了部分能够抓顶的位置。

表 4.1 部分能够抓顶的位置

抓顶位置	1轴(角度)	2轴(角度)	3轴(角度)	4轴(角度)	5轴(角度)	6轴(角度)	推进(米)
17行、9列	323.087	271.753	160.179	191.151	109.308	346.528	0.722
13行、9列	333.937	267.894	159.771	206.119	111.035	328.050	0.722
9行、9列	345.186	261.263	159.746	213.850	113.202	317.973	0.722
28行、8列	298.860	269.470	159.850	139.308	116.415	51.6618	0.722
24行、8列	305.869	273.716	159.765	155.994	110.587	29.2919	0.722
12行、8列	335.875	269.073	159.751	211.504	112.438	321.093	0.722
27行、7列	298.896	273.299	159.815	143.957	114.132	45.0501	0.722

13行、 7列	332.160	272.997	159.755	211.952	112.580	320.505	0.722
26行、 6列	299.079	276.859	159.798	149.649	112.149	37.4270	0.722
18行、 6列	317.457	279.658	159.863	196.648	109.531	339.852	0.722
23行、 5列	303.962	281.709	159.791	169.142	108.887	13.0931	0.722
19行、 5列	313.863	282.387	159.829	194.557	109.258	342.406	0.722
28行、 4列	292.600	279.083	159.781	143.226	114.394	46.0836	0.722
20行、 4列	310.139	284.881	159.892	191.702	109.059	345.878	0.722
25行、 3列	296.814	284.961	159.898	160.538	109.953	23.6373	0.722
19行、 3列	311.860	287.096	159.862	200.384	110.056	335.230	0.722
26行、 2列	293.319	286.173	159.780	155.825	110.641	29.5006	0.722
24行、 2列	297.750	288.036	159.788	169.412	108.866	12.7671	0.722
10行、 2列	349.513	364.961	152.009	235.804	112.850	290.641	0.652
25行、 1列	293.248	289.879	160.307	163.738	109.951	19.7435	0.727
15行、 1列	334.481	276.351	151.730	233.382	110.077	295.079	0.650
13行、 1列	340.426	273.123	151.730	235.703	112.028	291.214	0.650

通过对抓顶位置的所有尝试得出其抓顶位置如图 4.8 所示。

其中阴影部分为能抓顶区域，图中横坐标表示抓顶位置的列，纵坐标表示抓顶位置的行。

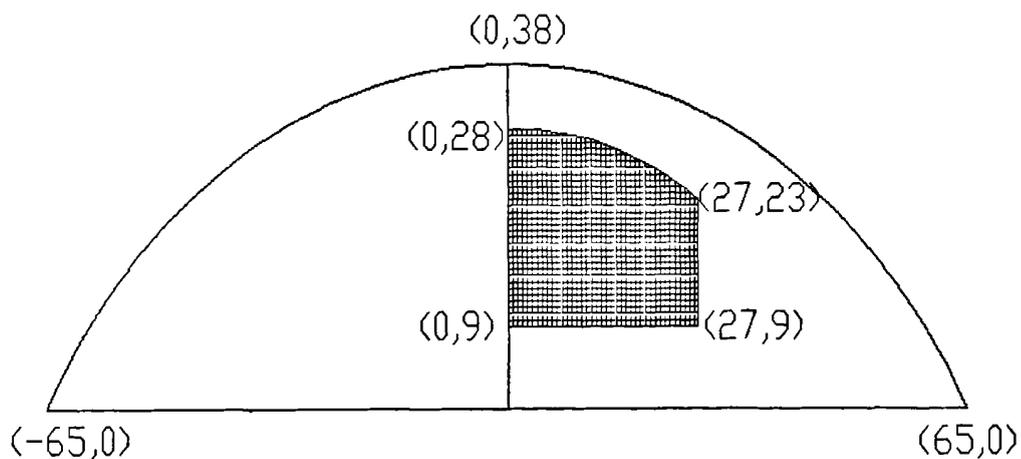


图 4.8 抓顶位置区域示意图

由于电机转速、工作环境及机械臂自身结构限制等原因导致右端抓顶范围受到限制，因此影响了机械臂的运动范围，也即影响了路径规划是采样空间对应的实际运行空间。

给定机械臂的初始点和目标点位姿，以下各图中分别以蓝色圆点表示起始点，红色圆点表示目标点。由第二章讲述的机械臂逆解方法解出机械臂的各关节角度如表 4.2 所示。

首先给出机械臂的起始点和目标点位姿，解算后得到机械臂的位形分别如图 4.9、图 4.11 所示。为了更为直观地观察规划的路径，给出了规划过程的一个中间位姿，如图 4.10 所示。用两阶段法对路径进行规划，得到的仿真曲线如图黄色线条所示。

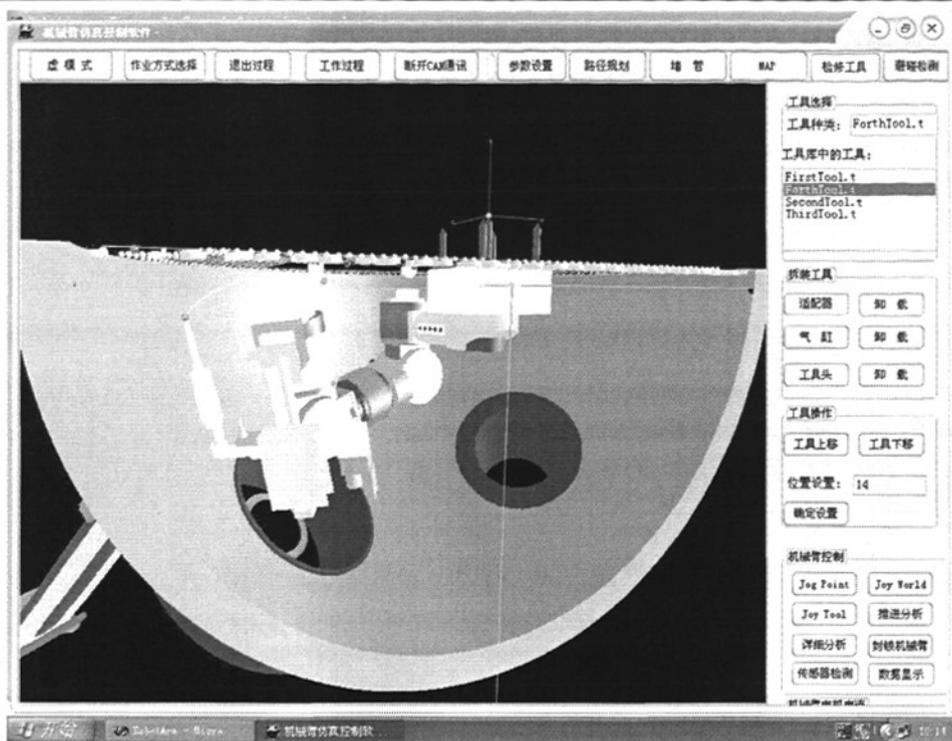


图 4.9 起始点位形图

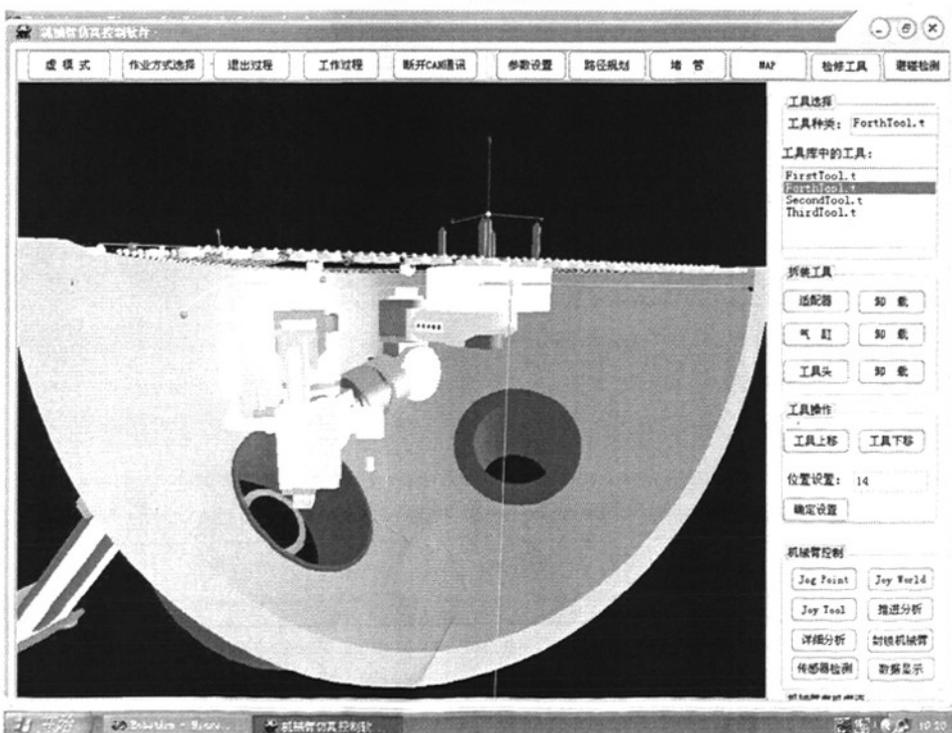


图 4.10 中间点位形图

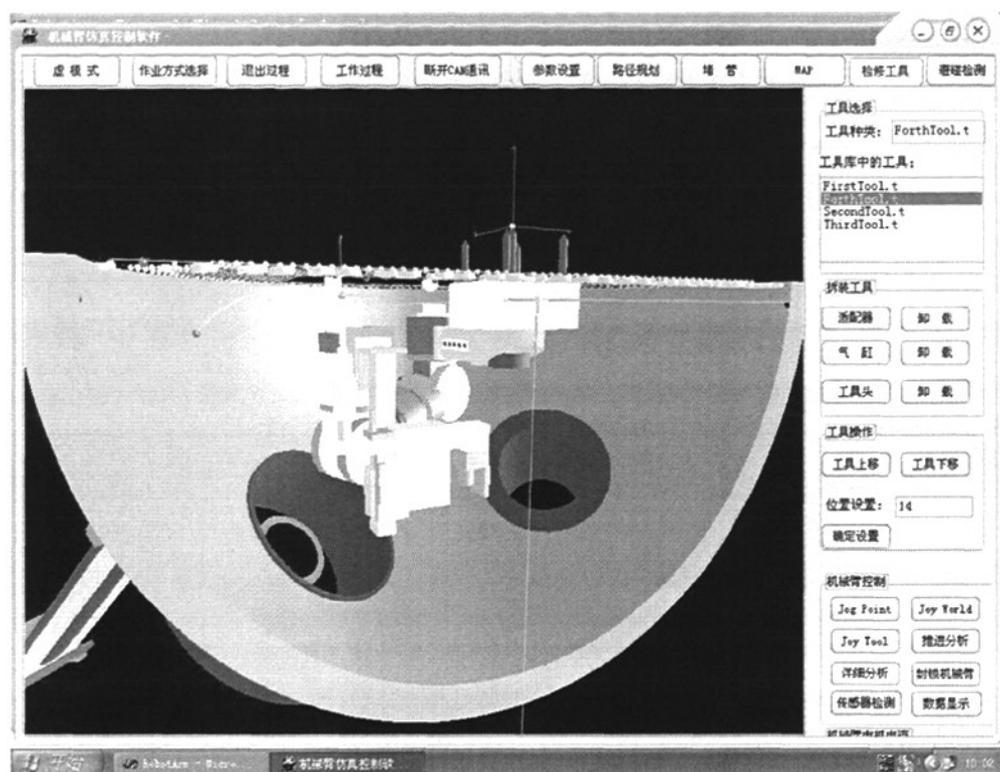


图 4.11 目标点位形图

表 4.2 中给出了起始点、中间点和目标点的位形数据，其中 x 、 y 、 z 为工具末端点坐标， α 、 β 、 γ 为工具末端（手端）姿态角， $\alpha_1 \sim \alpha_6$ 为一至六关节转角。

表 4.2 始末及采样点位形数据表

	起始点	中间点	目标点
x	0.266	0.2586	0.141
y	-0.591	-0.4920	-0.343
z	0.112	0.0435	0.043
α	185.047	180.027	180.031
β	100.986	89.9275	89.901
γ	37.929	46.3107	69.214
α_1	352.738	352.738	356.290

α_2	263.051	263.052	250.394
α_3	17.091	18.131	18.127
α_4	357.873	359.954	359.931
α_5	354.080	341.805	341.802
α_6	213.944	223.210	236.901

为了说明在两阶段法中在学习阶段进行扩展的必要，以下描述了应用第四章中提出的局部规划器、碰撞检测器和距离函数进行路径规划的仿真性能，对是否进行路径扩展进行了对比。规划器的参数如下：

T_C ：构建阶段的时间；

T_E ：扩展阶段的时间；

D_{max} ：局部规划器可能尝试连接的节点间的最大距离；

eps：在碰撞检测前用于离散化局部路径的常量；

T_{RB_expand} ：扩展阶段用于路径回弹操作持续的时间；

N_{RB_query} ：考虑连接初始和目标位形进行的回弹操作的最大数目；

T_{RB_query} ：在查询阶段用于路径回弹操作的持续时间。

后两个参数能够决定路径查询的时间。

我们首先手动输入一组位形，设定为测试序列。对于固定的 T_C 和 T_E ，我们分别建立起始于随机产生器产生的不同值的不同路标图。在此处的例子中，我们只是保留路标图的最大连接部分。之后对于每一路标图，我们尝试连接相同的位形，记录下规划器在给定的时间(2.5秒)下连接成功的几率。这种估计的成功率可能用于计算在测试序列中对任意两个位形查询成功的几率。特别地，结果不但反映了运行良好的情况，也包含了糟糕的情况。对于不同的 T_C 和 T_E 我们分别重复进行了相同的实验。对上面描述的其它参数，基于之前的实验结果，在整个试验中我们选择固定值。人工选择测试用的位形是很重要的。很明显，随机生成与学习阶段相似的位形点易于产生容易连接的路标图。人工处理允许我们选择那些感兴趣的位形，例如位于工作空间狭窄通道中的位形，而学习阶段的随机产生器不可能生成众多这样的位形。

表 4.3 给出在不同学习时间下的路标图序列中位形连接成功的几率。学习时间 T_L ，在第一列。它分为 T_C 和 T_E 两个部分，如 2 列和 3 列所示， $T_E = T_C / 2$ 。表格中的其它参数为： $D_{max} = 0.4$ ， $eps = 0.01$ ， $maxneighbors = 30$ ， $T_{RB_expand} = 0.03sec$ ， $T_{RB_query} = 0.5sec$ ， $N_{RB_query} = 45$ 。

表 4.3 的每一行我们分别生成了 10 个路标图，每一个有独立的学习时间。不同行的路标图分别计算，也就是说，在某些行中没有路标图进行重构。

表 4.3 中的第 4 列给出对于每一个不同学习时间下的路标图碰撞检测的平均数目。该数目可以认为是规划器计算相似性的估计。碰撞检测不但沿着机器人的路径进行，因为对于大多数的规划器，也要在试图猜测随机自由位形的时候进行。大多数这些随机的猜测的位形是不合理的，因为会与障碍或者自身发生碰撞。平均来说，会快速检测不合理的位形到并将大多数位形舍弃。第 5 列给出节点的平均数目，经过了 10 次运行，在学习阶段结束时的最大路标图构成。每一个路标图的最大连接用于查询处理。第 6 列给出了对于 10 个产生的路标图的相应位形，在 2.5 秒以内尝试连接成功的几率。从表 4.3 中可以看出，在学习时间超过 60 秒或更长的情况下，能成功地连接到产生的路标图上。

表 4.4 给出了不进行扩展成功与路标图成功的几率。与表 4.3 的相应的列相对应，表 4.4 给出了相同的学习时间的结果。我们同样对 10 个不同的路标图进行了实验。当学习时间更短的时候不同点更能显示出来。如果我们想尽快得到可行的路径，在学习阶段分配部分时间进行扩展要比完全花费在构建阶段效果要好。

表 4.3 含扩展阶段的不同学习时间内路标连接成功的几率

T_L	T_C	T_E	碰撞检测 平均数目	平均连接 节点	成功率
30.2	19.7	10.5	1289741	1882	68.1
40.2	26.5	13.7	1524102	2449	89.5
50.1	32.2	17.9	1748791	3010	96.4
60.3	39.2	21.1	1954712	3654	100.0
70.3	45.9	24.4	2215489	4233	100.0
80.2	51.1	29.1	245142	4841	100.0

表 4.4 无扩展阶段的不同学习时间内路标连接成功的几率

T_L	T_C	T_E	碰撞检测 平均数目	平均连接 节点	成功率
30.2	30.2	0.0	1139231	1878	57.8
40.2	40.2	0.0	1425664	2472	80.2
50.1	50.1	0.0	1614587	3041	88.5
60.3	60.3	0.0	1847912	3666	93.9
70.3	70.3	0.0	2102145	4257	100.0
80.2	80.2	0.0	2315478	4878	100.0

4.5 本章小结

本章采用基于两阶段法的随机路标法对机械臂进行避障前提下的路径规划。随机路标法的实施分为学习和查询两个阶段。学习阶段从自由空间中随机选择位形构建一个路标图，作为位形的集合。查询阶段使用学习阶段得到的路标图进行快速查询，得到可行路径。采用基于 Bezier 曲线的路径平滑技术对得到的路径进行平滑处理，可以得到平滑的运动轨迹。根据两阶段法编制相应软件，针对特定的抓顶位置进行了路径规划仿真，验证了算法的适用性。随着学习时间的延长规划成功的几率越大。通过仿真分析给出了学习阶段进行扩展的好处。

第 5 章 基于贪婪碰撞检测的启发式双向路径搜索

第 4 章讲述的基于两阶段法的避障路径规划方法在考虑学习时间时, 实时性不高, 为了提高路径规划的实时性, 本章提出了采用基于贪婪碰撞检测的启发式双向路径搜索对机械臂运动路径进行实时规划。

本章所讲的贪婪碰撞检测的启发式双向路径搜索策略有如下几个特点: 不需要预计算路标图, 即无需进行第 4 章 4.1 节中讲述的学习阶段, 这样节省了大量的预计算时间; 进行双向采样, 逐步增加以输入查询位形为根的两棵路标树来建立路标图, 两棵树的交替采样能保持两棵树的平衡; 采取贪婪碰撞检测, 延迟碰撞检测操作直到完全需要检测, 可以节省规划时间。

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, 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。

在综合分析路径规划研究的基础上, 采用基于 PRM(概率地图)的路径规划方法, 运用启发式搜索算法对机械臂运动路径进行实时规划。在搜索过程中, 以位姿点密度作为权重使路径向自由空间扩散, 避免过度采样。为提高路径规划速度, 采用贪婪碰撞检测策略, 可有效降低计算量。

BohlinR 等人提出位姿点间的碰撞检测可以延迟到找到最佳路径后再进行, 这样可以减少大部分无谓的计算^[59]。Sánchez G 等人通过试验发现, 如果 2 个位姿点均属于自由空间, 且两点间归一化距离小于 0.1 时, 该线段属于自由空间 F 的概率大于 90%。因此采用启发式双向路径搜索碰撞检测延迟算法, 找到最短路径后再对连接路径的各节点间线段进行碰撞检测, 如果发现碰撞, 则将该线段删除, 重新搜索最短路径, 直到找到无碰撞路径或无解返回^{[59][60]}。这一方法使路径规划计算更有效、实时。

为了实现实时快速的路径规划,必须解决采样方式、路径延伸方法、动态碰撞检测、局部路径安全性检验、桥接测试是否成功等问题。

5.1 桥接策略

文献[61]中讲述了桥接测试,参考其原理我们对其进行了部分改进。将起始位姿点 q_s 和目标位姿点 q_g 作为树根,向外延伸路径。为了叙述方便,这里用图 5.1 所示的二维位形空间来说明桥接策略。首先从基本位姿点开始,在其邻域内按照均匀分布随机采样,生成新的采样点,并进行碰撞检测,如果满足不碰撞条件,则将采样点作为位姿点加入到路径中。每延伸一步,都将生成一个新的位姿点,重复执行后形成路径树 T_s 和 T_g 。如果 2 个路径树出现连接(图 4.6 虚线部分),则停止规划。

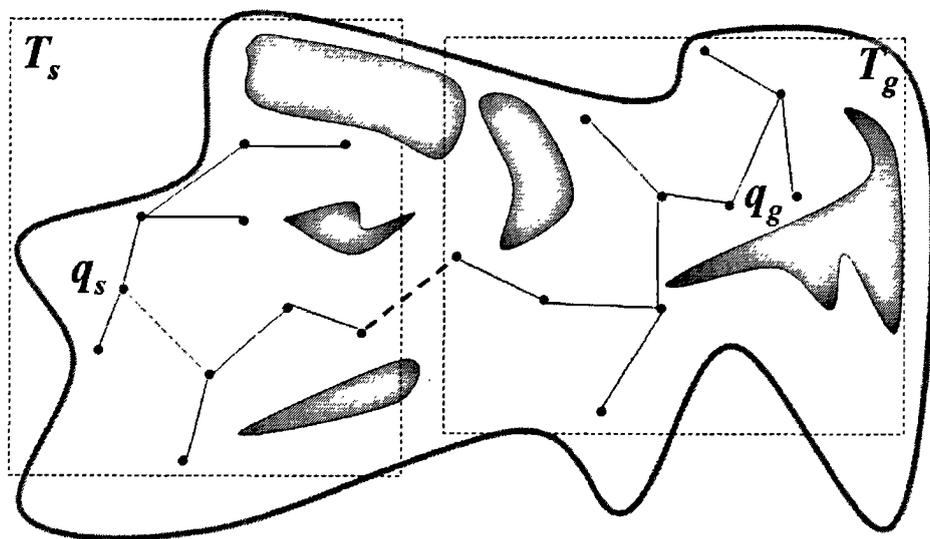


图 5.1 路径延伸示意图

桥接策略中关键问题是:如何保证路径树向外延伸,避免在局部区域过采样。为解决这个问题,规划器给每个位姿点定义一个权重参数 $w(q)$ 。 $w(q)$ 表示在位姿点 q 邻域内位姿点的密度,而 q 被选的概率为^[62]。

$$pr(q) = 1/w(q) \quad (5-1)$$

由式(5-1)可见, q 附近的位姿点越少,则被选中的概率越大。新增采样点被选为位姿点的条件与此相同。

这是一个网络多目标优化问题,目前还没有成熟的数学方法来解决。为

了求解此问题，本文采用一种启发式方法，把上述数学模型转化为以下数学模型：

$$\min \sum \frac{1}{w(q)}, \quad q \in C_f \quad (5-2)$$

此目标函数的含义就是在网络中寻找一条路径，其经过的所有位姿的权重的倒数和最小。

路径延伸执行步骤为：

(1) 分别以 1/2 的概率选择 T_s 和 T_g 作为路径延伸树 T ，在 T 中随机选取 q ，计算其邻域内位姿点密度，并将密度低的点作为待延伸节点。如图 5.2a 所示，此时 $w(q)=3$ ，虚框表示邻域大小。

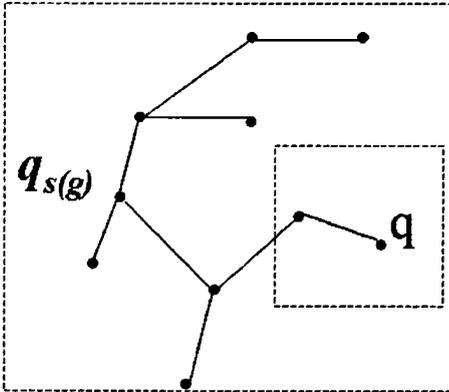


图 5.2(a) 选定低密度位姿点图

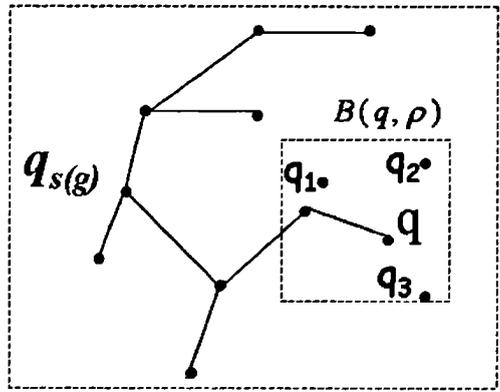


图 5.2(b) 均匀采样

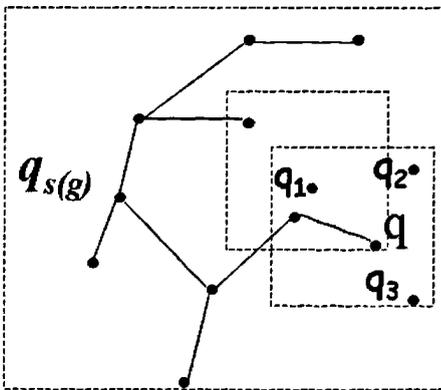


图 5.2(c) 计算新位姿点密度

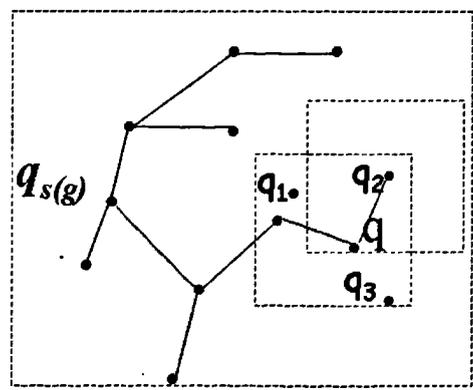


图 5.2(d) 路径延伸

(2) 位姿空间局部采样：在 q 点邻域内按均匀分布随机采样，并对采样点进行动态碰撞检测，无碰撞点构成局部自由空间，如图 5.2(b) 所示。通常

邻域的初值为 $B(q, \rho)$ ，位姿点 q 在位姿空间各轴上的邻域范围为 Q 。如果未找到新的位姿点，则收缩邻域范围 $B(q, \rho/i)$ ， $(i = 1, 2, \dots)$ ，直到有新的位姿点产生。这种收缩策略可以在狭窄的自由空间区域增加采样点密度，保证路标点的分布沿着树的组成空间沿着两颗树根逐渐分散。

(3) 分别计算局部自由空间采样点周围位姿点密度，如图 5.2(c) 中所示， $w(q_1) = 3$ ， $w(q_2) = 1$ ， $w(q_3) = 2$ 。 q_2 密度最小，则接入路径树，将其作为 q 的孩子，如图 5.2(d) 所示。

(4) 对新加入的位姿点 q 在其领域内检测是否存在其他树的位姿点，如果存在，则路径找到，停止路径延伸；否则继续。

路径延伸阶段我们采用贪婪碰撞检测策略。距离较近的路标点间发生碰撞的概率很低，碰撞检测上的大部分时间花费在连接测试上，大多数无碰撞的连接都不是最终路径的子路径，测试连接比碰撞检测要耗费更多的时间，这样，推迟连接测试直到完全需要时为止可以节省规划时间。

为了防止其中一棵树生长过快，路径延伸计算应在两者之间轮流进行，即所谓的双向采样^{[54][55]}。新生成局部路径不作碰撞检测，当其成为候选路径后才进行检测，平均来说，从新的位姿点到 q 的距离比开放自由空间区域要大^[58]，这样可以最大程度地节约计算时间。

5.2 路径安全性检验

路径延伸过程中，每个子节点只有一个父节点和它相连。这样连接起始和目标位姿的路径 τ 是唯一的，其中各个位姿点属于自由空间 C_f ，而位姿点间的局部路径未做碰撞检测，因此需要一个局部路径规划器。设路径 τ 由 p 个线段 u_1, u_2, \dots, u_p 组成。如果每个线段均是无碰撞的，则路径 τ 是安全的。

这里将线段离散化成一系列等长度的中间点。设线段长度向量为 L ，离散程度用 k 表示， $k = 1, 2, \dots$ ，则离散后的线段长度向量为 $L/2^k$ 。由于路径延伸计算时已对各位姿点做过碰撞检测，因此局部规划器不必检测线段两 endpoint。当 $k=1$ 时，线段被分割成 2 段，产生一个中间点 q_i 需进行碰撞检测。如果 $q_i \notin C_f$ ，则返回该段路径碰撞信息；若 $q_i \in C_f$ ，则检查 $L/2^k$ 是否小于给定的线段最小离散化阈值 ε ，如果条件成立则认为该线段是安全的；否则 $k+1$ ，继续离散线段，这样当碰撞确实发生时可以及时返回，节约检测时间。为

保证分层离散时，已经检测的上一层位姿点不再重复检测，用下面的公式保证只检测新生成的中间点。

$$q_i = q_1 + 2i\Delta L + \Delta L \quad (i = 0, 1, \dots, 2^{k-1}) \quad (5-3)$$

其中， $\Delta L = L/2^k$ 。

局部路径规划器一旦发现有碰撞线段，则将该线段删除，位姿点保持不动，此时路径断开。如果碰撞发生在路径树 T_s 和 T_g 的连接处，则将断开处作标记后重新进行树的延伸；如果发生在其他线段，则应从断开处将其子树由一棵树转移到另一棵树，其父子关系发生相应的变化，然后重新进行路径延伸计算，找到新的路径。找到的新路径仍执行路径碰撞检测。

反复执行上述过程，直到最后找到连接起始位姿点和目标位姿点的安全路径。如果采样的位姿点数超过给定最大采样点数 s ，则认为没有安全路径存在，路径规划失败。在进行路径碰撞检测时，将已检测安全的局部路径存储在数据库中，新的候选路径中如果包含这些局部路径，不必重复检测。

通过上述方法得到的路径一般不会是最优路径，如果直接驱动机器人会导致机械臂多余操作造成磨损，为此要对路径进行平滑处理，方法如第 4 章 4.3 节所述。

5.3 桥接测试

令 q_s' 、 q_g' 分别为在树 T_s 、 T_g 中最新创建的路标点，测试两点间的距离，如果两者距离比给定的常数 σ 小，将它们连接起来，称之为桥，并进行碰撞检测，若无碰撞返回连接 s 和 g 的路径 τ ，若碰撞则将发现碰撞的部分去掉，将路标图分成两个部分。文献[63]中介绍的 Repair-Conf 算法，对修正自由空间中的路标点给出了可行的方法。我们扩大搜索范围，若无成功连接，则路径规划失败。

5.4 软件设计及实现

5.4.1 软件设计流程

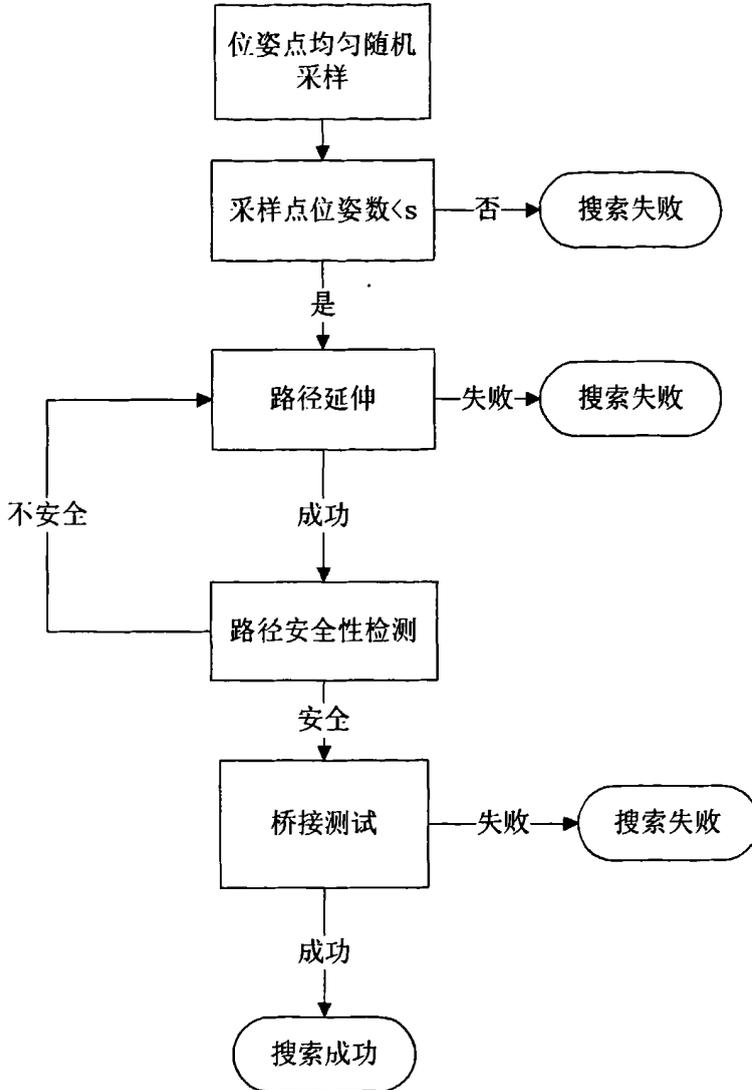


图 5.3 软件设计总体流程图

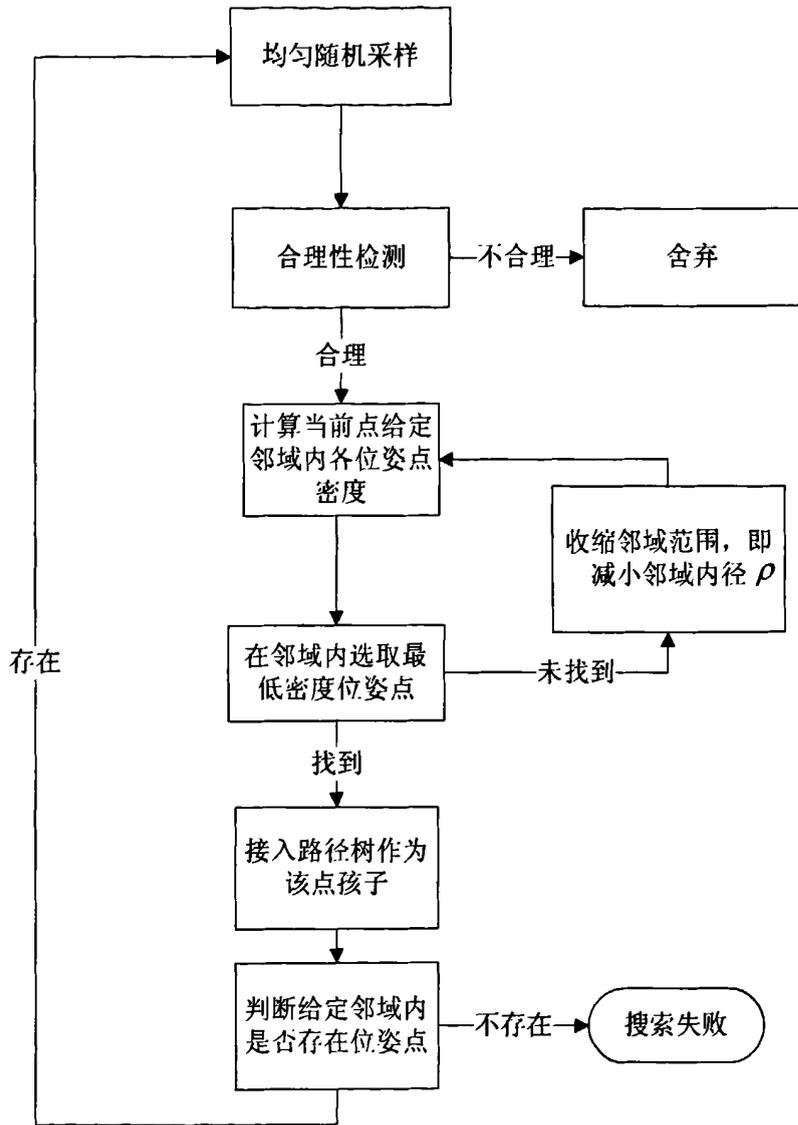


图 5.4 路径延伸流程图

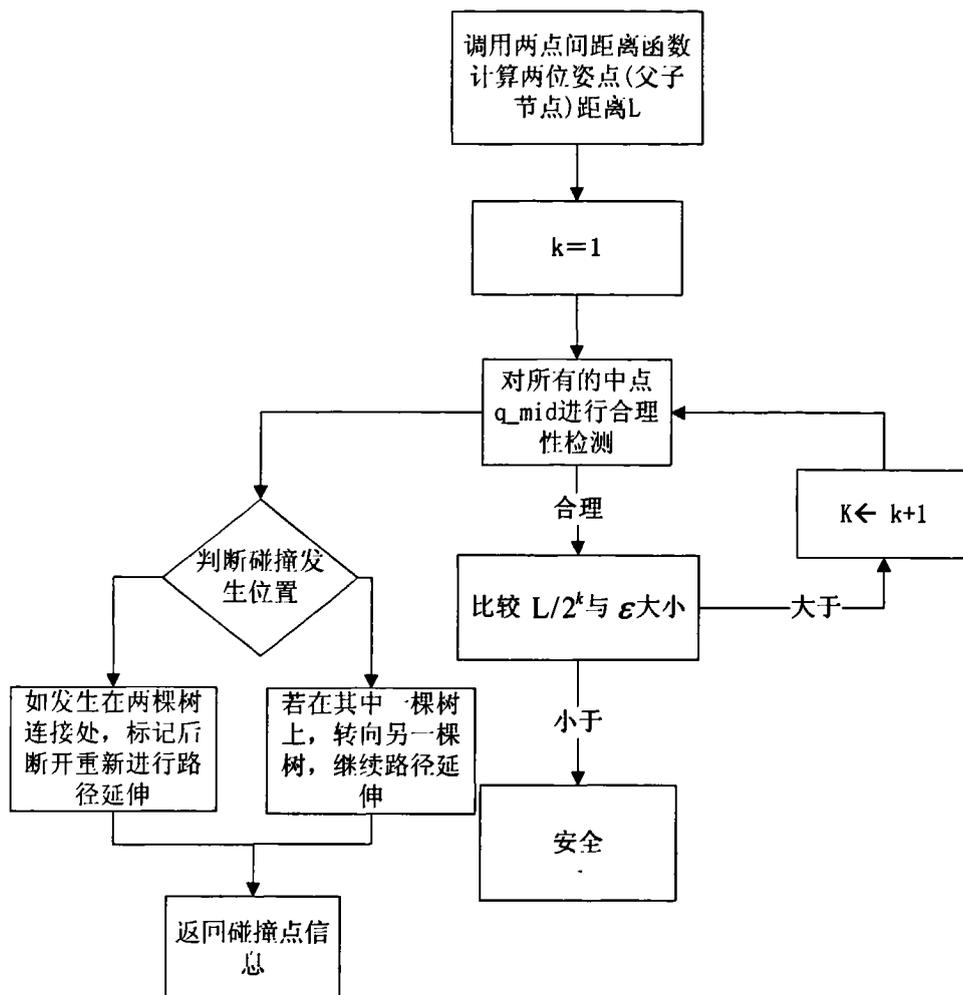


图 5.5 路径安全性检验流程图

路径延伸和路径安全性检验是基于贪婪碰撞检测的启发式双向路径搜索方法的难点，以上只给出了软件设计总体流程、路径延伸流程和路径安全性检验流程。

5.4.2 仿真结果及分析

为了与两阶段法进行对比，本节机械臂的起始点和目标点位姿，与 4.4.2 节中给出的完全相同。解算后得到机械臂的位形分别如图 5.6、图 5.8 所示。给出规划过程的一个中间位姿，如图 5.7 所示。用基于贪婪碰撞检测的启发式双向路径搜索方法对路径进行规划，得到的仿真曲线如图中绿色线条所示。

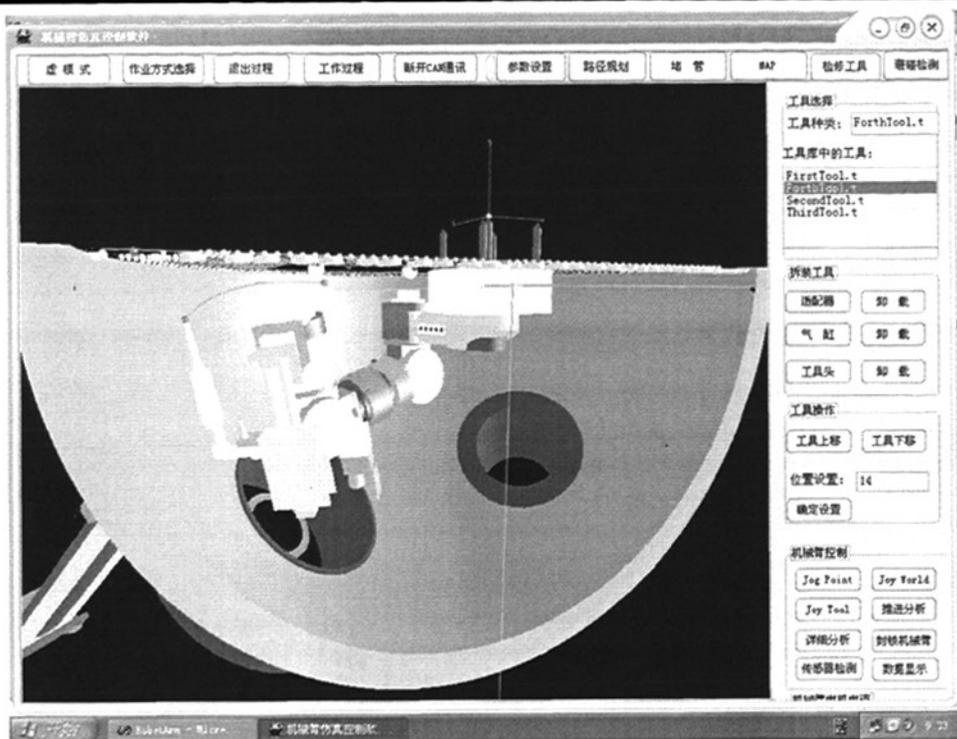


图 5.6 起始点位形图

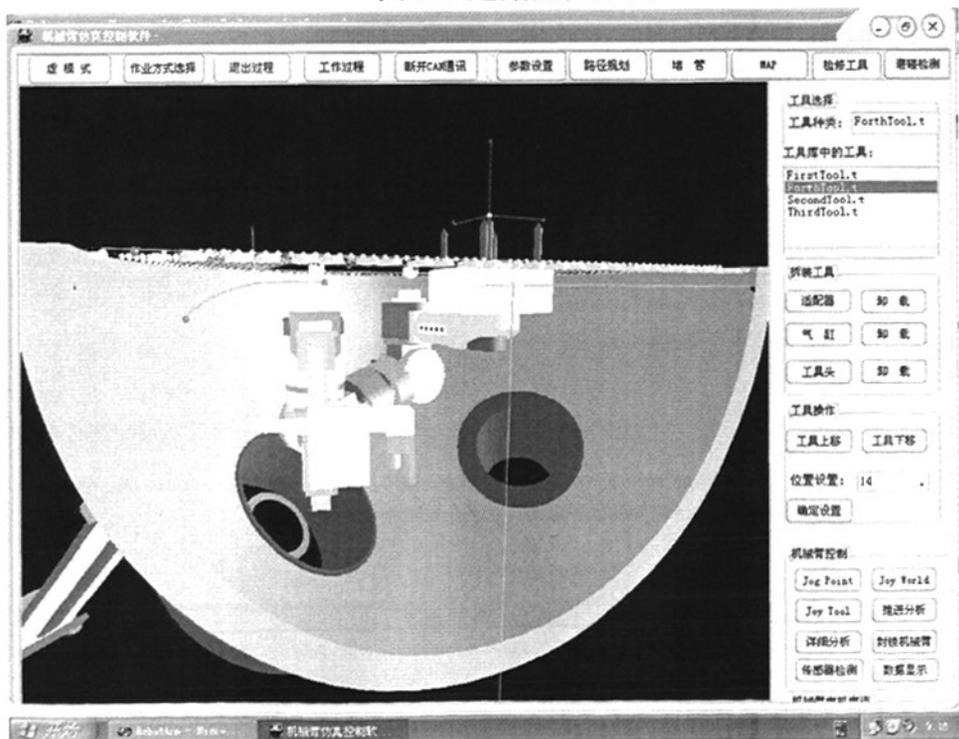


图 5.7 中间点位形图

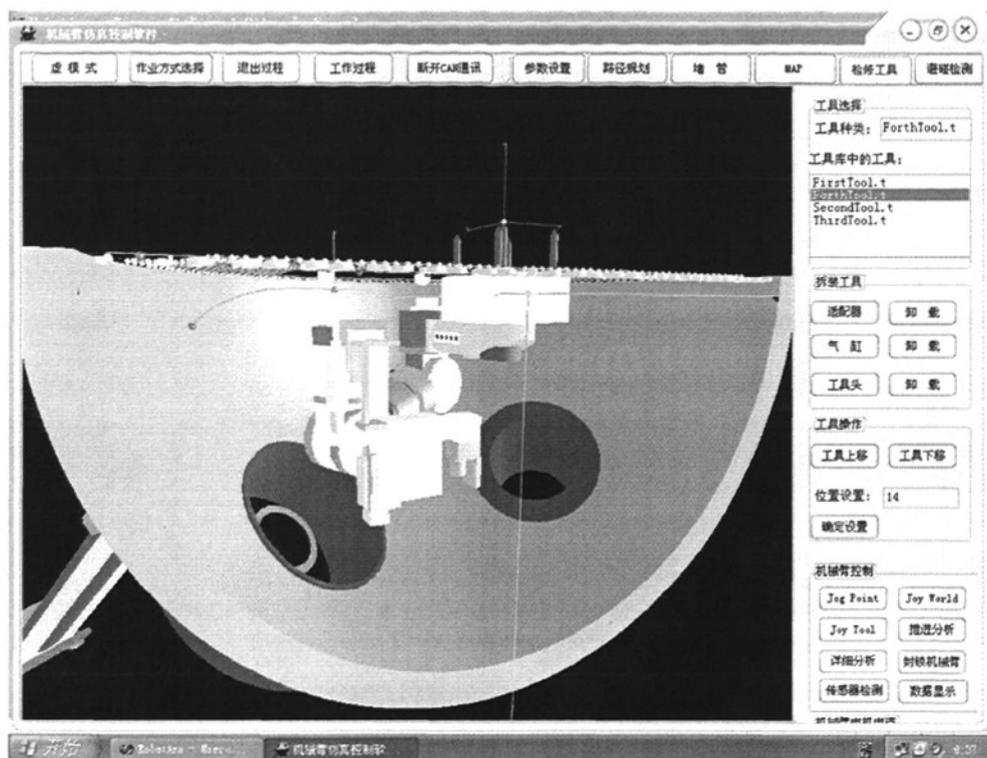


图 5.8 目标点位形图

表 5.1 中给出了起始点、中间点和目标点的位形数据，其中 x 、 y 、 z 为工具末端点坐标， α 、 β 、 γ 为工具末端（手端）姿态角， $\alpha_1 \sim \alpha_6$ 为一至六关节转角。

表 5.1 始末及采样点位形数据表

	起始点	中间点	目标点
x	0.266	0.258	0.141
y	-0.591	-0.513	-0.343
z	0.112	0.051	0.043
α	185.047	180.398	180.031
β	100.986	91.3938	89.901
γ	37.929	43.2896	69.214
α_1	352.738	351.766	356.290
α_2	263.051	260.178	250.394

α_3	17.091	18.055	18.127
α_4	357.873	357.969	359.931
α_5	354.080	343.318	341.802
α_6	213.944	230.493	236.901

5.5 两种路径规划方法性能比较

表 5.2 给出了不同规划时间下利用两阶段法进行避障路径规划成功的几率（其中 T_T 表示路径规划总耗时），从表中可以看出，规划时间越长，成功的几率越高，当超过一分钟时规划失败的可能性很小。

表 5.2 不同规划时间下两阶段法成功几率

T_T	T_L	T_Q	成功率
30.32	30.2	0.12	68%
40.34	40.2	0.14	88%
50.25	50.1	0.15	97%
60.48	60.3	0.18	97%
70.49	70.3	0.19	97%
80.40	80.2	0.20	97%

采用基于贪婪碰撞检测的启发式双向路径搜索进行路径规划所用时间约为 0.25s，随着始末距离的增加规划时间略有增长，相对于用传统的耗时数分钟去寻找路径方法相比，该方法实时性显著，达到了实时的要求。但是其失败的几率往往高于两阶段法。因此，前者成功率高，后者实时性强，实际规划时应依据需要采用不同的方法。

5.6 本章小结

在考虑学习时间时，两阶段法实时性不高，为了提高路径规划的实时性，本章提出了采用基于贪婪碰撞检测的启发式双向路径搜索对机械臂运动路径进行实时规划。不需要预计算路标图，采用贪婪碰撞检测策略，可以节省大

量的规划时间。两种方法的规划机理及特点不同，针对不同的问题可采用不同的方法处理。为了增强对比，与第四章仿真在相同的起始点和目标点位形下实施避障路径规划，实验表明该法都成功实现检修机械臂的在线路径规划。最后对两种方法的效能进行了仿真比较，前者成功率高，后者实时性更强。

第 6 章 结 论

本文欲实现在狭窄空间内多关节检修机械臂的避障路径规划。SG 机械臂的工作空间一般较为狭窄，但是其较大的关节力矩要求限制了机械臂尺寸无法进一步缩小，而且机械臂结构形式复杂，导致机械臂的碰撞约束非常复杂，从而使得机械臂在 SG 内进行无碰撞自动路径规划变得相当困难。

机械臂的运动学是避障路径规划研究的基础，本文对机械臂的运动学模型进行了建模。该机械臂虽然具有六自由度，但由于其结构和工作性质的特殊性，使其具有冗余机械臂的特性，因此其逆解变得更加复杂，因为它实质上是解一个非线性方程的问题，目前尚没有通用的解法。根据该机械臂结构的冗余度低，工作环境受限的实际情况，本文给出了一种简单易行的几何图解法，该算法不但稳定，而且简单、执行效率高，可用于在线求解，在 VC++ 仿真平台下对其运动学进行了仿真实验。

在机械臂进行检修工作时，如何判断和避免机械臂碰撞是实现自动作业的前提。采用基于包围盒和等效空间圆柱体相结合的网络包围法，通过简单有效的几何等效，把机械臂的碰撞检测问题转化为空间解析几何问题，大大简化了问题的求解，由机械臂运动学的正解可以实时快速地解出各个碰撞检测点的空间坐标，建立有效的信息存储数据结构，调用该碰撞检测算法即可以保证检修工作的无碰撞。该算法复杂度较低、实用性强，利于在线求解。实际工作验证了该算法的有效性、快速性，很好地解决了实时性和精确性的矛盾，为机械臂的安全检修提供了切实可靠的保障。

本文采用随机路标法在避障前提下对机械臂进行了路径规划。随机路标法的实施分为学习和查询两个阶段。学习阶段从自由空间中随机选择位形构建一个路标图，作为位形的集合。查询阶段使用学习阶段得到的路标图进行快速查询，得到可行路径。采用基于 Bezier 曲线的路径平滑技术对得到的路径进行平滑处理，可以得到平滑的运动轨迹。仿真结果表明，随着学习时间的延长，路径规划成功的几率就越高，当学习时间超过一分钟时，规划失败的可能性很小。

在考虑学习时间时，两阶段法实时性不高，为了提高路径规划的实时性，

文中提出了采用基于贪婪碰撞检测的启发式双向路径搜索对机械臂运动路径进行实时规划。关键是解决采样方式、路径延伸方法、动态碰撞检测、局部路径安全性检验、桥接测试是否成功等问题。该法不需要预计算路标图，采用贪婪碰撞检测策略，可以节省大量的规划时间。仿真实验表明，该法在实时性显著，规划时间明显少于两阶段法。但是，在规划成功性上要比两阶段法差。

两种方法的规划机理不同，效果也不同，前者更可靠，后者实时性更高，针对不同的问题可采用不同的方法处理。

方法改进的几点建议：

1.在保证成功率的前提下，为了能提高两阶段法的规划效率，可以对学习阶段和查询阶段交叉进行。

2.基于贪婪碰撞检测的启发式双向路径搜索明显提高路径规划的规划效率，但成功率不高，进一步提高避障路径规划的可靠性和规划效率是今后研究的重点。

3.另外，在不考虑计算机性能约束的前提下，可以采用一种我们称之为射线束逆向搜索策略的规划方法。即在启动避碰前提下从各关节点发射射线（可定为 30 条），采样位姿，从一关节到六关节，查找六关节旋转后最终位姿与目标位姿的近似程度，选取最近似位姿，逆向搜索路径即各轴转角。该法有一定理论意义，也是今后路径规划可尝试的一种方法。

参考文献

- [1] 孙增圻. 智能控制理论与技术. 北京: 清华大学出版社, 1997: 3-45页
- [2] J. C. Latombe. Robot Motion Planning. Norwell: Kluwer, 1991. 143-176
- [3] 王仲民, 岳宏. 一种移动机器人全局路径规划新型算法. 机器人, 2003, 25 (2): 152-155
- [4] S. Baase, A. V. Gelder. Computing Algorithms. Beijing: Higher Education Press
- [5] 姜松开, 夏凯, 吕林. 机器人路径规划. 机器人. 1989, 3(1): 6-11页
- [6] L. Aleksandrov, A. Maheshwari, J. R. Sack. Approximation algorithms for geometric shortest path problems. In Proceedings of the 32nd Annual ACM Symposium on Theory of Computing, 2000, pages 286-295
- [7] Z. Sun, J. H. Reif. BUSH WHACK: An approximation algorithm for minimal paths through pseudo-Euclidean spaces. In Proceedings of the 12th Annual International Symposium on Algorithms and Computation, volume 2223 of Lecture Notes on Computer Science, 2001, pages 160-171
- [8] 陈阳, 吴裕树, 史万明. 机器人路径规划的凸点法. 北京理工大学学报, 1995, 15(5): 17-20页
- [9] S. Baase, A. V. Gelder. Computing Algorithms. Beijing: Higher Education Press, 2001. 235-267
- [10] 钱敏平, 龚光鲁. 从数学角度看计算智能. 科学通报, 1998, 43 (16): 1681-1695
- [11] 戴光明. 避障路径规划的算法研究. 华中科技大学博士学位论文
- [12] Lozano-Perez. T. Automatic planning of manipulator transfer movement[J]. IEEE Trans on Systems, Man and Cybernetics, 1981, 11(10): 681-698

- [13] Maciejewski AA, Fox JJ. Path planning and topology of configuration space [J]. IEEE Trans on Robotics and Automation, 1993, 9(4): 444-456
- [14] 宋 晖, 张 华, 高小明. 智能机器人路径规划及算法研究[J]. 机器人. 2006 年第 22卷. 第11-2期: 244-246页
- [15] Metea MB. Planning for intelligence autonomous land vehicles using hierarchical terrain representation [C]. In: Proc IEEE Int Conf on Robotics and Automation, 1987. 1947-1952
- [16] Watanabe K, Izumi K. A survey of robotic control systems constructed by using evolutionary computations[C]. In: Proc IEEE Int Conf on Systems, Man and Cybernetics, Tokyo, Japan, 1999, 758-763
- [17] Kazuo Sugihara, John Smith. Genetic Algorithms for Adaptive Motion Planning of an Autonomous Mobile Robot[A]. Proceedings 1997 IEEE International Symposium on Computational Intelligence in Robotics and Automation[C]. 1997
- [18] K Sugihara. A genetic algorithm for 3-D Path Planning of a Mobile robot. Technical Report Univ. of Hawaii Manoa. 1996
- [19] K Sugihara. GA-based on-line path planning for SAUVIM. Lecture Notes in Artificial Intelligence. Springer-Verlag. 1998
- [20] 恽为民, 席裕庚. 基于遗传算法的机器人关节空间最优运动规划[J]. 机器人. 1995, 7(4): 206-216页
- [21] Nelson HC, Ye Y C. An intelligent mobile vehicle navigator based on fuzzy logic and reinforcement learning [J]. IEEE Trans on Systems, Man and Cybernetics, Part B: Cybernetics, 1999, 29(2): 314-321
- [22] 邢军, 王杰. 神经网络在移动机器人路径规划中的应用研究[J]. 微计算机信息, 2005, 22: 110-111
- [23] D. E. Koditschek. Exact robot navigation by means of potential functions: some topological considerations. Proc. IEEE Int.

- Conf. Robotics and Automation, 1-6, Raleigh, NC, 1987
- [24] J. Barraquand and J. C. Latombe. Robot motion planning: a distributed representation approach. *Int. J. Robotics Research*, 10:628, 649, 1991
- [25] E. Rimon and D. E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Tans. on Robotics and Automation* 8: 501—518, 1992
- [26] M. Overmars. A random approach to motion planning, Tech. Rep. RUU-CS-92-32, Dept. Comput. Sci. Utrecht Univ, Utrecht, The Netherlands, October 1992
- [27] M. Overmars and P. Svestka. A probabilistic learning approach to motion planning. In *Algorithmic Foundations of Robotics*, K. Goldberg et al. (eds.) A K Peters, Wellesley, MA, 19-37, 1995
- [28] P. Svestka. A probabilistic approach to motion planning for car-like robots. Tech. Rep. RUU-CS-93-18, Dept. Comput. Sci. Utrecht Univ, Utrecht, The Netherlands, April 1993
- [29] P. Svestka and M. Overmars. Motion planning for car-like robots using a probabilistic learning approach. Tech. Rep. RUU-CS-94-33, Dept. Comput. Sci. Utrecht Univ, Utrecht, The Netherlands, May 1994
- [30] P. Svestka and M. Overmars. Coordinated motion planning for multiple car-like robots using probabilistic roadmaps. To appear in the *Proc. IEEE Int. Conf. Robotics and Automation*, Nagoya, Japan, 1995
- [31] L. E. Kavraki and J. C. Latombe. Randomized preprocessing of configuration space for fast path planning. Tech. STAN-CS-93-1490, Dept. Comput. Sci. Stanford Univ, Stanford, CA, September 1993
- [32] L. E. Kavraki and J. C. Latombe. Randomized preprocessing of configuration space for fast path planning. *Proc. IEEE Int.*

- Conf. Robotics and Automation, 2138, 2145, San Diego, CA, 1994
- [33] L. E. Kavraki. Random networks in configuration space for fast path planning. Ph. D. Thesis, Tech. Rep. STAN-CS-95-1535, Dept. Comput. Sci. Stanford, Univ. Stanford, CA, January 1995
- [34] L. E. Kavraki. J. C. Latombe. R. Motwani. and P. Raghavan. Randomized query processing in robot path planning. To appear in STOC95, Las Vegas, NV, 1995
- [35] 蔡自兴. 机器人学[M]. 北京: 清华大学出版社, 2000
- [36] Warren C W. Multiple robot path coordination using artificial potential fields [A]. Proceedings of IEEE Int Conference on Robotics and Automation [C]. Sandiego, 1990. 500-505
- [37] Buckley S J. Fast motion planning for multiple moving robots [A]. Proceedings of IEEE Int Conf on Robotics and Automation [C]. Scottsdale, Arizona, 1989. 322-326
- [38] Liu YH, Kuroda S, Naniw T, et al. A practical algorithm for planning collision-free coordinated motion of multiple mobile robots [A]. Proceedings of IEEE Int Conf on Robotics and Automation [C]. Scottsdale, Arizona, 1989. 1427-1432
- [39] MARTNEZ -SALVADOR B, PEREZ-FRANC ISCOM, DEL POBL A. P. Collision detection between robot arms and people [J]. J Intell and Robotic Syst, 2003, 38 (1): 105-119
- [40] CHAN G C, CHANG M J, BIEN Z. Collision-free motion planning for two articulated robot arms using minimum distance functions [J]. Robotica, 1990, 8 (1) : 137-144
- [41] 熊勇刚, 周友行, 林峰. 多臂机器人关节间的碰撞检测研究[J]. 机械科学与技术. 2004, 23 (7) : 866-869页
- [42] 王志强, 洪嘉振, 杨辉. 碰撞检测问题研究综述[J]. 软件学报. 1999, 10(5): 545-551页
- [43] 陆军, 刘其端, 朱齐丹. 六自由度检修机械臂监控软件设计与实现[J]. 自动化技术与应用. 2008. 12

- [44] 陆军, 刘其端. 蒸汽发生器检修机器人监控软件设计[J]. 计算机仿真. 2008. 3
- [45] 和平鸽工作室编著. OpenGL三维图形系统开发和实用技术[M]. 重庆: 重庆大学出版社, 2003: 9-1页, 21-23页
- [46] 范晓磊. 混合包围盒碰撞检测算法研究. 青岛大学硕士学位论文
- [47] 魏迎梅, 王涌, 吴泉源, 石教英. 碰撞检测中的层次包围盒方法[J]. 计算机应用. 第20卷增刊2000年8月
- [48] 陈诚, 孙妮芳, 杨志强. 基于凸分解与OBB层次结构的碰撞检测方法[J]. 计算机技术与发展. 第17卷第11期2007年11月
- [49] David Hsu, Gildardo S´anchez-Ante, Ho-lun Cheng and Jean-Claude Latombe. Multi-Level Free-Space Dilation for Sampling Narrow Passages in PRM Planning
- [50] Mitul Saha and Jean-Claude Latombe. Finding Narrow Passages with Probabilistic Roadmaps: The Small Step Retraction Method [J]. 2006
- [51] Fabian Schwarzer, Mitul Saha, Jean-Claude Latombe. Exact Collision Checking of Robot Paths. Dept. of Computer Science, Stanford University, Stanford, CA 94305
- [52] 周友行, 邹湘伏, 何清华. 多臂机器人关节间的碰撞检测研究[J]. 工程机械基础理论、关键技术研究
- [53] Chirikjian G S, Burdick J.W. A hyper-redundant manipulator [J]. IEEE Robotics & Automation Magazine, 1994, (4): 22-29
- [54] Kavraki L E, Latombe J C, Overmars M, Probabilistic roadmap for path planning in high-dimensional configuration spaces[J]. IEEE Transactions on Robotics and Automation, 1996, 12(4): 566-580
- [55] Boor V, Overmars M H, Stappen A F van der. The gaussian sampling strategy for probabilistic roadmap planners[A], Proceeding of the 1999 IEEE International Conference on Robotics and Automation[C], [s. l.]: IEEE Robotics and Automation Society IEEE, 1999, 1018-1023

- [56] Shingo Ando. A fast collision-free path planning method for a general robot manipulator[J]. Proceedings of the 2003 IEEE Internatiooal Conference on Robotics & Automation Taipei, Taiwan, September 14-19, 2003: 2871-2877P
- [57] 高晶. Bezier 曲线的算法研究[J]. 辽宁师专学报. 2008年第2 期
- [58] BohlinR , Kavraki L E. Path planning using lazy PRM[C]. Proceeding sIEEE International Conference on Robotics and Automation, SanFrancisco, CA, USA, 2000: 521-528
- [59] Snchez G, Latombe J.C. Single-query bi-directional motion planning with lazy collision checking[C]. Procl0th International Symposium on Robotics Research, Lorne, Australia, 2001: 403-418
- [60] Sánchez G, Latombe J C. On delaying collision checking in PRM planning application to multi-robot coordination[J]. Int. J. of Robotics Research, 2002, 21 (1): 5-26
- [61] D. Hsu, T. Jiang, J. Reif, and Z. Sun. The Bridge Test for Sampling Narrow Passages with Probabilistic Roadmap Planners. Proc. IEEE Int. Conf. Robotics and Automation. 2003: 4420-4426
- [62] Hsu D. Randomized single query motion planning in expansive spaces [D]. California Stanford University. CA. 2000
- [63] Mitul Saha and Jean-Claude Latombe. Finding Narrow Passages with Probabilistic Roadmaps: The Small Step Retraction Method. Artificial Intelligence Laboratory Stanford University, Stanford, California, 94305, USA

攻读硕士学位期间发表的论文和取得的科研成果

- [1] 朱齐丹.周芳.赵国良.包卫卫.基于反步法和滑模观测器的船舶航向控制.华中科技大学(自然科学版).已录用
- [2] 陆军.包卫卫.蒸汽发生器检修机械臂碰撞检测研究.哈尔滨工程大学自动化学术会议论文集.已录用
- [3] 陆军.包卫卫.朱齐丹.多关节检修机械臂动态碰撞检测研究.电子技术及应用.已投稿

致 谢

在论文完成之际，首先要感谢我的导师陆军教授，从论文的选题到撰写的全部工作都是在陆老师的悉心指导和无私的帮助下完成的。在就读研究生的期间，得到了导师陆军教授在学习、生活中给予的极大帮助。陆老师在日常工作和生活中的为人师表、坦诚豁达，在研究工作中的缜密细致、精益求精，在学术上的见解独到以及对新知识的敏锐的洞察力，都给我留下了深刻的印象，使我受益良多，他的热情鼓励和谆谆教诲，我将永远铭记。陆老师特别注重工作方法、工作方向的指导，使我的工作能力得到了很大的提高。

其次要感谢朱齐丹老师、孟浩老师、栗鹏老师、原新老师、苏丽老师、张智老师、蔡成涛老师、王立辉老师、吕晓龙老师，还有给予我极大帮助的实验室同学尤其是机械臂课题组的同学，正是这个团结、奋进的集体使我在紧张的研究生生活中始终保持着乐观向上的精神状态、充满着勇于面对困难的决心和勇气。

感谢我的父母、亲人和朋友，正是他们的支持、鼓励、和帮助才使我完成学业。特别要感谢我尊敬和爱戴的父母，他们的无私奉献和默默的支持是激励我不断追求的动力！

最后，对本文所有被引用文献的作者表示感谢，也感谢各位专家和教授对本文的评阅和赐教。